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. For the safety precautions of the programmable controller system, refer to the MELSEC iQ-R Module Configuration Manual.

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.
- Configure a circuit so that the external power supply is turned off first and then the programmable controller. If the programmable controller is turned off 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. Doing so may result in malfunction due to electromagnetic interference. Keep a distance of 100 mm or more between those cables.
- 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.
- When changing the operating status of the CPU module from external devices (such as the remote RUN/STOP functions), select "Do Not Open by Program" for "Opening Method" of "Module Parameter". If "Open by Program" is selected, an execution of the remote STOP function causes the communication line to close. Consequently, the CPU module cannot reopen the line, and external devices cannot execute the remote RUN function.

[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 (IB-0800525). 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, until the hook(s) located at the top snaps into place. 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 component or wire, short circuit, or malfunction. Overtightening can damage the screw and/or module, resulting in drop, short circuit, or malfunction. For the specified torque range, refer to the MELSEC iQ-R Module Configuration Manual.
- 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.
- Beware that the module could be very hot while power is on and immediately after power-off.
- Do not directly touch any conductive parts and electronic components 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 before powering on the system for operation. Also, attach an extension connector protective cover[*1] to each unused extension cable connector as necessary. Directly touching any conductive parts of the connectors while power is on may result in electric shock.
  *1 For details, please consult your local Mitsubishi Electric representative.

[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. Failure to do so may result in malfunction due to noise. Keep a distance of 100 mm or more between those cables.
- 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.
- When a protective film is attached to the top of the module, remove it before system operation. If not, inadequate heat dissipation of the module may cause a fire, failure, or malfunction.
- 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
    - Connecting/disconnecting the extension cable to/from the base unit
- 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. Wearing a grounded antistatic wrist strap is recommended. Failure to discharge the static electricity may cause the module to fail or malfunction.

[Startup and Maintenance Precautions]

⚠CAUTION

- After unpacking, eliminate static electricity from the module to prevent electrostatic discharge from affecting the module. If an electrostatically charged module comes in contact with a grounded metal object, a sudden electrostatic discharge of the module may cause failure.
  For details on how to eliminate static electricity from the module, refer to the following.
  Antistatic Precautions Before Using MELSEC iQ-R Series Products (FA-A-0368)
- Use a clean and dry cloth to wipe off dirt on the module.
- 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.

[Computer Connection Precautions]

⚠CAUTION

- When connecting a personal computer to a module having a USB interface, observe the following precautions as well as the instructions described in the manual for the personal computer used. Failure to do so may cause the module to fail.
  (1) When the personal computer is AC-powered
  When the personal computer has a 3-pin AC plug or an AC plug with a grounding wire, connect the plug to a grounding receptacle or ground the grounding wire. Ground the personal computer and the module with a ground resistance of 100 ohms or less.
  When the personal computer has a 2-pin AC plug without a grounding wire, connect the computer to the module by following the procedure below. For power supplied to the personal computer and the module, using the same power source is recommended.
  1. Unplug the personal computer from the AC receptacle.
  2. Check that the personal computer is unplugged. Then, connect the personal computer to the module with a USB cable.
  3. Plug the personal computer into the AC receptacle.
  (2) When the personal computer is battery-powered
  The personal computer can be connected to the module without taking specific measures.
  For details, refer to the following.
  Cautions When Using Mitsubishi Programmable Controllers or GOTs Connected to a Personal Computer With the RS-232/USB Interface (FA-A-0298)
  When the USB cable used is the GT09-C30USB-5P manufactured by Mitsubishi Electric, specific measures are not required to connect the AC-powered personal computer to the module. However, note that the signal ground (SG) is common for the module and its USB interface. Therefore, if an SG potential difference occurs between the module and the connected devices, it causes failures of the module and the connected devices.

[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.

- 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

PART 1 BASIC SPECIFICATIONS

CONTENTS

PART 2 MOTION CONTROL

PART 3 MOTION CONTROL

CONTENTS

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
Motion Module Quick Start Guide [L03191ENG]	Describes system startup, parameter settings, and programming methods for first-time users of the Motion module	e-Manual PDF
Motion Module Quick Start Guide (PLC CPU Ladder Program) [L03194ENG]	Describes system startup, parameter settings, and programming methods for first-time users of the Motion module	e-Manual PDF

For programs, refer to the following.

MELSEC iQ-R Programming Manual (Program Design)

Point

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

e-Manual has the following features:

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

TERMS

Unless otherwise specified, this manual uses the following terms.

Term	Description
Aborting, Buffered, Blending	Various modes of the buffer mode
Actual position	A position address returned by a device station assigned to an axis that is rounded by the ring counter. It is affected by a current position change.
Actual velocity	A value obtained by converting the actual value from the device station 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
Axis error	An error or a warning related to an axis
Axis warning	
Axes group error	
Axes group warning	
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
Cam table	An operation profile used for cam control (Example) Operation profile data of cam data format or operation profile data of cam data format for rotary cutter
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 path data	Positioning data belonging to the "Continuous path" operation pattern type.
Continuous update	Continuously reflecting input values to the control while the ContinuousUpdate input of the motion control FB is TRUE
Control slave axis	A device that has the servo parameter "Driver communication setting - Slave - Master axis 1 - Control slave axis No. setting (PD23.1)" set to "1 to 8 (Master axis control station)" in a slave axis of the master-slave operation.
Cumulative current position	A command position address that is not rounded by a ring counter. It is affected by a current position change.
Cyclic transmission	A function by which data are periodically exchanged among stations on the same network
Dedicated instruction	An instruction for using functions of the module
Device	Various memory data in a module. There are devices handled in each bit and in each word.
Disconnection	A process of stopping data link if a data link error occurs
Driver communication	Using the master-slave operation function this function controls the master axis with the motion system, while the slave axes are controlled by data communication between servo amplifiers (driver communication) without using the motion system.
Drive unit error	An error or a warning occurred in the drive unit
Error reset	Clearing the detection status of errors and warnings
Execute	Setting TRUE to the Execute/Enable input for the motion control FB.
External signal high-accuracy input	An input signal linked with signal detection time of the device station
First point of cam table	The meaning is different depending on the interpolation method specification for the cam data • For Section interpolation, Spline interpolation: Point of Table No.1 • Linear interpolation: Point of No.1
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

Term	Description
Label	A variable used for a program
Link device	A device in a module on CC-Link IE
Link refresh	Automatic data transfer between a link device of the Motion module and a device in a CPU module
Machine feed value	A commanded position address that is not rounded by the ring counter. It is based on the home position, and it is not affected by a current position change.
Master axis	A device that sends commands to the slave axis of the master-slave operation. This applies to real drive axes.
Master axis (input) absolute coordinate	Another term combining input variable Master axis absolute coordinate (MasterAbsolute) of MC_CamTableSelect (Cam Table Selection)
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 device station to exchange cyclic data by a slave label and motion control
Motion control FB	An FB that is related to the motion control. Its name starts with MC_ or MCv_.
Motion FB	A motion control FB that takes an axis or an axes group for the argument and changes the axis status or the axes status by execution
Motion module	Another term for the RD78G(H)
Motion service processing	Processing in the motion system that does not require real-time
Motion synchronization station	Another term for motion control station
MR-J5-G	Servo amplifier model MR-J5-_G_(-RJ)
MR-J5D-G	Servo amplifier model MR-J5D_-_G_
MR-J5W-G	Servo amplifier model MR-J5W_-_G
MR-JET-G	Servo amplifier model MR-JET-G
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 axes interpolation data	Positioning data belonging to the "Positioning (multiple axes interpolation)" control method type.
Multiple start	Executing another motion control FB to the same axis while a motion control FB is executed
Network area	Software that performs the network control in the Motion module
Next FB	Another motion control FB that starts while an axis and an axes group are operated
Normal axis	A device with no master-slave operation setting.
Object	Various data of a device station compatible with CANopen
On-going FB	Motion control FB that is in execution on the axis and the axes group (FB that Busy is TRUE)
Output variable	An output argument of FB
Positioning complete data	Positioning data when the operation pattern is "Positioning complete".
Positioning continue data	Positioning data belonging to the "Positioning continue" operation pattern type.
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 device station on network
Real drive axis	An axis that is linked with a device station 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 device station. It is used for the master axis of the single axis synchronous control.
Reconnection	A process of restarting data link when a station recovers from an error
Retrigger	Starting up the Execute input again while executing an FB to a motion control FB that has the Execute input
Safety communications	A function to exchange safety data between safety stations on the same network
Service task	A dedicated task for executing the motion service processing
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)
Slave axis	A device that receives commands from a master axis under master slave operation. This applies to Motion control stations that have not been assigned to axes.

Term	Description
Slave axis (output) absolute coordinate	Another term combining input variable Slave axis absolute coordinate (SlaveAbsolute) of MC_CamTableSelect (Cam Table Selection)
Software reboot	Resetting only motion area and network area without a hardware reboot
Standard station	A device station other than motion synchronization (control) stations
Start	Executing a motion FB for the first time to an axis or an axes group
Switching speed	• The set velocity of the on-going motion control FB when specifying Aborting • The set velocity when the on-going motion control FB reached the target position when specifying Blending
System basic cycle	A basic cycle of fixed cycle processing such as motion operation processing
System error	An error or a warning not related to an axis When Axis information (Axis) and Axes Group Information (AxesGroup) of the Motion control FBs are not correct, the system error will occur.
System start	Carrying out an initialization processing of the Motion module by power ON or a reboot of the system
System memory	A generic term for memory areas which are available add-ons in the motion system. The areas do not include the memory area used for baseSystem.
Target position	A commanded position address that is specified in the motion FB
Transient transmission	A function of data communication unperiodically among nodes (station) on network. A function used to send message to the target station when requested by a link dedicated instruction or the engineering tool Communication is available with station on another network via relay station, or gateway.
Under-control FB	Motion control FB that is in execution on the axis and the axes group (FB that Active is TRUE)
Virtual axis	An axis that is not linked with a device station on network
Virtual encoder axis	An axis that generates commanded position from a variable. It is used for the master axis of the single axis synchronous control.
Virtual drive axis	A virtual axis that can generate a command virtually
Virtual linked axis	An axis that connects FBs and transmits a command in the single axis synchronous control

GENERIC TERMS AND ABBREVIATIONS

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

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

MEMO

PART 1 BASIC SPECIFICATIONS

This part consists of the following chapters.

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.

Operation of this function for each system status

○: Possible, △: Possible (restricted), ×: Not possible

System status	Operation availability
STOP	○
RUN	○
Moderate error	△ (Operation may not be possible according to the error status.)*1
Major error	×

*1 When a required add-on is stopped due to "Add-on Library Load Error (error code: 3205H)" or "Insufficient Add-on System Memory (RAM) (error code: 3209H)", operation will not be possible.

Relevant variables

Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
AxisStatus	Axis Status	Displays the current axis status. • -1: Axis variable uninitialization/Axis parameter error (Invalid) • 0: Axis disabled (Disabled) • 1: Stopping on error (ErrorStop) • 2: Decelerating to stop (Stopping) • 3: During homing (Homing) • 4: Standby (Standstill) • 5: During positioning operation (DiscreteMotion) • 6: During continuous operation (ContinuousMotion) • 7: During synchronous operation (SynchronizedMotion)

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.

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 1024 axes can be set. Settable number of axis varies depending on the system memory capacity setting.*2 Axis assignment method is as follows. • Set on the Engineering tool (Axis setting window)						
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.
<Example> For a 2-axis drive unit the number of axes counted is 2 axes.
*2 For the memory capacity, refer to the following.
☞ Page 60 Memory usage

If the axis setting exceeds the maximum number of controlled axes, "Maximum Setting Axis Number Over Warning (warning code: 0F0BH)" occurs.
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 Axis Status (AxisName.Md.AxisStatus) to "-1: Axis variable uninitialization/Axis parameter error (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. Re-importing to the control data is not executed after axis variables initialization.
Pr		Stores the axis parameter data. Opens the default value at the axis variable initialization. Re-importing to the control data is executed after axis variables 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.

Precautions

- Do not assign axis variables to each other. Writing an expression on a program that assigns axis variables to each other will copy all of the axis variables of the assignment source to the assignment target when the program is executed. Note that operating the axis variables of the assignment target after the assignment will not affect the internal status of the assignment source axis variables. (Operating the parameter (Pr)/control data (Cd) of the assignment target does not operate the parameter (Pr)/control data (Cd) of the assignment source.)
- Since obtaining all of the axis variables by assignment greatly affects the program processing time when referencing multiple monitor data, alternative methods such as defining the axis monitor variables (such as AXIS_REAL_MONI type/ AXIS_VIRTUAL_MONI type) to be obtained in assignments or copying only the required data are recommended.

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 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 does not turn to ON. For the label initialization processing when the PLC READY 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 then reconnected. (The axes can be emulated without network connection. (☞ Page 395 Axis Emulation))

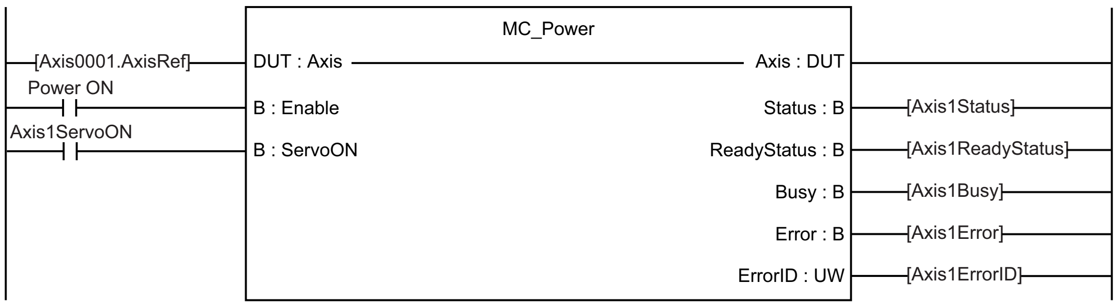
How to specify in a user program

When specifying the axis by the Motion control FB, set the AXIS_REF type member (AxisName.AxisRef) of the axis variable to the I/O variables which the data type is "AXIS_REF".

Ex.

When executing the MC_Power (Operation Available) with the following conditions:

- Axis type: Real drive axis
- Axis No.: 1
- Axis name: Axis0001



Axis status

Axis state transition

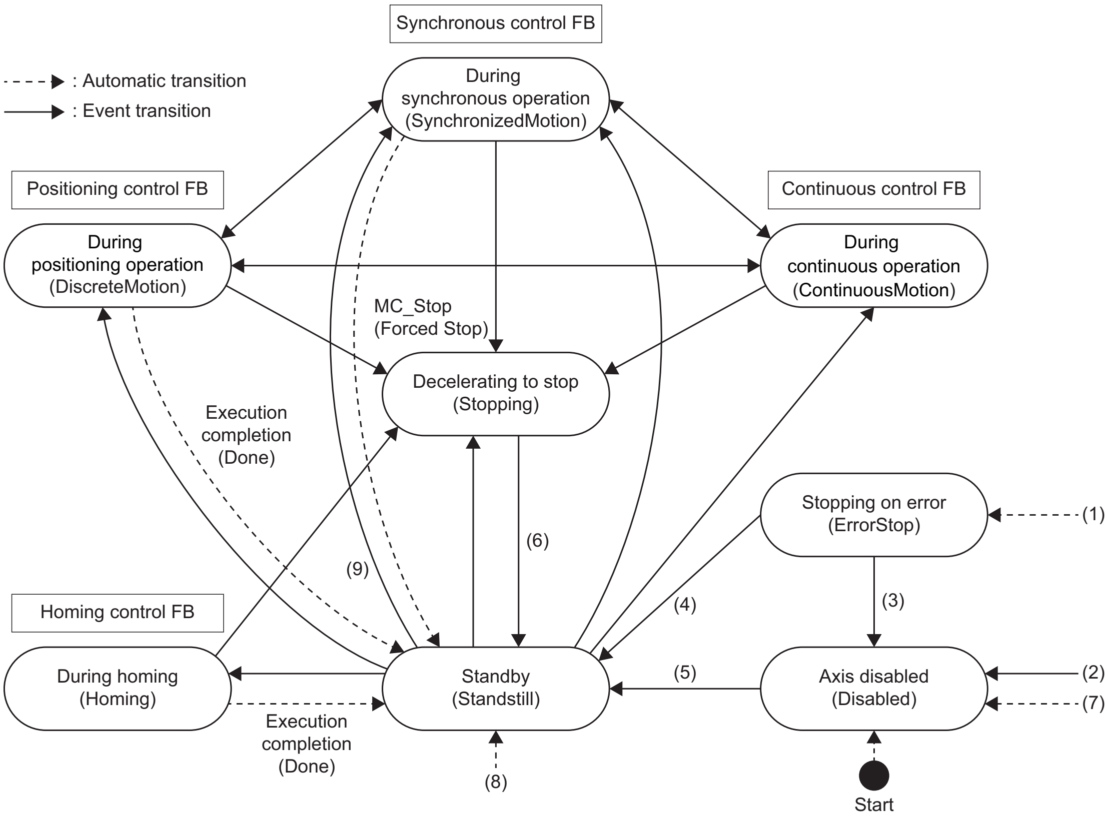
The following shows the status that a single axis can be in.

The current status can be confirmed in Axis Status (AxisName.Md.AxisStatus).

Status	Description
-1: Axis variable uninitialization/Axis parameter error (Invalid)	Axes cannot be used in this status due to axis variable uninitialization (parameter error occurrence at an initialization, etc.).
0: Axis disabled (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 device station is disconnected from the network.
1: Stopping on error (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: Decelerating to stop (Stopping)	The status where the deceleration stop has been executed by MC_Stop (Forced Stop). After the axis stop is completed, this status continues until the Execute command (Execute) of MC_Stop (Forced Stop) changes to FALSE. The Motion control FB cannot be executed in this status.
3: During homing (Homing)	The status where the homing has been executed in the driver or the motion system by the homing FB.
4: Standby (Standstill)	The status where the Motion control FB can be used. In case of a real drive axis, the axis is servo ON stop.
5: During positioning operation (DiscreteMotion)	The status where the positioning operation has been executed to the target position by the positioning control FB.
6: During continuous operation (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: During synchronous operation (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: Operating (GroupMoving)" status.

The status transits as shown below according to the start of the Motion control FB. The state transition may varies depending on the axis type. For the specifications of each axis type, refer to the following.

☞ Page 37 Axis Type



No.	State transition description
(1)	From any state, transits when an error in the axis occurred.
(2)	• From any state, transits when Enable (Enable) of MC_Power (Operation Available) is FALSE, and no error occurs in the axis. • Transits from "4: Standby (Standstill)" when Enable (Enable) of MC_Power (Operation Available) is TRUE, Servo ON request (ServoON) is FALSE, and no error occurs in the axis. • Transits when Process Selection at Servo OFF Command During Operation (AxisName.Pr.StopMode_ServoOff) is changed to which Servo ON request (ServoON) is FALSE from "0: Ignore (Ignore)" and "2: Decelerating to stop (Stopping)"*1.
(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 Enable (Enable) of MC_Power (Operation Available) is TRUE, Servo ON request (ServoON) is TRUE, and Operable (Status) of MC_Power (Operation Available) is TRUE.
(6)	Transits when Execution completion (Done) of MC_Stop (Forced Stop) is TRUE and Execute command (Execute) of MC_Stop (Forced Stop) is FALSE.
(7)	• Transits from "4: Standby (Standstill)" when the forced servo OFF by the driver is executed and the axis is enabled. • Transits when a communication with the driver is failed from 4: Standby (Standstill).
(8)	Transits automatically when the virtual axis is enabled.
(9)	Transits when the axes group used as configuration axes changes to "5: Operating (GroupMoving)".

*1 The axis status does not transit until the axis is stopped completely. After the axis stops completely the status transits to "0: Axis disabled (Disabled)". For details, refer to the following.

☞ Page 161 Stop

Precautions

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

Relevant add-ons

The following add-ons are required to use this function.

- Axis
- MotionEngine
- NetworkDriver_CCIETSN*1
- ServoDriver_CANopen*1

*1 When using the real axis

System memory capacity

For details, refer to the following.

☞ Page 60 Memory usage

1.2 Axis Type

Relevant variables

Real drive axis data type AXIS_REAL members (excluding AxisRef)

■ Axis parameter constant (AxisName.PrConst.)

Variable/Structure name	Name	Details
AxisType	Axis Type Setting	Sets the axis type. ☞ Page 56 Axis type
AddressOfStation	Station Address Setting	Sets the station address (IP address) with a character string. ☞ Page 57 Station address
PosRestoration_AbsPosBase	Absolute Position Reference Setting	When using the absolute position system, this variable sets the current position to be the basis of the current position restoration. ☞ Page 57 Absolute position control setting
OperationCycle	Control Cycle Setting	Sets the control operation cycle. ☞ Page 57 Control cycle
SlaveObject	Slave Object Data	Sets the slave object data to the real drive axis ☞ Page 59 Slave object settings

■ Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
AxisStatus	Axis Status	Displays the current axis status. • -1: Axis variable uninitialization/Axis parameter error (Invalid) • 0: Axis disabled (Disabled) • 1: Stopping on error (ErrorStop) • 2: Decelerating to stop (Stopping) • 3: During homing (Homing) • 4: Standby (Standstill) • 5: During positioning operation (DiscreteMotion) • 6: During continuous operation (ContinuousMotion) • 7: During synchronous operation (SynchronizedMotion)
UseInGroup	Using Axes Group	Displays whether the drive unit is being used in the axes group or not. • FALSE: Not used • TRUE: Used
Io_PosActualValue	Object Data_PosActualValue	Indicates a value of object data_PosActualValue. ☞ Page 59 Slave object settings
Io_Statusword	Object Data_Statusword	Indicates a value of object data_Statusword. ☞ Page 59 Slave object settings
Io_TargetPos	Object Data_TargetPos	Indicates a value of object data_TargetPos. ☞ Page 59 Slave object settings

Real encoder axis data type AXIS_ENCODER members (excluding AxisRef)

■ Axis parameter constant (AxisName.PrConst.)

Variable/Structure name	Name	Details
AxisType	Axis Type Setting	Sets the axis type. ☞ Page 56 Axis type
AddressOfStation	Station Address Setting	Via drive unit: Sets 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 (Drive)
Encoder_CounterDisableSignal	Counter Disabling Signal	Sets the signal to switch to the counter disable. ☞ Page 88 Follow Up

■ Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
AxisStatus	Axis Status	Displays the current axis status. • 0: Axis disabled (Disabled) • 1: Stopping on error (ErrorStop) • 2: Decelerating to stop (Stopping) • 3: During homing (Homing) • 4: Standby (Standstill)
Encoder_CounterDisable	Counter Disabled	Becomes 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)

■ Axis parameter constant (AxisName.PrConst.)

Variable/Structure name	Name	Details
AxisType	Axis Type Setting	Sets the axis type. ☞ Page 56 Axis type
PosRestoration_AbsPosBase	Absolute Position Reference Setting	When using the absolute position system, this variable sets the current position to be the basis of the current position restoration. ☞ Page 57 Absolute position control setting
OperationCycle	Control Cycle Setting	Sets the control operation cycle. ☞ Page 57 Control cycle

■ Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
AxisStatus	Axis Status	Displays the current axis status. • -1: Axis variable uninitialization/Axis parameter error (Invalid) • 0: Axis disabled (Disabled) • 1: Stopping on error (ErrorStop) • 2: Decelerating to stop (Stopping) • 3: During homing (Homing) • 4: Standby (Standstill) • 5: During positioning operation (DiscreteMotion) • 6: During continuous operation (ContinuousMotion) • 7: During synchronous operation (SynchronizedMotion)
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)

■ Axis parameter constant (AxisName.PrConst.)

Variable/Structure name	Name	Details
AxisType	Axis Type Setting	Sets the axis type. ☞ Page 56 Axis type
Encoder_CounterDisableSignal	Counter Disabling Signal	Sets the signal to switch to the counter disable. ☞ Page 88 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 specified -2147483648 to 2147483647

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

■ Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
AxisStatus	Axis Status	Displays the current axis status. • 0: Axis disabled (Disabled) • 1: Stopping on error (ErrorStop) • 2: Decelerating to stop (Stopping) • 3: During homing (Homing) • 4: Standby (Standstill)
Encoder_Connected	Connection Status	Displays the virtual encoder axis connection status. • FALSE: Not connected • TRUE: Connected
Encoder_CounterDisable	Counter Disabled	Becomes 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.

■ Axis control data (AxisName.Md.)

Variable/Structure name	Name	Details
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.

Virtual linked axis data type AXIS_VIRTUAL_LINK members (excluding AxisRef)

■ Axis parameter constant (AxisName.PrConst.)

Variable/Structure name	Name	Details
AxisType	Axis Type Setting	Sets the axis type. ☞ Page 56 Axis type
PosRestoration_AbsPosBase	Absolute Position Reference Setting	When using the absolute position system, this variable sets the current position to be the basis of the current position restoration. ☞ Page 57 Absolute position control setting
OperationCycle	Control Cycle Setting	Sets the control operation cycle. ☞ Page 57 Control cycle

■ Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
AxisStatus	Axis Status	Displays the current axis status. • -1: Axis variable uninitialization/Axis parameter error (Invalid) • 0: Axis disabled (Disabled) • 1: Stopping on error (ErrorStop) • 2: Decelerating to stop (Stopping) • 3: During homing (Homing) • 4: Standby (Standstill) • 5: During positioning operation (DiscreteMotion) • 6: During continuous operation (ContinuousMotion) • 7: During synchronous operation (SynchronizedMotion)

Real drive axis

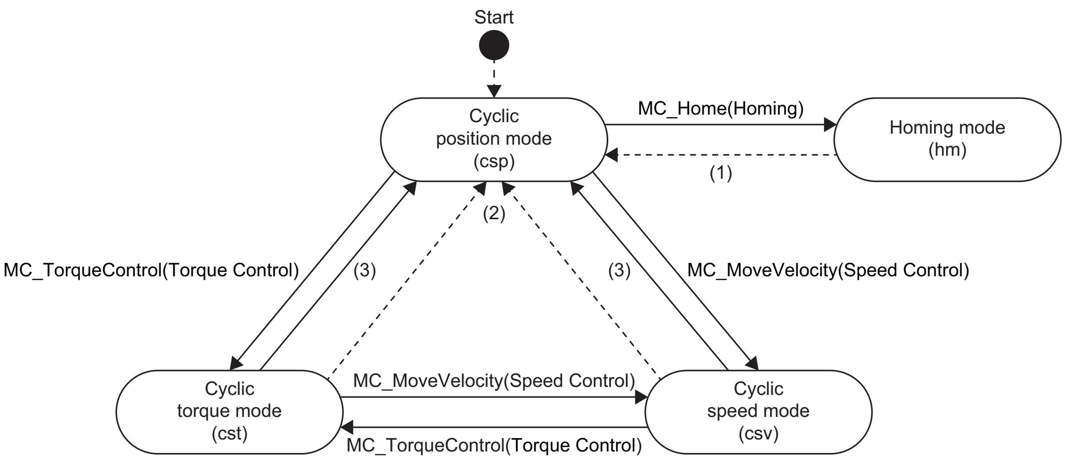
This axis uses a drive unit which is compatible with the CiA402 drive profile connected to CC-Link IE TSN. It is counted as the number of control axes.

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 Driver control mode (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 195 Driver homing method
Cyclic position mode (csp)	8: csp	Executes control following the sequential positioning command from the controller in each communication cycle.	☞ Page 274 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 "Cyclic position mode (csp)" when the driver is connected.
Control mode switching of the driver is executed at the same time when the Motion control FB is executed. The state transition is shown below.



No.	State transition description
(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 Motion control FB other than MC_MoveVelocity (Speed Control)/MC_TorqueControl (Torque Control) 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 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.

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

Point

When using linear encoders and A/B/Z-phase differential output type rotary encoders that are connected to the drive unit, use them for a virtual encoder axis. For details, refer to the following.
☞ Page 48 Virtual encoder axis

Parameter setting items

The following items are required to set up in order to use the real encoder axis.
○: Necessary, △: Optional, —: Unnecessary, ●: Necessary for the same station address real drive axis

Setting item		Real encoder axis type Via drive unit	Reference
Axis No.		○	☞ Page 56 Axis No.
Axis type		○	☞ Page 56 Axis type
Real encoder axis type setting		○	☞ Page 43 Real encoder axis type setting
Station address		○	☞ Page 43 Station address ☞ Page 57 Station address
Absolute position control setting		○	☞ Page 57 Absolute position control setting
Control cycle		○	☞ Page 57 Control cycle
Slave object setting	PosActualValue	—	☞ Page 43 Slave object setting ☞ Page 59 Slave object settings
	Encoder status 2	●	
	Scale ABS counter	●	
	Scale cycle counter	●	
	Scale measurement encoder Resolution	●	
	Scale measurement encoder reception status	●	
Driver unit conversion numerator		○	☞ Page 360 Compensation Function
Driver unit conversion denominator		○	
Position command unit setting		○	☞ Page 74 Technical Units
Velocity command unit setting		○	
Command unit string		○	
Ring counter enabled selection		○	☞ Page 80 Positioning Range
Ring counter upper limit value		△	
Ring counter lower limit value		△	
Counter disabling signal		△	☞ Page 88 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 (Drive)	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 43 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, "Synchronous Encoder Via Drive Unit Setting Incorrect (error code: 1A90H)" occurs.
		Scale ABS counter*2	Specify the object ID storing the current synchronous encoder multiple revolution counter. ■Setting range Handled as -32768 to 32767.
		Scale cycle counter*2	Specify the object ID storing the current synchronous encoder position within one revolution. ■Setting range 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 385 TARGET_REF structure
*2 If all of the slave objects are not assigned, "Synchronous Encoder Via Drive Unit Setting Incorrect (error code: 1A90H)" occurs.

Required functions to use

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

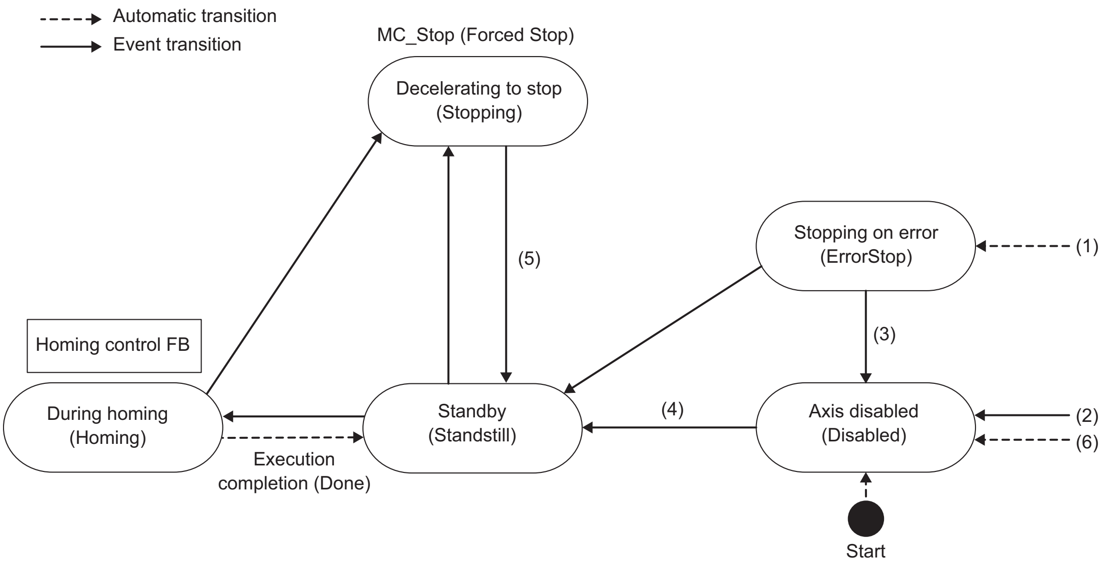
Function		Reference	Axis type specific detailed information
Axis assignment		☞ Page 56 Axis Assignment	—
Technical units		☞ Page 74 Technical Units	—
Compensation function	Driver unit conversion function	☞ Page 360 Compensation Function	—
Absolute position control		☞ Page 93 Absolute Position Control	—
Operation cycle		☞ Page 105 Operation Cycle	—
State transition	Single axis state transition	☞ Page 50 Single axis state transition	—
Follow up (counter enabled/counter disabled)		☞ Page 88 Follow Up	—
Servo ON/OFF		☞ Page 85 Servo ON/OFF	—
Ring counter setting		☞ Page 80 Positioning Range	—
Control change function	Current position change function	☞ Page 298 Current Position Change Function	☞ Page 46 Control change function
Command filter	Smoothing filter	☞ Page 363 Command Filter	—
	Moving direction restriction 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.



No.	State transition description
(1)	Transits from any state when an error of the axis occurs.
(2)	• Transits from any state when Enable (Enable) of MC_Power (Operation Available)/MCv_AllPower (All Axes Operation Available) is FALSE and no error occurs in the axis. • Transits from "4: Standby (Standstill)" when Enable (Enable) of MC_Power (Operation Available)/MCv_AllPower (All Axes Operation Available) is TRUE, Servo ON request (ServoON) is FALSE, and no error occurs in the axis.
(3)	Transits when the status error is cleared by the error reset command during servo OFF. (The state is the current position restoration not executed when the axis status transits to "0: Axis disabled (Disabled)" because the follow up is not executed in the "1: Stopping on error (ErrorStop)" state.)
(4)	After the real drive axis which is the same station address is connected, transits when Enable (Enable) of MC_Power (Operation Available)/ MCv_AllPower (All Axes Operation Available) is TRUE, Servo ON request (ServoON) is TRUE, and Operable (Status) of MC_Power (Operation Available) is TRUE.
(5)	Transits when Execution completion (Done) of MC_Stop (Forced Stop) is TRUE and Execute command (Execute) of MC_Stop (Forced Stop) is FALSE.
(6)	Transits from any state other than the "1: Stopping on error (ErrorStop)" state when the real drive axis which is the same station address is not to be connected.

Point

Follow-up is disabled during Decelerating to stop (Stopping). For details on the set position when follow-up is disabled, refer to the following.

☞ Page 90 For real encoder axis/virtual encoder axis

Control change function

When starting the current position change control unconditionally, use MC_SetPosition (Current Position Change).

Point

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

Setting examples

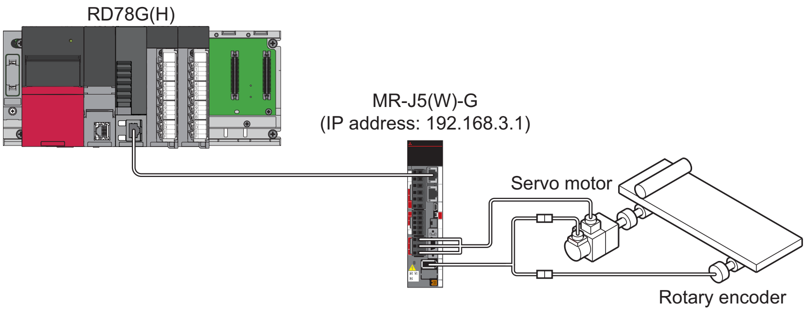
The setting example of the real encoder axis is shown below.

[Via drive unit]

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

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: Real Drive Axis (DriveAxis)	2: Real Encoder Axis (EncoderAxis)
Real encoder axis type setting	Unnecessary	1: Via Drive Unit (Drive)
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: Automatic Setting (Acquire from Connected Device) (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: Automatic Setting (Acquire from Connected Device) (Auto)" is set, acquires the absolute position system setting from the driver side.

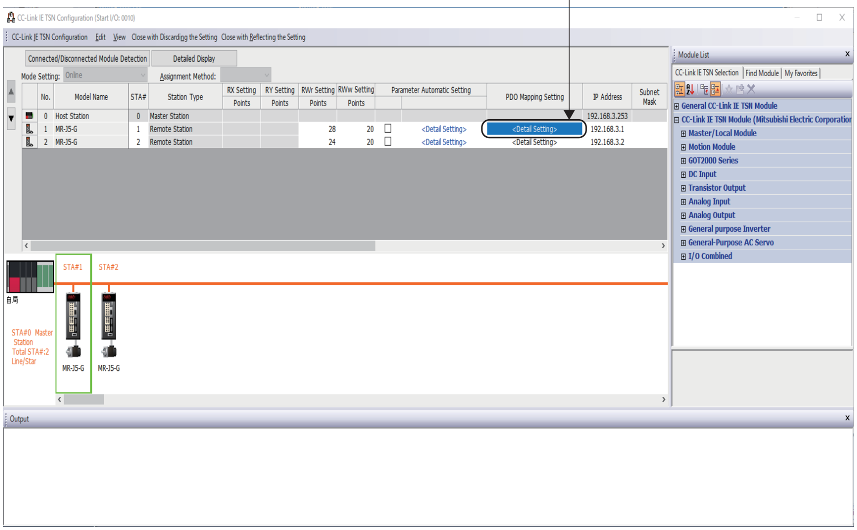
*2 The data refresh cycle of the synchronous encoder is the control cycle of the connected real drive axis and the axis control may not be executed by each data refresh cycle when it differs with the real encoder axis setting. Thus, It is recommended that the real encoder axis control cycle is set as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

■ PDO mapping setting

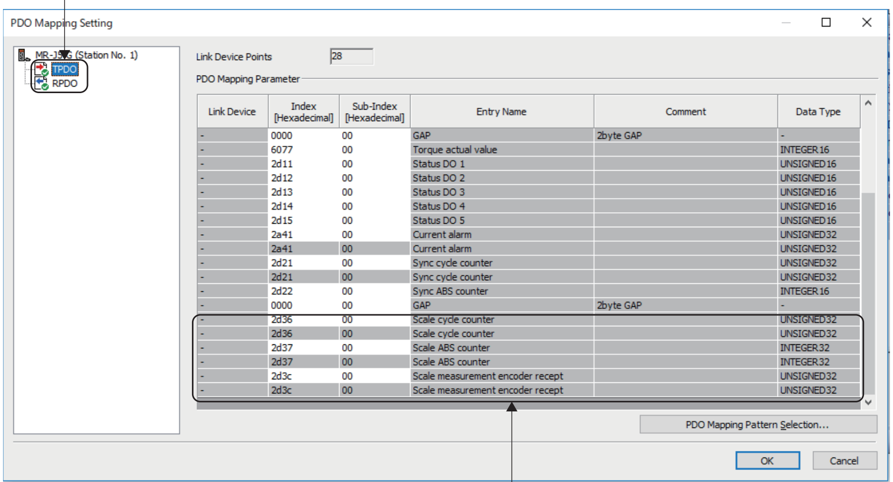
Regarding the "CC-Link IE TSN Configuration" screen under network configuration setting of GX Works3, set all three objects of the following to the PDO mapping (TxPDO) of the real drive axis 1 (MR-J5(W)-G station address 192.168.3.1).

- Scale cycle counter (Index: 2d36, subindex: 00)
- Scale ABS counter (Index: 2d37, subindex: 00)
- Scale measurement encoder reception status (Index: 2d3c, subindex: 00)

Executes PDO mapping setting to the first station to use the scale measurement device of the first station as an input of the real encoder axis 2.



The setting screen of TPDO and RPDO can be switched here.



Add required objects via drive unit.

■ 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 specification.

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

*1 Applicable to MR-J5(W)-G (A5 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 axis emulation function to the real axis drive. For details, refer to the following.

☞ Page 395 Axis 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 48 Slave object setting
Driver unit conversion numerator		○	☞ Page 360 Compensation Function
Driver unit conversion denominator		○	
Position command unit setting		○	☞ Page 74 Technical Units
Velocity command unit setting		○	
Command unit string		○	
Ring counter enabled selection		○	☞ Page 80 Positioning Range
Ring counter upper limit value		△	
Ring counter lower limit value		△	
Encoder ring counter upper limit value		○	☞ Page 49 Encoder ring counter upper limit value/ lower limit value
Encoder ring counter lower limit value		○	
Counter disabling signal		△	☞ Page 88 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, "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, "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, "Out of Parameter Range (Axis) (error code: 1D80H)" occurs. • When [VAR] type is used, do not specify the local label. Use "Encoder Input Value (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 385 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, "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.

$$\text{One operation cycle movement amount} < \frac{|\text{the encoder ring counter upper limit value - the encoder ring counter lower limit value + 1}|}{2}$$

- 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.

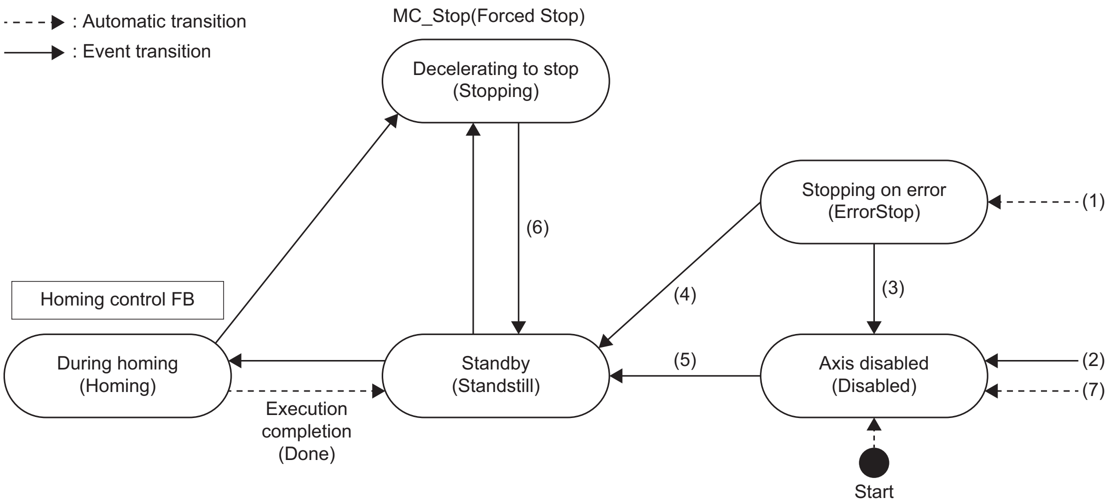
Function		Reference	Axis type specific detailed information
Axis assignment		☞ Page 56 Axis Assignment	—
Technical units		☞ Page 74 Technical Units	—
Compensation function	Driver unit conversion function	☞ Page 360 Compensation Function	—
Absolute position control		☞ Page 93 Absolute Position Control	—
Operation cycle		☞ Page 105 Operation Cycle	—
State transition	Single axis state transition	☞ Page 50 Single axis state transition	☞ Page 45 State transitions
Follow up (counter enabled/counter disabled)		☞ Page 88 Follow Up	—
Ring counter setting		☞ Page 80 Positioning Range	—
Control change function	Current position change function	☞ Page 298 Current Position Change Function	☞ Page 46 Control change function
Command filter	Smoothing filter	☞ Page 363 Command Filter	—
	Moving direction restriction filter		—

For other functions, refer to each function, check the required slave object for the function. 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.



No.	State transition description
(1)	Transits from any state when an error of the axis occurs.
(2)	• Transits from any state when Enable (Enable) of MC_Power (Operation Available)/MCv_AllPower (All Axes Operation Available) is FALSE and no error occurs in the axis. • Transits from "4: Standby (Standstill)" when Enable (Enable) of MC_Power (Operation Available)/MCv_AllPower (All Axes Operation Available) is TRUE, Servo ON request (ServoON) is FALSE, and no error occurs in the axis.
(3)	Transits when the status error is cleared by the error reset command during servo OFF.
(4)	Transits when the status error is cleared by the error reset command during servo ON.
(5)	After Connection Command (AxisName.Cd.Encoder_Connect) is set to TRUE, transits when Enable (Enable) of MC_Power (Operation Available)/ MCv_AllPower (All Axes Operation Available) is TRUE, Servo ON request (ServoON) is TRUE, and Operable (Status) of MC_Power (Operation Available) is TRUE.
(6)	Transits when Execution completion (Done) of MC_Stop (Forced Stop) is TRUE and Execute command (Execute) of MC_Stop (Forced Stop) is FALSE.
(7)	Transits from any state other than the "1: Stopping on error (ErrorStop)" state when Connection Command (AxisName.Cd.Encoder_Connect) is FALSE.

Point

Follow-up is disabled during Decelerating to stop (Stopping). For details on the set position when follow-up is disabled, refer to the following.

☞ Page 90 For real encoder axis/virtual encoder axis

Control change function

When starting the current position change control unconditionally, use the MC_SetPosition (Current Position Change).

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

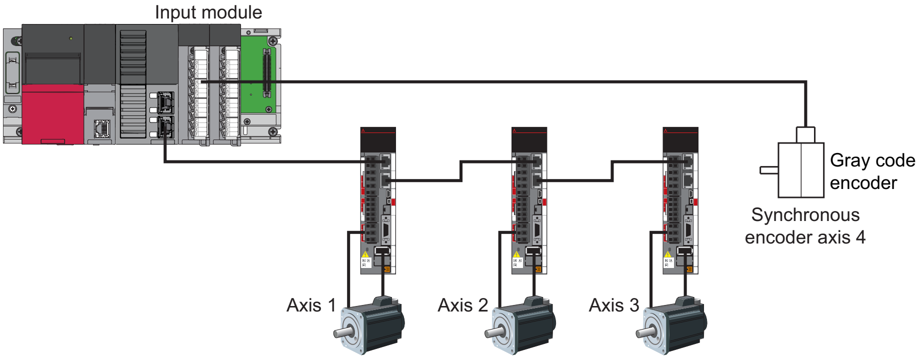
The setting example of the virtual encoder axis and the method of use are shown below.

■ When using the encoder value of the gray code encoder connected to the input module

Ex.

When the axis 4 is used as the virtual encoder axis

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



- Axis setting

Setting items	Setting value
AxisNo.	4
AxisType	4: Virtual Encoder Axis (VirtualEncoderAxis)
Absolute position control setting	0: Disable Absolute Position System (ABSDisabled)
Encoder ring counter upper limit value	4095
Encoder ring counter lower limit value	0
Slave object data_PosActualValue	[DEV](DINT)G11478000

- How to use virtual encoder axis

1. Set Command (AxisName.Cd.Encoder_Connect) to TRUE.
2. Check Current Position Restoration Status (AxisName.Md.PosRestoration_Status) is "2: Restoration Completed in Incremental System (RestoredInIncSystem)", and Connection Status (AxisName.Md.Encoder_Connected) is TRUE.
3. Set Enable (Enable) of MC_Power (Operation Available)/MCv_AllPower (All Axes Operation Available) and Servo ON request (ServoON) to TRUE.
4. Check Axis Status (AxisName.Md.AxisStatus) is "4: Standby (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.)

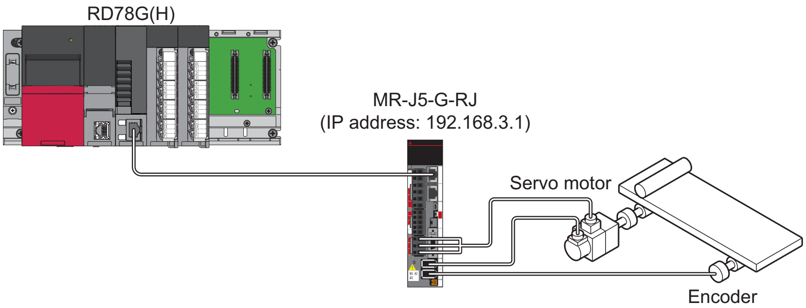
■ When using the encoder value of an encoder connected with the driver

Ex.

When the axis 4 is used as the virtual encoder axis

(The encoder value of A/B/Z-phase differential output type rotary encoder (incremental type) (resolution: 5000 [pulse/rev]) is used to control)

The axis to which the encoder is connected (axis 1) is used as the real drive axis.



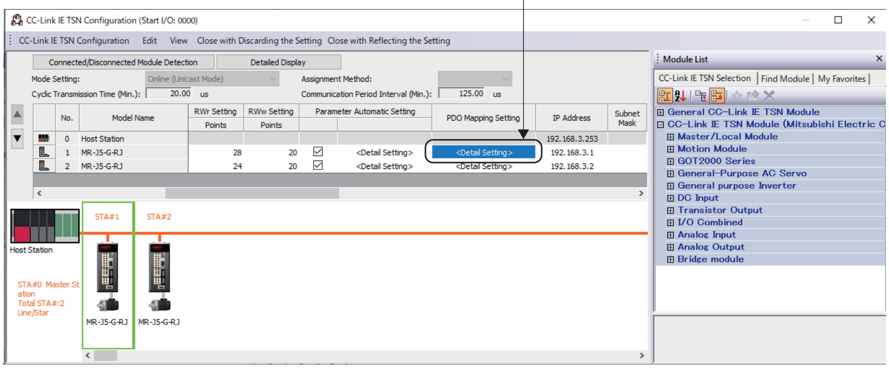
- PDO mapping setting

  In the "CC-Link IE TSN Configuration" screen under Network Configuration Settings in GX Works3, set the following objects to the PDO mapping (TxPDO) for the real drive axis 1 (MR-J5(W)-G station address "192.168.3.1").

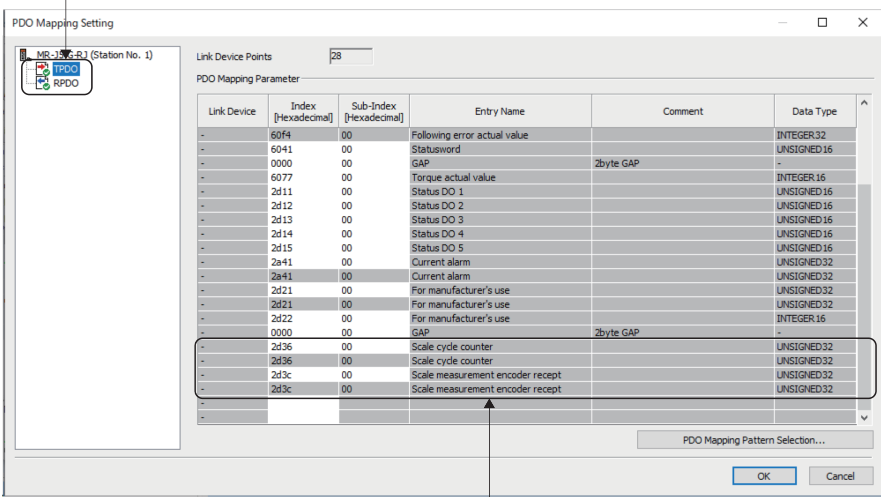
  · Scale cycle counter (Index: 2d36, Sub-index: 00)

  · Scale measurement encoder reception status (Index: 2d3c, Sub-index: 00)

Set the PDO mapping setting for the first station to use the scale measurement device of the first station as an input of the virtual encoder axis 4.



Switch between the setting screen of TPDO and RPDO here. Select TPDO.



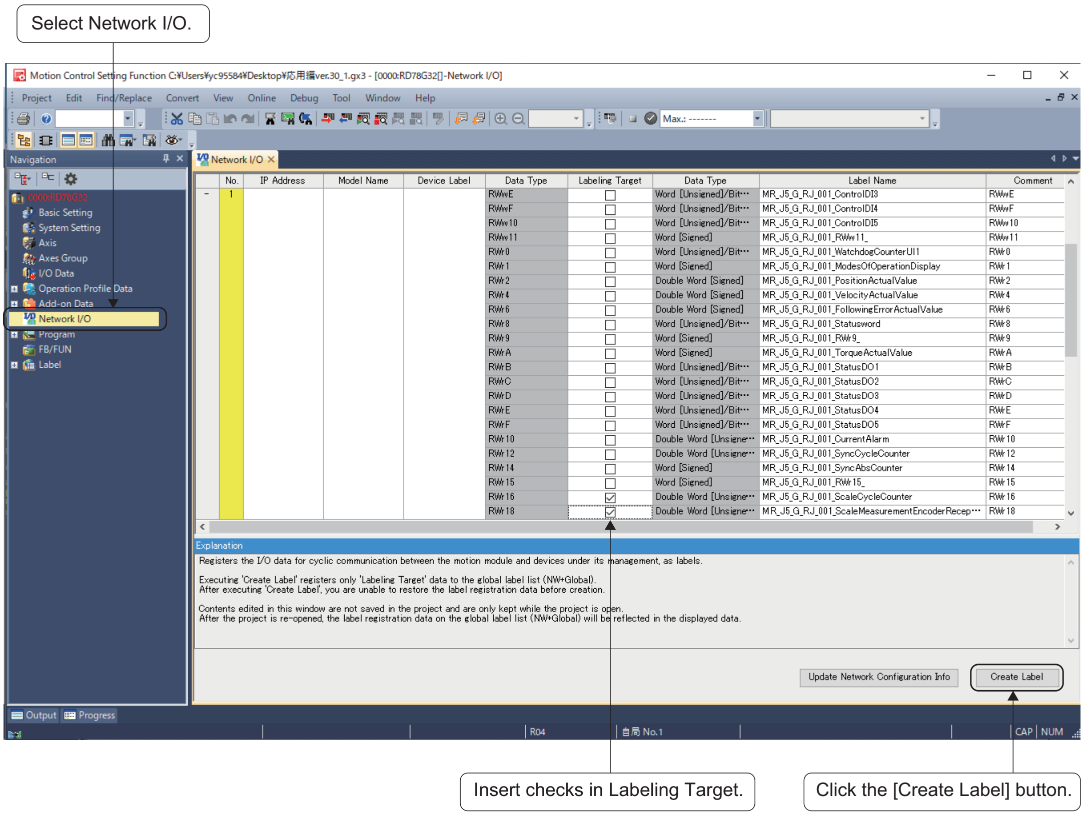
Add the required objects via drive unit.

- Network I/O setting

In the "Network I/O" screen under Motion Control Setting Function in GX Works3, set the following slave labels.

· Scale cycle counter (Index: 2d36, Sub-index: 00)

· Scale measurement encoder reception status (Index: 2d3c, Sub-index: 00)



- Driver side (MR-J5(W)-G) setting

To enable the scale measurement function on MR-J5(W)-G, set the parameters below.

For details on the parameters and connection method of the encoder, refer to the driver device specification.

No.	Symbol	Name	Overview
PA22.3*1	**PCS	Scale measurement function selection	Select scale measurement function. • 0: Disabled (initial setting) • 1: Use with absolute position detection system • 2: Use with incremental system
PE51*2	**EDV2	Load-side encoder resolution setting	Set "0". When using A/B/Z-phase differential output type rotary encoders, if a value other than "0" is set, Scale cycle counter may change rapidly.

*1 Supported by MR-J5(W)-G (version A5) or later

*2 Supported by MR-J5(W)-G (version B2) or later

Point

When using A/B/Z-phase differential output type rotary encoders, if the "Load-side encoder resolution setting (PE51)" servo parameter is set to a value other than "0", the following operations will occur.

- Scale cycle counter changes rapidly when the Z-phase is passed for the first time.
- If an incorrect encoder resolution is set to "Load-side encoder resolution setting (PE51)", the value of Scale cycle counter is invalid.

- Axis setting

Setting items	Setting value	
	Real drive axis	Virtual encoder axis
Axis No.	1	4
Axis type	0: Real Drive Axis (DriveAxis)	4: Virtual Encoder Axis (VirtualEncoderAxis)
Station address	192.168.3.1	Unnecessary
Encoder ring counter upper limit value*1	Unnecessary	0
Encoder ring counter lower limit value*1	Unnecessary	0
Absolute position control setting	-1: Automatic Setting (Acquire from Connected Device) (Auto)	0: Disable Absolute Position System (ABSDisabled)
Control cycle	0	0
Slave object data_PosActualValue	Unnecessary	[VAR](DWORD)MR_J5_G_RJ_001_ScaleCycleCounter

*1 Set the value within the range of Scale cycle counter.
When the range of Scale cycle counter is "0 to 4294967295", the axis can be used by setting "Encoder ring counter upper limit value = Encoder ring counter lower limit value".
<Example>
Encoder ring counter upper limit value and Encoder ring counter lower limit value are set to "0" so that the range of Scale cycle counter becomes "0 to 4294967295".

- How to use virtual encoder axis

1. Set Enable (Enable) of MC_Power (Operation Available)/MCv_AllPower (All Axes Operation Available) and Servo ON request (ServoON) to TRUE.
2. Check that Driver Status (AxisName.Md.Driver_State) of the real drive axis (Axis 1) with a scale measurement encoder connected is "6: Operation Enable".
3. Set Connection Command (AxisName.Cd.Encoder_Connect) of the virtual encoder axis (Axis 4) to TRUE.
4. Check that Current Position Restoration Status (AxisName.Md.PosRestoration_Status) of the virtual encoder axis (Axis 4) is "2: Restoration Completed in Incremental System (RestoredInIncSystem)", and Connection Status (AxisName.Md.Encoder_Connected) is TRUE.
5. Check that Axis Status (AxisName.Md.AxisStatus) of the virtual encoder axis (Axis 4) is "4: Standby (Standstill)". (The virtual encoder axis will be controlled according to the encoder value of the scale measurement encoder.)

Point

The virtual encoder axis will continue to be controlled even when an alarm occurs on the scale measurement encoder.
To stop controlling the virtual encoder axis, set Connection Command (AxisName.Cd.Encoder_Connect) of the virtual encoder axis (Axis 4) to FALSE when the value of MR_J5_G_RJ_001_ScaleMeasurementEncoderReceptionStatus becomes "other than 0 (an encoder alarm occurs)".

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 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 is turned ON to OFF.)

How to generate an axis

Set it according to axis assignment. For the axis assignment 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 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.

Operation of this function for each system status

○: Possible, ×: Not possible

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

Setting items

The items set to the axis type are shown below.

○: Necessary, —: Unnecessary

Axis type	Setting items					
	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, "Axis No. Setting Duplication Error (error code: 1A91H)" will occur.

Axis setting	Axis No.
User specification	1 to 10000
Virtual 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 32 Data type

Station address

- Specify the station address (IP address) of the target device to connect a network device as a real axis. Also, specify the axis ID (multidrop No.) for connecting a device which can control multiple axes with one device (multiple axes device).

Ex.

When specifying the C-axis of a multiple axes device (MR-J5W-G)

192.168.3.1#2
Multi-drop No.*1
IP address

*1 The multidrop No. is specified as "# + No. (decimal format)". When omitted, it is regarded as "#0".
· #0: A-axis
· #1: B-axis
· #2: C-axis

- When a station address of a device that cannot be a real axis is specified, "SLMP Communication Error (error code: 1C43H)" or "Station Address Setting Incorrect (error code: 1A95H)" occurs. (Example: When a station address of an I/O unit is specified)
- 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 385 TARGET_REF structure
- Even though the station address have not been set ("") in the real axis, by setting Axis Emulation Enabled (AxisName.PrConst.SlaveEmulate_Enable) to TRUE, it enables to use as the axis emulate function. If axis emulate function is not used, "Station Address Setting Incorrect (error code: 1A95H)" occurs 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 93 Absolute Position Control

Control cycle

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

When the specified operation cycle does not support axis control, "Cycle Assignment Incorrect (Axis) (error code: 1AF9H)" occurs.

☞ Page 105 Operation Cycle

The following shows the communication cycles that can be specified as the control cycle of an axis.

Add-on Axis version	Communication cycle
Earlier than "1.31"	31.25 μs, 62.5 μs, 125 μs, 250 μs, 500 μs, 1.0 ms, 2.0 ms, 4.0 ms, 8.0 ms
"1.31" or later	31.25 μs, 62.5 μs, 125 μs, 250 μs, 500 μs, 1.0 ms, 1.5 ms, 2.0 ms, 2.5 ms, 3.0 ms, 3.5 ms, 4.0 ms, 4.5 ms, 5.0 ms, 5.5 ms, 6.0 ms, 6.5 ms, 7.0 ms, 7.5 ms, 8.0 ms

When setting the communication cycle to a value other than "31.25×2^n [μs]", set it in "Communication Period Interval Setting (Set it in units of 1 μs)" under "Basic Settings" of Module Parameter (Network).

For the setting method, refer to "Communication Period Setting" of the following manual.

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

<When using MR-J5(W)-G series>

For details on the models and versions of MR-J5(W)-G series that support communication cycles other than "31.25×2^n [μs]", refer to the manual of MR-J5(W)-G series.

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

Axis information (AxisName.AxisRef.)

Variable/Structure name	Name	Details
AxisNo	Axis No.	Sets the axis No. • 0: Not set • 1 to 10000: Setting axis No.

Axis parameter constant (AxisName.PrConst.)

Variable/Structure name	Name	Details
AxisType	Axis Type Setting	Sets the axis type. • 0: Real Drive Axis (DriveAxis) • 2: Real Encoder Axis (EncoderAxis) • 3: Virtual Drive Axis (VirtualDriveAxis) • 4: Virtual Encoder Axis (VirtualEncoderAxis) • 5: Virtual Linked Axis (VirtualLinkAxis)
AddressOfStation	Station Address Setting	Sets network address of the device station with a character string.
PosRestoration_AbsPosEnable	Absolute Position Control Setting	Sets the absolute position control setting. • 0: Disable Absolute Position System (ABSDisabled) • 1: Enable Absolute Position System (Enabled) • -1: Automatic Setting (Acquire from Connected Device) (Auto)
OperationCycle	Control Cycle Setting	Sets Control cycle setting. For details, refer to the following. ☞ Page 105 Operation Cycle
SlaveObject	Slave Object Data	Sets the object of the device station. *: Availability depends on the axis type. For details, refer to the following. ☞ Page 37 Axis Type

Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
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.
Drive_RPDO[1..64]	RPDO (Motion System → Device Station) Mapping	Indicates the object index specified to the nth place in the RDPO/TPDO mapping settings of the network parameter. H□□□□□□□□ → Object size (bits) → Object subindex → Object index • The object size displays 1 byte, 2 bytes, 4 bytes, and 8 bytes in bits. • When the size of the nth object is 1 byte, a padding object is inserted in the nth+1 place.
Drive_TPDO[1..64]	TPDO (Device Station → Motion System) Mapping	

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 device station by setting the data related to the device station 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. Usable data differs depending on the axis type. For details, refer to the following. ☞ Page 37 Axis Type For details on the format to specify the data, refer to the following. ☞ Page 385 TARGET_REF structure *: For mapping availability to cyclic data when using [OBJ], refer to the following. ☞ Page 61 Slave object list *: For target modification availability, refer to the following. · Real encoder axis (☞ Page 43 Slave object setting) · Virtual encoder axis (☞ Page 48 Slave object setting)

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 61 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, "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, "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 61 Slave object list
- When "no setting" is set to the slave object setting, functions that need the applicable slave object cannot be used. For details, refer to the following.
  ☞ Page 64 Motion system function and 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: Device → 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 → Device	(Setting location data) < (Slave object)	Stores a slave object value as it is to 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 device 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, "Necessary Slave Object Unset (error code: 1AA8H)" will occur.)

Mapping	Necessary slave object name
Output (RPDO)	Control 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 device station 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 Maximum RAM Size (System.PrConst.Addon_Axis.RamSizeMax).
When the axis data memory usage exceeds the capacity at initializing the axis variable, "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*1
- ServoDriver_CANopen*1
- SignalIO*2
- ExternalSignal*2

*1 For using real drive axis
*2 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 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 device station to the master station, the motion system can read the data from the device station in fixed-cycle. • RxPDO: By specifying the link device outputted from the master station to the device station, the motion system can write the data from the device stations in fixed-cycle.
Default	Indicates the default value of the slave object for the real drive axis.

The slave object lists are shown below.

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	Data Type	Access	PDO mapping	Default	Description	Slave object name				
						Real drive axis	Real Encoder axis[*1]	Virtual drive axis	Virtual Encoder axis	Virtual linked axis
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		○	×	×	×	×
Control DI 3	U16	rw	RxPDO	2D030010		○	×	×	×	×
Control DI 4	U16	rw	RxPDO	2D040010		○	×	×	×	×
Control DI 5	U16	rw	RxPDO	2D050010		○	×	×	×	×
Control DI 6	U16	rw	No	2D060010		○	×	×	×	×
Control DI 7	U16	rw	No	2D070010		○	×	×	×	×
Statusword	U16	ro	TxPDO	60410010	Gets the status of the driver device.	○	×	×	×	×

Slave object name	Data Type	Access	PDO mapping	Default	Description	Slave object name				
						Real drive axis	Real Encoder axis[*1]	Virtual drive axis	Virtual Encoder axis	Virtual linked axis
Status DO 1	U16	ro	TxPDO	2D110010	Sets the output device in driver device.	○	×	×	×	×
Status DO 2	U16	ro	TxPDO	2D120010		○	×	×	×	×
Status DO 3	U16	ro	TxPDO	2D130010		○	×	×	×	×
Status DO 4	U16	ro	TxPDO	2D140010		○	×	×	×	×
Status DO 5	U16	ro	TxPDO	2D150010		○	×	×	×	×
Status DO 6	U16	ro	No	2D160010		○	×	×	×	×
Status DO 7	U16	ro	No	2D170010		○	×	×	×	×
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 driver device.	○	×	×	×	×
Torque actual value	I16	ro	TxPDO	60770010	The current torque of the driver device.	○	×	×	×	×
Polarity	U8	rw	No	607E0008	Sets the rotation direction selection of driver device	○	×	×	×	×
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		○	×	×	×	×
Supported Control DI 3	U16	ro	No	2D000310		○	×	×	×	×
Supported Control DI 4	U16	ro	No	2D000410		○	×	×	×	×
Supported Control DI 5	U16	ro	No	2D000510		○	×	×	×	×
Supported Control DI 6	U16	ro	No	2D000610		○	×	×	×	×
Supported Control DI 7	U16	ro	No	2D000710		○	×	×	×	×

Slave object name	Data Type	Access	PDO mapping	Default	Description	Slave object name				
						Real drive axis	Real Encoder axis*1	Virtual drive axis	Virtual Encoder axis	Virtual linked axis
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		○	×	×	×	×
Supported Status DO 3	U16	ro	No	2D100310		○	×	×	×	×
Supported Status DO 4	U16	ro	No	2D100410		○	×	×	×	×
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		○	×	×	×	×
Supported Status DO 7	U16	ro	No	2D100710		○	×	×	×	×
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 velocity 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.	○	×	×	×	×
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.	○	×	×	×	×

Slave object name	Data Type	Access	PDO mapping	Default	Description	Slave object name				
						Real drive axis	Real Encoder axis*1	Virtual drive axis	Virtual Encoder axis	Virtual linked axis
Scale ABS counter	I16	ro	No	2D370010	Gets the multiple revolution counter of the scale measurement encoder from the driver device.	○	×	×	×	×

*1 For the slave object settings of the real encoder axis, refer to the following.
☞ Page 42 Real encoder axis

Motion system function and slave object

When no slave object is registered to the axis the errors and functions with restrictions are shown in the following table.

◎: 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 (Speed Control)	TargetVelocity	◎	"Necessary Slave Object Unset (error code: 1AA8H)" occurs.
MC_TorqueControl (Torque Control)	TargetTorque	◎	
Torque limit function	PositiveTorqueLimitValue NegativeTorqueLimitValue	○	The command is ignored.
MC_Home (Homing)	HomeOffset	—	Operates with data set method.

Precautions

Relevant add-ons

The following add-on is required to use this function.

- Axis

System memory capacity

For details, refer to the following.

☞ Page 60 Memory usage

1.4 Axes Group

Axes group is used for multiple axes control such as linear interpolation control and circular interpolation control, etc.
The maximum setting number of axes groups depends on the memory. If an axes group cannot be set for memory shortage, "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.

Operation of this function for each system status

○: Possible, △: Possible (restricted), ×: Not possible

System status	Operation availability
STOP	△ (Only monitoring is available. When the axes group status is Standby, the axes group is disabled.)
RUN	○
Moderate error	△ (Operation may not be possible according to the error status.)*1
Major error	×

*1 When a required add-on is stopped due to "Add-on Library Load Error (error code: 3205H)" or "Insufficient Add-on System Memory (RAM) (error code: 3209H)", operation will not be possible.

Relevant variables

Axes group information (AxesGroupName.AxesGroupRef.)

Variable/Structure name	Name	Details
GroupNo	Axes Group No.	Sets axes group No. in the motion system. • 0: Not set • 1 and later: Setting axes group No.

Axes group parameter (AxesGroupName.Pr.)

Variable/Structure name	Name	Details
Axis[1..16]	Configuration Axis	Sets Axis No. (AxisNo) of Axis Information (AxisName.AxisRef) which configures the axes group.

Axes group monitor data (AxesGroupName.Md.)

Variable/Structure name	Name	Details
GroupStatus	Axes Group Status	Displays the current axes group status. • -1: Axes group variable uninitialization/Axes group parameter error (Invalid) • 0: Axes group disabled (GroupDisabled) • 1: Stopping on error (GroupErrorStop) • 2: Decelerating to stop (GroupStopping) • 4: Standby (GroupStandby) • 5: Operating (GroupMoving)
GroupName	Axes Group Name	Stores the axes group name.
Axis[1..16]	Configuration Axis	Sets Axis No. (AxisNo) of Axis Information (AxisName.AxisRef) which configures the axes group.

Relevant FBs

For details on Motion control FBs, refer to the following.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

Motion control FB	Name	Description
MC_GroupEnable	Axes Group Enabled	Transits the specified axes group status from "0: Axes group disabled (GroupDisabled)" to "4: Standby (GroupStandby)".
MC_GroupDisable	Axes Group Disabled	Transits the specified axes group status to "0: Axes group disabled (GroupDisabled)".

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 72 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 Axis No. (AxisNo) of Axis Information (AxisName.AxisRef) 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 72 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. Re-importing to the control data is executed after axes group variables initialization. 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 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 it 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 does not turn ON. For the label initialization processing when the PLC READY turns OFF to ON, refer to "Label initialization function" in the following manual. 📖MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

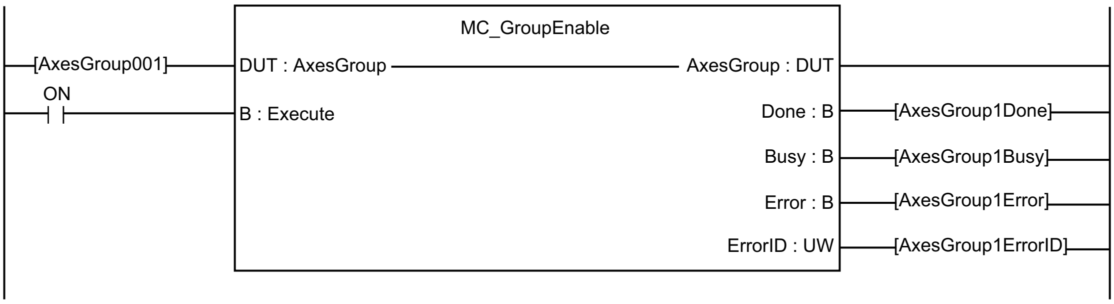
How to specify in a user program

When specifying the axes group by the Motion control FB, set the AXES_GROUP_REF type member (AxesGroupName.AxesGroupRef) of each axes group variable to the I/O variables which the data type is "AXES_GROUP_REF".

Ex.

When executing the MC_GroupEnable (Axes Group Enabled) with the following settings

- Axes group No.: 1
- Axes group name: AxesGroup001



Axes group status

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

Axes group state transition

The following shows status that an axes group can enter.

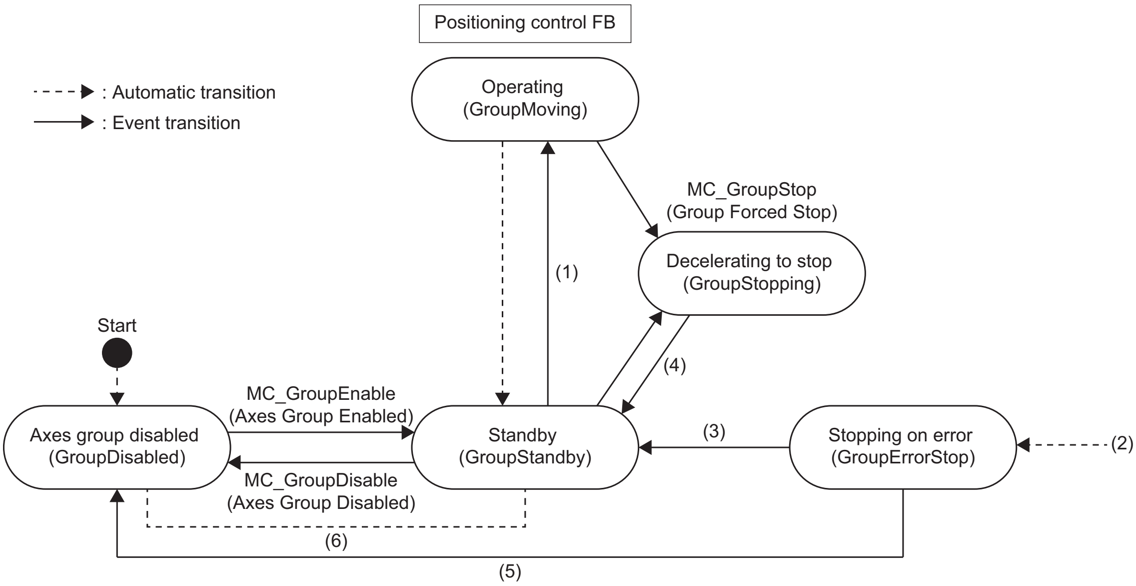
The current status can be confirmed with Axes Group Status (AxesGroupName.Md.GroupStatus).

When the axes group changes to "4: Standby (GroupStandby)", Using Axes Group (AxisName.Md.UseInGroup) of the configuration axis becomes TRUE.

When the motion FB is executed for the axis (the axis of which Using Axes Group (AxisName.Md.UseInGroup) is TRUE) used in the axes group, "Start Not Possible (error code: 1AADH)" occurs. Disable the axes group by MC_GroupDisable (Axes Group Disabled).

Status	Description
-1: Axes group variable uninitialization/Axes group parameter error (Invalid)	Axes group is disabled because of the axes group variables are not initialized (such as an parameter error occurrence at initialization).
0: Axes group disabled (GroupDisabled)	Axes group is disabled. Motion control FB cannot be executed in this status. The axis which is being controlled will be stopped immediately to transit this status.
1: Stopping on error (GroupErrorStop)	Axes group is in the deceleration stop or the stop status by an error occurrence. When Axis Status (AxisName.Md.AxisStatus) is set to "1: Stopping on error (ErrorStop)" in any of the configuration axes, the axes group also enters this status.
2: Decelerating to stop (GroupStopping)	The deceleration stop is executed by MC_GroupStop (Group Forced Stop) . After the axis stop is completed, this status will be maintained until the Execute command (Execute) of MC_GroupStop (Group Forced Stop) becomes FALSE. The Motion control FB cannot be executed in this status.
4: Standby (GroupStandby)	Axes group is enabled and stand-by. (Not depending on the servo ON/OFF status of the configuration axis.)
5: Operating (GroupMoving)	The positioning control FB is executed in an axes group. When transiting into this status, Axis Status (AxisName.Md.AxisStatus) of all configuration axes turn to "7: During synchronous operation (SynchronizedMotion)".

The status is transited as the following figure according to start of the Motion control FB.



No.	State transition description
(1)	All group motion FB
(2)	• Transits when an error caused by an axes group occurs in the status other than "0: Axes group disabled (GroupDisabled)". • Transits when an error caused by an configuration axes occurs in the status other than "0: Axes group disabled (GroupDisabled)".
(3)	Transits when status error is cleared by axes group error reset. For details of axes group error reset, refer to the following. ☞ Page 775 Axes group error reset
(4)	Transits when Execution completion (Done) of MC_GroupStop (Group Forced Stop) is TRUE and Execute command (Execute) MC_GroupStop (Forced Stop) is FALSE.
(5)	Transits when the axes group is disabled at executing MC_GroupDisable (Axes Group Disabled) or clearing the status error by axes group error reset.
(6)	Transits when a stop of program execution is completed by RUN status → STOP status.

Axes group enabled

This section describes enabling an axes group.

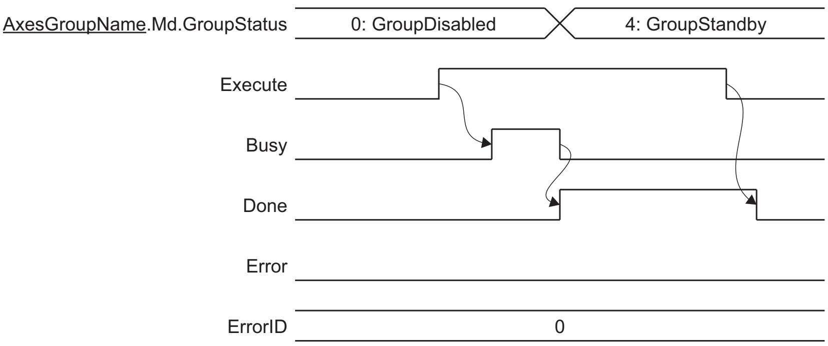
Control details

Specify Axes Group Information (AxesGroup) and change Axes Group Status (AxesGroupName.Md.GroupStatus) to "4: Standby (GroupStandby)".

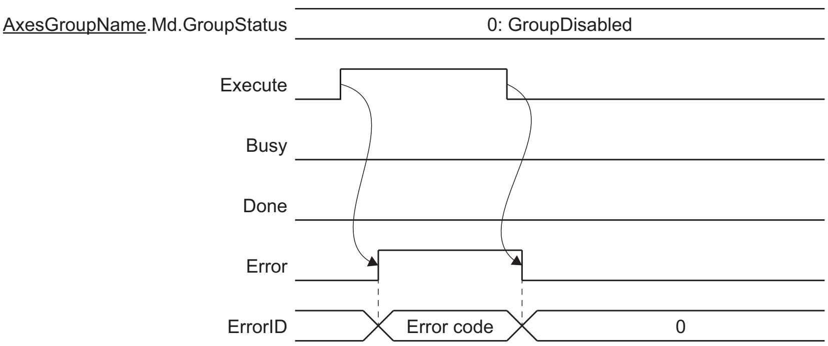
- The axes group can start the motion FB only when Axes Group Status (AxesGroupName.Md.GroupStatus) is "4: Standby (GroupStandby)".
- MC_GroupEnable (Axes Group Enabled) can be executed only when Axis Status (AxisName.Md.AxisStatus) of all configuration axes is "4: Standby (Standstill)" or "0: Axis disabled (Disabled)".
- When Axes Group Status (AxesGroupName.Md.GroupStatus) changes to the "4: Standby (GroupStandby)", Using Axes Group (AxisName.Md.UseInGroup) of the configuration axis becomes TRUE.
- When MC_GroupEnable (Axes Group Enabled) is executed by specifying another axes group including the configuration axis of which Using Axes Group (AxisName.Md.UseInGroup) is TRUE, "Axes Group Configuration Axis Is in Use (error code: 3496H)" occurs.
- If the error occurs in any of the configuration axis, Axes Group Status (AxesGroupName.Md.GroupStatus) will be "1: Stopping on error (GroupErrorStop)".
- When MC_GroupEnable (Axes Group Enabled) is executed for an axes group that has already been enabled, Execution completion (Done) becomes TRUE and the execution ends.

■ Timing chart

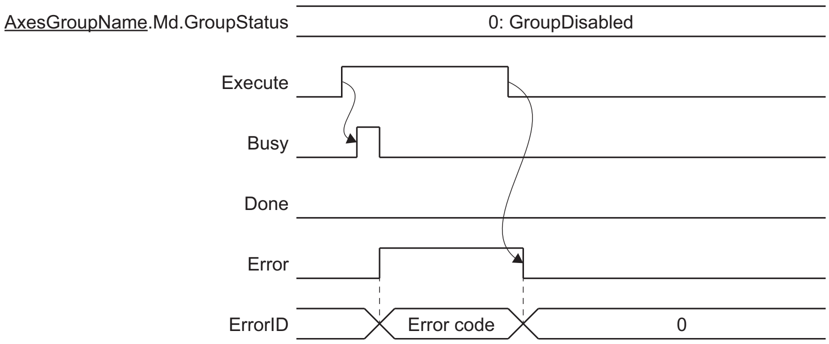
- When the operation is normally completed



- When there is an I/O variable error



• When an error occurs



Axes group disabled

This section describes disabling an axes group.

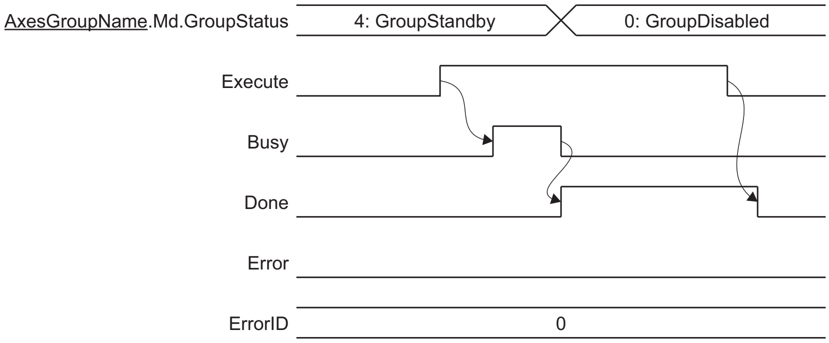
Control details

Axes Group Status (AxesGroupName.Md.GroupStatus) can be changed to "0: Axes group disabled (GroupDisabled)" by specifying Axes Group Information (AxesGroup).

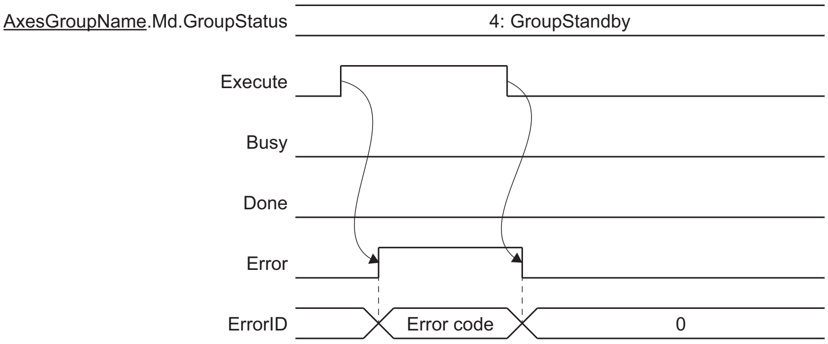
- An axes group cannot execute the motion FB when Axes Group Status (AxesGroupName.Md.GroupStatus) is "0: Axes group disabled (GroupDisabled)".
- MC_GroupDisable (Axes Group Disabled) can only be executed when Axes Group Status (AxesGroupName.Md.GroupStatus) is "4: Standby (GroupStandby)" or "1: Stopping on error (GroupErrorStop)". When executing this FB while Axes Group Status is "1: Stopping on error (GroupErrorStop)" and the axes group is in operation, the axes group status transits to "0: Axes group disabled (GroupDisabled)" after the stop completes. When this FB is executed while Axes Group Status (AxesGroupName.Md.GroupStatus) is other than "4: Standby (GroupStandby)" or "1: Stopping on error (GroupErrorStop)", "Axes Group Status Incorrect (When Axes Group is Disabled) (error code: 1A97H)" occurs. The positioning control FB stops. For details on stop processing, refer to the following.
  ☞ Page 161 Stop
- When Axes Group Status (AxesGroupName.Md.GroupStatus) changes to "0: Axes group disabled (GroupDisabled)", Using axes group (AxisName.Md.UseInGroup) of the configuration axis becomes FALSE.
- When MC_GroupDisable (Axes Group Disabled) is executed for an axes group that has already been disabled, Execution completion (Done) becomes TRUE and the execution ends.

■ Timing chart

• When the operation is normally completed



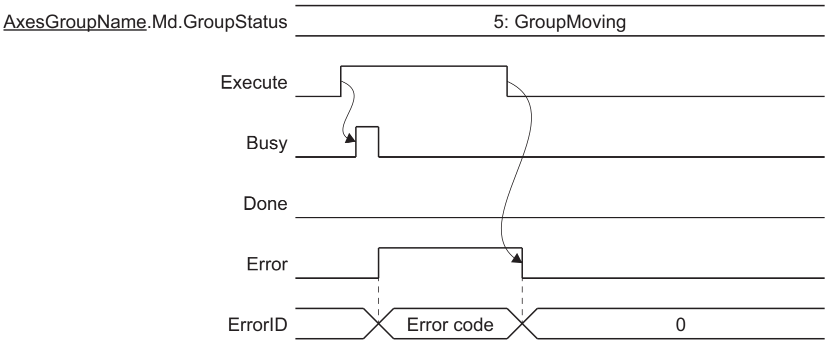
- When there is an I/O variable error



- When an error occurs

Ex.

When executing Axes Group Disable while Axes Group Status (AxesGroupName.Md.GroupStatus) is "5: Operating (GroupMoving)"



Precautions

If the parameter error occurs at axes group variable initialization, "Out of Parameter Range (Axes Group) (error code: 1D81H)" occurs. In this case, Axes Group Status (AxesGroupName.Md.GroupStatus) changes to "-1: Axis variable uninitialization/Axis parameter error (Invalid)". Refreshing of the monitor data, etc. is not executed for axes groups whose status is " -1: Axis variable uninitialization/Axis parameter error (Invalid)". If the axes group is specified by the user program, "Out of Axes Group No. Range (error code: 3402H)" occurs.

Relevant add-ons

The following add-on is required to use this function.

- Axis
- MotionEngine
- MotionControl_General

1.5 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 turns ON.

Relevant variables

Axes group monitor data (AxesGroupName.AxesGroupRef.)

Variable/Structure name	Name	Details
GroupNo	Axes Group No.	Sets the axes group No. • 0: No setting • 1 to 10000: Setting axes group No.

Axes group monitor data (AxesGroupName.Pr.)

Variable/Structure name	Name	Details
Axis[1..16]	Configuration Axis	Sets Axis No. (AxisNo) of Axis Information (AxisName.AxisRef) which configures the axes group.

Axes group monitor data (AxesGroupName.Md.)

Variable/Structure name	Name	Details
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 Axis No. (AxisNo) of Axis Information (AxisName.AxisRef) 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, "Axes Group No. Setting Duplication Error" (error code: 1A58H) will occur.

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

Item	Setting range
Axes group No.	1 to 10000

■ 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 Configuration Axis (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.

"No axis (error code: 1A99H)" and "Axes Group Configuration Axis Operation Cycle Incorrect (error code: 1A59H)" occur 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 Maximum RAM Size (System.PrConst.Addon_Axis.RamSizeMax).

When the axis data memory usage exceeds the capacity at initializing the axes group variable, "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

Axis parameter (AxisName.Pr.)

Variable/Structure name	Name	Details
Unit_Position	Position Command Unit	Sets the position command unit to be used in the motion control. ☞ Page 75 Position Command Unit (AxisName(AxesGroupName).Pr.Unit_Position)
Unit_Velocity	Velocity Command Unit	Sets the velocity command unit to be used in the motion control. ☞ Page 76 Velocity Command Unit (AxisName(AxesGroupName).Pr.Unit_Velocity)
Unit_PositionString	Position Command Unit 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).

Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
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.

Axes group parameter (AxesGroupName.Pr.)

Variable/Structure name	Name	Details
Unit_Position	Position Command Unit	Sets the position command unit to be used in the motion control. ☞ Page 75 Position Command Unit (AxisName(AxesGroupName).Pr.Unit_Position)
Unit_Velocity	Velocity Command Unit	Sets the velocity command unit to be used in the motion control. ☞ Page 76 Velocity Command Unit (AxisName(AxesGroupName).Pr.Unit_Velocity)
Unit_PositionString	Position Command Unit 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).

Axes group monitor data (AxesGroupName.Md.)

Variable/Structure name	Name	Details
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 setting method is shown below.

Position Command Unit (AxisName(AxesGroupName).Pr.Unit_Position)

H□□□□□□□□

Position command unit

Prefix

■ 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 Position Command Unit String (AxisName(AxesGroupName).Pr.Unit_PositionString). (This can also be set as blank ("").)

■ 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 ($\times 10^0$).

Setting value	Prefix
H00	$\times 10^0$
HFD	$\times 10^{-3}$ [m]
HFA	$\times 10^{-6}$ [μ]
HF7	$\times 10^{-9}$ [n]

■ Display example

The following shows display examples of Position Command Unit Display (AxisName(AxesGroupName).Md.Unit_PositionDisplay) according to the setting value of Position Command Unit (AxisName(AxesGroupName).Pr.Unit_Position).

Ex.

The Setting value of Position Command Unit (AxisName(AxesGroupName).Pr.Unit_Position) and Position Command Unit Display (AxisName(AxesGroupName).Md.Unit_PositionDisplay).

Position Command Unit (AxisName(AxesGroupName).Pr.Unit_Position)	Position Command Unit Display (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 Position Command Unit (AxisName(AxesGroupName).Pr.Unit_Position) is "HFD010000", the position will be regarded as "1.0 mm".

Velocity Command Unit (AxisName(AxesGroupName).Pr.Unit_Velocity)

H□□□□□□□□

Velocity command unit

Prefix

■ Setting value of the velocity command unit

Setting value	Velocity command unit
H03	s
H47	min

■ Setting value of prefix

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

■ Display example

The following shows display examples of Velocity Command Unit Display (AxisName(AxesGroupName).Md.Unit_VelocityDisplay) according to the setting value of Velocity Command Unit (AxisName(AxesGroupName).Pr.Unit_Velocity).

Ex.

The Setting value of Velocity Command Unit (AxisName(AxesGroupName).Pr.Unit_Velocity) and Velocity Command Unit Display (AxisName(AxesGroupName).Md.Unit_VelocityDisplay).

Velocity Command Unit (AxisName(AxesGroupName).Pr.Unit_Velocity)	Velocity Command Unit Display (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 Position Command Unit (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 Velocity Command Unit Display (AxisName(AxesGroupName).Md.Unit_VelocityDisplay).
- If "1" is set in the program when Velocity Command Unit (AxisName(AxesGroupName).Pr.Unit_Velocity) is HFD000300, the velocity will be regarded as "1.0 [position command unit]/ms".

- The following shows display examples of Velocity Command Unit Display (AxisName(AxesGroupName).Md.Unit_VelocityDisplay) according to the setting value of Position Command Unit (AxisName(AxesGroupName).Pr.Unit_Position) and Velocity Command Unit (AxisName(AxesGroupName).Pr.Unit_Velocity).

Ex.

The Setting value of Position Command Unit (AxisName(AxesGroupName).Pr.Unit_Position) and Velocity Command Unit (AxisName(AxesGroupName).Pr.Unit_Velocity) and Velocity Command Unit Display (AxisName(AxesGroupName).Md.Unit_VelocityDisplay).

Position Command Unit (AxisName(AxesGroupName).Pr.Unit_Position)	Velocity Command Unit (AxisName(AxesGroupName).Pr.Unit_Velocity)	Velocity Command Unit Display (AxisName(AxesGroupName).Md.Unit_VelocityDisplay)
HFD010000	HFD000300	[mm/ms]

Combination of command units during multiple axes positioning control

During multiple axes positioning control, set command units of the axes group and the interpolation axes to be matched.
When the command units are mismatched, specify the input value of FB as shown below.

- For the input (Target position (Position), Movement amount (Distance), etc.) regarding position in multiple axes positioning control, specify the value based on the position command unit of the each interpolation axes necessarily.
- For the input (Velocity (Velocity), Acceleration (Acceleration), Deceleration (Deceleration), and Jerk (Jerk)) of multiple axes positioning control, specify the value based on the velocity command unit used for each control necessarily.
- When the combination of the position command unit between the axes group and the interpolation axes 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 velocity command unit and the FB being executed at multiple start, the warning "Velocity Command Unit Mismatch Warning" (warning code: 0D1EH) will occur at multiple start.

Velocity command units used for each control are shown below.
The following operations depend on interpolation controls.

■ Linear interpolation control (MCv_MoveLinearInterpolateAbsolute (Absolute Value Linear Interpolation Control), MCv_MoveLinearInterpolateRelative (Relative Value Linear Interpolation Control))

VelocityMode	Operation
0: Vector Velocity (VectorSpeed)	• The movement amount on the linear path (combined movement amount) is calculated from the value of current position and Target position (Position) or Movement amount (Distance). The position command unit of the axes group and the interpolation axes is not affected. • Unit of the axes group is used for the velocity command unit in controlling.
1: Long Axis Velocity (LongAxisSpeed)	• The axis whose movement amount value is the largest is regarded as the long axis from the value of current position and Target position (Position) or Movement amount (Distance). The position command unit of the axes group and the interpolation axes is 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: Reference Axis Velocity (ReferenceAxisSpeed)	• The unit of the reference axis (First element of Linear interpolation axes (LinearAxes)) is used for the velocity command unit in controlling.

- For details of linear interpolation control, refer to the following.

☞ Page 211 Linear interpolation control

■ Circular interpolation control (MCv_MoveCircularInterpolateAbsolute (Absolute Value Circular Interpolation Control), MCv_MoveCircularInterpolateRelative (Relative Value Circular Interpolation Control))

- Calculates circular path from the value of current position, Sub point (AuxPoint), and End point (EndPoint). The position command unit of the axes group and the configuration axes is not affected.
- The unit of the axes group is used for the velocity command unit in controlling.
- For details of circular interpolation control, refer to the following.

☞ Page 216 Circular interpolation control

■ Operation when axes group unit and interpolation axes 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.

Ex.

Operation example

- Axis setting and Axes group setting

Setting item	Setting details	Setting value
Position/Velocity command unit of the axes group	mm/ms	Position Command Unit (AxesGroupName.Pr.Unit_Position): HFD010000 Velocity Command Unit (AxesGroupName.Pr.Unit_Velocity): HFD000300
Position/Velocity command unit of the configuration axis 1	mm/s	Position Command Unit (AxisName.Pr.Unit_Position): HFD010000 Velocity Command Unit (AxisName.Pr.Unit_Velocity): H00000300
Position/Velocity command unit of the configuration axis 2	degree/min	Position Command Unit (AxisName.Pr.Unit_Position): H00410000 Velocity Command Unit (AxisName.Pr.Unit_Velocity): H00004700

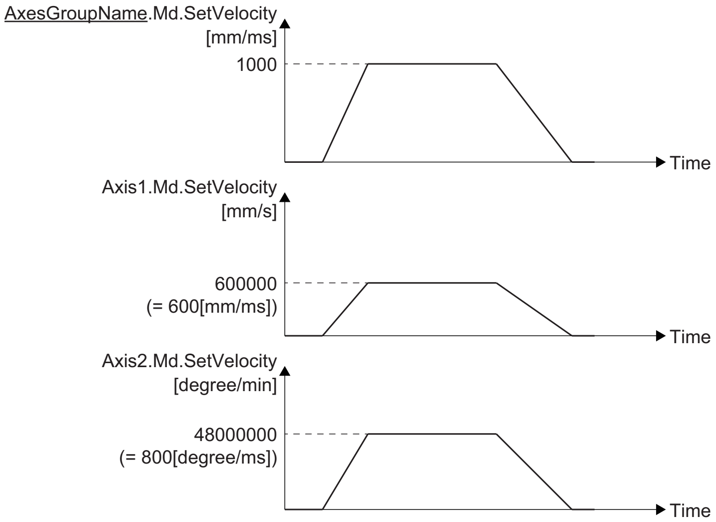
- FB input (MCv_MoveLinearInterpolateRelative (Relative Value Linear Interpolation Control))

Setting item	Setting value
Linear interpolation axes (LinearAxes[0])	1
Linear interpolation axes (LinearAxes[1])	2
Linear interpolation axes (LinearAxes[2])	0
Movement amount (Distance[0])	300
Movement amount (Distance[1])	400
Velocity (Velocity)	1000
Velocity Mode (VelocityMode)	0: Vector Velocity (VectorSpeed)

<Operation>

- Linear interpolation is performed by Movement amount (Distance) "300 [mm]" of the configuration axis 1, and "400 [degree]" of the configuration axis 2.
- Position and velocity command units of axes group are used for the unit of Velocity (Velocity) as a vector velocity.It will be controlled by the vector velocity at "1000 [mm/ms]".
- The motion system calculates a "combined movement amount $\sqrt{(300^2 + 400^2)}$ = 500 [mm]" from each value of the configuration axis movement amount. Each axis movement amount of every operation cycle is calculated from the combined movement amount (500 [mm]) and vector velocity (1000 [mm/s]). The calculated amount is commanded as a movement amount in the position command unit of each axis.
  The motion system calculates positioning speed of each axis such as follow.
  A monitor value of each axis speed is stored by the velocity command unit of each axis.

Positioning speed of each axis	Calculated value
Positioning speed of configuration axis 1	1000 × 300 / 500 = 600 [mm/ms] (= 600000 [mm/s])
Positioning speed of configuration axis 2	1000 × 400 / 500 = 800 [degree/ms] (= 48000000 [degree/min])



Precautions

- When "Out of Position Command Unit Range Warning (warning code: 0D2DH)" occurs, the initial value is stored in Position Command Unit Display (AxisName(AxesGroupName).Md.Unit_PositionDisplay) (The default value of position command unit: pulse)
- When "Out of Velocity Command Unit Range Warning (warning code: 0D2EH)" occurs, the initial value is stored in Velocity Command Unit Display (AxisName(AxesGroupName).Md.Unit_VelocityDisplay). (The default value of velocity command unit: pulse/s)
- Even if Position Command Unit (AxisName(AxesGroupName).Pr.Unit_Position), Velocity Command Unit (AxisName(AxesGroupName).Pr.Unit_Velocity) and Position Command Unit String (AxisName(AxesGroupName).Pr.Unit_PositionString) are changed, the homing request will not become 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 Set Position (AxisName.Md.SetPosition) is determined by Ring Counter Upper Limit Value (AxisName.PrConst.RingCount_UpperValue) and Ring Counter Lower Limit Value (AxisName.PrConst.RingCount_LowerValue).

Relevant variables

Axis parameter constant (AxisName.PrConst.)

Variable/Structure name	Name	Details
RingCount_Enable	Ring Counter Enabled Selection	Selects the ring counter enable/disable. • FALSE: Disabled[*1] • TRUE: Enabled
RingCount_UpperValue	Ring Counter Upper Limit Value	Sets the ring counter upper limit value.[*2] • -10000000000.0 to 10000000000.0
RingCount_LowerValue	Ring Counter Lower Limit Value	Sets the ring counter lower limit value.[*2] • -10000000000.0 to 10000000000.0

*1 Positioning range is "-10000000000.0 ≤ positioning range < 10000000000.0".
*2 The value is not fetched when the ring counter is disabled.

Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
SetPosition	Set Position	Stores the current position address.[*1] 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. The ring address is "-10000000000.0 . CumulativePosition < 10000000000.0".
FeedMachinePosition	Machine Feed Value	Stores the current position address by the machine coordinate. The accumulative value from the homing completed position is set. The address is not changed even if the current position is changed. The ring address is "-10000000000.0 . Machine feed value < 10000000000.0".
CommandedPosition	Commanded Position	Stores Target position (Position)/Movement amount (Distance) imported to the positioning FB being controlled. Stores "0.0" after the positioning is completed.
ActualPosition	Actual Position	Stores the actual current position. The value of "current position × driver unit conversion denominator / driver unit conversion numerator" is set. For details, refer to the following. ☞ Page 360 Driver unit conversion function The actual position is a value which is rounded with the ring counter range.

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

Set position

Set Position (AxisName.Md.SetPosition) can be set to an arbitrary ring address by setting Ring Counter Upper Limit Value (AxisName.PrConst.RingCount_UpperValue) and Ring Counter Lower Limit Value (AxisName.PrConst.RingCount_LowerValue). The ring counter upper/lower limit value can be set within the range of -10000000000.0 to 10000000000.0.

2

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.

Example	Description
1	When the positioning of relative position specification such as MC_MoveRelative (Relative Value Positioning) is executed repeatedly, it may accumulate the truncation error between the set position and the specified movement value. If the truncation error is a problem, execute the positioning of absolute position specification such as MC_MoveAbsolute (Absolute Value Positioning). In addition, refer to the following setting example to suppress the truncation error.
2	During positioning control such as automatic deceleration at low speed, "Overrun Warning (warning code: 0D10H)" or "Overrun Error (error code: 1A7EH)" may occur due to truncation error when the axis reaches the target position. Refer to the following setting example to suppress the truncation error.

■ Setting example

Setting real number type parameters that are related to Motion control functions in a way that does not include numbers after the decimal point can suppress truncation errors because all data being processed within the function is in real numbers. In the following example, when the truncation error is a problem in setting (A), setting (B) can be used alternatively.

- 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]

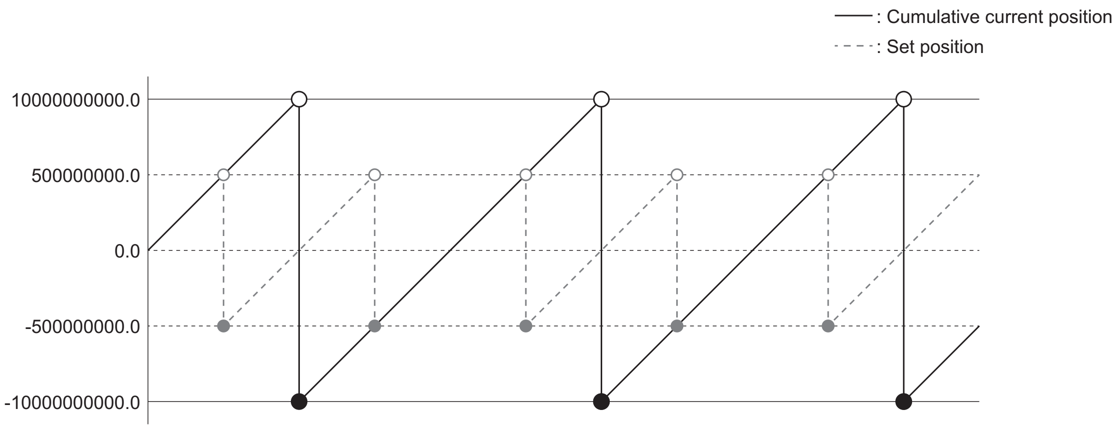
*1 For details on the setting method, refer to the following.
☞ Page 74 Technical Units

Control details

Set Position (AxisName.Md.SetPosition) to be the ring address is calculated based on Cumulative Current Position (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 Ring Counter Enabled Selection (AxisName.PrConst.RingCount_Enable) is "TRUE (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

Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
SetVelocity	Set Velocity	Stores the set output velocity.*1 This velocity is calculated from the difference of the set position. The value is positive when moving to the positive direction (Address increase direction), and the value is negative when moving to the negative direction (Address decrease direction). Stores the set velocity of the configuration axis during the axes group operation.
CommandedVelocity	Commanded Velocity	Stores commanded velocity imported to the motion FB being controlled. Stores "0.0" after the control is completed. Stores "0.0" in Commanded Velocity (AxisName.Md.CommandedVelocity) of the configuration axis during the axes group operation.
TargetVelocity	Target Velocity	Stores the actual target velocity whose override and velocity limit value are considered. Stores "0.0" after the control is completed. Stores "0.0" in Target Velocity (AxisName.Md.TargetVelocity) of the configuration axis during axes group operation. ■For the positioning control FB The value becomes more than "0.0" regardless of the movement direction. ■For other than the positioning control FB The value is positive when moving to the positive direction (Address increase direction), and the value is negative when moving to the negative direction (Address decrease direction)
ActualVelocity	Actual Velocity	Stores the actual velocity This velocity is calculated from the difference of the actual position. The value is positive when moving to the positive direction (Address increase direction), and the value is negative when moving to the negative direction (Address decrease direction).

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

Axes group monitor data (AxesGroupName.Md.)

Variable/Structure name	Name	Details
SetVelocity	Set Velocity	Stores the set output velocity during the axes group operation. The value becomes more than "0.0" as it is regardless of the movement direction of the configuration axes. Stores "0.0" after the control is completed. *: When "1: Long Axis Velocity (LongAxisSpeed)" or 2: Reference Axis Velocity (ReferenceAxisSpeed) is specified in the linear interpolation control, stores the absolute value of set velocity of the relevant axis.
CommandedVelocity	Commanded Velocity	Stores the commanded velocity imported to the motion FB being controlled. Stores "0.0" after the control is completed. *: When "1: Long Axis Velocity (LongAxisSpeed)" or 2: Reference Axis Velocity (ReferenceAxisSpeed) is specified on Velocity mode (VelocityMode) in the linear interpolation control, stores the commanded velocity of relevant axis.
TargetVelocity	Target Velocity	Stores the actual target velocity whose override and velocity limit value are considered. The value becomes more than "0.0" regardless of the movement direction of the configuration axis. Stores "0.0" after the control is completed. *: When "1: Long Axis Velocity (LongAxisSpeed)" or 2: Reference Axis Velocity (ReferenceAxisSpeed) is specified on Velocity mode (VelocityMode) in the linear interpolation control, stores the target velocity of relevant axis.
ActualVelocity	Actual Velocity	Stores the actual velocity. It is the vector velocity of the actual velocity of the configuration axes. The value becomes more than "0.0" as it is regardless of the movement direction of the configuration axes. Stores "0.0" when the axes group is invalid.

Velocity set range

The set velocity in the motion system is as follow.

Name	Description	Range
Commanded velocity	Set velocity that is specified in the motion FB.	0.0, ±0.0001 to ± 2500000000.0
Target velocity	The actual set velocity whose override and velocity limit value are considered with the commanded velocity.	

- For the valid range of the commanded velocity of each motion FB, refer to the each FB specifications.
- The operation that the target velocity be out of the range because of using override function, refer to the following.
  ☞ Page 349 Override Function
- When "0.0" is set to the commanded velocity in the FB to execute multiple start, the commanded velocity of previous FB will be set. For details, refer to the following.
  ☞ Page 132 Multiple Start (Buffer Mode)
- To perform floating-point operation, the lower limit value of the commanded velocity is limited by the following restrictions. If the speed that is converted from commanded velocity to operation cycle is less than 0.00001, "Out of Operation Cycle Converted Velocity Range (error code: 1AE4H)" (when changing the velocity, "Out of Operation Cycle Converted Velocity 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 Position Command Unit (AxisName(AxesGroupName).Pr.Unit_Position) or Velocity Command Unit (AxisName(AxesGroupName).Pr.Unit_Velocity).

Ex.

When the operation cycle is "0.5 [ms]" and the commanded velocity is set with the following units

If specifying the commanded velocity to "0.0001" with the following units, the calculation cycle converted speed is as follows.

Variable name	Setting value	Units
Position command unit (AxisName(AxesGroupName).Pr.Unit_Position)	H00000000	1×10^0 [pulse]
Velocity command unit (AxisName(AxesGroupName).Pr.Unit_Velocity	H00000300	1×10^0 [pulse/s]

Calculation Cycle Converted Speed = 0.0001 [pulse/s] $\div$ 1000 $\times$ 0.5 [ms] = 0.00000005 [pulse]

"Out of Operation Cycle Converted Velocity Range (error code: 1AE4H)" occurs.

If changing Position command unit (AxisName(AxesGroupName).Pr.Unit_Position) from " [pulse] (H00000000)" to "$\times 10^{-3}$ [pulse](HFD000000)", and changing commanded velocity to "0.1", the calculation cycle converted speed is as follows.

Calculation Cycle Converted Speed = 0.1×10^{-3} [pulse/s] $\div$ 1000 $\times$ 0.5 [ms] = 0.00005×10^{-3} [pulse]

Thus an error can be avoided by changing the commanded velocity to match the change of the position command unit.

Precautions

When "0.0 < velocity after velocity override < 0.0001", "Out of Velocity Range Clamping Warning (warning code: 0D34H)" occurs and becomes "0.0".

2.4 Servo ON/OFF

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.

Relevant variables

Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
Driver_ReadyOn	Driver Ready On Status	Displays the connected driver status. Becomes TRUE when Driver Status (AxisName.Md.Driver_State) of the connected driver is as follows. • 2: Fault Reaction Active (FaultReactionActive) • 5: Switched On (SwitchedOn) • 6: Operation Enable (OperationEnable) • 7: Quick Stop Active (QuickStopActive)
Driver_ServoOn	Driver Servo On Status	Displays the connected driver status. Becomes TRUE when Driver Status (AxisName.Md.Driver_State) of the connected driver is as follows. • 6: Operation Enable (OperationEnable) • 7: Quick Stop Active (QuickStopActive)

Relevant FBs

For details on Motion control FBs, refer to the following.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

Motion control FB	Name	Description
MC_Power	Operation possible	Switches a specified axis to the operation possible status.
MCv_AllPower	All axes operation possible	Switches every axis to the operation possible status.

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 87 Control details

■ MC_Power (Operation Available) 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 (Enable) and Servo ON request (ServoON) of MC_Power (Operation Available).

Input variables		Output variables		Servo ON/OFF status	Driver Status (AxisName.Md.Driver_State)
Enable (Enable)	Servo ON request (ServoON)	Ready ON status (ReadyStatus)	Operable (Status)		
TRUE	TRUE	TRUE	TRUE	Servo ON	6: Operation Enable (OperationEnable)
	FALSE	TRUE	FALSE	Servo OFF	5: Switched On (SwitchedOn)
FALSE	TRUE	FALSE	FALSE	Servo OFF	3: Switch On Disabled (SwitchOnDisabled)
	FALSE	FALSE	FALSE	Servo OFF	3: Switch On Disabled (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 driver device specification.
- While the drive unit error occurs, Enable (Enable) and Servo ON request (ServoON) are not required to be turned from FALSE to TRUE again since MC_Power (Operation Available) 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 (Enable) and Servo ON request (ServoON) of MCv_AllPower (All Axes Operation Available).

Input variables		Servo ON/OFF status	Driver Status (AxisName.Md.Driver_State)
Enable (Enable)	Servo ON request (ServoON)		
TRUE	TRUE	Servo ON	6: Operation Enable (OperationEnable)
	FALSE	Servo OFF	5: Switched On (SwitchedOn)
FALSE	TRUE	Servo OFF	3: Switch On Disabled (SwitchOnDisabled)
	FALSE	Servo OFF	3: Switch On Disabled (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 driver device specification.
- While the drive unit error occurs, since MCv_AllPower (All Axes Operation Available) has been sent to the driver, Enable (Enable) and Servo ON request (ServoON) are not required to be turned from FALSE to TRUE again.

Point

To individually execute servo OFF when using MCv_AllPower (All Axes Operation Available), use MC_Power (Operation Available) together.

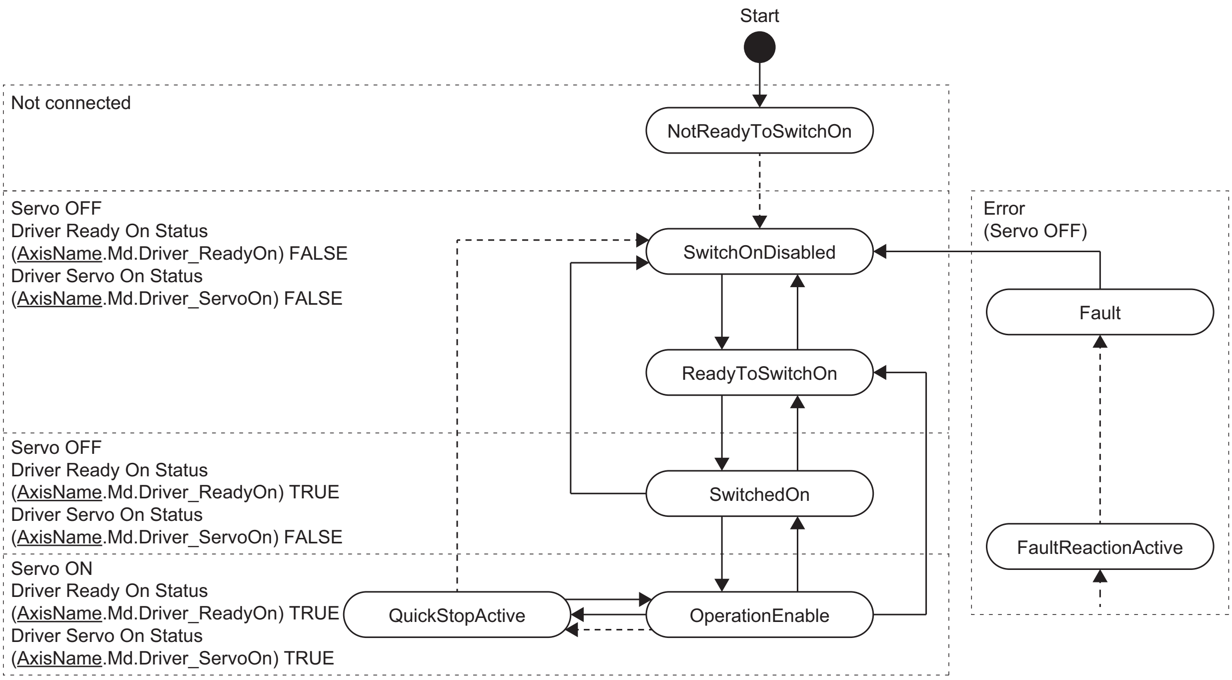
When MCv_AllPower (Operation Available) and MC_Power (All Axes Operation Available) are used together, the command of MC_Power (Operation Available) is given priority.

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 Driver Status (AxisName.Md.Driver_State).

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



Precautions

- When MC_Power (Operation Available) calls the instruction first after the PLC READY is switched to ON, the axis information is determined. Even if the axis information is changed while Enable (Enable) is FALSE, the change is not reflected.
- Do not set two or more MC_Power (Operation Available) for one axis. The operation is not guaranteed if those are set.
- The servo ON/OFF status when stopping the program is maintained while the READY is OFF.

2.5 Follow Up

Follow up is the function to reflect the input (current position) from the device station assigned to an axis 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.

Axis type	Description
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 (Speed Control)/MC_TorqueControl (Torque Control) are in execution. • The driver control mode of the driver device is connected by motion system axis control unsupported mode. • MC_Home (Homing) is in execution. (When "1: Follow up enabled" is selected in Follow-up enabled/disabled selection (Options (Options): Bit 16))
Real 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.
Virtual Encoder axis	

Operation of this function for each system status

○: Possible, △: Possible (restricted), ×: Not possible

Status	Operation availability
STOP	○
RUN	○
Moderate error	△*1
Major error	×

*1 Operation may not be possible depending on the add-on and device station status.

Relevant variables

Axis parameter constant (AxisName.PrConst.)

Variable/Structure name	Name	Details
Encoder_CounterDisableSignal	Counter Disabling Signal	Sets a signal to switch to the counter disable. The following shows the specific setting and the operation of Counter Disabling Signal (Encoder_CounterDisableSignal). ■Target (Target) Only [VAR], [DEV] and [CONST] can be specified to data type for Target. ■Signal detection method (Detection) Only the following level detections can be set. • 0: Detection at TRUE (HighLevel) • 1: Detection at FALSE (LowLevel) ■Compensation time (CompensationTime) Only "0.0 [s]" can be specified ■Filter time (FilterTime) The setting range is "0.0 to 5.0 [s]" *: When a value outside the range is set, "Out of Parameter Range (Axis) (error code: 1D80H)" will occur.

Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
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	Becomes 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.

Axis control data (AxisName.Cd.)

Variable/Structure name	Name	Details
FollowupDisable	Follow-up 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

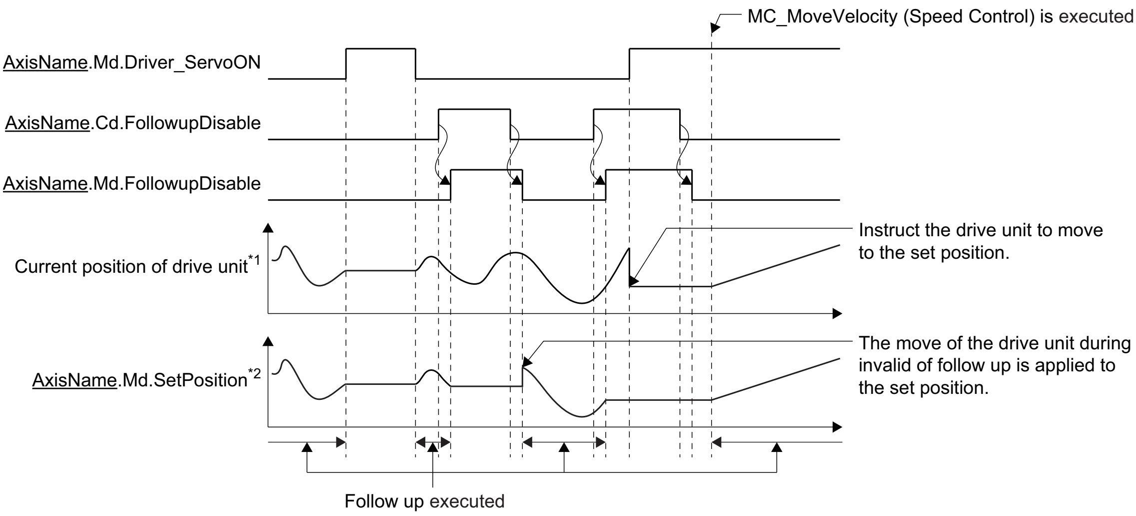
The follow up enabled/disabled can be switched for each axis (default status: enabled). Input (the current position) from the device station assigned to an axis is disabled and not reflected in the set position by switching the follow up to enabled.

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

Follow-up Disabled (AxisName.Md.FollowupDisable) becomes TRUE by setting Follow-up Disabled (AxisName.Cd.FollowupDisable) to TRUE, and the follow up is disabled.



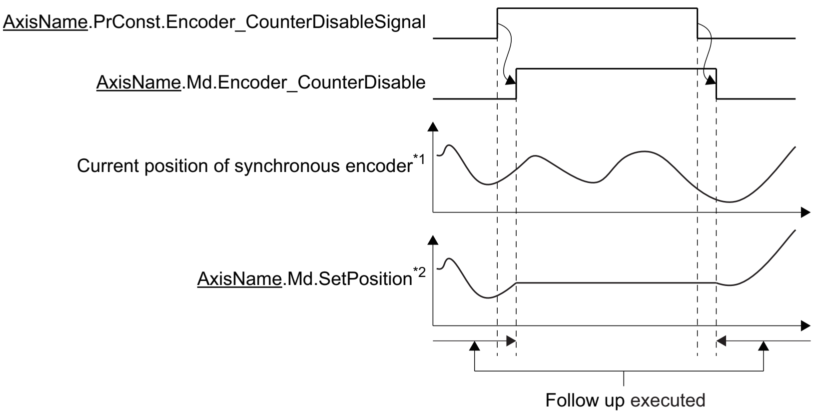
*1 When checking with slave label

*2 When driver unit conversion (Numerator / Denominator) is (1 / 1).

For real encoder axis/virtual encoder axis

■ When Counter Disabling Signal (AxisName.PrConst.Encoder_CounterDisableSignal) is "no setting"

Counter Disabled (AxisName.Md.Encoder_CounterDisable) becomes TRUE by setting Counter Disabled (AxisName.Cd.Encoder_CounterDisable) to TRUE and the input from the synchronous encoder will be disabled.

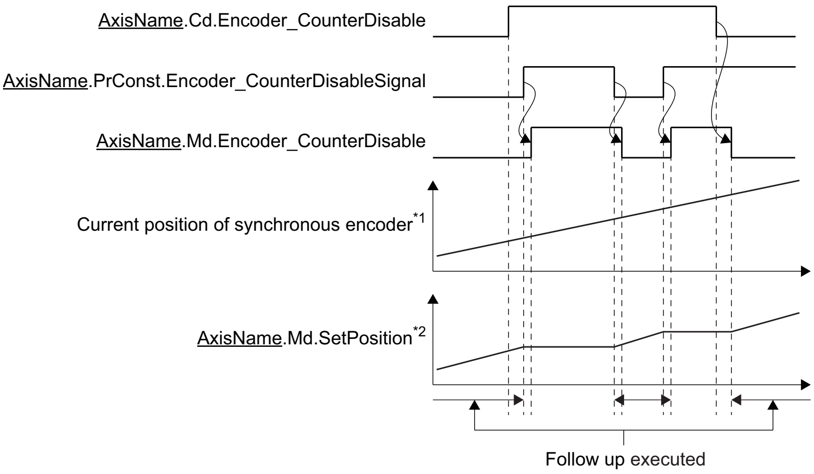


*1 When checking with the slave label

*2 When driver unit conversion (Numerator / Denominator) is (1 / 1).

■ **When Counter Disabling Signal (AxisName.PrConst.Encoder_CounterDisableSignal) is set**

While Counter Disabled (AxisName.Cd.Encoder_CounterDisable) is TRUE, the counter disabling signal will be valid. Counter Disabled (AxisName.Cd.Encoder_CounterDisable) becomes TRUE while the signal is in detection, and becomes the counter disable. While Counter Disabled (AxisName.Cd.Encoder_CounterDisable) is FALSE, the counter disable signal becomes invalid.



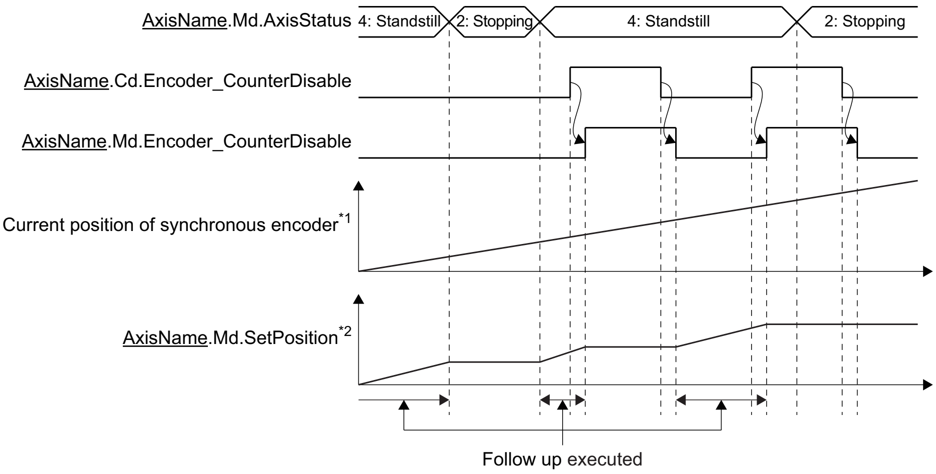
*1 When checking with the slave label

*2 When driver unit conversion (Numerator / Denominator) is (1 / 1).

■ **When Axis Status (AxisName.Md.AxisStatus) is "2: Decelerating to stop (Stopping)"**

Follow-up is disabled while Axis Status (AxisName.Md.AxisStatus) is "2: Deceleration to stop (Stopping)".

At this time, Counter Disabled (AxisName.Md.Encoder_CounterDisable) does not change.



*1 When checking with slave label

*2 When driver unit conversion (Numerator / Denominator) is (1 / 1).

Precautions

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.

For real encoder axes/virtual encoder axes

- The follow-up may not operate depending on the axis status. For details refer to the following.

☞ Page 42 Real encoder axis
☞ Page 48 Virtual encoder axis

For all axes types

- When the movement amount of every operation cycle exceeds the following range, the follow-up does not operate normally because of the movement amount of axis are different with the movement amount of device station. Use the device stations so that their movement amount is within the following range.

Axis type	Valid range for available follow-up per operation cycle
Real drive axis	(Movement amount per operation cycle) [Driver unit] within signed 32bit integral number range
Real encoder axis (Real encoder axis type: via drive unit)	- ((Multiple revolution counter maximum value) / 2 × (Encoder resolution)) ≤ (Movement amount per operation cycle) [Encoder pulse unit] ≤ ((Multiple revolution counter maximum value) / 2 × (Encoder resolution) -1)
Virtual encoder axis	- (Encoder ring counter upper limit value - Encoder ring counter lower limit value + 1) / 2 < (Movement amount per operation cycle) [Encoder pulse unit] < (Encoder ring counter upper limit value - Encoder ring counter lower limit value + 1) / 2

Relevant add-ons

The following add-ons are required to use this function.

- Axis
- MotionEngine

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 101 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 device station assigned to an axis 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 95 Current position restoration

Relevant variables

Axis parameter constant (AxisName.PrConst.)

Variable/Structure name	Name	Details
PosRestoration_AbsPosEnable	Absolute Position Control Setting	Sets the absolute position control setting. • 0: Disable Absolute Position System (ABSDisabled) • 1: Enable Absolute Position System (Enabled) • -1: Automatic Setting (Acquire from Connected Device) (Auto)
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: Feed Machine Position (FeedMachinePosition)

Axis parameter (AxisName.Pr.)

Variable/Structure name	Name	Details
Homing_Required	Homing Required or Not	Sets whether homing is required or not. Sets FALSE when the homing is not required. If FALSE is set, the homing request will not become TRUE at the current position restoration. • FALSE: Homing Not Required • TRUE: Homing Required

Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
Homing_Required	Homing Required or Not	Displays whether the axis requires the homing or not. If FALSE is set, the homing request will not become TRUE at the current position restoration. • FALSE: Homing Not Required • TRUE: Homing Required
PosRestoration_Status	Current Position Restoration Status	Displays the current position restoration status. • 0: Not Executed (NotExecute) • 1: Waiting for Restoration Request (WaitingRequest) • 2: Restoration Completed in Incremental System (RestoredInIncSystem) • 3: Restoration Completed in Absolute Position System (Homing Uncompleted) (RestoredInAbsSystemUnHomed) • 4: Restoration Completed in Absolute Position System (RestoredInAbsSystem) *: "0: Not Executed (NotExecute)" is displayed when the device station is disconnected.
Homing_Request	Homing Request	Displays whether homing is required or not. The timing of becoming TRUE differs depending on the absolute position control setting. For details, refer to the following. ☞ Page 97 Homing request Becomes FALSE when the homing is completed. • FALSE: No homing request • TRUE: Homing request
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	Machine Feed Value	Stores the current position address by the machine coordinate. It is the cumulative current value from the homing completed position. The address is not change even though the current position is changed.

System monitor data (System.Md.)

Variable/Structure name	Name	Details
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 canceled • 4: Restored data saving • 5: Restored data saving completed

System control data (System.Cd.)

Variable/Structure name	Name	Details
BackupRestore	Backup/Restore Request	Executes backup stop and restored data saving. • 0000H: No request • ABCDH: Backup stop request • FFFEH: Restored data saving request • Other than the above: No operation

Current position restoration

Specify whether the absolute position system is used for the current position restoration or not in the absolute position control setting (AxisName.PrConst.PosRestoration_AbsPosEnable).

Absolute position control setting (AxisName.PrConst.PosRestoration_AbsPosEnable)	Details
0: Disable absolute position system (ABSDisabled)	☞ Page 95 Incremental system
1: Enable absolute position system (Enabled)	☞ Page 96 Absolute position system
-1: Automatic setting (Acquire from connected device) (Auto)	☞ Page 96 Connected device setting

- "-1: Automatic Setting (Acquire from Connected Device) (Auto)" can be set when the connected device has the ENCODER_ABS_STATUS object.
- When "-1: Automatic Setting (Acquire from Connected Device) (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, "Absolute Position Control Setting Mismatch (error code: 1A9AH)" will occur at the current position restoration.)

Ex.

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: Enable Absolute Position System (Enabled)" for Absolute Position Control Setting (AxisName.PrConst.PosRestoration_AbsPosEnable) 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 101 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 machine feed value) becomes "0.0", and the homing request (AxisName.Md.Homing_Request) becomes TRUE. (The set position may not become "0.0" depending on the ring counter setting. (☞ Page 80 Positioning Range))

When restoration is completed, the axis current position restoration status (AxisName.Md.PosRestoration_Status) is "2: Restoration Completed in Incremental System (RestoredInIncSystem)".

If the machine position establishment with the device station is not required, set to FALSE in the homing request or not. (AxisName.Pr.Homing_Required) This prevents the homing request (AxisName.Md.Homing_Request) from becoming 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 machine feed value) is restored based on backed up absolute position data. When restoration is completed, the axis current position restoration status (AxisName.Md.PosRestoration_Status) becomes "4: Restoration Completed in Absolute Position System (RestoredInAbsSystem)".
However, if "Absolute Position Data Incorrect Warning (warning code: 0D00H)" has been detected at restoration, the axis current position restoration status (AxisName.Md.PosRestoration_Status) becomes "3: Restoration Completed in Absolute Position System (Homing Uncompleted) (RestoredInAbsSystemUnHomed)". The current position is restored with the same content as the incremental system and the homing request (AxisName.Md.Homing_Request) becomes 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 (AxisName.Md.PosRestoration_Status) becomes "4: Restoration Completed in Absolute Position System (RestoredInAbsSystem)".)
When the absolute position system is used, the restoration content differs depending on the axis type. For details, refer to the following.
☞ Page 101 Current position restoration for each axis type

In the following case, "Absolute Position Data Incorrect Warning (warning code: 0D00H)" occurs.

- 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.
- For real drive axes, an error occurs by the following causes.
  - "Absolute position erased" is detected in the driver side.
  - "Polarity (607EH)" b7: position polarity of the slave object is changed.
  - "HomeOffset (607CH)" of the slave object is changed.
  - The connected driver device is changed.
  - The encoder resolution of the driver device is changed.
- For real encoder axes (via drive unit), an error occurs by the following causes.
  - "Absolute position erased" is detected in the scale measurement encoder.
  - The encoder resolution of the scale measurement encoder is changed.
- For virtual encoder axes, an error occurs by the following causes.
  - The upper limit value/the lower limit value of the encoder ring counter is changed.

Connected device setting

For a real axis, the connected device with the ENCODER_ABS_STATUS object acquires the absolute position system setting by the SDO communication, and the current position is restored.
Set the absolute position system setting in the connected device side.
For a real axis, if the connected device does not have the ENCODER_ABS_STATUS object, it operates as the incremental system without an error.
For a virtual axis, it operates as the incremental system.

Homing request

The homing request (AxisName.Md.Homing_Request) will be TRUE in the following cases.

Point

The reason why the homing request (AxisName.Md.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 ON
	The driver power turns on.*3 [The real drive axis only]		At current position restoration
Disable absolute position system	The current position restoration is executed.		At the current position restoration
Enable 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 slave 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 occurring
	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 encoder ring counter is changed.	

*1 For the "-1: Automatic Setting (Acquire from Connected Device) (Auto)" setting, refer based on the connected device setting.

*2 When the homing required or not (AxisName.Pr.Homing_Required) is FALSE, the homing request (AxisName.Md.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: Enable Absolute Position System (Enabled)" is set and the homing is completed at resetting, the homing request will not become 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. ("Absolute Position Data Incorrect Warning (warning code: 0D00H)" occurs.)
At the current position restoration, if the setting values match with the axis variable and the axis type of the absolute position data, the absolute position data will be taken over even when the variable names are different.

Ex.

When the following operations are performed, Axis0002 takes over the absolute position data of Axis0001.
· Deletes an axis (Variable name: Axis0001, Axis No.: 1, Axis type: Real drive axis) and executes writing
· Adds an axis (Variable name: Axis0002, Axis No.: 1, Axis type: Real drive axis) and executes writing
· Executes the current position restoration

Backup is executed while the homing request (AxisName.Md.Homing_Request) is FALSE. (When the homing request (AxisName.Md.Homing_Request) becomes TRUE, the absolute position data is discarded.)
Backups will take place in intervals of 2 [s] from the previous backup.

- Storage folder and file name
  The following shows the storage folder and file name of the backed up absolute position data.
  · 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 Maximum Backup RAM Size (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, "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.

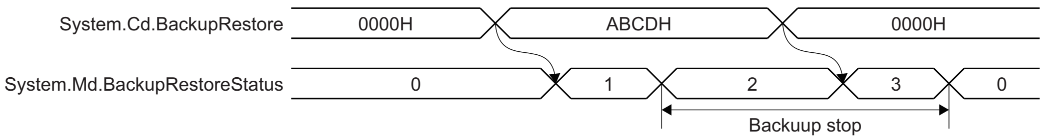
After restoring is completed, software reboot is required to enable the restored absolute position data. (The backup is maintained stop and it can not be cleared until software reboot is carried out.)

For real drive axis, the homing request (AxisName.Md.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 Backup/Restore Request (System.Cd.BackupRestore).
2. Confirm Backup/Restore Information (System.Md.BackupRestoreStatus) changes to "1: Backup stop required" 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 "0000H: No request" to Backup/Restore Request (System.Cd.BackupRestore) after confirming file transfer (Execution data backup).
5. Confirm Backup/Restore Information (System.Md.BackupRestoreStatus) changes to "3: Backup stop canceled" and "0: Backup restore not executed".



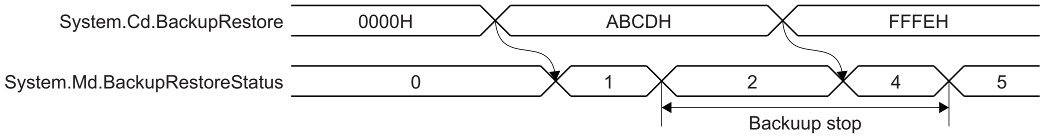
Point

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



Point

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 Backup/Restore Information (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
		Absolute Position Reference Setting (AxisName.PrConst.PosRestoration_AbsPosBase) is "3: Machine Feed Value (FeedMachinePosition)"
Machine feed value	Restored in "0.0".	Differs depending on the restoration method. For details, refer to the following. ☞ Page 93 Absolute position system
Cumulative current position	The same value as restored machine feed value.	
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 device station. For details on the absolute position system to execute the 64-bit current position restoration, refer to the following.

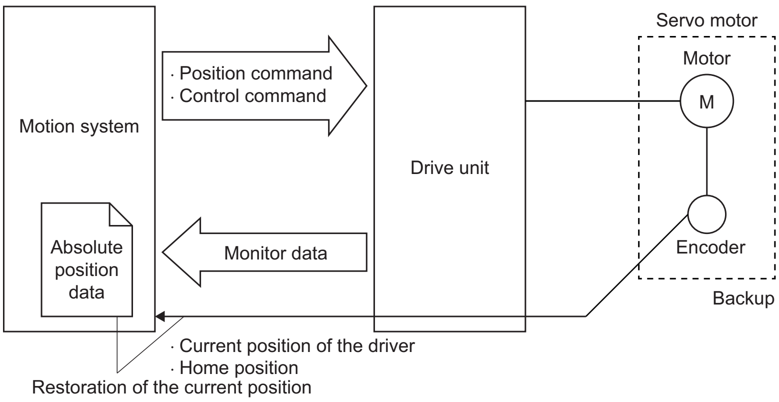
☞ Page 93 Absolute position system

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.

For MR-J5(W)-G: 📖MR-J5-G/MR-J5W-G User’s Manual (Parameters)

- 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 device station has been changed.

*1 Actual restoration possible range is shown below.
- ((Multiple revolution counter maximum value) / 2 × (Encoder resolution)) to ((Multiple revolution counter maximum value) / 2 × (Encoder resolution) - 1)

- The unit of the restoration possible range from the reference position in 64-bit restoration is shown below
  · Driver unit or motor encoder unit
  · For the motor encoder 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 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.
  If any one of the above parameters is not specified, 32-bit restoration is executed.

Setting item Slave Object Data (AxisName.PrConst.SlaveObject) (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 position 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.

Precautions

- Even if the above parameters have been specified, "Driver Restoration Data Incorrect (error code: 1AE0H (detail code: 0001H))" will occur and the current position restoration may not be completed. Execute reconnecting after correcting the parameter of the drive unit.
- For MR-J5(W)-G
  When the drive unit cannot restore the current position, "Driver Restoration Data Incorrect (error code: 1AE0H (detail code: 0001H))" will occur in the motion system, and the current position restoration is not completed. The current position may be restored by checking the status of the drive unit and reconnecting it. During reconnection, when homing request becomes TRUE, execute homing again.

■ 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 of the servo parameter, "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: Automatic Setting (Acquire from Connected Device) (Auto)" or "1: Enable Absolute Position System (Enabled)" in Absolute Position Control Setting(AxisName.PrConst.PosRestoration_AbsPosEnable) 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 "0.0".	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 Real Encoder Axis Type Setting (AxisName.PrConst.Encoder_AxisType) 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 unit] 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 30 Axis

*1 Actual restoration possible range is shown below.
- ((Multiple revolution counter maximum value) / 2 × (Encoder resolution)) to ((Multiple revolution counter maximum value) / 2 × (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 "0.0".	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 33 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 "0.0".	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 Connection Command (AxisName.Cd.Encoder_Connect) becomes TRUE.

Virtual linked axis

Each current position is restored as follows.

Current position	Incremental system	Absolute position system
Cumulative current position	Restored in "0.0".	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 33 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, "Absolute Position Control Setting Mismatch (error code: 1A9AH)" will occur at current position restoration.)
- If the homing required or not (AxisName.Pr.Homing_Required) is set to TRUE, address information stored in the motion system is not guaranteed.
- During backup/restoration by the engineering tool, do not write in Backup/Restore Request (System.Cd.BackupRestore).
- A backup file created with a different model can be used. However, current position restoration can only be executed for the number of axes of the backup file.

Ex.

When a backup file is created with a 4-axis model, and the backup file is moved to an 8-axis model.
Current position restoration is executed for 4 axes and the not for the remaining 4 axes.

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.

Operation of this function for each system status

○: Possible, ×: Not possible

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

Relevant variables

Axis parameter constant (AxisName.PrConst.)

Variable/Structure name	Name	Details
OperationCycle	Control Cycle Setting	Sets the control operation cycle. 0: Operate in the first operation cycle
FastOperationMode*1	Fast Operation Mode Setting	Switch the operation process of the set axis to fast operation mode.*2 • 5FE2H: Fast operation mode • Others: Normal mode

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

*2 Only the real drive axis is supported.

Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
OperationCycle	Control Cycle	Stores the control operation cycle of the axis. • 0: Not initialized/Outside the range of the operation cycle setting • 1: Operate in the first operation cycle

System parameter constant (System.PrConst.)

Variable/Structure name	Name	Details
OperationCycle[1].Cycle	Operation Cycle Setting (cycle setting)	Sets the first operation cycle. Setting and operations specific to Operation cycle setting (OperationCycle[1]) are shown below. ■Cycle setting (Cycle) • 0: Synchronizes with the basic cycle of network
BuffermemoryRefreshCycle.Cycle	Buffer Memory Refresh Cycle Setting (cycle setting)	Sets the buffer memory refresh cycle. Setting and operations specific to Buffer memory refresh cycle setting (BuffermemoryRefreshCycle) are shown below. ■Cycle setting (Cycle) • 0: The nth operation cycle ("n" is the slowest operation cycle among operation cycles whose setting is enabled.) • -118192 to -110001: Multiplication by n of the first operation cycle (1 to 8192)
Link_MotionStationRefreshType	Motion Synchronization Station Send/ Receive Data Refresh Method	Sets the send/receive data refresh method of the motion synchronization station. • 0: Response preferred method (EmphasisResponse) • 1: Operation cycle preferred method (EmphasisOperationCycle)
FastOperationMode*1	Fast Operation Mode Setting	Switch the operation process of the entire system to fast operation mode. • 50EFH: Fast operation mode • Others: Normal mode

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

System monitor data (System.Md.)

Variable/Structure name	Name	Details
SystemBaseCycle_Counter	System Basic Cycle Counter	Stores the system basic cycle counter. Counts up every time when executing system basic processing.

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 Operation Cycle Setting (System.PrConst.OperationCycle[1]).
"Synchronize with the network setting" must be set to control real axes connected to the network.

A rough standard for the operation cycle settings

The following shows a reference of the operation cycle settings.

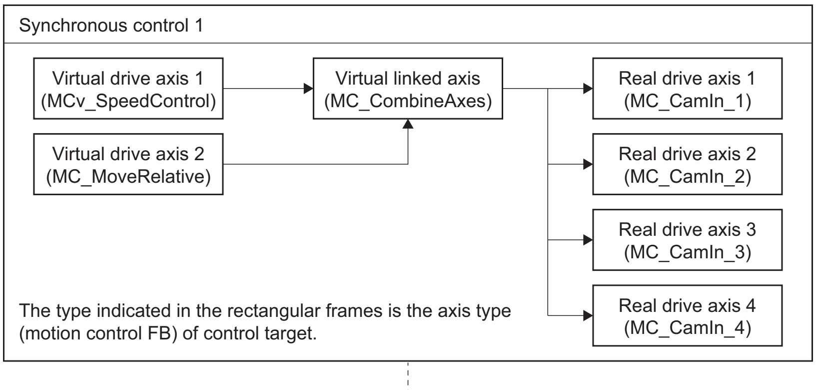
■ A rough standard for the number of axes and the operation cycle settings when using the positioning control

A rough standard is shown below when using real axis set as the axes group of two axes each and performing positioning control repeatedly.

Operation cycle	RD78G4	RD78G8	RD78G16	RD78G32	RD78G64	RD78GHV	RD78GHW
31.25 [μs]	—					—	
62.5 [μs]	—					—	
125 [μs]	—					2 axes	
250 [μs]	2 axes					8 axes	
500 [μs]	4 axes	8 axes				22 axes	
1.0 [ms]	4 axes	8 axes	16 axes	20axes		44 axes	
1.5 [ms]	4 axes	8 axes	16 axes	32 axes		66 axes	
2.0 [ms]	4 axes	8 axes	16 axes	32 axes	40 axes	74 axes	
3.0 [ms]	4 axes	8 axes	16 axes	32 axes	64 axes	96 axes	
4.0 [ms]	4 axes	8 axes	16 axes	32 axes	64 axes	128 axes	
5.0 [ms]	4 axes	8 axes	16 axes	32 axes	64 axes	128 axes	
6.0 [ms]	4 axes	8 axes	16 axes	32 axes	64 axes	128 axes	
7.0 [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

■ A rough standard for the number of axes and the operation cycle settings when using the synchronous control

A rough standard is shown below when Synchronous control 1 in the following configuration example is one unit, Virtual drive axis 1 and Virtual drive axis 2 are controlled, and four Real drive axes are synchronized. (The number of using axes shown in the table indicates the number of real drive axes in the configuration example.)



Operation cycle	RD78G4	RD78G8	RD78G16	RD78G32	RD78G64	RD78GHV	RD78GHW
31.25 [μs]	—					—	
62.5 [μs]	—					—	
125 [μs]	—					—	
250 [μs]	—					8 axes	
500 [μs]	4 axes	8 axes				16 axes	
1.0 [ms]	4 axes	8 axes	16 axes			28 axes	
1.5 [ms]	4 axes	8 axes	16 axes			38 axes	
2.0 [ms]	4 axes	8 axes	16 axes	28 axes		48 axes	
3.0 [ms]	4 axes	8 axes	16 axes	32 axes		66 axes	
4.0 [ms]	4 axes	8 axes	16 axes	32 axes	48 axes	84 axes	
5.0 [ms]	4 axes	8 axes	16 axes	32 axes	64 axes	128 axes	
6.0 [ms]	4 axes	8 axes	16 axes	32 axes	64 axes	128 axes	
7.0 [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	152 axes

■ A rough standard for the number of axes and the operation cycle settings for the configuration minimized system load

A rough standard when using axis operation process fast operation mode and executing MCv_Jog (JOG Operation) by each real drive axis is shown below.

Operation cycle	RD78G4	RD78G8	RD78G16	RD78G32	RD78G64	RD78GHV	RD78GHW
31.25 [μs]	—					2 axes	
62.5 [μs]	1 axes					4 axes	
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	
1.5 [ms]	4 axes	8 axes	16 axes	32 axes	64 axes	82 axes	
2.0 [ms]	4 axes	8 axes	16 axes	32 axes	64 axes	100 axes	
3.0 [ms]	4 axes	8 axes	16 axes	32 axes	64 axes	128 axes*1	
4.0 [ms]	4 axes	8 axes	16 axes	32 axes	64 axes	128 axes*1	
5.0 [ms]	4 axes	8 axes	16 axes	32 axes	64 axes	128 axes*1	
6.0 [ms]	4 axes	8 axes	16 axes	32 axes	64 axes	128 axes*1	
7.0 [ms]	4 axes	8 axes	16 axes	32 axes	64 axes	128 axes*1	
8.0 [ms]	4 axes	8 axes	16 axes	32 axes	64 axes	128 axes*1	256 axes*1

*1 Using Normal mode.

Point

- Adjust the operation cycle setting depending on the system load status.
- The following functions increase the operation cycle, so use them depending on the system load status.
  · Public label
  · Device label setting
  · Instantiating and execution of the Motion control FB
  · Logging
- Set execution scan of ST programs depending on the system load status as executing ST programs by fixed scan increases operation time.
- The operation time increases when the function block starts. The increase of operation time can be reduced by changing the start timing. Note the following for parameter setting values, and adjust them depending on the system load status.
- The operation time increases at output "Cycle Over Warning (warning code: 0F08H)".
  (1) Set 0 to number of cycle over warning detections (System.PrConst.OperationCycle[1].NumOfCycleOverWngDetectTimes), so "Cycle Over Warning (warning code: 0F08H)" will not be output.
  (2) User needs to check the normal operation range of the number of cycle over error detections (System.PrConst.OperationCycle[1].NumOfCycleOverErrDetectTimes), and change values to avoid "Cycle Over (error code: 1C80H)" and "Cycle Over (error code: 320CH)".
- When the operation cycle is set to 31.25 [μs], specify "1: Operation cycle preferred method (EmphasisOperationCycle)" for Operation Cycle Preferred Method (System.PrConst.Link_MotionStationRefreshType).

Operation cycle assignment

An operation cycle of the axis can be set by Control Cycle Setting (AxisName.PrConst.OperationCycle).

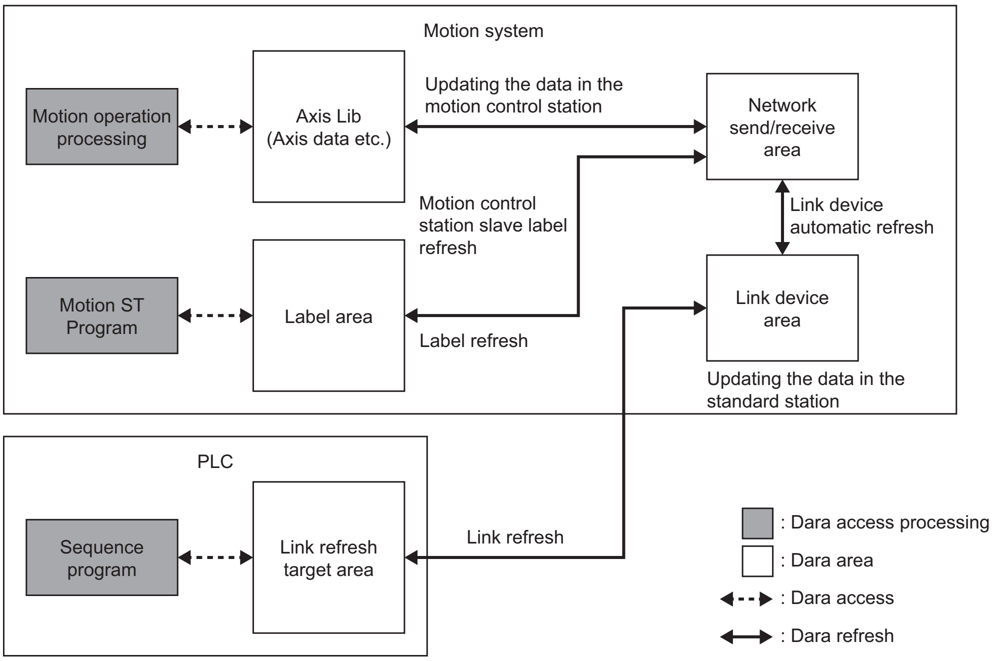
Send/Receive data refresh of network connected devices

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

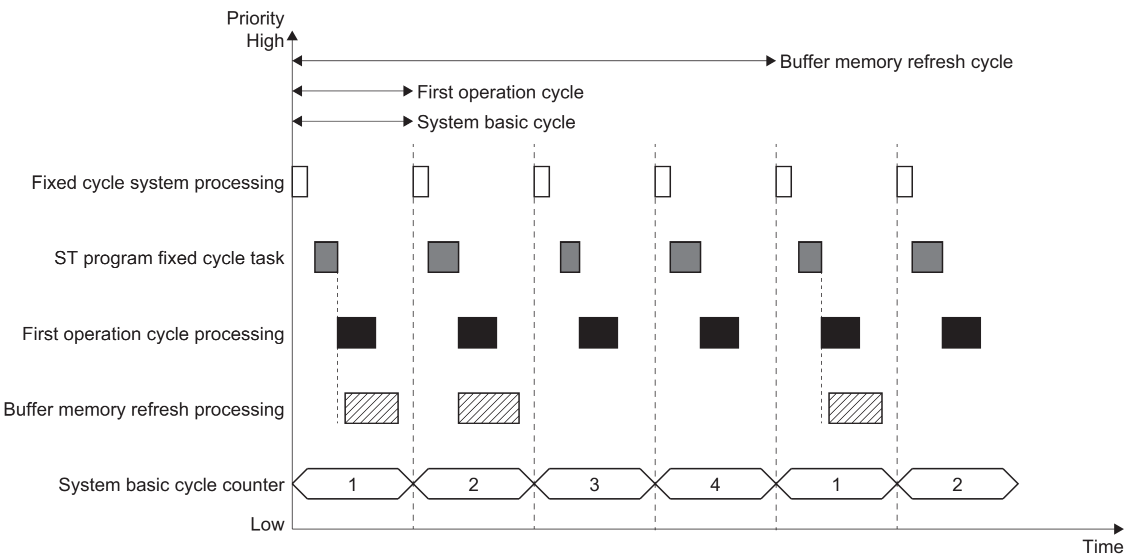
Station	Explanation
Motion synchronization station	• The motion system refreshes data by directly synchronizing the network send/receive with the operation cycle. • The target link device is not refreshed.
Standard station*1	• Stations other than motion synchronization stations are standard stations.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

*1 Control by the programmable controller

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



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



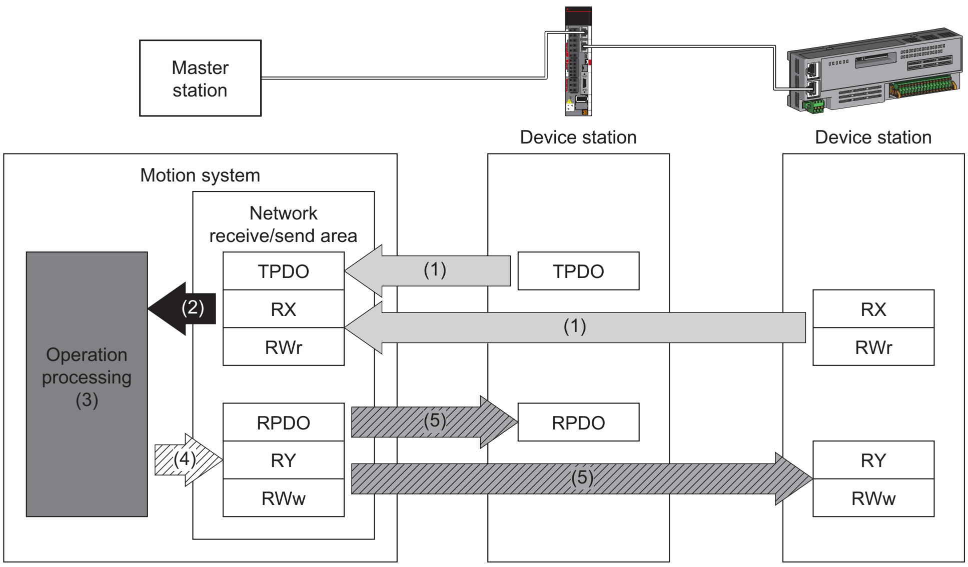
Motion synchronization station

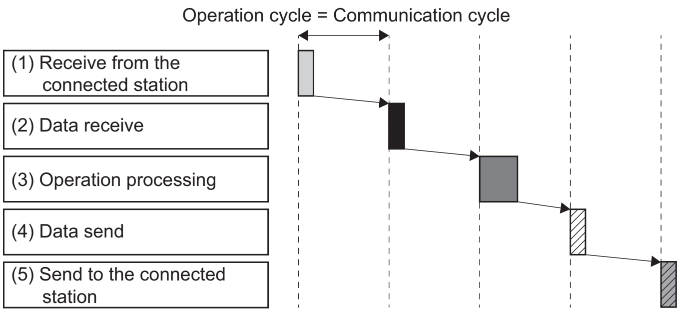
Select a data refresh method from the following for motion synchronization station.
It can be set in Motion Synchronization Station Send/Receive Data Refresh Method (System.PrConst.Link_MotionStationRefreshType). It is common in all of the operation cycles.

Motion synchronization station send/receive data refresh method	Description
Operation cycle preferred method (default)	• Data receive, motion operation, and data send are executed in different operation cycles.
Response preferred method	• Data receive, motion operation, and data send are executed in the same operation cycles. • The timing of data update is earlier compered to operation cycle preferred method so that the responsiveness to feedback is high. • More processing steps are needed in one operation cycle compared to operation cycle preferred method so that it takes long for the operation cycle processing time.

■ Operation cycle preferred method

With operation cycle preferred method, data receive, operation processing, 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.





Point

Slave label executes the refresh at the following timing.
- Receive data (RX, RWr): (3) Operation processing (reflect data received before two cycles from the connected station)
- Send data (RY, RWw): (4) Data send (send to the connected station after one cycle)

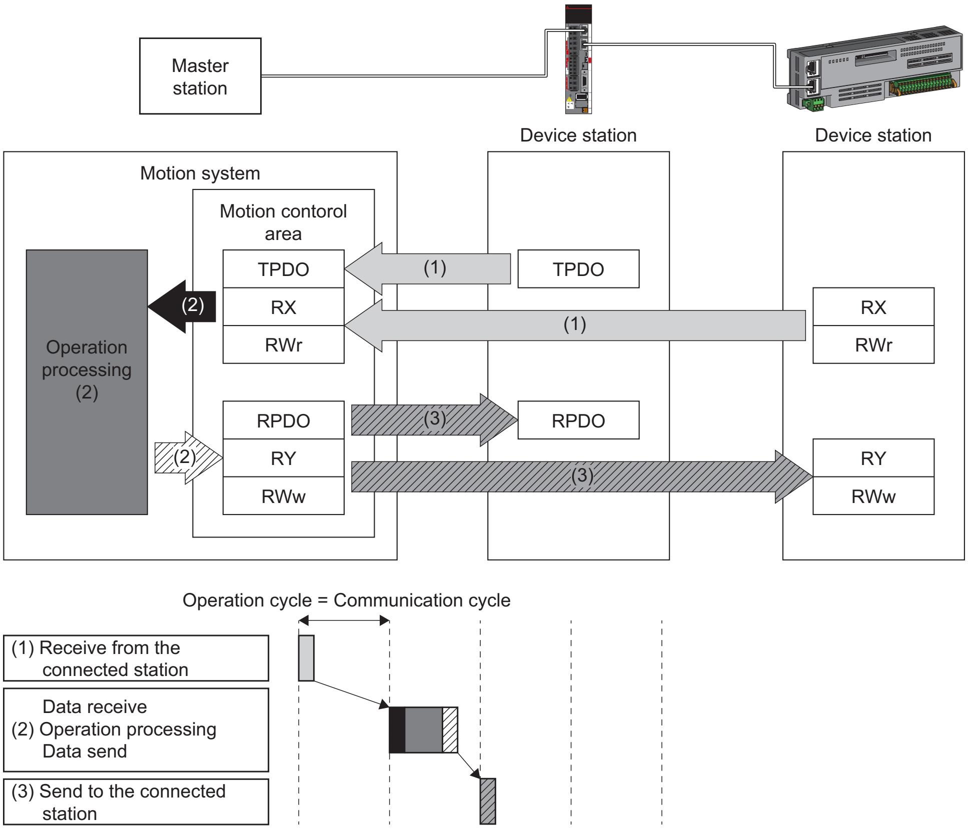
Real drive axis/Real encoder axis executes the refresh at the following timing.
- Receive data (TPDO): (3) Operation processing (reflect data received before two cycles from the connected station)
- Send data (RPDO): (4) Data send (send to the connected station after one cycle)

■ Response preferred method

With response preferred method, data receive, operation processing, and data send are executed in one operation cycle. For the connected device, it takes three cycles to receive a feedback to the data sent from the connected device.

Even though the timing of data update is earlier compared to operation cycle preferred method so that the responsiveness to feedback is high, it takes long for the operation cycle processing time because more processing steps are needed in one operation cycle.

When 31.25 [μs] is selected in the communication cycle setting and the motion synchronization station send/receive data refresh method is set to response preferred method, "Motion Synchronization Station Send/Receive Data Refresh Method Setting Combination Error (error code: 3229H)" will occur.



Point

Slave label executes the refresh at the following timing.

- Receive data (RX, RWr): (2) Data receive/Operation processing/Data send (reflect data received before one cycle from the connected station)
- Send data (RY, RWw): (2) Data receive/Operation processing/Data send (send to the connected station after one cycle)

Real drive axis/Real encoder axis executes the refresh at the following timing.

- Receive data (TPDO): (2) Data receive/Operation processing/Data send (reflect data received before one cycle from the connected station)
- Send data (RPDO): (2) Data receive/Operation processing/Data send (send to the connected station after one cycle)

■ Slave label

The I/O data which the device station of motion synchronization 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 831 Connectable device to CC-Link IE TSN

Standard station

The following stations update the data.

- Programmable controller

For details, refer to "CC-Link IE TSN Network Synchronous Communication Function" in the following manual.

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

Buffer memory refresh

Data which are used to issue the Motion control FB by programmable controller and to get axis monitor information, etc. are refreshed in the buffer memory refresh cycle.

The target of the buffer memory refresh is as follows.

- Area for the Motion control FB
- 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 Buffer Memory Refresh Cycle Setting (System.PrConst.BuffermemoryRefreshCycle).

Axis operation process fast operation mode

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

Axis operation process fast operation mode restrictions

In the fast operation 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 fast operation 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. • Set Acceleration (AxisName.Md.SetAcceleration) • Actual Velocity (AxisName.Md.ActualVelocity) • Actual Position (AxisName.Md.ActualPosition) (For real encoder axis) • Object Data_TargetPos(AxisName.Md.Io_TargetPos) • Object Data_PosActualValue (AxisName.Md.Io_PosActualValue) • Object Data_VelActualValue (AxisName.Md.Io_VelActualValue) • Object Data_TargetVelocity (AxisName.Md.Io_TargetVelocity) • Object Data_TorqueActualValue (AxisName.Md.Io_TorqueActualValue) • Object Data_Statusword (AxisName.Md.Io_Statusword) • Drive Unit Error Detection (AxisName.Md.DriverError) • Drive Unit Error Code (AxisName.Md.DriverErrorID) • Drive Unit Error Detail Code (AxisName.Md.DriverErrorDetailID) • Velocity Override Factor (AxisName.Md.VelocityOverride) • Acceleration Override Factor (AxisName.Md.AccelerationOverride) • Jerk Override Factor (AxisName.Md.JerkOverride) • Positive Direction Torque Limit Value (AxisName.Md.TorqueLimit_Positive) • Negative Direction Torque Limit Value (AxisName.Md.TorqueLimit_Negative) • Driver Ready On Status (AxisName.Md.Driver_ReadyOn) • Driver Servo On Status (AxisName.Md.Driver_ServoOn) • Velocity Override Factor (AxesGroupName.Md.VelocityOverride) • Acceleration Override Factor (AxesGroupName.Md.AccelerationOverride)
4	A part of axis command unusable	The input of the following command will be ignored. • Homing Request Clear (AxisName.Cd.Homing_ClearRequest) • Positive Direction Torque Limit Value (AxisName.Cd.TorqueLimit_Positive) • Negative Direction Torque Limit Value (AxisName.Cd.TorqueLimit_Negative) • Velocity Override Factor (AxisName.Cd.VelocityOverride) • Acceleration Override Factor (AxisName.Cd.AccelerationOverride) • Velocity Override Factor (AxesGroupName.Cd.VelocityOverride) • Acceleration Override Factor (AxesGroupName.Cd.AccelerationOverride) 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 fast operation mode axis.

Precautions

- In the high speed-mode, only the position control can be executed.
  If used by direct control (velocity control, torque control) "Necessary Slave Object Unset (error code: 1AA8H)" occurs.
- The fast operation mode of the axes group becomes valid when axes set to fast operation mode are included in configuration axes.
- Note the description in the following Point.
  ☞ Page 107 Operation cycle settings

Operation cycle 31.25 setting (Only when using 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	AbsSystem	×
2	Axis	○
3	baseSystem	○
4	ExternalSignal	×
5	FileTransfer	×
6	Logging	×
7	MotionControl_AxisFilter	○
8	MotionControl_General	○
9	MotionControl_Sync	○
10	MotionEngine	○
11	MotionEventHist	×
12	NetworkDriver_CCIETSN	○
13	PackagingApp	×
14	PlcInstruction	○
15	ProfileControl	×
16	Program_ST	○
17	ServoDriver_CANopen	○
18	ServoSystemRecorder	×
19	SignalIO	○
20	SimpleMotion	×
21	MotionControl_AdvancedSync	×

■ Parameter setting

Set the axis to be operated by 31.25 [μs] to the fast operation mode. For details, refer to the following.

☞ Page 114 Axis operation process fast operation mode

Point

The system load status can be confirmed by Processing Time (System.Md.OperationCycle[1].ProcessingTime).
Do not let Processing Time (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 usable. 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 system 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

- Syst settingem
  The initial value is applied for the setting value of parameters not listed below.

Item		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

- Axis parameter
  The initial value is applied for the setting value of parameters not listed below.

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

- Axes group parameter
  For the setting value of parameters other than configuration axis, the initial value is applied.

■ Program

Executes the following Motion control FBs by normal execution type program of the motion ST program.

- MCv_AllPower (All Axes Operation Available)
- MC_GroupEnable (Axes Group Enabled)
- MCv_MoveLinearInterpolateRelative (Relative Value Linear Interpolation Control)

Precaution

Combination of the version

- Response preferred method support

Tool name	Add-on name	
Motion control setting function	NetworkDriver_CCIETSN	
	"1.18" or earlier	"1.19" or later
"1.021X" or earlier	—	— (Only operation cycle preferred method can be selected)
"1.030G" or later	Operates as operation cycle preferred method regardless of the setting values of Motion Synchronization Station Send/ Receive Data Refresh Method (System.PrConst.Link_MotionStationRefreshType).	—

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 879 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.

System parameter constant (System.PrConst.)

Variable/Structure name	Name	Details
Addon_AddonLibraryName	Add-on AddonLibraryName Parameter	Sets the maximum capacity for the memory used in Add-on AddonLibraryName parameter (Addon_AddonLibraryName). ADDON_PARAM structure For add-on used for Addon_AddonLibraryName, refer to the "system variables" in the following. 📖MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

System monitor data (System.Md.)

Variable/Structure name	Name	Details
Addon_AddonLibraryName	Add-on AddonLibraryName Monitor	Displays the maximum capacity for the memory used in Add-on AddonLibraryName parameter (Addon_AddonLibraryName). ADDON_MON structure For add-on used for Addon_AddonLibraryName, refer to the "system variables" in the following. 📖MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

ADDON_PARAM

Variable/Structure name	Name	Details
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 system memory (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

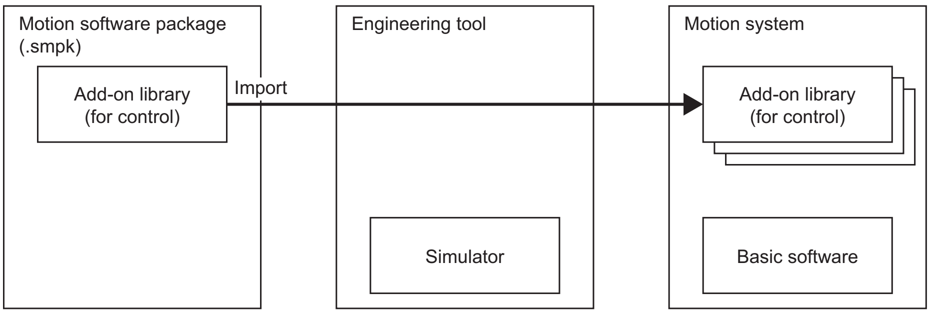
ADDON_MONI

Variable/Structure name	Name	Details
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 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.
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 752 Software file configuration of motion system

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.)

For details on configuration of the motion software package, refer to the following.

☞ Page 752 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 751 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

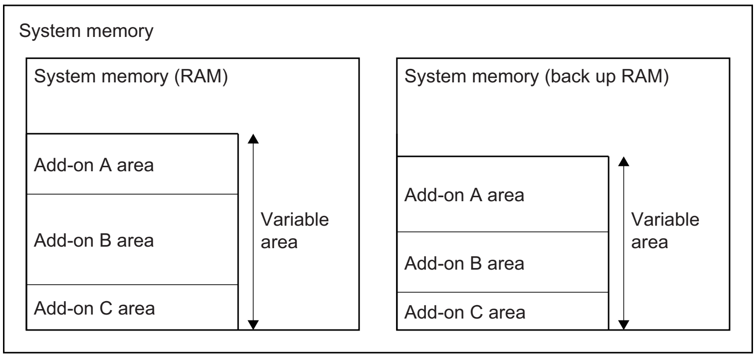
Loading add-on library

Add-on libraries are loaded when the motion system is initialized. When the loading can not be performed because of the followings, "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).



Operation of this function for each system status

○: Possible

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.

System parameter constant (System.PrConst.)

Variable/Structure name	Name	Details
Addon_AddonLibraryName	Add-on AddonLibraryName Parameter	Sets the maximum capacity for the memory used in Add-on AddonLibraryName parameter (Addon_AddonLibraryName). ADDON_PARAM structure For add-on used for Addon_AddonLibraryName, refer to the "system variables" in the following. MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

System monitor data (System.Md.)

Variable/Structure name	Name	Details
Addon_AddonLibraryName	Add-on AddonLibraryName Monitor	Displays the maximum capacity for the memory used in Add-on AddonLibraryName parameter (Addon_AddonLibraryName). ADDON_MONI structure For add-on used for Addon_AddonLibraryName, refer to the "system variables" in the following. MELSEC iQ-R Programming Manual (Motion Control Function Blocks)
MemoryUsage	System Memory Usage	Stores the current usage amount and the maximum usage amount for the system memory (RAM) and system memory (backup RAM). ADDON_MON structure
MemorySize	System Memory Size	Stores the maximum usage amount for the system memory (RAM) and system memory (backup RAM). ADDON_PARAM structure

ADDON_PARAM

Variable/Structure name	Name	Details
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

Variable/Structure name	Name	Details
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 current amount of system memory (backup RAM) used in the add-on in k bytes.
BackupRamUsage	Backup RAM Usage	Stores the maximum 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.
Version	Version	Stores the version information of the add-on.

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	96M bytes	256M bytes

System memory (RAM) usage setting

The amount of system memory (RAM) available in each add-on library is set by Maximum RAM Size (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 usage amount and the maximum usage amount for the system memory (RAM) used in the add-ons can be monitored by the following variables. The value is stored in k bytes.
  - RAM Usage (System.Md.Addon_AddonLibraryName.RamUsage)
  - RAM Usage Maximum (System.Md.Addon_AddonLibraryName.RamMaxUsage)
- The current usage amount and the maximum usage amount for system memory (RAM) used in the all add-ons can be monitored by the following variables. The value is stored in k bytes.
  - RAM Usage (System.Md.MemoryUsage.RamUsage)
  - RAM Usage Maximum (System.Md.MemoryUsage.RamMaxUsage)

■ 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.)

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	232k bytes	472k bytes

3

System memory (backup RAM) usage setting

The amount of system memory (backup RAM) available in each add-on library is set by Maximum Backup RAM Size (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 usage amount or the maximum usage amount for the system memory (RAM) used in add-on can be monitored by the following variables. The value is stored in k bytes.
  - · Backup RAM Usage (System.Md.Addon_AddonLibraryName.BackupRamUsage)
  - · Backup RAM Usage Maximum (System.Md.Addon_AddonLibraryName.BackupRamMaxUsage)
- The current usage amount and the maximum usage amount for system memory (RAM) used in the all add-ons can be monitored by the following variables. The value is stored in k bytes.
  - · Backup RAM Usage (System.Md.MemoryUsage.BackupRamUsage)
  - · Backup RAM Usage Maximum (System.Md.MemoryUsage.BackupRamMaxUsage)

■ Examples of adjustable memory

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

Point

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 request, the system restarts and all data of the system is deleted.
This function is used to fetch 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 Software Reboot Request (System.Cd.SoftRebootRequest).

Action	Software Reboot Request (System.Cd.SoftRebootRequest)	Target	Details
Reset	MOTION_RESET	Motion area	Resets parameters and control data.
Clear	MOTION_CLEAR	Motion area	Clears built-in memory.
Quick clear	MOTION_QCLEAR	Motion area	Clears built-in memory quickly.

Software Reboot Enabled (System.PrConst.SoftReboot_Enable) setting must be allowed(TRUE) to execute the software reboot. If the software reboot is executed with the setting prohibited (FALSE), the software reboot will not be executed and "Software Reboot Disabled Warning (warning code: 0F00H)" will be detected.
Register event history (type: operation power ON/reset) after the software reboot.

Relevant variables

System parameter constant (System.PrConst.)

Variable/Structure name	Name	Details
SoftReboot_Enable	Software Reboot Enabled	Sets the software reboot execution enable/disable. When prohibited, the software reboot is not executed. • FALSE: Disabled • TRUE: Enabled

System control data (System.Cd.)

Variable/Structure name	Name	Details
SoftRebootRequest	Software Reboot Request	Executes the software reboot. • MOTION_RESET: Executes a reset of the motion area only. • MOTION_CLEAR: Executes a clear of the memory after rebooting the motion area. • MOTION_QCLEAR: Executes a quick clear of the memory after rebooting the motion area. • Others: 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
Parameter	Module parameter (Motion/ Network)	Holds parameters during operation	Imports again
	Basic settings	Imports again	
	Control data		
	Global label		
	Global label initial value		
	Program		
	Program component		
Variable/ Program	Label	Sets the default value	Sets the default value
	Device	Clear*1	Clear
	Program	Equivalent to STOP → RUN	Equivalent to STOP → RUN
Network communication		All stations are disconnected and then reconnected after restart	All stations are disconnected and then reconnected after restart
File transmission by script file		Execute	Execute
Base system software/add-on		Reload	Reload
CPU module		Continues the operation	Reset
RAM drive		Clear	Clear
Errors/warnings*2		Clear*3	Clear

*1 The outside data of the reset target is maintained.
*2 Check the event history for the error details before the software reboot.
*3 Moderate errors and major errors cannot be cleared.

- 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 complete a low-level format of the drive. Approximate completion times are shown below.

Motion module	Time
RD78G	Approximately 4 minutes (could take up to 30 minutes depending on the operating environment.)
RD78GH	Approximately 9 minutes (could take up to 60 minutes depending on the operating environment.)

- 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 OFF.
- When data reset by the software reboot is executed, the variables and status of the device station 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.
- When software reboot is executed while monitoring the engineering tool, incorrect variable values may appear on the monitor. In such a case, stop the monitor once after software reboot completion and restart.
- Do not turn OFF the power of the motion system while clear or quick clear is executed. "Drive Error (error code: 3207H)" may occur along with format failure of the drive (only for add-on baseSystem version "1.5" or later). In such a case, execute clear again.

PART 2 MOTION CONTROL

This part consists of the following chapters.

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 FBs 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 Axis Status (AxisName.Md.AxisStatus) of "4: Standby (Standstill)" or Axes Group Status (AxesGroupName.Md.GroupStatus) of "4: Standby (GroupStandby)" is called "start" of the axis.
Function blocks have Execute command (Execute) type and Enable (Enable) type, and these are started at the rising edge of Execute command (Execute) or Enable (Enable) which are input variables of the Motion control FB.

Operation of this function for each system status

○: Possible, ×: Not possible

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

Relevant variables

Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
AxisStatus	Axis Status	Displays the axis status. • 0: Axis disabled (Disabled) • 1: Stopping on error (ErrorStop) • 2: Decelerating to stop (Stopping) • 3: During homing (Homing) • 4: Standby (Standstill) • 5: During positioning operation (DiscreteMotion) • 6: During continuous operation (ContinuousMotion) • 7: During synchronous operation (SynchronizedMotion)
UseInGroup	Using Axes Group	Displays whether the axis is being used in the axes group or not. • FALSE: Not used • TRUE: Used
Homing_Request	Homing Request	Displays whether homing is required or not. Becomes FALSE when the homing is completed. • FALSE: No homing request • TRUE: Homing request
HwStrokeLimit_FlsStatus	Upper Limit Signal Status	Displays the detection status of the upper limit signal (FLS). • FALSE: Detecting the upper limit signal (FLS)} • TRUE: Not detecting the upper limit signal (FLS)
HwStrokeLimit_RlsStatus	Lower Limit Signal Status	Displays the detection status of the lower limit signal (RLS). • FALSE: Detecting the lower limit signal (RLS) • TRUE: Not detecting the lower limit signal (RLS)
ForcedStop_Released	Forced Stop Cancelling	Displays the forced stop cancel status. • FALSE: Forced stop • TRUE: Forced stop release
StopStatus	Stop Status	Displays the input status of the stop signal (STOP). • FALSE: Stop processing released • TRUE: During stop processing

Axes group monitor data (AxesGroupName.Md.)

Variable/Structure name	Name	Details
GroupStatus	Axes Group Status	Displays the axes group status • 0: Axes group disabled (GroupDisabled) • 1: Stopping on error (GroupErrorStop) • 2: Decelerating to stop (GroupStopping) • 4: Standby (GroupStandby) • 5: Operating (GroupMoving)

System monitor data (System.Md.)

Variable/Structure name	Name	Details
Ready	Ready	Displays the ON/OFF status of the READY. • FALSE: OFF (Not READY) • TRUE: ON (READY)
Sync	Synchronization flag	Displays the ON/OFF status of the synchronization flag. • FALSE: OFF (Module access disabled) • TRUE: ON (Module access enabled)

System control data (System.Cd.)

Variable/Structure name	Name	Details
SequenceReady	PLC Ready	Specifies RUN/STOP of PLC READY. • FALSE: STOP • TRUE: RUN *: Becomes TRUE/FALSE automatically by rising/falling of the Motion module.

Point

Note the following when switching STOP/RUN by operating PLC Ready (System.Cd.SequenceReady).

- When setting PLC Ready (System.Cd.SequenceReady) to the public label, do not operate PLC Ready 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 Motion control FB 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. "CPU operating status of own station (SB004C)" or "CPU status of own station (SW004B)" shows the CPU operating status. For "CPU operating status of own station (SB004C)" and "CPU status of own station (SW004B)", 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 Status	4: Standby (Standstill)	AxisName.Md.AxisStatus

■ Signal state

Signal name		Signal state		Variable name
I/O signal	PLC Ready	ON	RUN	System.Cd.SequenceReady
	Ready	ON	Ready	System.Md.Ready
	Synchronization flag	ON	Module access enabled	System.Md.Sync
	Using 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 Status	4: Standby (Standstill)	AxisName.Md.AxisStatus
Axes Group Status	4: Standby (GroupStandby)	AxesGroupName.Md.AxesGroupStatus

■ Signal state

Signal name		Signal state		Variable name
I/O signal	PLC Ready	ON	RUN	System.Cd.SequenceReady
	Ready	ON	Ready	System.Md.Ready
	Synchronization flag	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 command (Execute) again)
- Continuous update (ContinuousUpdate)

With a retrigger/continuous update of the FB, input variables can be imported again to the on-going FB instance 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 command (Execute) becomes FALSE to TRUE and when Continuous update (ContinuousUpdate) becomes 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 Motion control FB, 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 Analyzing (AxisName.Md.Analyzing) is TRUE
  - · Immediately before termination: Immediately before Execution completion (Done) becomes 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 command (Execute), a retrigger can be executed by the re-rising edge of Execute command (Execute).

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

Continuous update

If Continuous update (ContinuousUpdate) is TRUE when Execute command (Execute) changes 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)

Motion control FBs can be executed continuously without stopping multiple Motion control FBs 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 Motion control FB with the axis and the axes group stop status ("4: Standby (Standstill)" or "4: Standby (GroupStandby)") is called "start". For details, refer to the following.
  ☞ Page 128 Start
- When Axis Status (AxisName.Md.AxisStatus) and Axes Group Status (AxesGroupName.Md.GroupStatus) are as the following, to execute the motion FB of another instance is called "multiple start".
  [Axis Status (AxisName.Md.AxisStatus) that can be multiple started]
  · 3: During homing (Homing) (Only MC_Stop (Forced Stop) is possible)
  · 5: During positioning operation (DiscreteMotion)
  · 6: During continuous operation (ContinuousMotion)
  · 7: During synchronous operation (SynchronizedMotion)
  [Axes Group Status (AxesGroupName.Md.GroupStatus) that can be multiple started]
  · 5: Operating (GroupMoving)
- Multiple start of the single axis control FB cannot be executed to an axis operated in the axes group. It will cause "Motion 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	Reference	
			During single axis operation	During multiple axes operation
0: mcAborting	Aborting	Aborts (cancels) the under-control FB and executes the next FB immediately.	☞ Page 140 Aborting during single axis operation	☞ Page 147 Aborting during multiple axes operation
1: mcBuffered	Buffered	Buffers the next FB on the under-control FB. If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after completion of the under-control FB.	☞ Page 141 Buffered during single axis operation	☞ Page 148 Buffered during multiple axes operation
2: mcBlendingLow	BlendingLow	Buffers the next FB on the under-control FB.*1 If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. The lower target velocity between the under-control FB and the buffering FB is used as the switching speed.	☞ Page 143 BlendingLow	☞ Page 150 BlendingLow
3: mcBlendingPrevious	BlendingPrevious	Buffers the next FB on the under-control FB.*1 If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. The switching speed changes to the target velocity of the under-control FB.	☞ Page 142 BlendingPrevious	☞ Page 149 BlendingPrevious
4: mcBlendingNext	BlendingNext	Buffers the next FB on the under-control FB.*1 If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. The switching speed changes to the target velocity of the buffering FB.	☞ Page 142 BlendingNext	☞ Page 149 BlendingNext
5: mcBlendingHigh	BlendingHigh	Buffers the next FB on the under-control FB.*1 If the under-control FB already has an FB buffering on it, subsequent FBs are buffered consecutively. (Up to 2.) Buffering FBs are executed in order after the under-control FB reaches the target position. The switching speed changes to the higher target velocity value between the under-control FB and the buffering FB.	☞ Page 143 BlendingHigh	☞ Page 150 BlendingHigh

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

Point

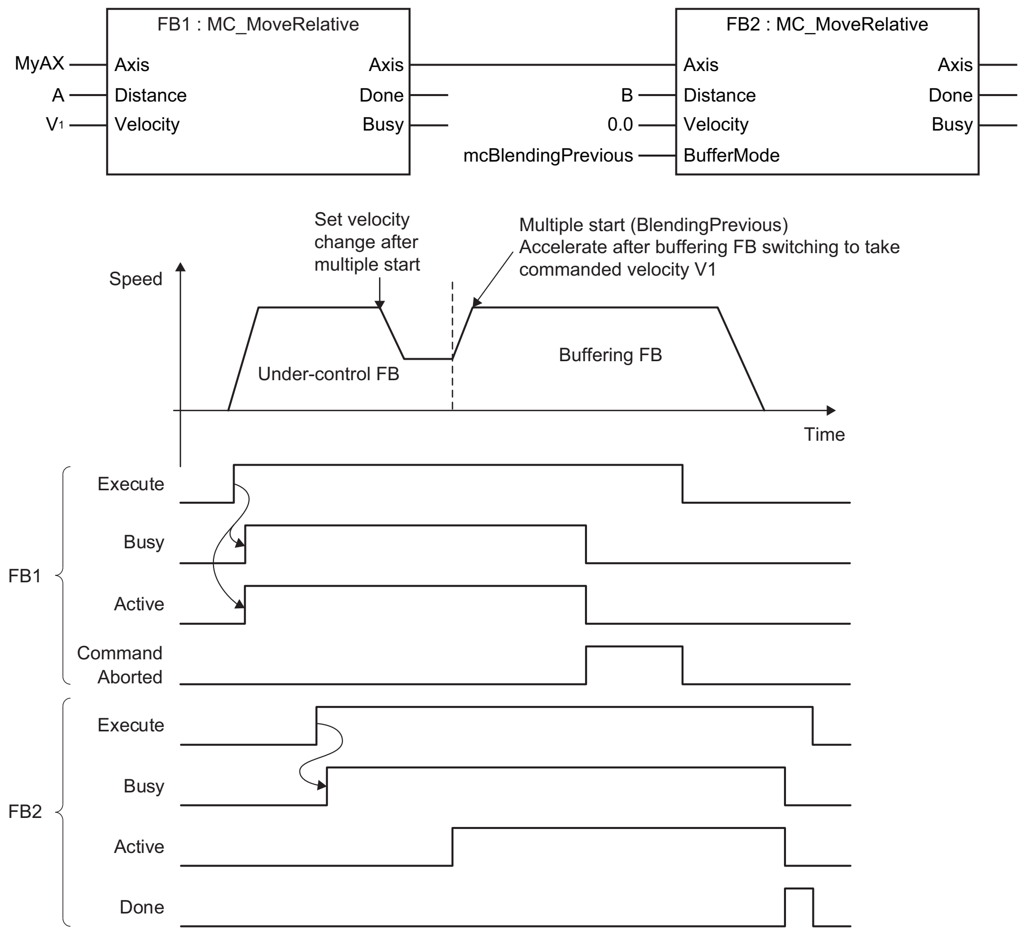
- Up to two motion FBs can be buffered after multiple start in one axis and an axes group. If multiple start is executed when two FBs have already been multiple started, "Warning Starting over Number of Buffering FBs (warning code: 0D22H)" occurs and waits the analysis of the buffering FB until under-control 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 765 Checking Errors and Warnings
When an error or a stop cause occurred in the under-control FB, FBs which are waiting for the analysis are interrupted.
- When "Warning starting over number of buffering FBs (warning code: 0D22H)" occurs, note that not to execute the multiple start until the under-control 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 the under-control 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 in the under-control FB, all buffering FBs are interrupted (the output of Abortion of execution (CommandAborted) becomes TRUE).

Command omission at multiple start

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

Point

When buffering with velocity 0.0 using Blending buffer mode type, the target velocity may not be reached when the velocity of the previous FB becomes 0.0 (switching speed), and the FB may not switch. When buffering with velocity 0.0, use Buffered buffer mode type.



- 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 317 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 (Executing (Busy) is TRUE).

Relevant variables

Axis parameter (AxisName.Pr.)

Variable/Structure name	Name	Details
OverrunOperation	Operation Setting at Overrun	Sets the operation settings when overrun occurs during operation. • 1: Immediate Stop (ImmediateStop) • 2: Keep Current Acceleration/Deceleration (KeepCurrentAcc)

Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
BufferingFBs	Number of Buffering FBs	Displays the number of buffering FBs (0 to 2)

Axes group parameter (AxesGroupName.Pr.)

Variable/Structure name	Name	Details
OverrunOperation	Operation Setting at Overrun	Sets the operation settings when overrun occurs during operation. • 1: Immediate Stop (ImmediateStop)

Axes group monitor data (AxesGroupName.Md.)

Variable/Structure name	Name	Details
BufferingFBs	Number of Buffering FBs	Displays the number of buffering FBs (0 to 2)

Multiple start during single axis operation

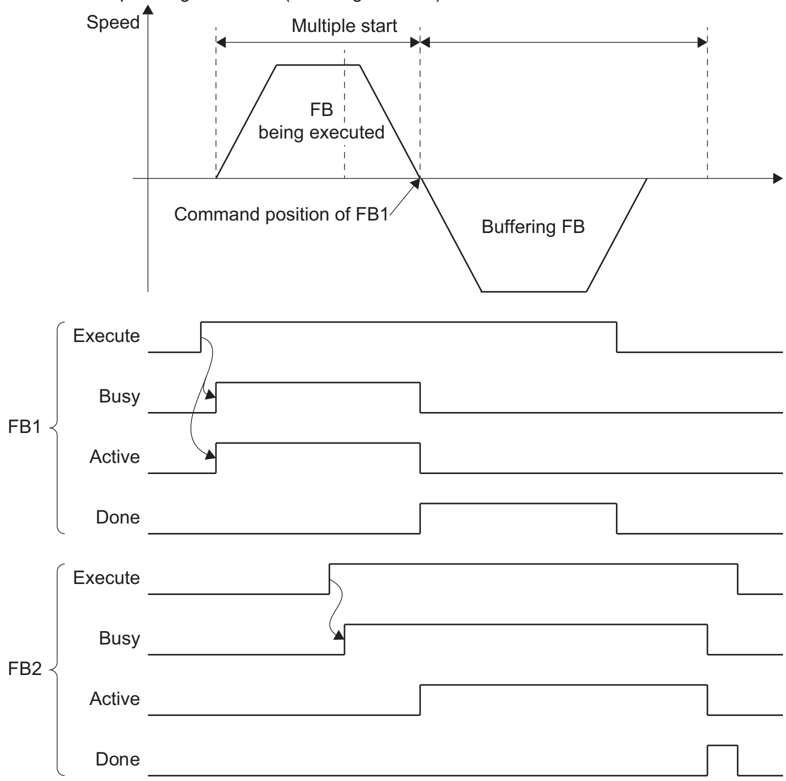
- When axis operation state is as the following, multiple start is enabled by starting a motion FB of another instance. (If the axis operation is in the "4: Standby (Standstill)" status, it is regarded as "start".)

■Axis Status (AxisName.Md.AxisStatus) that can be multiple started

  - 3: During homing (Homing) (Only MC_Stop (Forced Stop) is possible)
  - 5: During positioning operation (DiscreteMotion)
  - 6: During continuous operation (ContinuousMotion)
  - 7: During synchronous operation (SynchronizedMotion)

- When the buffering FB input includes Direction selection (Direction) and "4: Current Direction (mcCurrentDirection)" is selected, the previous FB operation direction is taken over.
- When the directions are different between the under-control FB and next FB in the single operation, the operation differs depending on the selection in case of having Reverse rotation permission selection (Options bit5). The operation is the same as the operation at reverse rotation permission in case of not having Reverse rotation permission selection (Options bit5).
- When the direction is reversed at while Blending is specified, a deceleration stop is executed at the target position of the under-control FB, and velocity change is executed to the target velocity of the buffering FB after the under-control FB is switched to the buffering FB. (It is the same operation as Buffered is specified.)

Deceleration stop setting at reverse (BlendingPrevious)

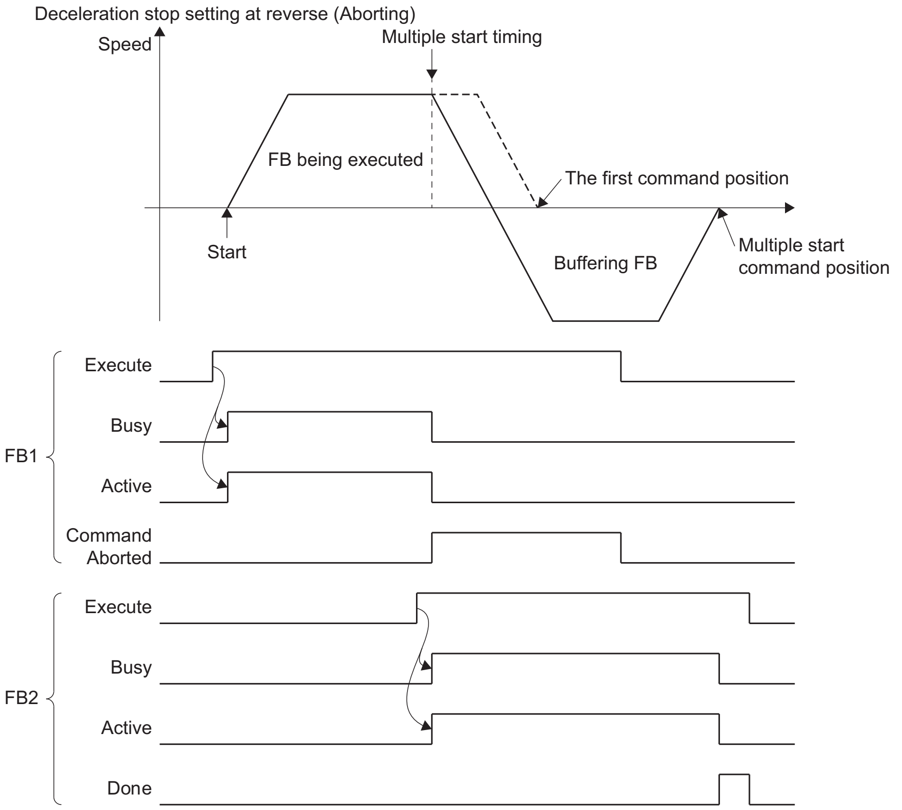


4

Operations with Reverse rotation permission selection (Options (Options): Bit 5)

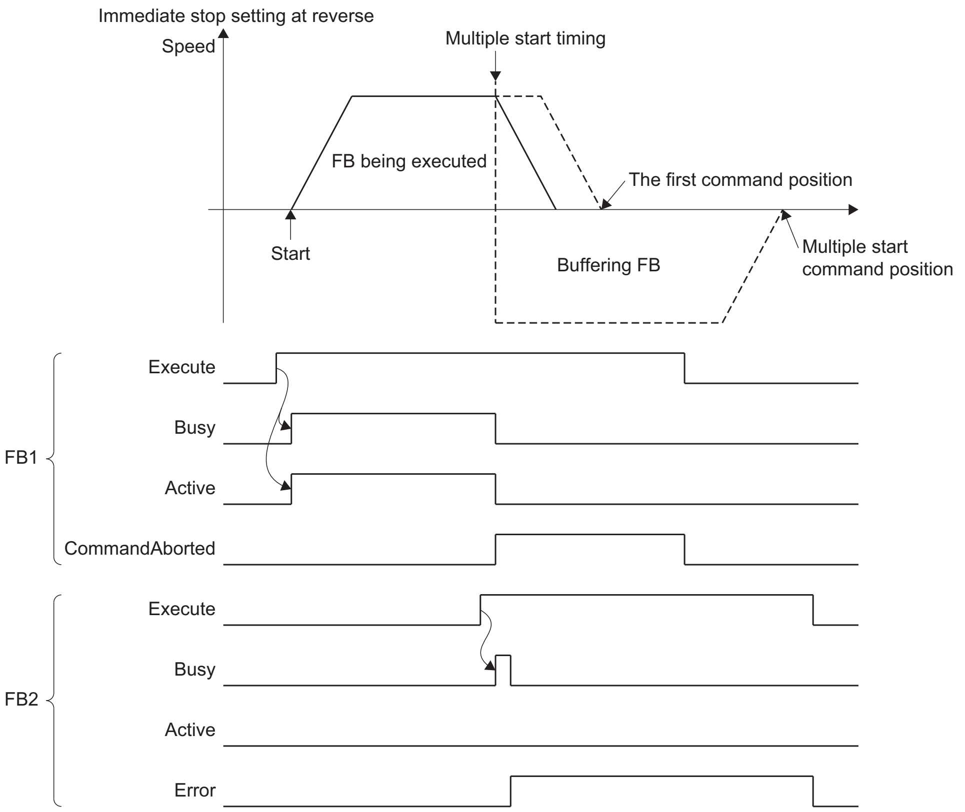
■ When reverse is enabled

When reverse rotation permission (0: Allow) is set in Reverse rotation permission selection (Options (Options): Bit 5), a deceleration stop is executed once.



■ When reverse is disabled

When reverse rotation no permission (1: Do not allow) is set in Reverse rotation permission selection (Options (Options): Bit 5), the error "Overrun Error (error code: 1A7EH)" occurs at switching and the axis decelerates to a stop.



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 motion FB		
	Aborting	Buffered	Blending		Aborting	Buffered	Blending
MC_Home (Homing)	—*1	—	—	Execution completion (Done)	○*2	—	—
MC_Stop (Forced Stop)	○	—	—	Execution completion (Done)	○*2	—	—
MC_MoveAbsolute (Absolute Value Positioning)	○	○	○	Execution completion (Done)	○	○	○
MC_MoveRelative (Relative Value Positioning)	○	○	○	Execution completion (Done)	○	○	○
MCv_Jog (JOG Operation)	—*1	—	—	Execution completion (Done)	○	○	—
MCv_SpeedControl (Speed Control (Including Position Loop))	○	○	○	Target velocity reached (InVelocity)	○	○	—
MC_MoveVelocity (Speed Control)	○	○	—	Target velocity reached (InVelocity)	○	○	—
MC_TorqueControl (Torque Control)	○	○	—	Target torque reached (InTorque)	○	○	—
MC_CamIn (Cam Operation Start)	○	○	—	Cam cycle completion (EndOfProfile)	○	○	—
MC_GearIn (Gear Operation Start)	○	○	—	Gear ratio reached (InGear)	○	○	—
MC_CombineAxes (Addition/ Subtraction Positioning)	○	○	—	In synchronization (InSync)	○	○	—
MCv_SmoothingFilter (Smoothing Filter)	—*1	—	—	—	○*2	—	—
MCv_DirectionFilter (Moving Direction Restriction Filter)	—*1	—	—	—	○*2	—	—
MCv_SpeedLimitFilter (Speed Limit Filter)	—*1	—	—	—	○*2	—	—
MCv_BacklashCompensationFilter (Backlash Compensation)	—*1	—	—	—	○*2	—	—
MCv_AdvancedSync (Advanced Synchronous Control)	—*1	—	—	Cam cycle completion (CycleZeroPoint)	○	○	—

*1 Multiple start can not be executed because this FB can be started only when Axis Status (AxisName.Md.AxisStatus) is "4: Standby (Standstill)".

*2 Only MC_Stop (Forced Stop) is possible.

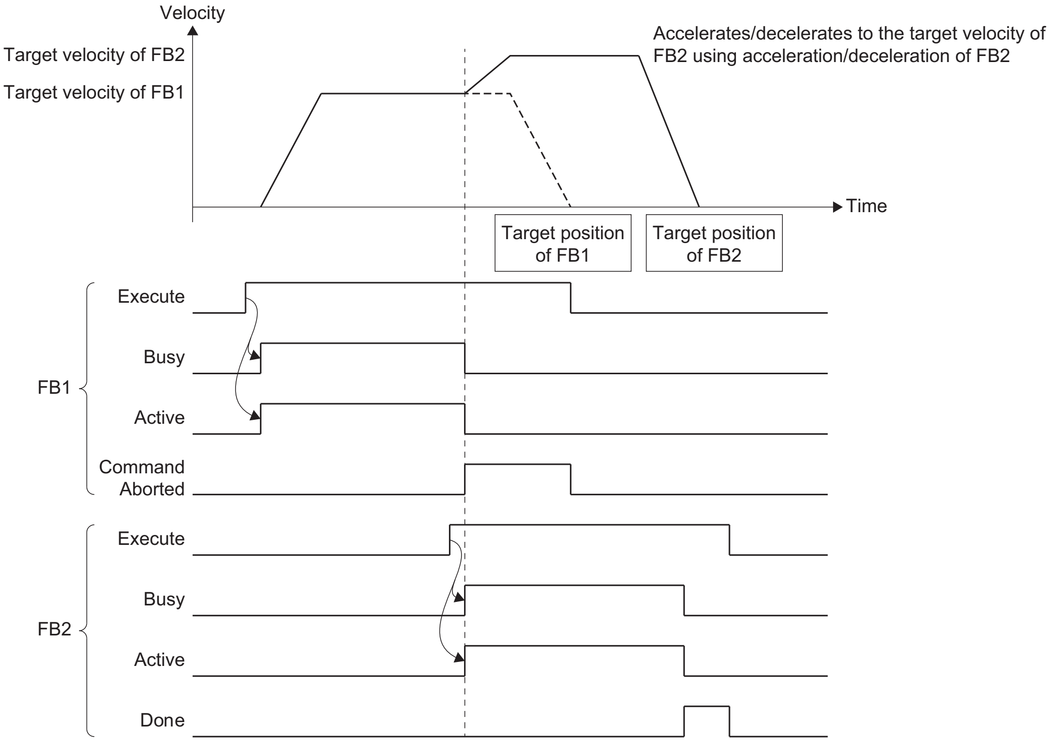
Aborting during single axis operation

The under-control FB is interrupted and the next FB is executed.

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

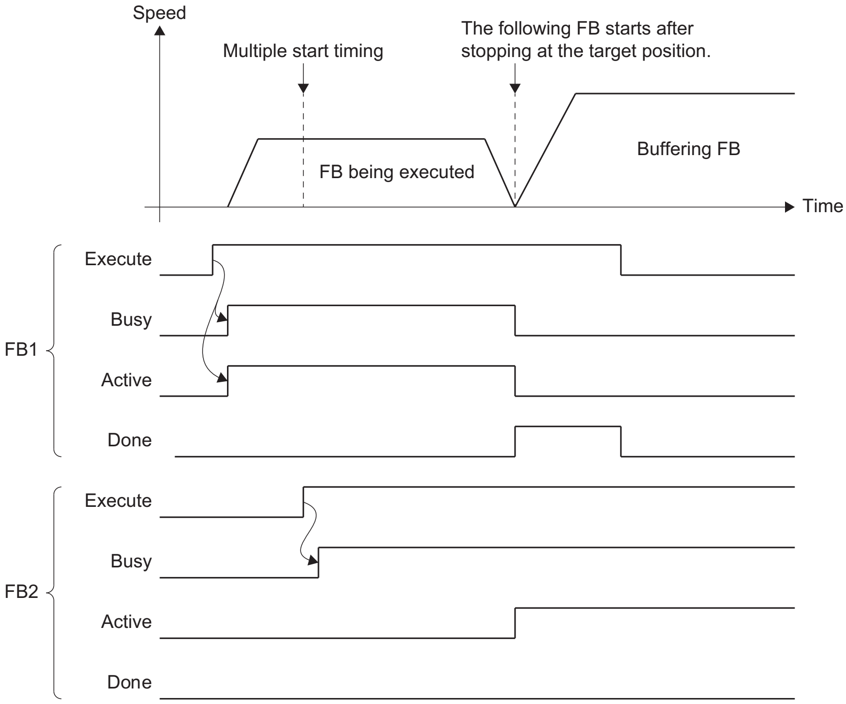
When omitting to specify the buffer mode, this operation is performed.

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



Buffered during single axis operation

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



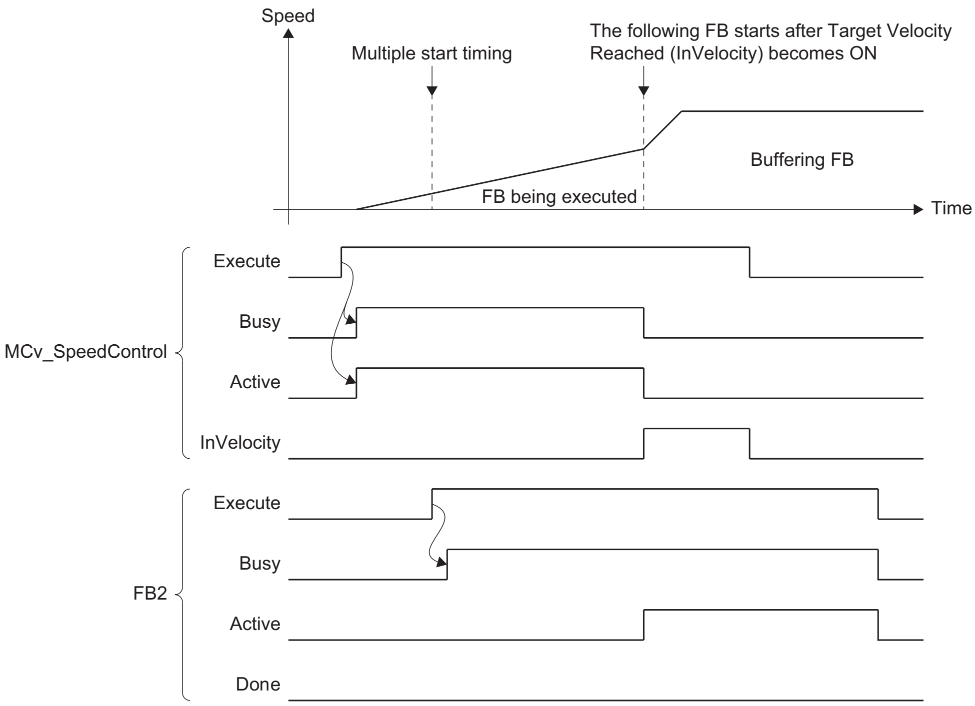
Conditions to determine completion are different depending on the Motion control FB, and the FB may not be in the stop status with speed 0 at completion. For details, refer to the following.

☞ Page 137 Multiple start during single axis operation

Ex.

When an FB is multiple started with Buffered in MCv_SpeedControl (Speed Control (Including Position Loop))

In MCv_SpeedControl (Speed Control (Including Position Loop)), the FB is determined that it is completed when Target velocity reached (InVelocity) becomes TRUE, and the following buffering FB is executed. (The current speed at this time is used as the switching speed.)

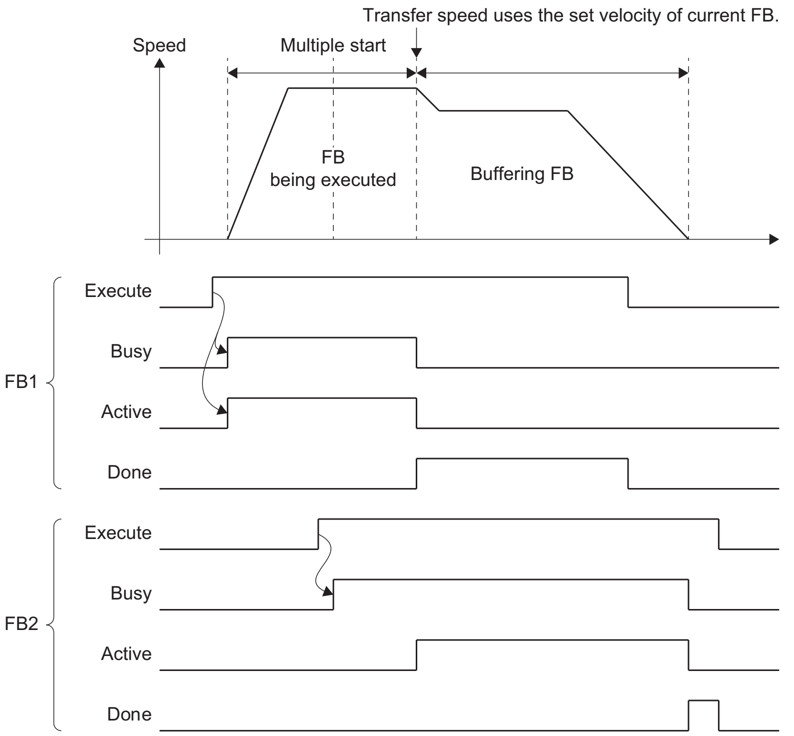


Blending during single axis operation

Multiple started FBs are buffered and wait until the under-control FB is completed. After the under-control 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

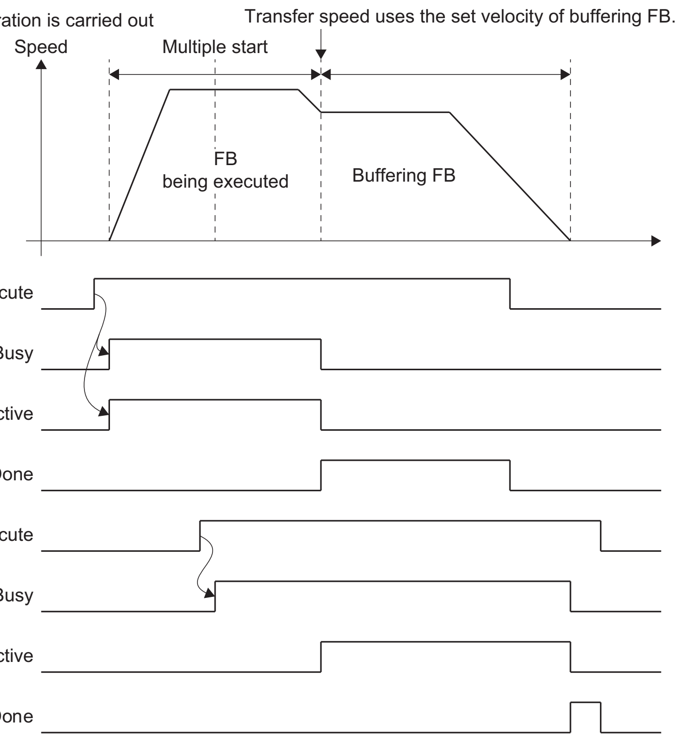
The FB is operated at the speed of the current under-control FB to the target position of the under-control FB. After switching to the FB that is during buffering, velocity change is executed to the target velocity of the buffering FB.



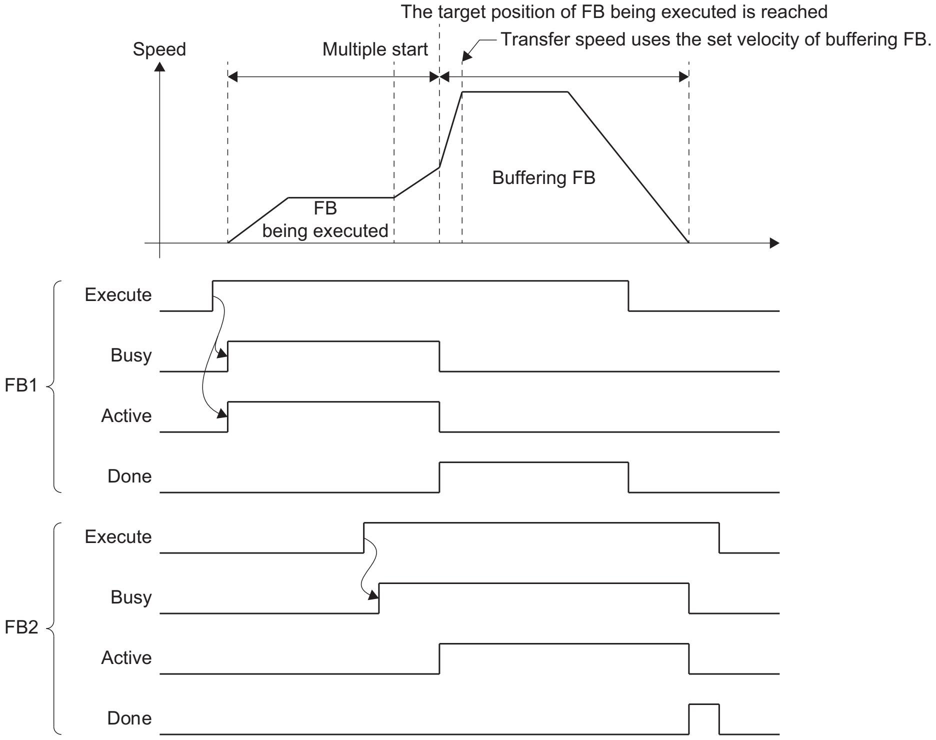
■ BlendingNext

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

When deceleration is carried out



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 under-control 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 under-control FB. If the target velocity of an under-control FB is reached during acceleration/deceleration, control is changed by the acceleration/deceleration of the buffering FB at that time.



■ BlendingLow

The slower target velocity between the under-control FB and the buffering FB is used as the switching speed.
The operation is same as BlendingPrevious when the under-control 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 under-control FB and the buffering FB is used as the switching speed.
The operation is same as BlendingPrevious when the under-control 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

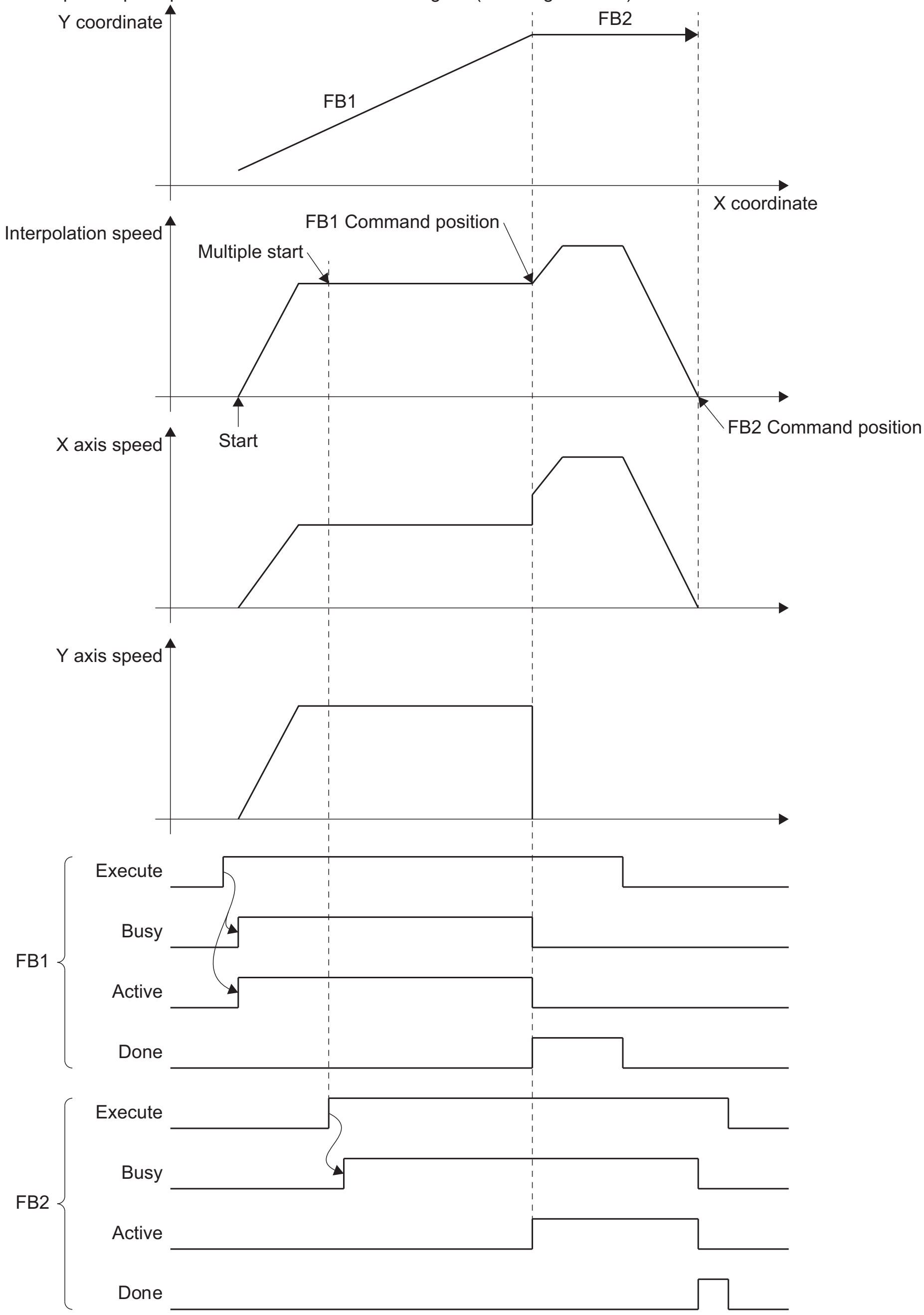
- When axes group operation state is as the following, multiple start is enabled by executing the motion FB of another instance. (If the axis operation is in the "4: Standby (GroupStandby)" status, it is regarded as "start".)

■Axes Group Status (AxesGroupName.Md.GroupStatus) that can be multiple started

- 5: Operating (GroupMoving)

- In multiple start during multiple axes operation, the current interpolation speed of the under-control 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.

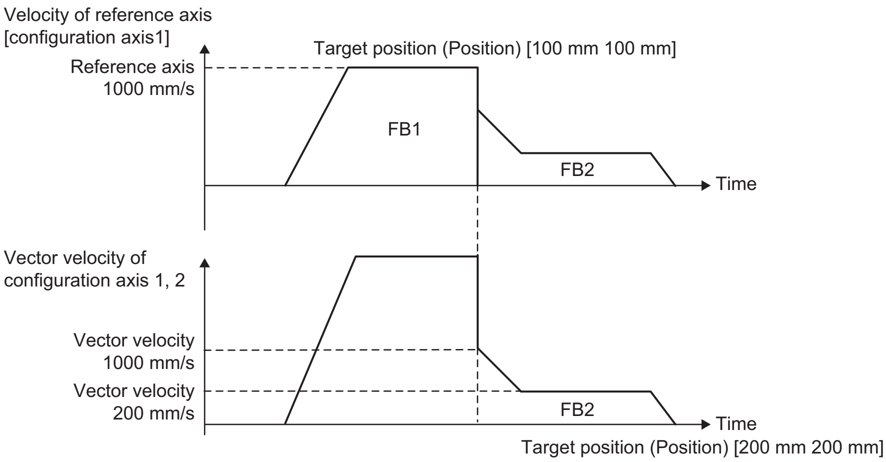
Example of speed pattern for each axis at switching FB (BlendingPrevious)



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

Ex.

When multiple start by specifying BlendingPrevious from FB1 with the reference axis velocity to FB2 with the vector velocity.



The operation is carried out so that the reference axis velocity of FB1 becomes the velocity set in Velocity (Velocity) (reference axis velocity 1000 [mm/s]). The reference axis velocity at the target position of FB1 is regarded as the switching speed and the current vector velocity value of FB2 without conversion, and it accelerates and decelerates the vector velocity (200 [mm/s]) set in Velocity (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, "Motion 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 motion FB		
	Aborting	Buffered	Blending		Aborting	Buffered	Blending
MC_GroupStop (Group Forced Stop)	—*1	—	—	Execution completion (Done)	○*2	—	—
MCv_MoveLinearInterpolateAbsolute (Absolute Value Linear Interpolation Control)	○	○	○	Execution completion (Done)	○	○	○
MCv_MoveLinearInterpolateRelative (Relative Value Linear Interpolation Control)	○	○	○	Execution completion (Done)	○	○	○
MCv_MoveCircularInterpolateAbsolute (Absolute Value Circular Interpolation Control)	○	○	○	Execution completion (Done)	○	○	○
MCv_MoveCircularInterpolateRelative (Relative Value Circular Interpolation Control)	○	○	○	Execution completion (Done)	○	○	○
MCv_MovePositioningData (Multiple Axes Positioning Data Operation)	○	○	—	Execution completion (Done)	○	—	—

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

*2 Only MC_GroupStop (Group Forced Stop) is possible.

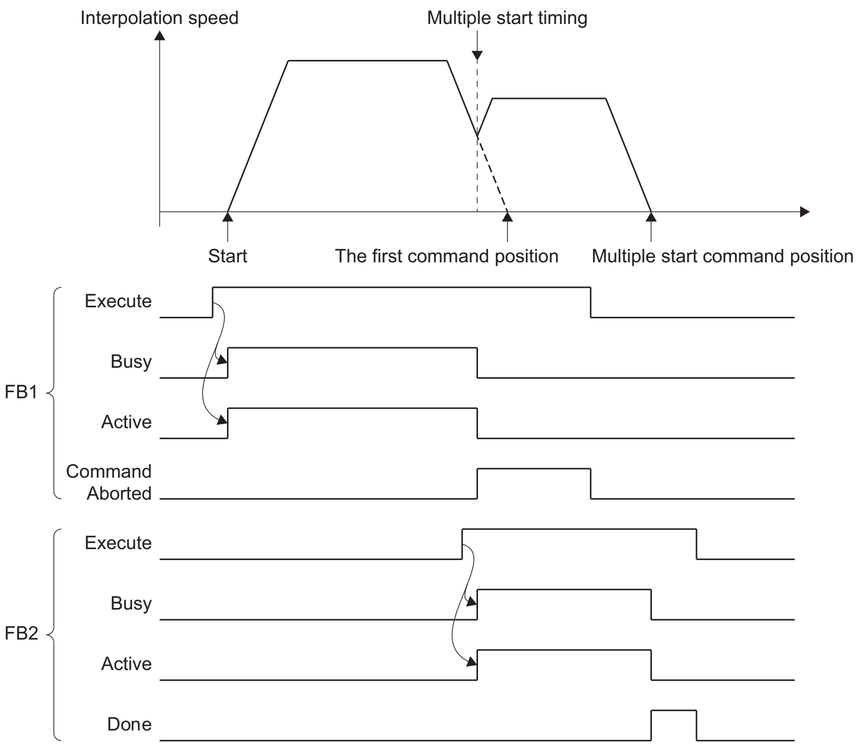
Aborting during multiple axes operation

The under-control 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 under-control 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 under-control FB1 with the linked start FB2 is shown below.

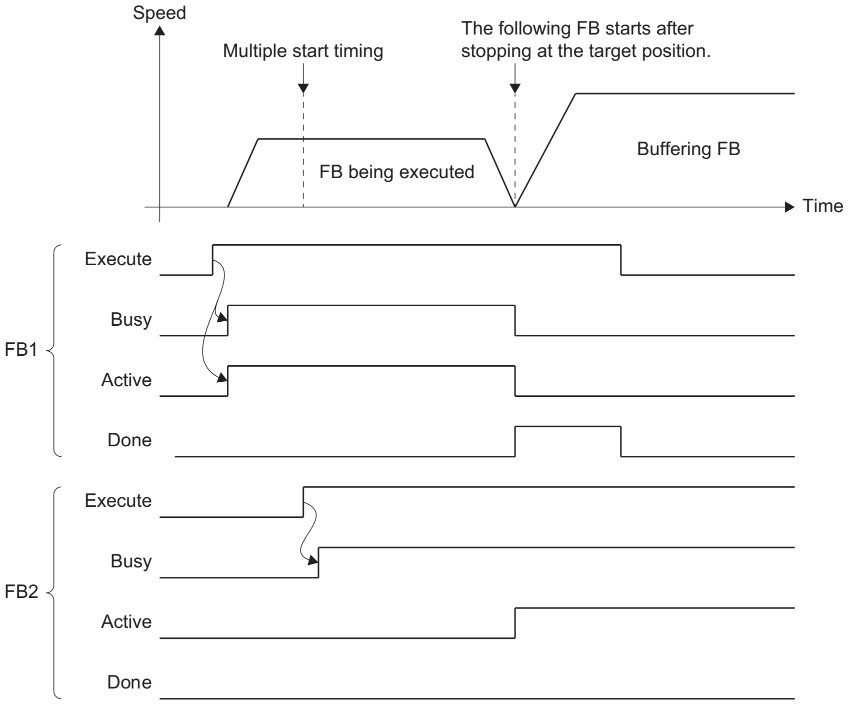


Point

For multiple start during multiple axes operation, when an axes group error such as "Software Stroke Limit Over (Target Position) (error code: 1A00H)" is detected at the same time as the switching of an FB, it may immediately stop regardless of the axes group Stop Selection at Stop Cause Occurrence (AxisGroupName.Pr.StopMode_General) setting.

Buffered during multiple axes operation

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

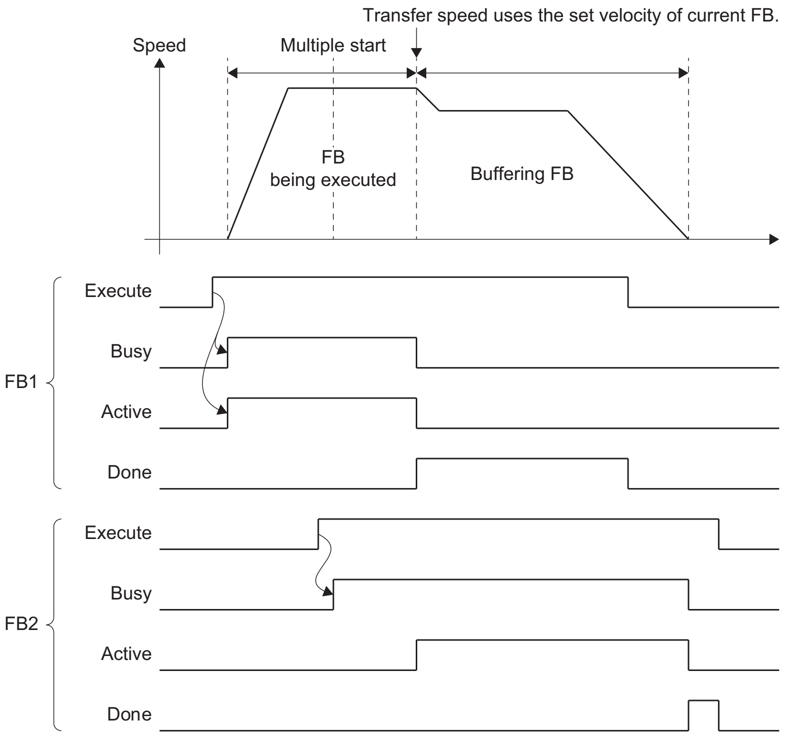


Blending during multiple axes operation

Multiple started FBs are buffered and wait until the under-control FB is completed. After the under-control 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.

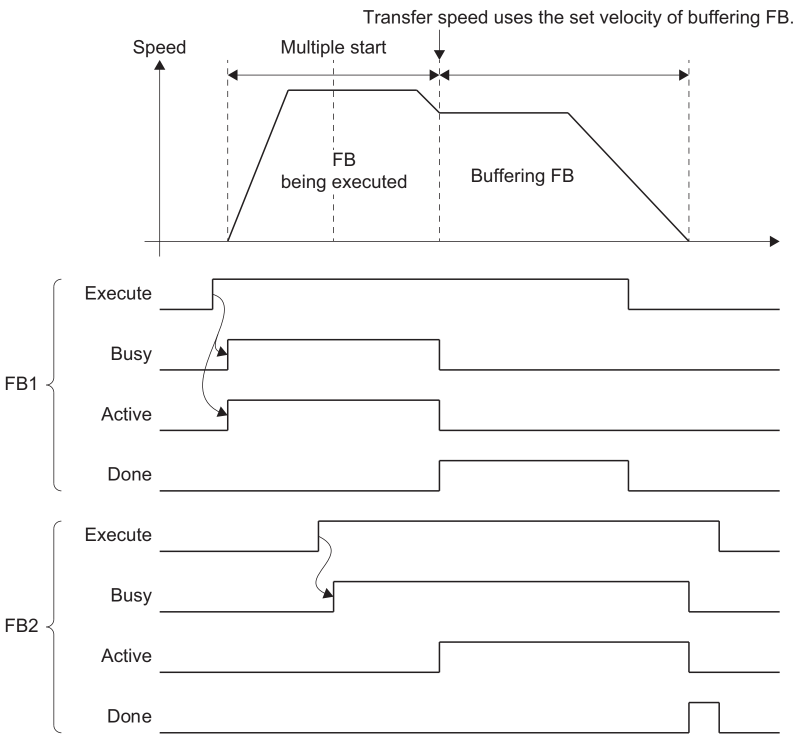
■ BlendingPrevious

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

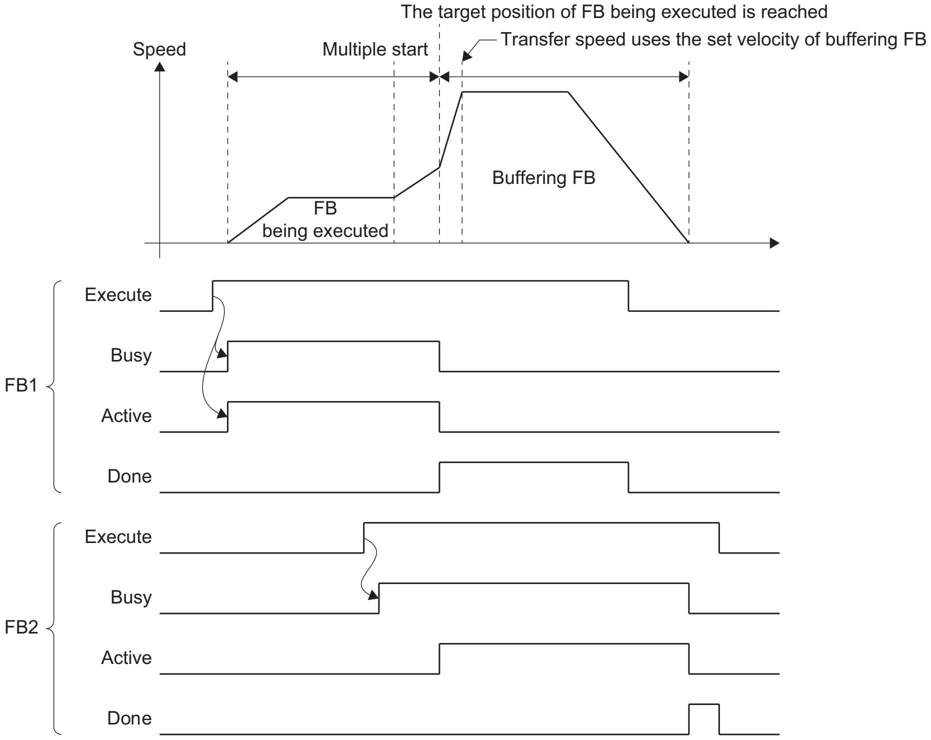


■ BlendingNext

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 under-control 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 under-control 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 under-control FB.
If the target velocity of an under-control FB is reached during acceleration/deceleration, control is changed by the acceleration/deceleration of the buffering FB at that time.



■ BlendingLow

The smaller target velocity between the under-control FB and the buffering FB is used as the switching velocity.
When the smaller switching velocity is the under-control FB, the control is the same as BlendingPrevious, and when the smaller switching velocity is the buffering FB, the control is the same as BlendingNext.

■ BlendingHigh

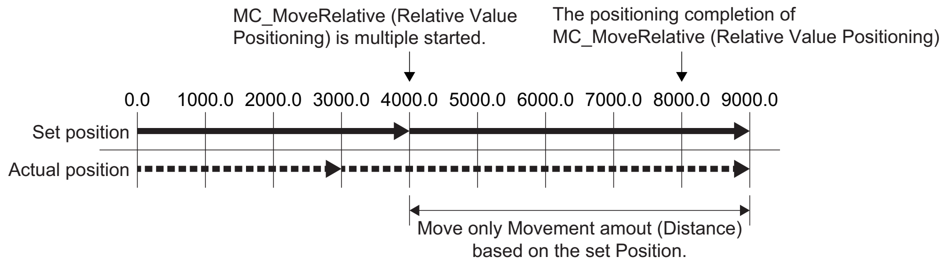
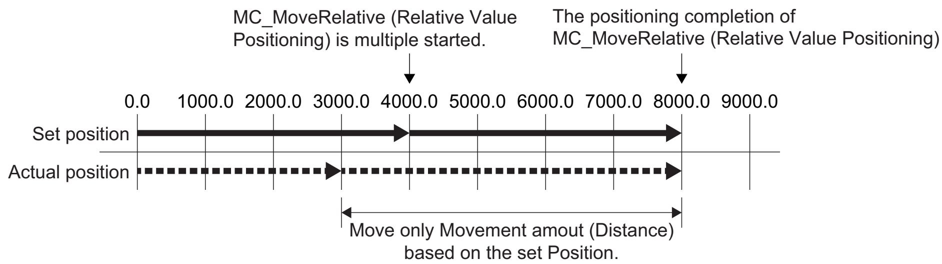
The larger target velocity between the under-control FB and the buffering FB is used as the switching velocity.
When the larger switching velocity is the under-control FB, the control is the same as BlendingPrevious, and when the larger switching velocity is the buffering FB, the control is the same as BlendingNext.

Option of buffering FB

Set the option of the function used for the buffering FB by Options (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 (Options): Bit 3)

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

<table>
<tr><th>Setting value</th><th>Description</th></tr>
<tr><td>0: Set position</td><td>The relative value positioning control from set position
<Example>
Multiple start with movement amount (Distance) set to "5000.0" and Options (Options) set to "00000000H (bit 3 is 0: Set position)"

</td></tr>
<tr><td>1: Actual position</td><td>The relative value positioning control from actual position
<Example>
Multiple start with movement amount (Distance) set to "5000.0" and Options (Options) set to "00000008H (bit 3 is 1: Actual position)"

</td></tr>
</table>

Reverse rotation permission selection (Options (Options): Bit 5)

In the single axis motion FB, reverse rotation is allowed when the operation direction differs between the under-control FB and the buffering FB. For details, refer to the following.
☞ Page 137 Multiple start during single axis operation

Setting value	Description
0: Allow	Reverse rotation is allowed. Performs a deceleration stop once, and starts operation in the changed direction after deceleration stop is completed.
1: Do not allow	Reverse rotation is not allowed. While switching, "Overrun Error (error code: 1A7EH)" occurs and a deceleration stop is performed.

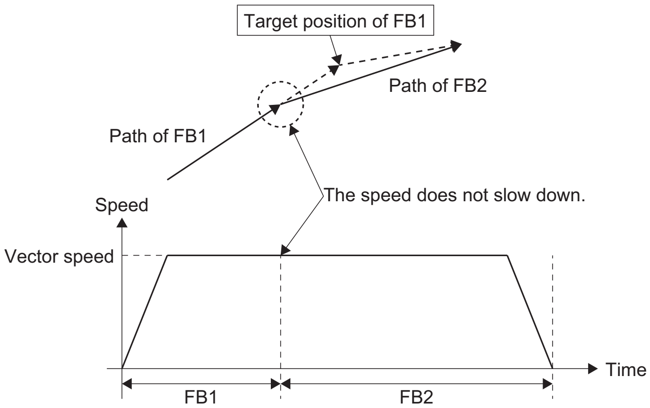
Near pass

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)



Precautions

- When executing multiple start by Blending, the set velocity will not reach the target velocity if the movement amount of the previous FB is small.

Combination with retrigger/continuous update

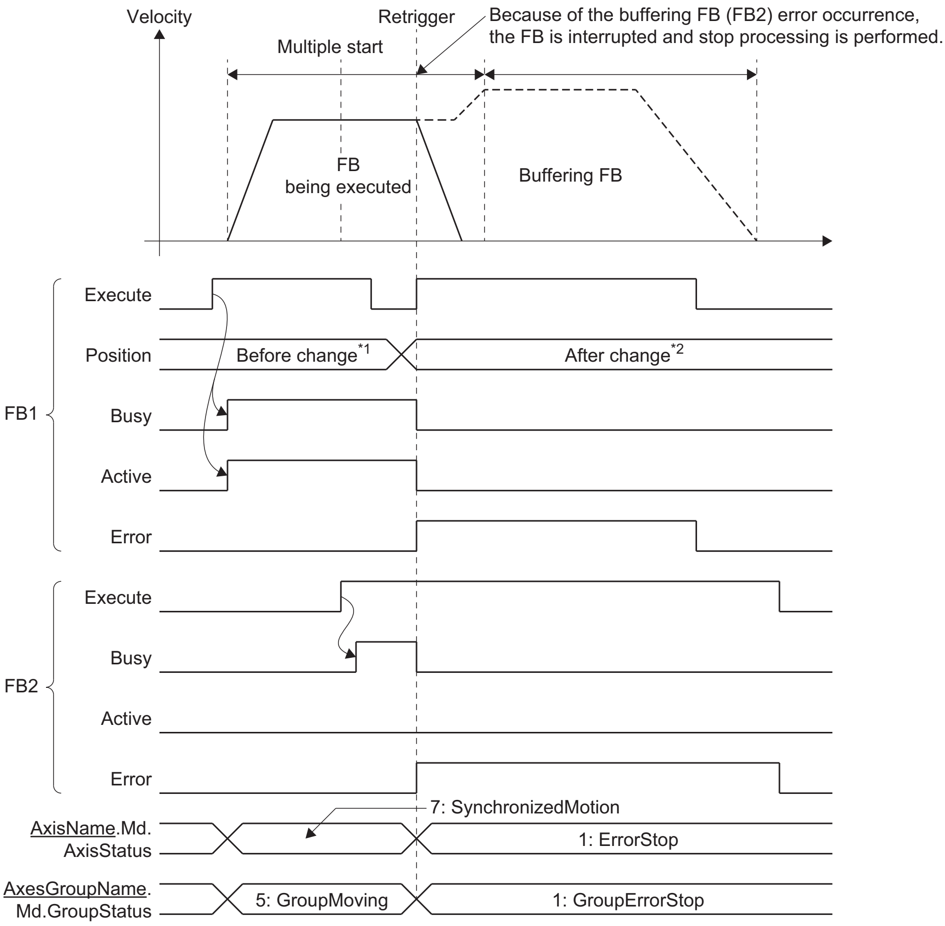
Reanalysis of the buffering FB by retrigger/continuous update

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

As a result of reanalysis, when an error occurs in any of the FBs, Axis Status (AxisName.Md.AxisStatus) becomes "1: Stopping on error (ErrorStop)" or Axes Group Status (AxesGroupName.Md.GroupStatus) becomes "1: Stopping on error (GroupErrorStop)" and the under-control 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 under-control FB.



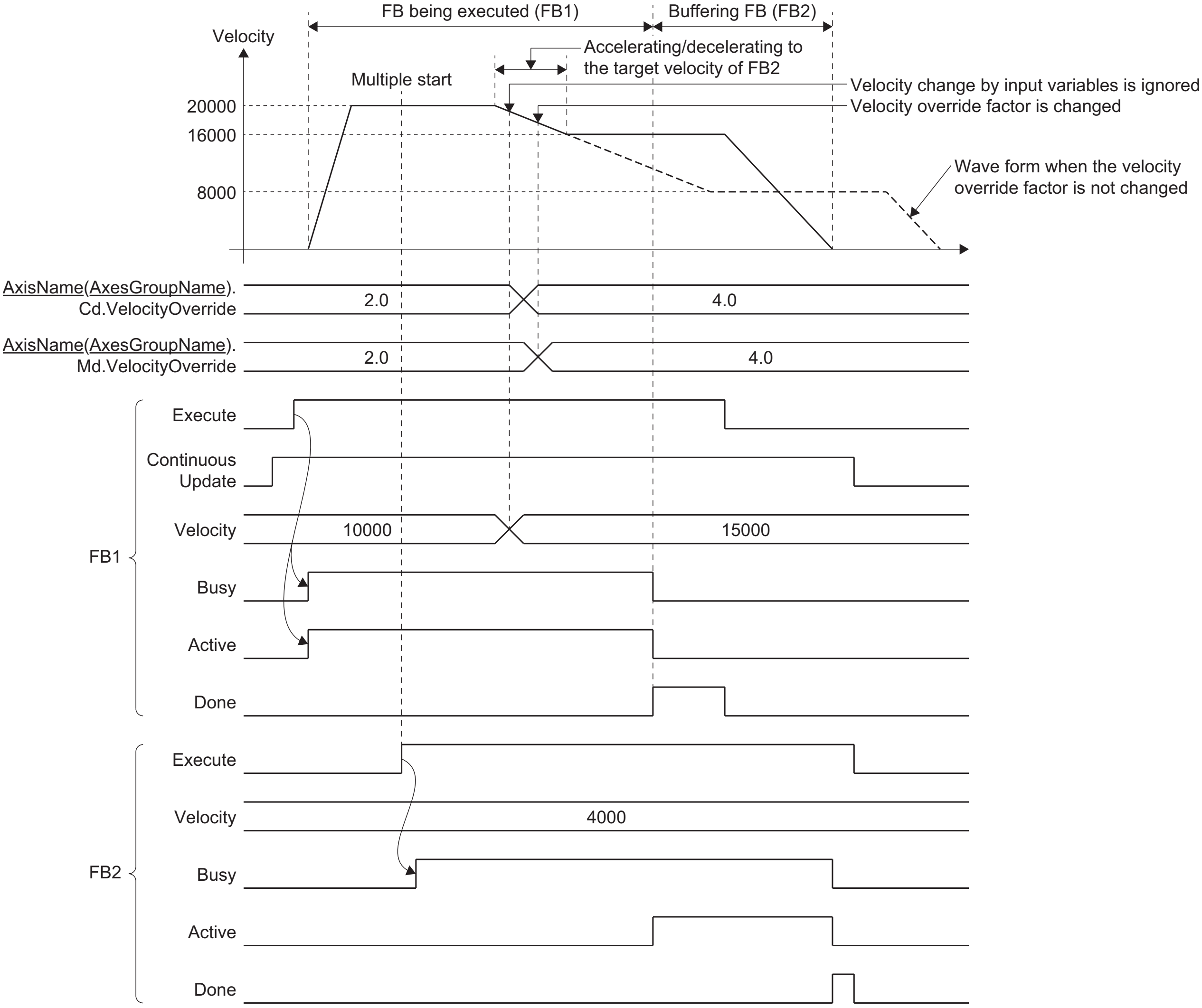
*1 The target position is within the software stroke limit range.

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

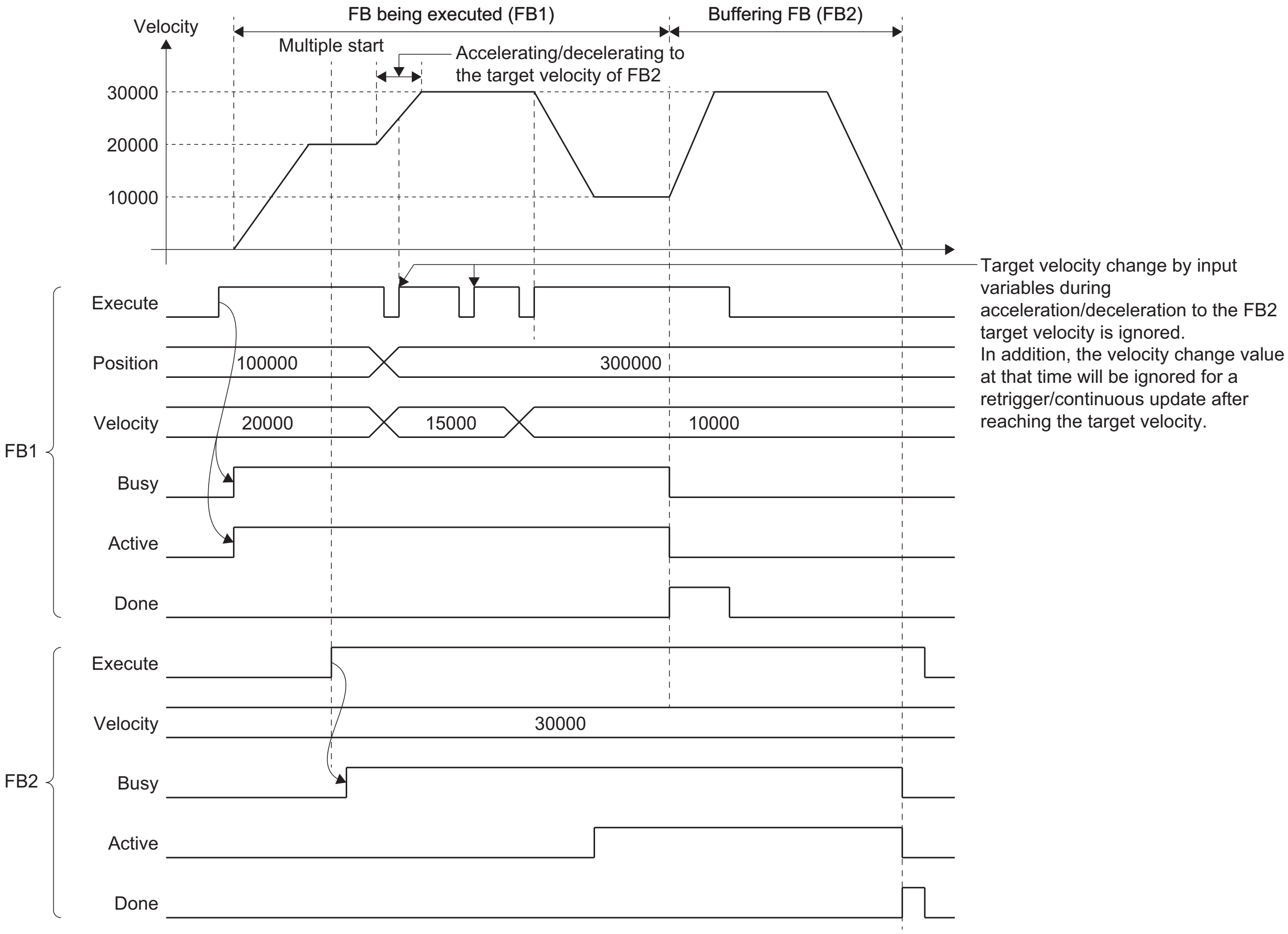
4

BlendingNext changes target velocity at accelerating/decelerating

When BlendingNext is specified and the multiple start is executed, target velocity change by input variables of the under-control FB is not allowed while the buffering FB is accelerating/decelerating to its target velocity (transfer speed). Target velocity change by changing the Velocity override factor (AxisName(AxesGroupName).Cd.VelocityOverride) is allowed.



After reaching the buffering FB target velocity using acceleration/deceleration by BlendingNext, velocity change by input variables of the under-control FB is allowed. However, the velocity change values during acceleration/deceleration by BlendingNext are ignored for a retrigger/continuous update after reaching the buffering FB target velocity. In addition, when a target velocity change is executed in the under-control FB, accelerate/decelerate to the target velocity of the buffering FB will occur after the FB has switched.



BlendingLow/BlendingHigh changes target velocity after multiple start

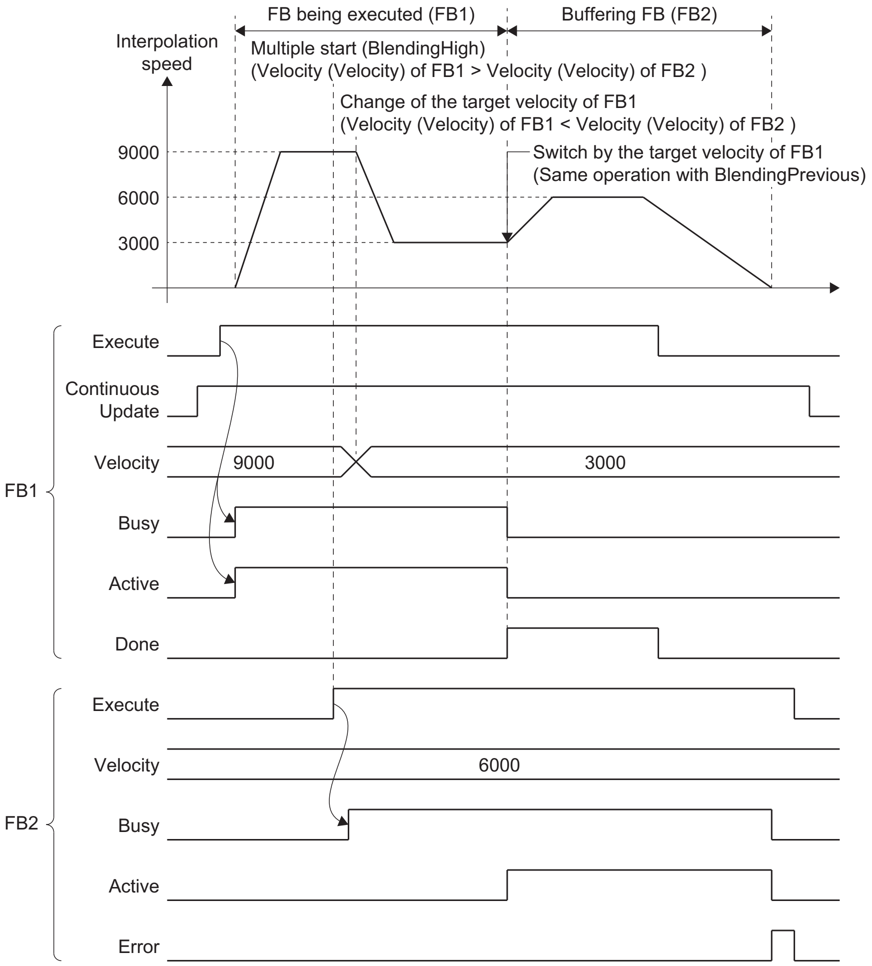
Multiple start by BlendingLow/BlendingHigh determines whether to switch with the velocity of the under-control FB or the buffering FB.

Therefore, if the size related to the target velocity between the under-control FB and the buffering FB is changed by target velocity change after multiple start, the switching velocity is still determined as the target velocity that was determined at multiple start.

When the velocity for the under-control FB and buffering FB is the same at multiple start, the velocity of the under-control FB is used for switching for both BlendingLow and BlendingHigh. Therefore, if the target velocity is changed after multiple start, the under-control FB velocity after the change is used for switching.

Ex.

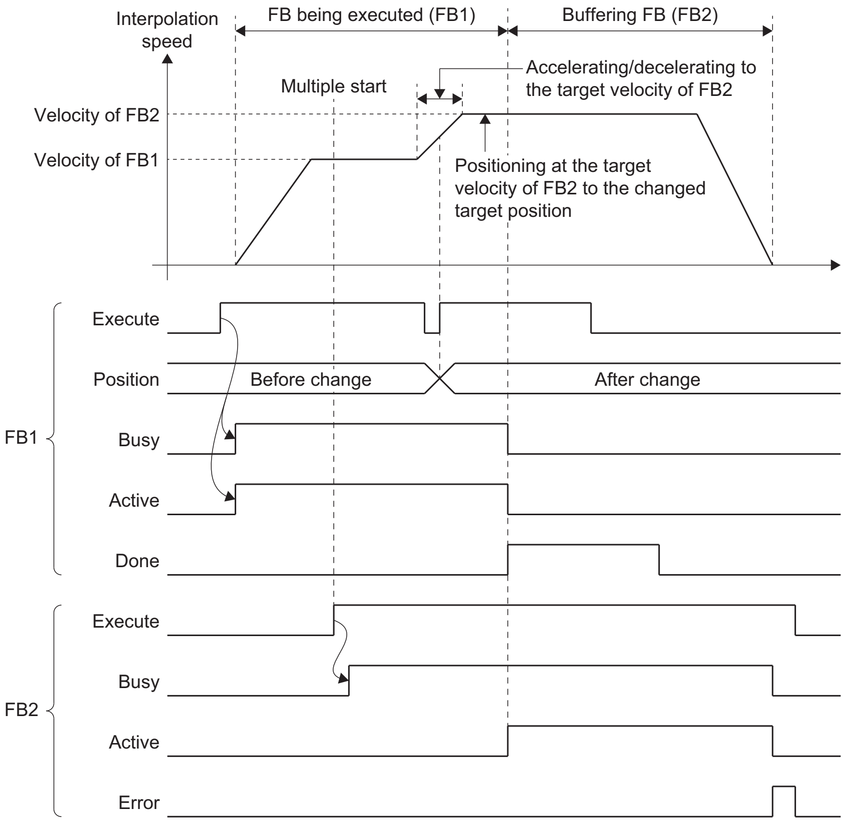
When the target velocity is changed after multiple start by BlendingHigh



BlendingNext changes target position/movement distance at accelerating/decelerating

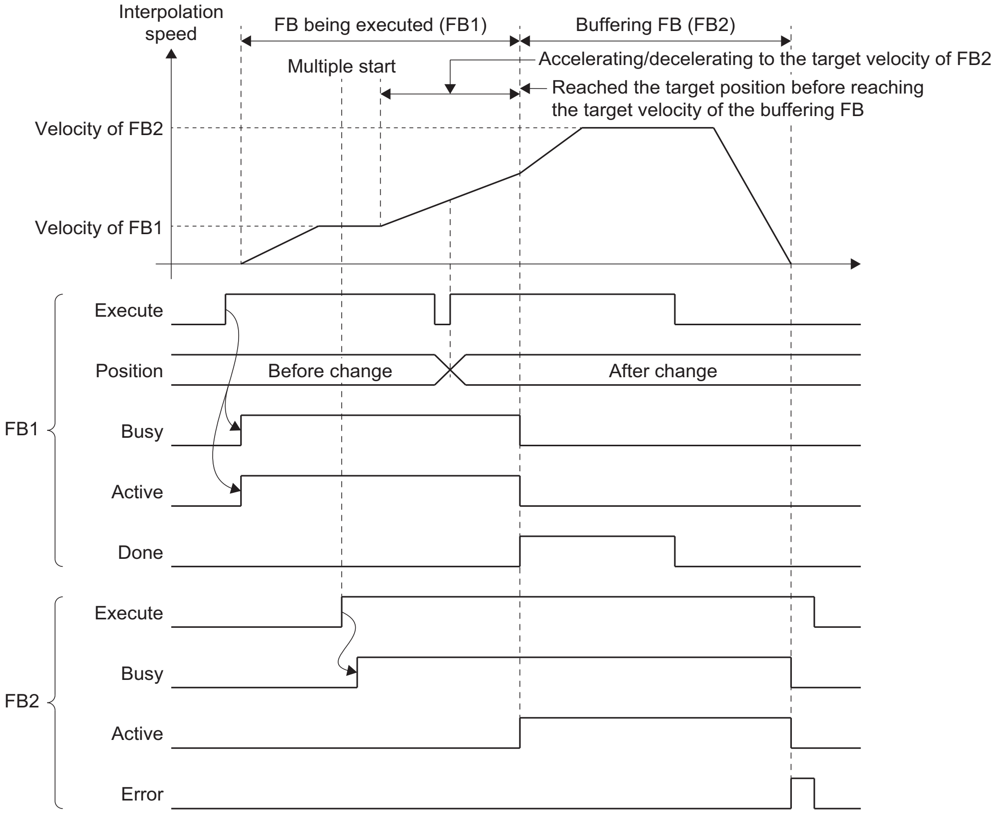
When multiple start is executed by specifying BlendingNext, the operation when the target position/movement distance change is executed during accelerating/decelerating to the target velocity (transfer speed) of the buffering FB is shown below.

- When the movement amount is increased by a target position/movement distance change, positioning to the changed target position/movement distance is executed at the target velocity (transfer speed) of the buffering FB.



• When the movement amount is decreased by a target position/movement distance change, the FB is switched while accelerating/decelerating to the target velocity (transfer speed) of the buffering FB. In addition, if the target position/ movement distance is changed to before the current position, the operation direction will be reversed. For the operation when reversed, refer to the following.

☞ Page 137 Multiple start during single axis operation

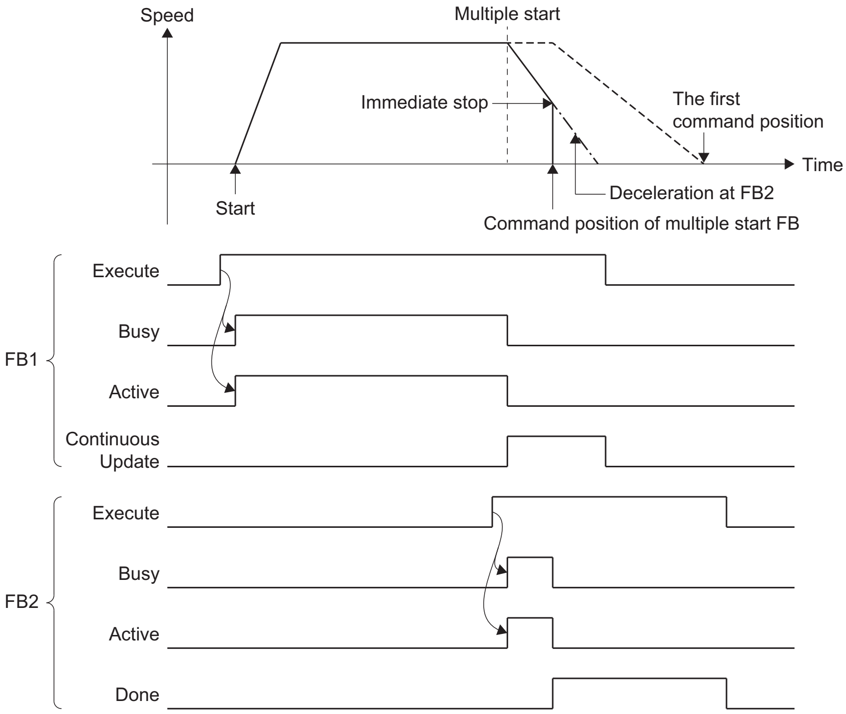


Precautions

- 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 setting of Operation Setting at Overrun (AxisName(AxesGroupName).Pr.OverrunOperation).

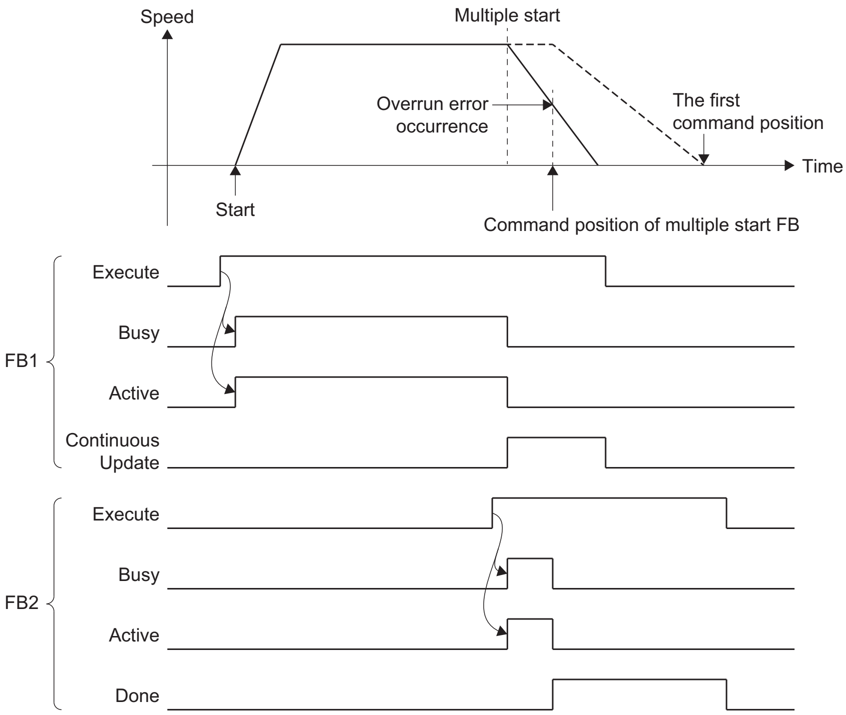
■For "1: Immediate Stop (ImmediateStop)"

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



■For "2: Keep Current Acceleration/Deceleration (KeepCurrentAcc)" (Only for single axis)

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



Point

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

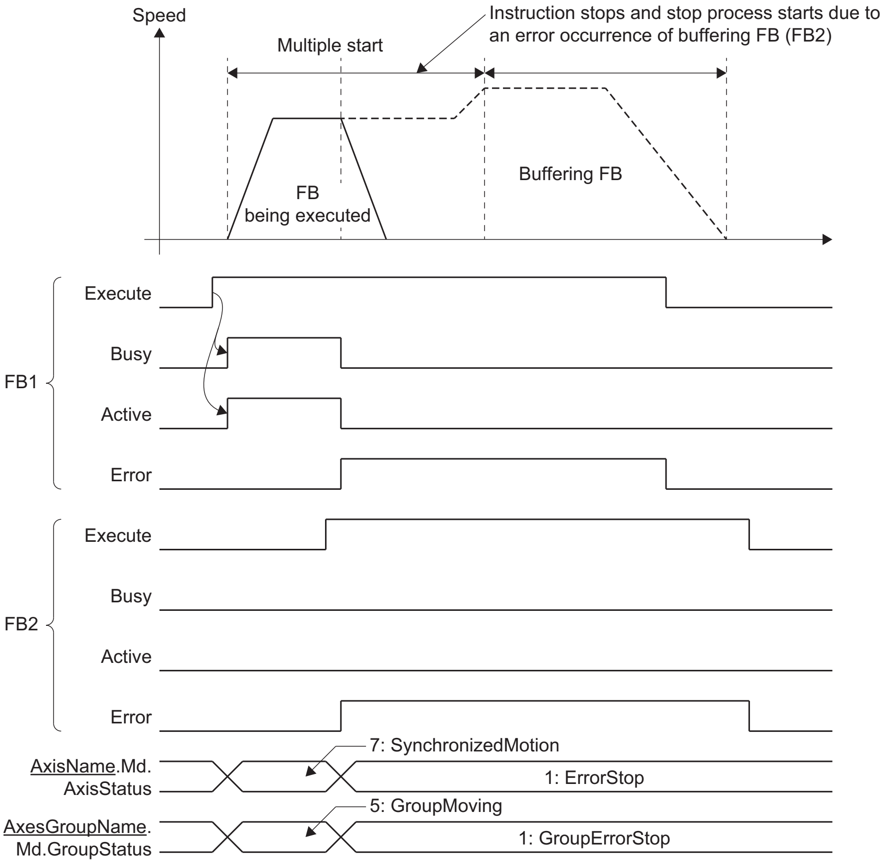
<When the Operation Setting at Overrun (AxisName.Pr.OverrunOperation) setting is "2: Keep Current Acceleration/Deceleration (KeepCurrentAcc)">

- If the target position has already been reached when the multiple start happens (when "Overrun Error (error code: 1A7EH)" occurs), the buffering 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 "Overrun Error (error code: 1A7EH)" occurrence.

<When the Operation Setting at Overrun (AxisName.Pr.OverrunOperation) setting is other than "2: Keep Current Acceleration/Deceleration (KeepCurrentAcc)">

- 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, Axis Status (AxisName.Md.AxisStatus) and Axes Group Status (AxesGroupName.Md.GroupStatus) will be "1: Stopping on error (GroupErrorStop)" and the under-control FBs will stop.



- If the acceleration/deceleration method settings are different between the under-control FB and the next FB, "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.
- Changes are reflected at the switching of FB when restart or continuous update is executed to the buffering FB after multiple start. If the warning occurs at the switching of FB, it operates with the value at multiple start without accepting the changes.

4.4 Stop

This section describes how to stop control.
Each control is stopped in the following cases.
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.

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 (Enable)" of MC_Power (Operation Available) is FALSE
- When "Servo ON request (ServoON)" of MC_Power (Operation Available) is FALSE
- When the hardware stroke limit upper/lower limit error occurs
- When a CPU module error occurs
- When the PLC READY 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 command (Execute)" of MC_Stop (Forced Stop) is TRUE
- When "The stop signal (STOP)" of external input signal is TRUE

Hardware stroke limit upper/lower limit error	Software stroke limit upper/lower limit error
• FLS Signal Detection (at Start) (error code: 1A2DH) • RLS Signal Detection (at Start) (error code: 1A2EH) • FLS Signal Detection (Controlling) (error code: 1A2FH) • RLS Signal Detection (Controlling) (error code: 1A30H)	• Software Stroke Limit Over (Forward Direction) (error code: 1A03H) • 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 command (Execute)" of MC_GroupStop (Group Forced Stop) is TRUE
- When "the stop signal (STOP)" of external input signal is TRUE

Operation of this function for each system status

○: Possible

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

Relevant variables

Axis parameter (AxisName.Pr.)

Variable/Structure name	Name	Details
StopMode_HwStrokeLimit	Stop Selection at Hardware Stroke Limit Error Occurrence	Selects operation when the hardware stroke limit upper/lower limit error occurs. • 1: Immediate Stop (ImmediateStop) (Initial value) • 2: Keep Current Acceleration/Deceleration (KeepCurrentAcc) • 3: Alternative Acceleration/Deceleration (AlternativeAcc) *: (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: Immediate Stop (ImmediateStop) (Initial value) • 2: Keep Current Acceleration/Deceleration (KeepCurrentAcc) • 3: Alternative Acceleration/Deceleration (AlternativeAcc) *: (Immediate stop will be executed when executing an instruction without deceleration specification.)
StopMode_General	Stop Selection at Stop Cause Occurrence	Selects operation when a stop cause for each axis other than the stop cause 1 and 2 occurs. • 1: Immediate Stop (ImmediateStop) • 2: Keep Current Acceleration/Deceleration (KeepCurrentAcc) • 3: Alternative Acceleration/Deceleration (AlternativeAcc) (initial value) *: (Immediate stop will be executed when executing an instruction without deceleration specification.)
StopMode_Deceleration	Deceleration at Stop	Specifies the deceleration/deceleration time 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: 0.000000, a positive number from 0.000001 to 8400.0 [s][*1] *: If "0.0" is set, immediately stops regardless of the operation selection at acceleration/deceleration 0 (AccelerationZeroBehavior) 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). • 1: Recreate Deceleration Curve (OverrideCurve) (Initial value)
StopMode_ServoOff	Process Selection at Servo OFF Command During Operation	Selects operation when Servo ON request (ServoON) of MC_Power (Operation Available) becomes FALSE during operation. • 0: Ignore (Ignore) (Initial value) • 4: Servo OFF After Immediate Stop (ServoOffAfterImmediateStop) • 5: Servo OFF After Deceleration to Stop (ServoOffAfterDecelStop)
OverrunOperation	Operation Setting at Overrun	Selects operation when stop position is reached during deceleration stop processing by a stop cause. • 1: Immediate Stop (ImmediateStop) (Initial value) • 2: Keep Current Acceleration/Deceleration (KeepCurrentAcc)

Variable/Structure name	Name	Details
StopSignal	Stop Signal	Sets a signal that uses the stop signal (STOP) of the external input signal. Stop signal (StopSignal) is a structure of SIGNAL_SELECT type. 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 381 External Signal Selection The specific setting and operation for this signal are shown below. ■I/O Number (StartIO) Ignores input values. ■Target (Target) If the target is unspecified, it is regarded as invalid signal and it is always the signal undetection status. Only [VAR] , [DEV] , and [CONST] can be specified for the data type. *: When unavailable data is set, "Out of Parameter Range (Axis) (error code: 1D80H)" occurs. ■Signal Detection Method (Detection) The specification is allowed at the following level detection only. • 0: Detection at TRUE (HighLevel) • 1: Detection at FALSE (LowLevel) *: When edge detection specification is set, "Out of Parameter Range (Axis)" (error code: 1D80H) (detail code: 0067H) occurs. ■Compensation Time (CompensationTime) Ignores the input value. ■Filter Time (FilterTime) Setting range of the filter time is 0.0 to +5.0. *: When a value outside the range is set, "Out of Filter Time Setting Range of Each Axis Signal Warning (warning code: 0D24H)" occurs) and the axis operates with 0.0 of the filter time.
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)

*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.

Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
StopSignal	Stop Signal	Displays input status of the stop signal (STOP) of the external input signal. Stop signal (StopSignal) is a structure of SIGNAL_SELECT type. For details on SIGNAL_SELECT type, refer to the following. ☞ Page 381 External Signal Selection The specific output for this signal is shown below. ■I/O Number (StartIO) 0 is always displayed. ■Target (Target) The fetch result of the parameter is displayed. ■Signal Detection Method (Detection) The fetch result of the parameter is displayed. ■Compensation Time (CompensationTime) 0.0 is always displayed. ■Filter Time (FilterTime) The fetch result of the parameter is displayed.
StopStatus	Stop Status	Displays the input status of the stop signal (STOP) among external input signals. • FALSE: Stop processing released • TRUE: During stop processing
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 Cause Occurrence	Displays input status of the stop selection at stop cause occurrence.
StopMode_HwStrokeLimit	Stop Selection at Hardware 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 Software Stroke Limit Error Occurrence	Displays input status of the stop selection at the software stroke limit error occurrence.

Variable/Structure name	Name	Details
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.

Axes group parameter (AxesGroupName.Pr.)

Variable/Structure name	Name	Details
StopMode_General	Stop Selection at Stop Cause Occurrence	Selects operation when a stop cause of the axes group occurs. • 1: Immediate Stop (ImmediateStop) • 2: Keep Current Acceleration/Deceleration (KeepCurrentAcc) • 3: Alternative Acceleration/Deceleration (AlternativeAcc) (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). • 1: Recreate Deceleration Curve (OverrideCurve) (Initial value)
OverrunOperation	Operation Setting at Overrun	Selects operation when stop position is reached during deceleration stop processing by a stop cause. If other than "1: Immediate Stop (ImmediateStop)" (Initial value) is set, an error will occur. • 1: Immediate Stop (ImmediateStop)
StopMode_ErrorInGroup	Configuration Axes Operation Selection at Axis Stop Cause Occurrence	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: Immediate Stop (ImmediateStop)
StopMode_Deceleration	Deceleration at Stop	Specifies the deceleration/deceleration time 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 "0.0" is set, immediately stops regardless of the operation selection at acceleration/deceleration 0 (AccelerationZeroBehavior) 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.

Axes group monitor data (AxesGroupName.Md.)

Variable/Structure name	Name	Details
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	Configuration Axes Operation Selection at Axis Stop Cause Occurrence	Displays input status of the configuration axes operation selection at axis stop cause occurrence.
StopMode_General	Stop Selection at Stop Cause Occurrence	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 parameter (System.Pr.)

Variable/Structure name	Name	Details
StopMode_All	Stop Selection at All Axes Stop Cause Occurrence	Selects whether to stop immediately or decelerate to stop when an all axes stop cause occurs. • 1: Immediate Stop (ImmediateStop) • 2: Keep Current Acceleration/Deceleration (KeepCurrentAcc) • 3: Alternative Acceleration/Deceleration (AlternativeAcc) *: (Immediate stop will be executed when executing an instruction without deceleration specification.)
StopMode_AllDeceleration	Deceleration at All Axes Stop	Specifies the deceleration/deceleration time at deceleration stop when an all axes stop cause occurred. ■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 "0.0" is set, immediately stops regardless of the operation selection at acceleration/deceleration 0 (AccelerationZeroBehavior) 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.

Relevant FBs

For details on Motion control FBs, refer to the following.

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

Motion control FB	Name	Description
MC_Stop	Forced stop	Decelerates specified axis to stop.
MC_GroupStop	Group forced stop	Decelerates specified axes group to stop.

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: Stopping on error (ErrorStop)	Immediate stop*2
2	Drive unit power supply is OFF	Each axis	—	1: Stopping on error (ErrorStop)	Immediate stop*2
	Drive unit network disconnection detection				
	Drive unit error				
	Forced stop input to the drive unit				
	Enable (Enable) of MC_Power (Operation Available) is FALSE				
	Servo ON request (ServoON) of MC_Power (Operation Available) is FALSE (When Process Selection at Servo OFF Command During Operation (AxisName.Pr.StopMode_ServoOff) is "4: Servo OFF After Immediate Stop (ServoOffAfterImmediateStop)")*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: Stopping on error (ErrorStop)	1: Stopping on error (ErrorStop)	Deceleration stop/immediate stop (Follows Stop Selection at Hardware Stroke Limit Error Occurrence (AxisName.Pr.StopMode_HwStrokeLimit))
4	CPU module error occurrence	All axes	1: Stopping on error (ErrorStop)	1: Stopping on error (ErrorStop)	Deceleration stop/immediate stop (Follows Stop Selection at All Axes Stop Cause Occurrence (System.Pr.StopMode_All))*7
	PLC READY 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: Stopping on error (ErrorStop)	1: Stopping on error (ErrorStop)	Deceleration stop/immediate stop (Follows Stop Selection at Software Stroke Limit Error Occurrence (AxisName.Pr.StopMode_SwStrokeLimit))
6	Axis error detection*4	Each axis	1: Stopping on error (ErrorStop)	1: Stopping on error (ErrorStop)	Deceleration stop/immediate stop (Follows Stop Selection at Stop Cause Occurrence (AxisName.Pr.StopMode_General))
7	Execute command (Execute) of MC_Stop (Forced Stop) is TRUE	Each axis	2: Decelerating to stop (Stopping)	2: Decelerating to stop (Stopping)*5	Deceleration stop/immediate stop (Follows the deceleration (Deceleration) set in the FB)
8	"The stop signal (STOP) " of external input signal is TRUE	Each axis	No change	4: Standby (Standstill)	Deceleration stop/immediate stop (Follows Stop Selection at Stop Cause Occurrence (AxisName.Pr.StopMode_General))

*1 For each operation of the stop process, refer to the following.
☞ Page 168 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 Process Selection at Servo OFF Command During Operation (AxisName.Pr.StopMode_ServoOff). For details, refer to the following.
☞ Page 173 Stop causes by Servo ON request (ServoON) input of MC_Power (Operation Available)

*4 If an axis error (the error that transits the state into the "1: Stopping on error (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 command (Execute) of MC_Stop (Forced Stop) is FALSE at stop completion, Axis Status (AxisName.Md.AxisStatus) after stop will be "4: Standby (Standstill)".

*6 Immediately stops the motion system side command, and follow up on the current position.
*7 The deceleration stop can be performed independently on the driver side depending on the driver device specifications. For details, refer to the manual of the driver.

Precautions

Provide the emergency stop circuits outside the servo system to prevent cases where danger may result from abnormal operation of the entire 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: Stopping on error (GroupErrorStop)	1: Stopping on error (GroupErrorStop)	Stop cause occurrence axis	Axes group deceleration stop/immediate stop*2 (For axes group deceleration stop, it follows Stop Selection at Stop Cause Occurrence (AxesGroupName.Pr.StopMode_General))
						Other configuration axes	Axes group deceleration stop/immediate stop (For axes group deceleration stop, it follows Stop Selection at Stop Cause Occurrence (AxesGroupName.Pr.StopMode_General))*3
2		"The stop signal (STOP)" of external input signal is TRUE	Axes group	No change	4: Standby (GroupStandby)	Axes group deceleration stop/immediate stop (Follows Stop Selection at Stop Cause Occurrence (AxesGroupName.Pr.StopMode_General))	
3	Axes group error detection*4		Axes group	1: Stopping on error (GroupErrorStop)	1: Stopping on error (GroupErrorStop)	Axes group deceleration stop/immediate stop (Follows Stop Selection at Stop Cause Occurrence (AxesGroupName.Pr.StopMode_General))	
4	Execute command (Execute) of MC_GroupStop (Group Forced Stop) is TRUE		Axes group	2: Decelerating to stop (GroupStopping)	2: Decelerating to stop (GroupStopping)*5	Axes group deceleration stop/immediate stop (Follows the deceleration (Deceleration) set in the FB)	

*1 For each operation of the stop process, refer to the following.
☞ Page 170 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 Configuration Axes Operation Selection at Axis Stop Cause Occurrence (AxesGroupName.Pr.StopMode_ErrorInGroup). For details, refer to the following.
☞ Page 180 Stop cause occurrence in the configuration axis
*4 If an axes group error (the error that transits the state into the "1: Stopping on error (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 command (Execute) of MC_GroupStop (Group Forced Stop) is FALSE at stop completion, Axes Group Status (AxesGroupName.Md.GroupStatus) after stop will be "4: Standby (GroupStandby)".

Types of stop processes

This section describes types of stop processes for a single axis and an axes group.

Stop operation by Stop Selection at Stop Cause Occurrence (AxisName(AxesGroupName).Pr.StopMode_General) and Deceleration at Stop (AxisName(AxesGroupName).Pr. StopMode_Deceleration) are explained in this section. For the stop operation by MC_Stop (Forced Stop) or MC_GroupStop (Group Forced Stop), refer to the following.

☞ Page 177 MC_Stop (Forced Stop)
☞ Page 184 MC_GroupStop (Group Forced Stop)

Stop processes for single axis

■ Deceleration stop

The operation decelerates from the speed during operation to speed 0. The deceleration differs depending on Stop Selection at Stop Cause Occurrence (AxisName.Pr.StopMode_General).

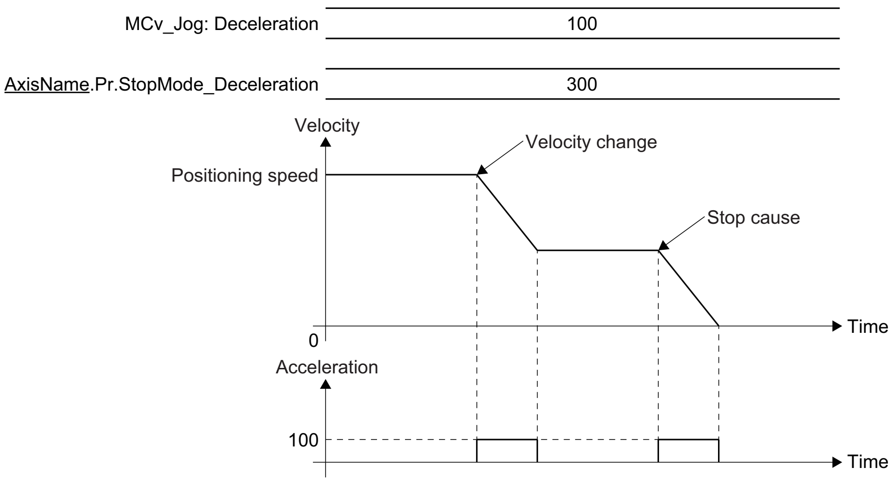
Stop Selection at Stop Cause Occurrence (AxisName.Pr.StopMode_General setting value)	Deceleration at stop
2: Keep Current Acceleration/Deceleration (KeepCurrentAcc)	Deceleration at instruction during operation
3: Alternative Acceleration/Deceleration (AlternativeAcc)	Deceleration at Stop (AxisName.Pr. StopMode_Deceleration)

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.

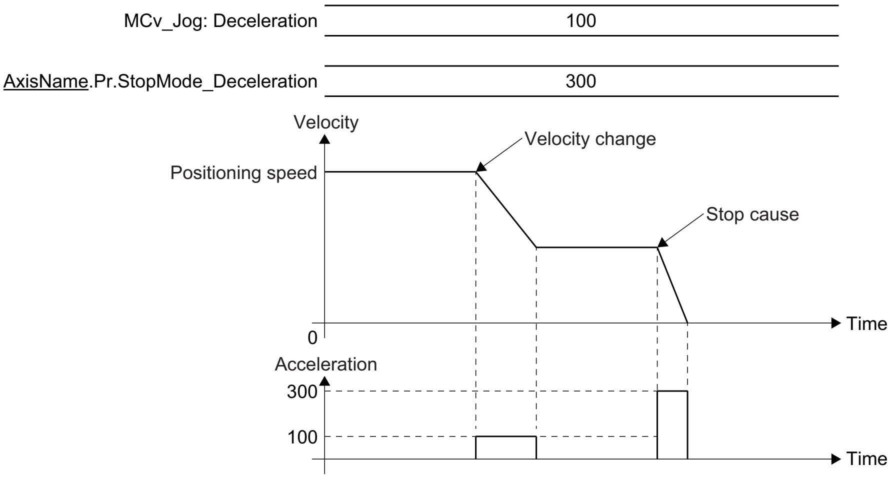
For details of each stop method, refer to the following.

☞ Page 317 Acceleration/deceleration Processing Function

- When "2: Keep Current Acceleration/Deceleration (KeepCurrentAcc)" is set to Stop Selection at Stop Cause Occurrence (AxisName.Pr.StopMode_General) (When the instruction in operation is MCv_Jog (JOG Operation))



• When "3: Alternative Acceleration/Deceleration (AlternativeAcc)" is set to Stop Selection at Stop Cause Occurrence (AxisName.Pr.StopMode_General) (When the instruction in operation is MCv_Jog (JOG Operation))

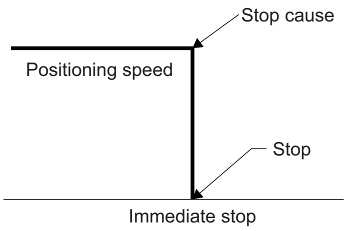


■ Immediate stop

The operation does not decelerate and immediately stops the command.

For the stop method of the drive unit, refer to each drive unit manual.

For MR-J5(W)-G: 📖MR-J5-G/MR-J5W-G User's Manual (Parameters)



Stop processes for axes group

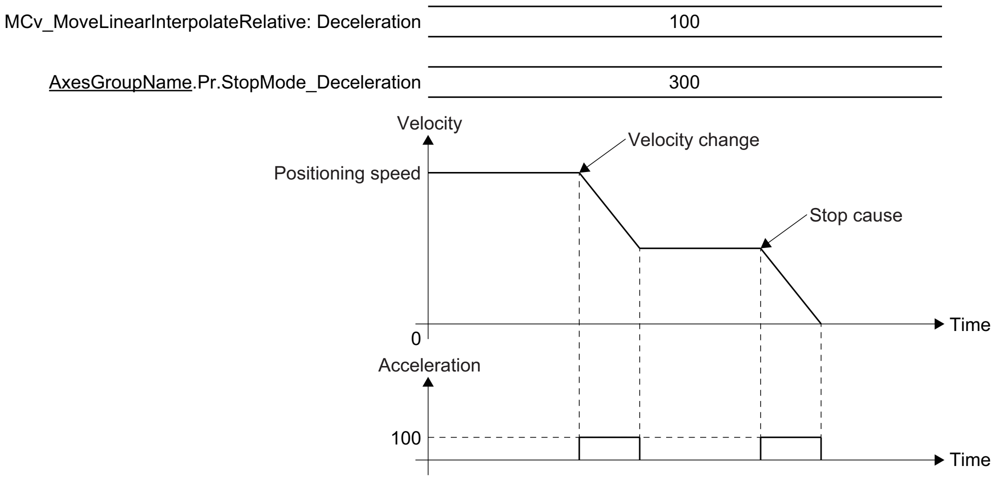
■ Deceleration stop

The operation decelerates from the interpolation speed during operation to the interpolation speed 0. The deceleration differs depending on Stop Selection at Stop Cause Occurrence (AxesGroupName.Pr.StopMode_General).

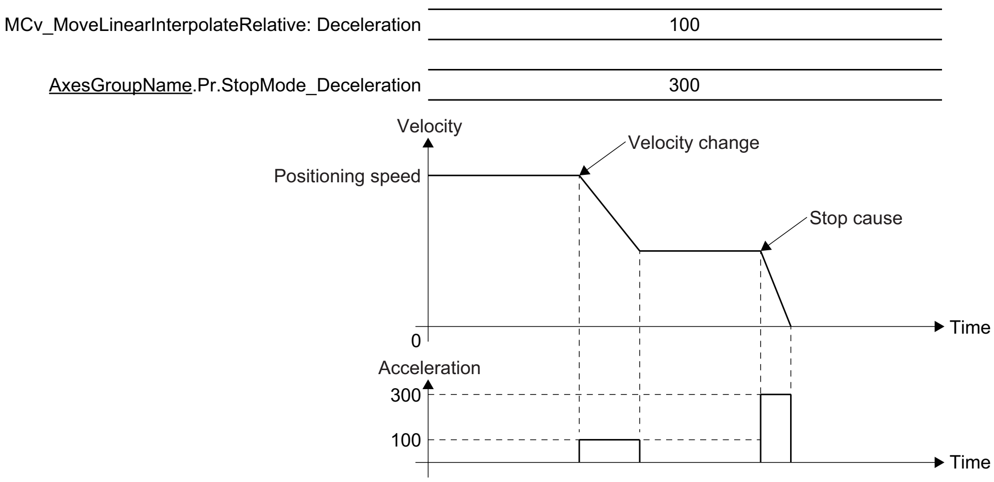
Stop Selection at Stop Cause Occurrence (AxesGroupName.Pr.StopMode_General) setting value	Deceleration at stop
2: Keep Current Acceleration/Deceleration (KeepCurrentAcc)	Deceleration at instruction during operation
3: Alternative Acceleration/Deceleration (AlternativeAcc)	Deceleration at Stop (AxesGroupName.Pr. StopMode_Deceleration)

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 317 Acceleration/deceleration Processing Function

- When "2: Keep Current Acceleration/Deceleration (KeepCurrentAcc)" is set to Stop Selection at Stop Cause Occurrence (AxesGroupName.Pr.StopMode_General (When the instruction in operation is MCv_MoveLinearInterpolateRelative (Relative Value Linear Interpolation Control))



- When "3: Alternative Acceleration/Deceleration (AlternativeAcc)" is set to Stop Selection at Stop Cause Occurrence (AxesGroupName.Pr.StopMode_General) (When the instruction in operation is MCv_MoveLinearInterpolateRelative (Relative Value Linear Interpolation Control))

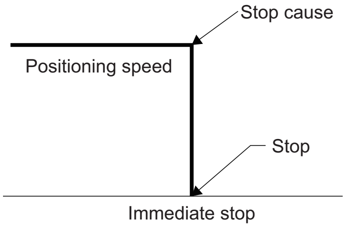


■ Immediate stop

The operation does not decelerate and immediately stops the command.

For the stop method of the drive unit, refer to each drive unit manual.

For MR-J5(W)-G: 📖MR-J5-G/MR-J5W-G User's Manual (Parameters)



4

Order of priority for stop process

This section describes the 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.

Item	Parameter	Setting value
Single axis	Stop Selection at Deceleration to Stop (AxisName.Pr.StopMode_DecelerationCurve)	1: Recreate Deceleration Curve (OverrideCurve)
Axes group	Stop Selection at Deceleration to Stop (AxesGroupName.Pr.StopMode_DecelerationCurve)	

Deceleration curve re-processing

■ 1: Recreate Deceleration Curve (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: Immediate Stop (ImmediateStop)" in Operation Setting at Overrun (AxisName.Pr.OverrunOperation).

For operation details, refer to the following.

☞ Page 172 Overrun at stop cause occurrence

Overrun at stop cause occurrence

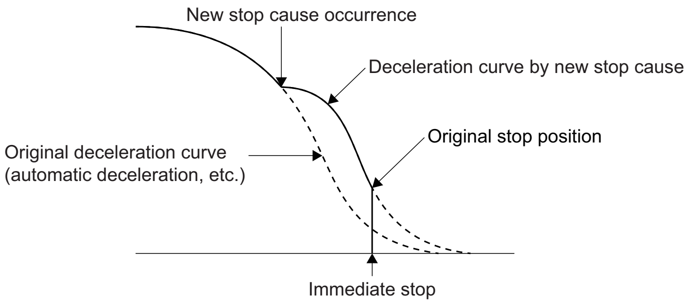
This function selects operation for the case that the positioning address of an instruction which was executed before a stop cause occurs is reached during deceleration stop and immediate stop process.
This function is enabled only when the target position exists at stop cause occurrence.
The following parameters for a single axis or an axes group are need to be set in this function.

Item	Parameter	Setting value
Single axis	Operation Setting at Overrun (AxisName.Pr.OverrunOperation)	1: Immediate Stop (ImmediateStop) 2: Keep Current Acceleration/Deceleration (KeepCurrentAcc)
Single axis	Operation Setting at Overrun (AxesGroupName.Pr.OverrunOperation)	1: Immediate Stop (ImmediateStop)

Immediate stop

■ 1: Immediate Stop (ImmediateStop)

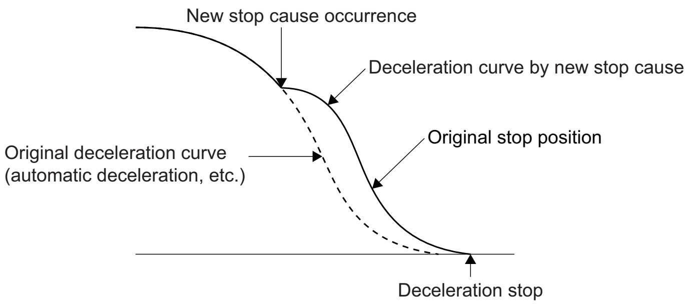
The operation is immediately stopped when reaching the original positioning address during deceleration by a stop cause. In this case, the operation does not exceed the positioning address set in the under-control FB.



Keep current acceleration/deceleration (Only for a single axis)

■ 2: Keep Current Acceleration/Deceleration (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 under-control 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.

For MR-J5(W)-G: 📖MR-J5-G/MR-J5W-G User's Manual (Parameters)

- "Forced stop input" is turned OFF from an external device[*1]
- Drive unit power supply OFF
- Drive unit network disconnection detection
- Drive unit error
- Forced stop input to drive unit
- "Enable (Enable)" of MC_Power (Operation Available) is FALSE

*1 The forced stop input is set in Forced Stop Signal (AxisName.Pr.ForcedStop_Signal). For operation details, refer to the following.
☞ Page 187 Forced Stop

Stop causes by Servo ON request (ServoON) input of MC_Power (Operation Available)

■ Process selection at servo OFF command during operation

When Servo ON request (ServoON) of MC_Power (Operation Available) is set to FALSE, the stop operation differs depending on the setting of Process Selection at Servo OFF Command During Operation (AxisName.Pr.StopMode_ServoOff). Process Selection at Servo OFF Command During Operation (AxisName.Pr.StopMode_ServoOff) has the following three functions.

Process selection at servo OFF command during operation (AxisName.Pr.StopMode_ServoOff)[*1]	Description
0: Ignore (Ignore)	During operation, this function ignores the servo OFF command even though it is input and continues on-going operation. If Servo ON request (ServoON) of MC_Power (Operation Available) is set to FALSE when Axis Status (AxisName.Md.AxisStatus) of the operation completion axis turns "4: Standby (Standstill)", servo OFF will be executed. (The axis will not stop.)
4: Servo OFF After Immediate Stop (ServoOffAfterImmediateStop)	During operation, the command is immediately stopped by "Immediate Stop by Servo OFF During Operation (error code: 1A3CH)" occurrence, and a servo OFF is issued to the driver.
5: Servo OFF After Deceleration to Stop (ServoOffAfterDecelStop)	During operation, when receiving the servo OFF command input, "Deceleration Stop by Servo OFF During Operation (error code: 1A3BH)" occurs and deceleration stop is executed. If Servo ON request (ServoON) of MC_Power (Operation Available) 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 .

*1 Process selection at servo OFF command during operation (AxisName.Pr.StopMode_ServoOff) during operation operates "4: Servo OFF After Immediate Stop (ServoOffAfterImmediateStop)".

■ Axis state transition

The axis state changes based on the input combination of Enable (Enable) and Servo ON request (ServoON) of MC_Power (Operation Available).

Axis status before input operation (AxisName.Md.AxisStatus)	Input		Process Selection at Servo OFF Command During Operation (AxisName.Pr.StopMode_ServoOff)	Axis Status (AxisName.Md.AxisStatus) after input operation
	Enable (Enable)	Servo ON request (ServoON)		
4: Standby (Standstill) 0: Axis disabled (Disabled)	TRUE	TRUE	Independent of the setting	4: Standby (Standstill)
		FALSE		0: Axis disabled (Disabled)
	FALSE	TRUE		
		FALSE		
1: Stopping on error (ErrorStop)	TRUE	TRUE		1: Stopping on error (ErrorStop)
		FALSE		
	FALSE	TRUE		
		FALSE		
Others	TRUE	TRUE		No transition
		FALSE	0: Ignore (Ignore)	No transition[*1*2]
			4: Servo OFF After Immediate Stop (ServoOffAfterImmediateStop)	1: Stopping on error (ErrorStop)[*4]
			5: Servo OFF After Deceleration to Stop (ServoOffAfterDecelStop)	1: Stopping on error (ErrorStop)[*3]
	FALSE	TRUE	Independent of the setting	1: Stopping on error (ErrorStop)[*4]
		FALSE		

*1 It is "0: Axis disabled (Disabled)" when Axis Status (AxisName.Md.AxisStatus) is "4: Standby (Standstill)" or "2: Decelerating to stop (Stopping)" after deceleration stop.

*2 Axis Status (AxisName.Md.AxisStatus) will not transit if Axis Status (AxisName.Md.AxisStatus) is "1: Stopping on error (ErrorStop)" after deceleration stop.

*3 If ServoON becomes FALSE during operation, "Deceleration Stop by Servo OFF During Operation (error code: 1A3BH)" will occur.

*4 When Enable (Enable)/Servo ON request (ServoON) becomes FALSE during operation, "Immediate Stop by Servo OFF During Operation (error code: 1A3CH)" occurs. In this case, the under-control 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 Stop Selection at Hardware Stroke Limit Error Occurrence (AxisName.Pr.StopMode_HwStrokeLimit).

For details, refer to the following.

☞ Page 311 Hardware Stroke Limit

CPU module error occurrence

When the stop error occurs in the control CPU module, PLC READY holds ON, and the motion side will be STOP status.

PLC READY OFF

When the PLC READY is turned OFF, the operation stops according to the process selected by Stop Selection at All Axes Stop Cause Occurrence (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 Stop Selection at All Axes Stop Cause Occurrence (System.Pr.StopMode_All).
According to the set values of Cycle Over Error Selection (System.PrConst.OperationCycle[1].CycleOverErrorType) and Cycle Over Error Selection (System.PrConst.BuffermemoryRefreshCycle.CycleOverErrorType), different error code is output in every axis.

Setting value	Error
2: Minor error (MinorError)	Cycle Over (error code: 1C80H)
3: Moderate error (ModerateError)	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

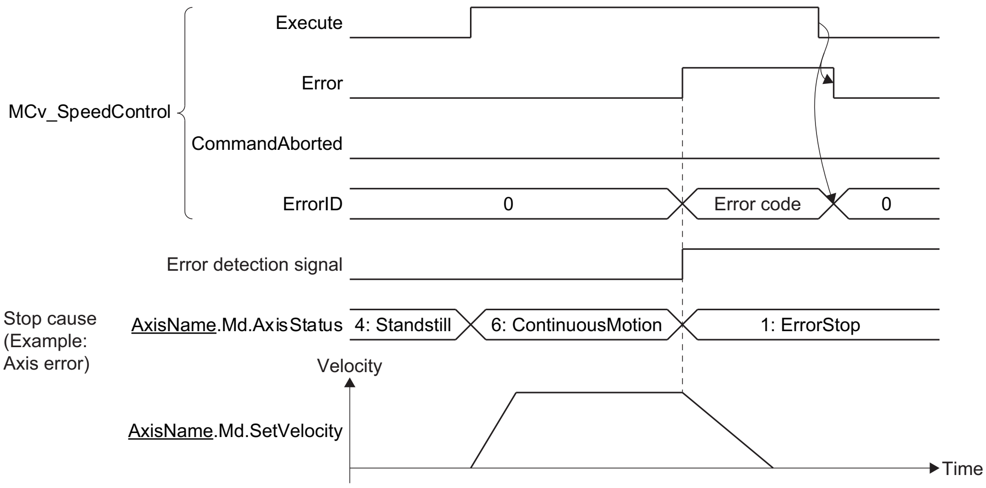
When the software stroke limit upper/lower limit error occurs, the operation stops according to the process selected by Stop Selection at Software Stroke Limit Error Occurrence (AxisName.Pr.StopMode_SwStrokeLimit).
For details, refer to the following.
☞ Page 304 Software Stroke Limit

Axis error detection

When an axis error other than the errors below is detected, the operation stops according to the process selected by Stop Selection at Stop Cause Occurrence (AxisName.Pr.StopMode_General). In this case, Axis Status (AxisName.Md.AxisStatus) transits to "1: Stopping on error (ErrorStop)".

- Hardware stroke limit upper/lower limit error
- The CPU module error
- An error by the PLC READY OFF
- The error "Cycle Over" (error code: 1C80H)
- The software stroke limit upper/lower limit error
- the drive unit error

- Operation of the FB that received a stop cause by axis error occurrence



■ Driver command discard detection

By setting Driver Command Discard Detection Setting (AxisName.Pr.StopOption_DriverTargetIgnored), when the driver unit Statusword Bit12 turns ON to OFF during axis operation at the real drive axis, "Driver Command Discard Detection (error code: 1AE6H)" is output and can stop the command.

When "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 Object Data_Statusword (AxisName.Md.Io_Statusword).

For details, refer to the following.

☞ Page 311 Hardware Stroke Limit

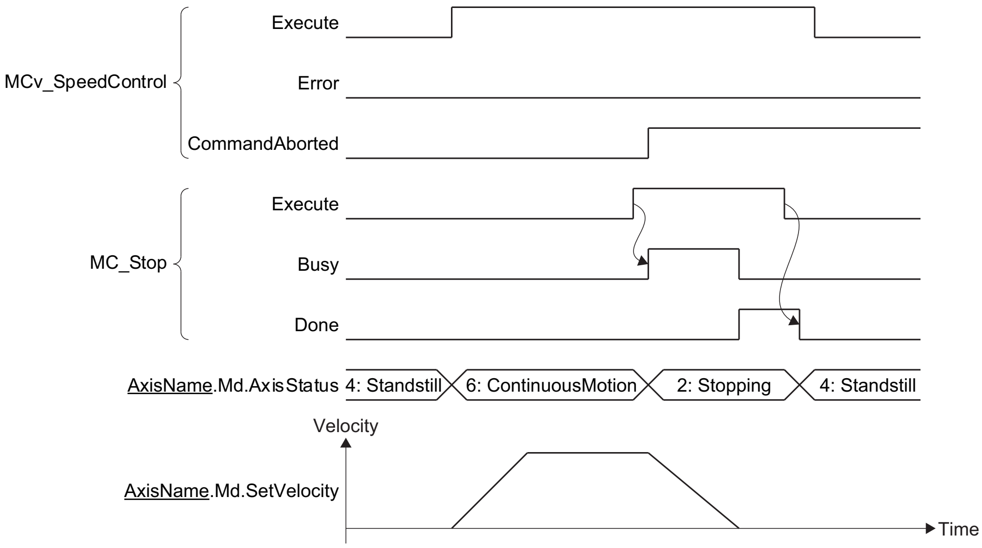
The details of Statusword (Bit12) differs depending on the connect driver unit control mode. Also, for Statusword changing conditions etc., refer to the connected device driver device specification.

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)

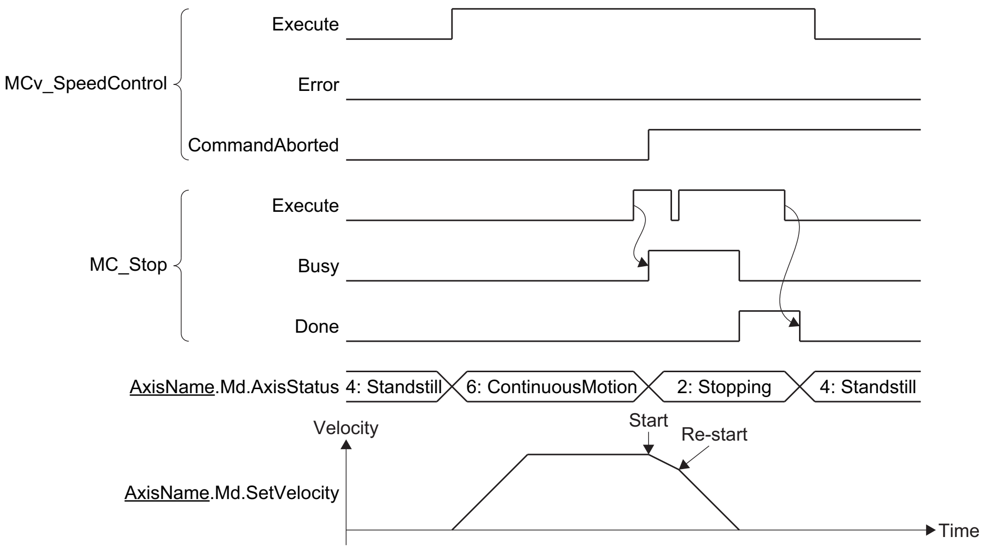
In MC_Stop (Forced Stop), Deceleration (Deceleration) is set and the under-control FB is decelerated to stop.
When MC_Stop (Forced Stop) is executed, Abortion of execution (CommandAborted) of the on-going FB becomes TRUE and Axis Status (AxisName.Md.AxisStatus) transits to the "2: Decelerating to stop (Stopping)" status. While Execute command (Execute) is TRUE or while the speed does not reach 0, the "2: Decelerating to stop (Stopping)" status is maintained. When Execution completion (Done) becomes TRUE and Execute command (Execute) becomes FALSE at stop completion, the axis status changes to the "4: Standby (Standstill)" status.

■ MCv_Stop (Forced Stop) execution during MC_SpeedControl (Speed Control (Including Position Loop))

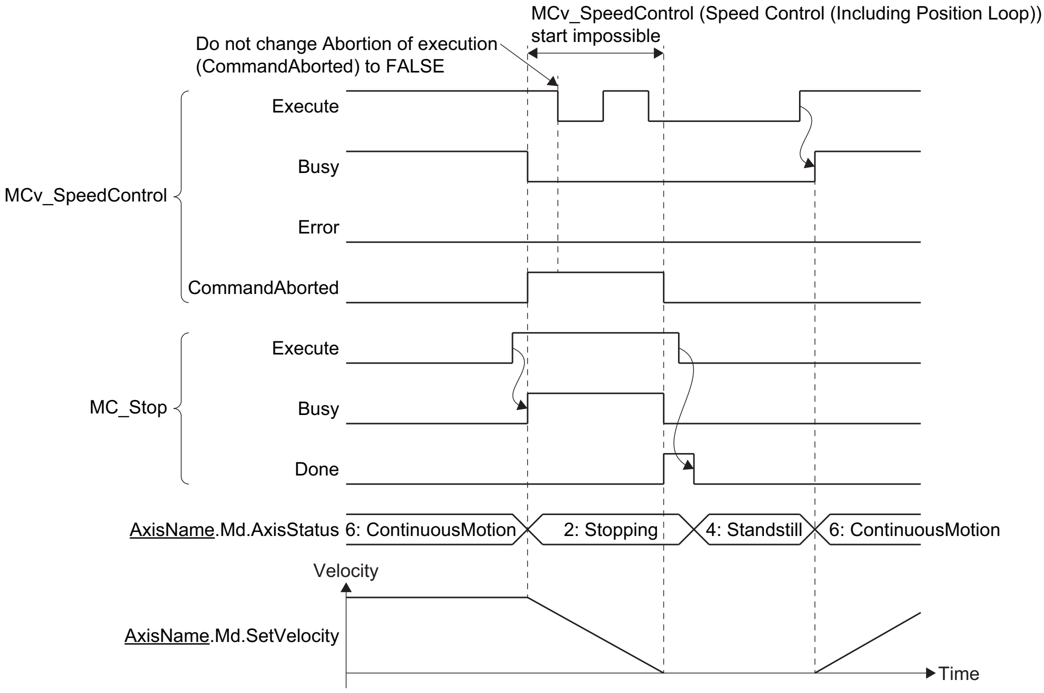


■ Operation description

- During deceleration to stop by this FB and while Execute command (Execute) of the input variable is TRUE, any operation command other than MC_Stop (Forced Stop) is not accepted.
- If the deceleration is set to "0.0" or omitted, immediate stop will be executed.
- The acceleration/deceleration method and the jerk take over the method specified in the control being executed and decelerate.
- If MC_GroupStop (Group Forced Stop) is executed during single axis synchronous control, synchronization to the master axis will be canceled.
- When the setting value of Deceleration (Deceleration) is changed and MC_Stop (Forced Stop) is restarted
  A deceleration stop is executed from the restart point based on Deceleration (Deceleration) set in MC_Stop (Forced Stop).



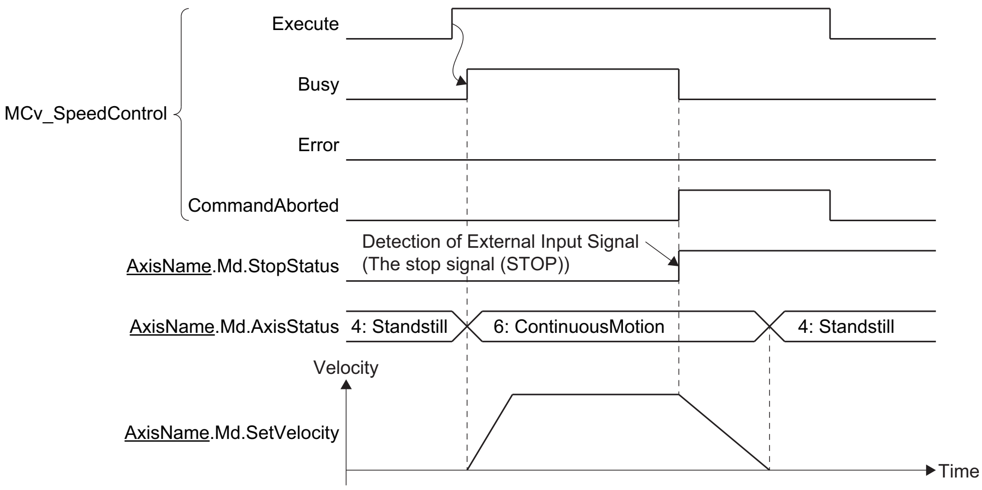
- When an FB that operates axes other than MC_Stop (Forced Stop) is executed while Axis Status (AxisName.Md.AxisStatus) is the "2: Decelerating to stop (Stopping)" status
"Start Not Possible (error code: 1AADH)" occurs and Error of MC_Stop (Forced Stop) becomes TRUE, and then Axis Status (AxisName.Md.AxisStatus) changes to "1: Stopping on error (ErrorStop)".
- After axis stop, the axis status can be transited to "4: Standby (Standstill)" by executing Axis Error Reset (AxisName.Cd.ErrorReset).
- Even if Execute command (Execute) of command aborted FB by MC_Stop (Forced Stop) set to FALSE, once the stop operation is started, TRUE of Abortion of execution (CommandAborted) will be continued until it is completed. After it stops completely, changes Abortion of execution (CommandAborted) to FALSE if Execute command (Execute) of command aborted FB is FALSE.



"The stop signal (STOP)" of external input signal is TRUE

The under-control FB is stopped by setting the stop signal (STOP) set by Stop Signal (AxisName.Pr.StopSignal) to TRUE.

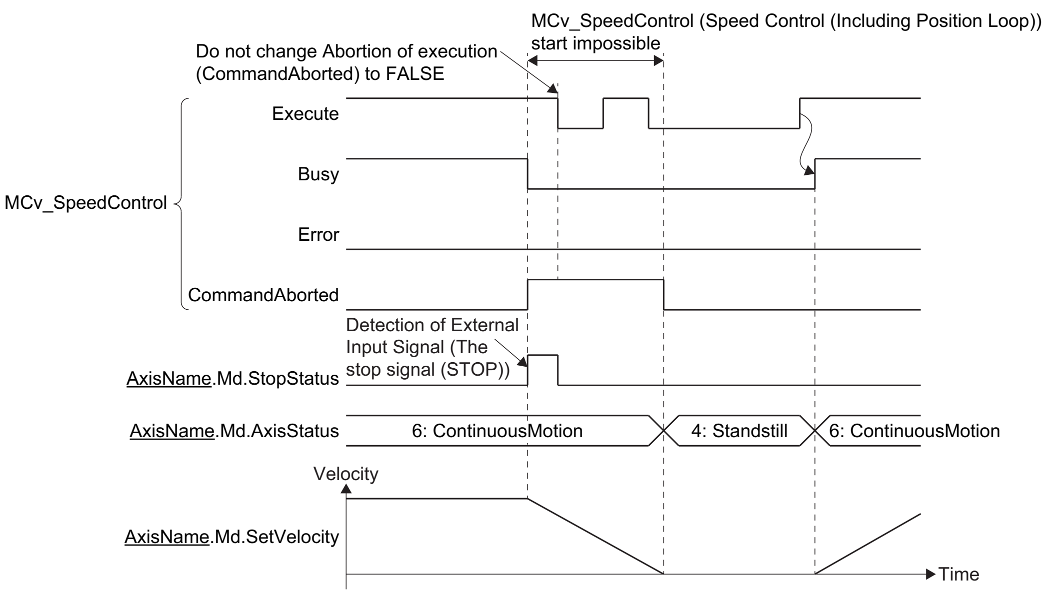
- The operation stops by the operation set in Stop Selection at Stop Cause Occurrence (AxisName.Pr.StopMode_General).
- The on-going FB sets Abortion of execution (CommandAborted) to TRUE by changing the stop signal (STOP) to TRUE. When the speed reached 0, Axis Status (AxisName.Md.AxisStatus) transits to "4: Standby (Standstill)".



Point

- When the process is stopped by TRUE of Stop Signal (AxisName.Pr.StopSignal), Axis Status (AxisName.Md.AxisStatus) after stop will not change.
- The deceleration can be changed by executing MC_Stop (Forced Stop) while process stop is executed by TRUE of Stop Signal (AxisName.Pr.StopSignal).
- The deceleration can be changed when Stop Signal (AxisName.Pr.StopSignal) is TRUE during MC_Stop (Forced Stop) execution. In this case, Axis Status (AxisName.Md.AxisStatus) after deceleration stop will not change.

- Even if Execute command (Execute) of command aborted FB is set to FALSE by changing the stop signal (STOP) to TRUE, once the stop operation is started, TRUE of Abortion of execution (CommandAborted) will be continued until it is completed. After it stops completely, changes Abortion of execution (CommandAborted) to FALSE if Execute command (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 describes the axes group operation when a stop cause of the configuration axis occurs.

Stop cause occurrence axis				Axes group	
Cause No.	**Stop cause**	**Axis Status (AxisName.Md.AxisStatus) at stop cause occurrence**	**Stop process**	**Axes Group Status (AxesGroupName.Md.GroupStatus)**	**Stop process**
1	"Forced stop input" is OFF	1: Stopping on error (ErrorStop)	Immediate stop*1	1: Stopping on error (GroupErrorStop)	■Stop cause occurrence axis Immediate stop ■Other configuration axes Axes group deceleration stop/immediate stop (Follows Configuration Axes Operation Selection at Axis Stop Cause Occurrence (AxesGroupName.Pr.StopMode_ErrorInGroup))
2	Drive unit power supply is OFF	1: Stopping on error (ErrorStop)			
	Drive unit network disconnection detection				
	Drive unit error				
	Forced stop input to the drive unit				
	Enable (Enable) of MC_Power (Operation Available) is FALSE				
	Servo ON request (ServoON) of MC_Power (Operation Available) is FALSE (When Process Selection at Servo OFF Command During Operation (AxisName.Pr.StopMode_ServoOff) is "4: Servo OFF After Immediate Stop (ServoOffAfterImmediateStop)")				
3	Hardware stroke limit upper/ lower limit error occurrence in the configuration axis	1: Stopping on error (ErrorStop)	Based on the axes group stop process		Operation differs depending on the setting value of Stop Selection at Hardware Stroke Limit Error Occurrence (AxisName.Pr.StopMode_HwStrokeLimit) of the configuration axis. ■When the configuration axis has an axis set to "1: Immediate Stop (ImmediateStop)" Immediately stop ■When the configuration axis does not have an axis set to "1: Immediate Stop (ImmediateStop)" Axes group deceleration stop/immediately stop (Follows Stop Selection at Stop Cause Occurrence (AxesGroupName.Pr.StopMode_General))
4	CPU module error occurrence	1: Stopping on error (ErrorStop)			Axes group deceleration stop/immediate stop (Follows Stop Selection at Stop Cause Occurrence (AxesGroupName.Pr.StopMode_General))
	PLC READY is OFF				
	Cycle over error occurrence				
5	Software stroke limit upper/lower limit error occurrence in the configuration axis	1: Stopping on error (ErrorStop)			Operation differs depending on the setting value of Stop Selection at Software Stroke Limit Error Occurrence (AxisName.Pr.StopMode_SwStrokeLimit) of the configuration axis. ■When the configuration axis has an axis set to "1: Immediate Stop (ImmediateStop)" Immediately stop ■When the configuration axis does not have an axis set to "1: Immediate Stop (ImmediateStop)" Axes group deceleration stop/Immediately stop (Follows Stop Selection at Stop Cause Occurrence (AxesGroupName.Pr.StopMode_General))

Stop cause occurrence axis				Axes group	
Cause No.	**Stop cause**	**Axis Status (AxisName.Md.AxisStatus) at stop cause occurrence**	**Stop process**	**Axes Group Status (AxesGroupName.Md.GroupStatus)**	**Stop process**
6	Axis error detection	1: Stopping on error (ErrorStop)	Based on the axes group stop process	1: Stopping on error (GroupErrorStop)	Axes group deceleration stop/immediate stop (Follows Stop Selection at Stop Cause Occurrence (AxesGroupName.Pr.StopMode_General))
7	Execute command (Execute of MC_Stop (Forced Stop) is TRUE	1: Stopping on error (ErrorStop)*2			
8	"The stop signal (STOP)" of external input signal is TRUE	No change		No change	

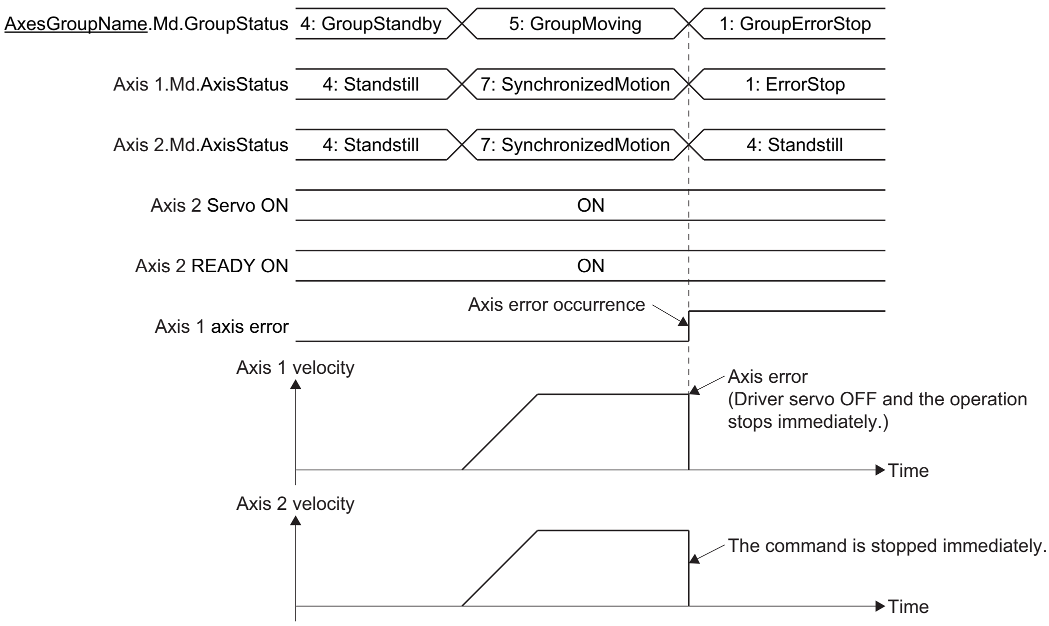
*1 The servo OFF is executed in the driver side and the axis immediately stops, and also the motion side command is stopped.
*2 If a single axis FB is executed during axes group operation, an axis error will occur and the single axis FB will be disabled. For details, refer to the following.
☞ Page 182 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)

"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 Configuration Axes Operation Selection at Axis Stop Cause Occurrence (AxesGroupName.Pr.StopMode_ErrorInGroup).
When a stop cause has occurred in a configuration axis during the axes group operation, the following methods show how to stop the other configuration axes.

- 1: Immediate Stop (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.



■ Axis error

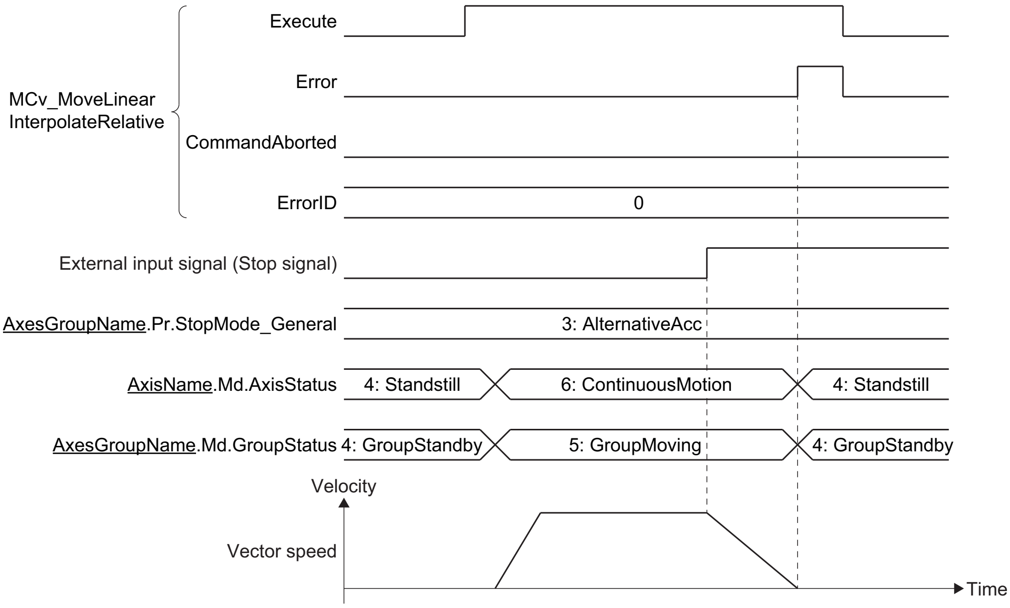
If an axis error has occurred in the configuration axis, "Stop Cause of Axes Group Configuration Axis (error code: 1A3DH)" will occur in the axes group. In this case, the axes group will pass through the path and stop according to the setting of Stop Selection at Stop Cause Occurrence (AxesGroupName.Pr.StopMode_General).

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: Stopping on error (ErrorStop)" status.

By an error occurrence in the configuration axis, "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 Stop Selection at Stop Cause Occurrence (AxesGroupName.Pr.StopMode_General).

■ "The stop signal (STOP)" of external input signal is TRUE

When Stop Signal (AxisName.Pr.StopSignal) is set to TRUE to the configuration axis, the axis stops according to the setting of Stop Selection at Stop Cause Occurrence (AxesGroupName.Pr.StopMode_General). The axes group will pass through the path and decelerate to a stop. When the speed reaches 0, Abortion of execution (CommandAborted will become TRUE, and Axes Group Status (AxesGroupName.Md.GroupStatus) will transit to "4: Standby (GroupStandby)". The value set in Deceleration at Stop (AxesGroupName.Pr.StopMode_Deceleration) will be used for the deceleration at this time.



Point

- When stopping by setting Stop Signal (AxisName.Pr.StopSignal) to TRUE, Axes Group Status (AxesGroupName.Md.GroupStatus) after stop will not change.
- When the stop processing is being executed by setting Stop Signal (AxisName.Pr.StopSignal) to TRUE, the deceleration can be changed by executing MC_GroupStop (Group Forced Stop).
- The deceleration can be changed by setting Stop Signal (AxisName.Pr.StopSignal) to TRUE during MC_GroupStop (Group Forced Stop) execution. In this case, 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, "Stop Cause of Axes Group Configuration Axis (error code: 1A3DH)" occurs in the axes group and its status changes to the "1: Stopping on error (GroupErrorStop)".

- When the stop cause "Hardware stroke limit error" is occurring in any axis of configuration axes, "1: Immediate Stop (ImmediateStop)" is set to Stop Selection at Hardware Stroke Limit Error Occurrence (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: Immediate Stop (ImmediateStop)" is set to Stop Selection at Software Stroke Limit Error Occurrence (AxisName.Pr.StopMode_SwStrokeLimit) in one or more axes.

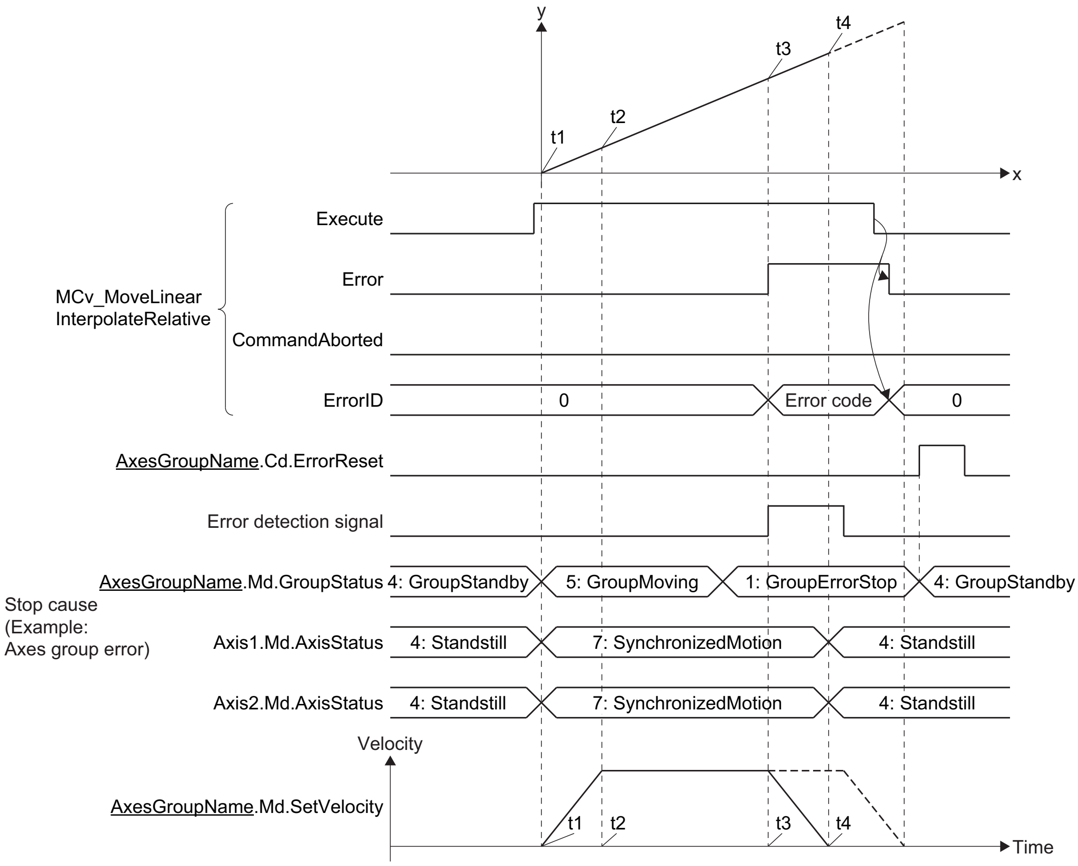
Hardware stroke limit error	Software stroke limit error
• FLS Signal Detection (at Start) (error code: 1A2DH) • RLS Signal Detection (at Start) (error code: 1A2EH) • FLS Signal Detection (Controlling) (error code: 1A2FH) • RLS Signal Detection (Controlling) (error code: 1A30H)	• Software Stroke Limit Over (Target Position) (error code: 1A00H) • Software Stroke Limit Over (Start Position) (error code: 1A01H) • Software Stroke Limit Over (Forward Direction) (error code: 1A03H) • Software Stroke Limit Over (Reverse Direction) (error code: 1A04H)

4

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 Stop Selection at Stop Cause Occurrence (AxesGroupName.Pr.StopMode_General). The value set in Deceleration at Stop (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

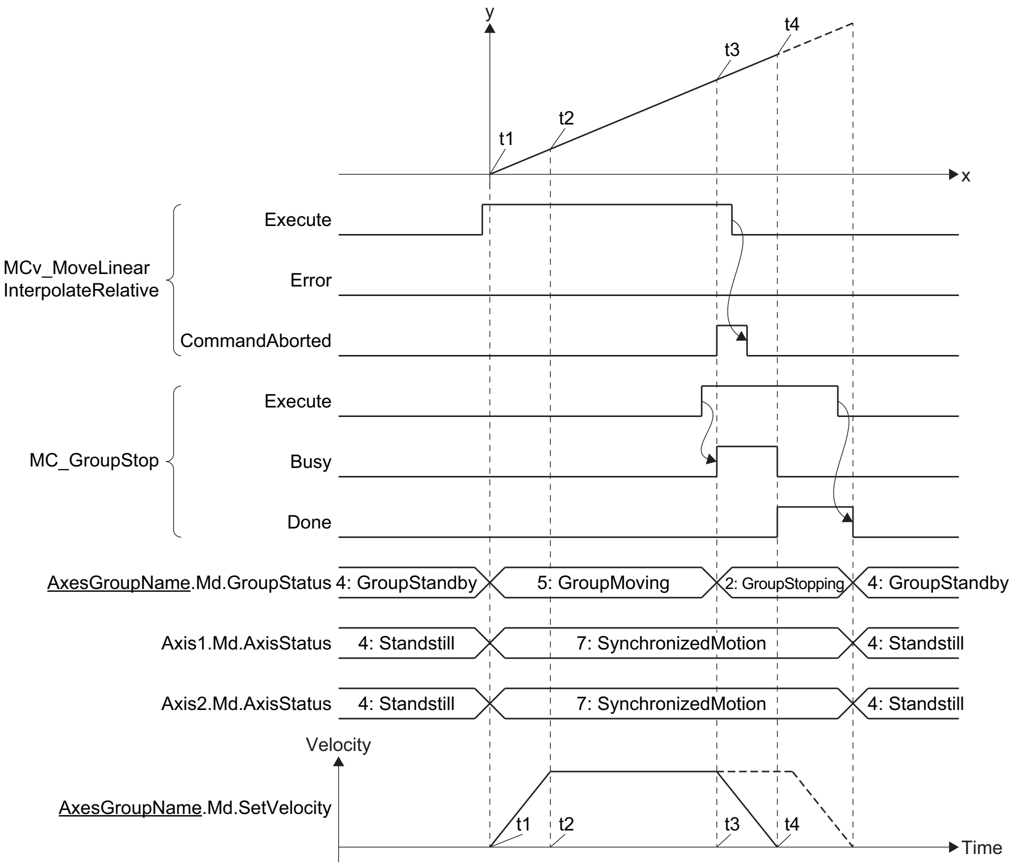


MC_GroupStop (Group Forced Stop)

In MC_GroupStop (Group Forced Stop), Deceleration (Deceleration) is set and the under-control FB is decelerated to stop. The axes group passes through the previous operation path and decelerates to a stop.

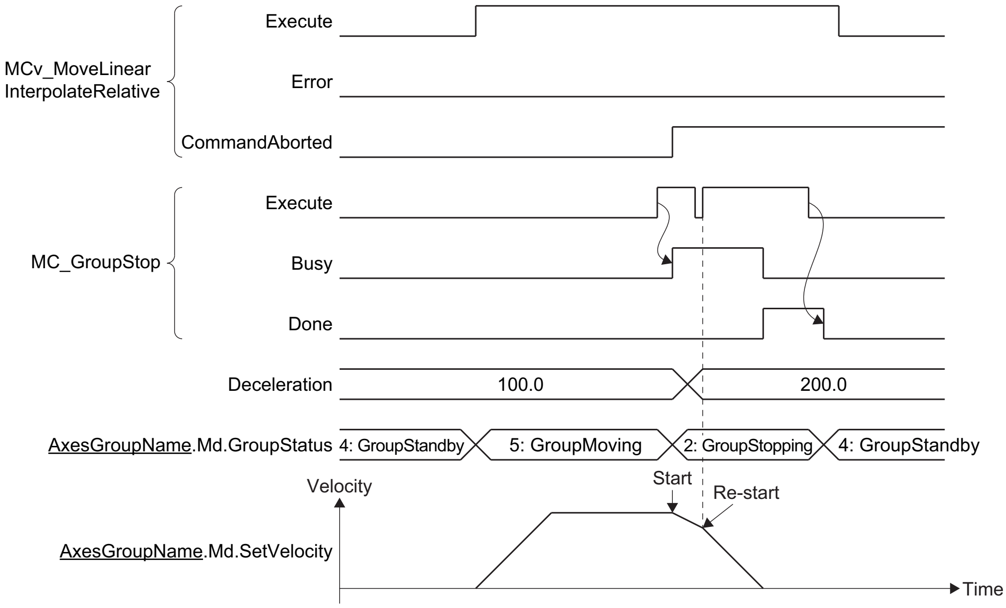
When MC_GroupStop (Group Forced Stop) is executed, Abortion of execution (CommandAborted) of the on-going FB becomes TRUE and Axes Group Status (AxesGroupName.Md.GroupStatus) transits to the "2: Decelerating to stop (GroupStopping)" status. While Execute command (Execute) is TRUE or while the speed does not reach 0, the "2: Decelerating to stop (GroupStopping)" status is maintained. When Execution completion (Done) becomes TRUE and Execute command (Execute) becomes FALSE at stop completion, the axes group status changes to the "4: Standby (GroupStandby)" status.

■ MC_GroupStop (Group Forced Stop) execution during MCv_MoveLinearInterpolateRelative (Relative Value Linear Interpolation Control)

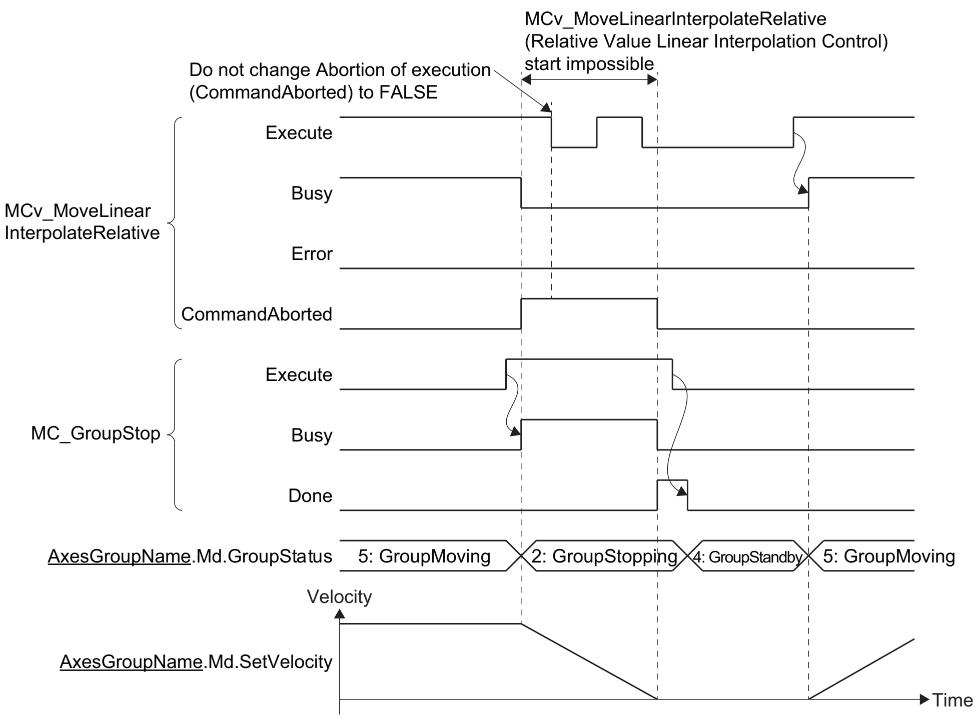


■ Operation description

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



- When an FB that operates axes other than MC_GroupStop (Group Forced Stop) is executed while Axes Group Status (AxesGroupName.Md.GroupStatus) is the "2: Decelerating to stop (GroupStopping)" status, "Start Not Possible (error code: 1AADH)" occurs and Error of MC_GroupStop (Group Forced Stop) becomes TRUE, and then Axes Group Status (AxesGroupName.Md.GroupStatus) changes to "1: Stopping on error (GroupErrorStop)".
- Even if Execute command (Execute) of command aborted FB is set to FALSE by MC_GroupStop (Group Forced Stop), once the stop operation is started, Abortion of execution (CommandAborted) will remain TRUE until it is completed. After it stops completely, changes Abortion of execution (CommandAborted) to FALSE if Execute command (Execute) of command aborted FB is FALSE.



Precautions

Relevant add-ons

The following add-on is required to use this function.

- Axis
- MotionEngine
- ExternalSignal[*1]
- MotionControl_General[*2]
- ServoDriver_CANopen[*3]

*1 For using the stop signal, the forced stop signal, the all axes forced stop signal, and the upper/lower limit signal

*2 For using the following FBs
· MC_Stop (Forced stop)
· MC_GroupStop (Group Forced Stop)

*3 For using the real drive axis or the real encoder axis

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 is valid for the following axis types which supports Forced Stop Signal (AxisName.Pr.ForcedStop_Signal).
  - · Real drive axis
  - · Virtual drive axis
- The all axes forced stop signal is valid for the following axes including the types which do not support Forced Stop Signal (AxisName.Pr.ForcedStop_Signal).
  - · Real drive axis
  - · Virtual drive axis
  - · Virtual linked axis
- Refer to the driver device specification 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 device station at the same time. For operation when "Quick Stop" is issued, refer to the manual of the device station.

Operation of this function for each system status

○: Possible, △: Possible (restricted)

Status	Operation availability
STOP	△*1
RUN	○
Moderate error	△*2
Major error	△*2

*1 Since the Forced Stop Signal (AxisName.Pr.ForcedStop_Signal) cannot be read, if it has never been set to RUN, operation will not be possible.

*2 Depending on the error, it may not be possible to issue QuickStop.
When a moderate or major error occurs, the motion system notifies the device station that an error has been detected in the CPU operating status. For details on the operation of the device station when an error is detected, refer to each device station manual.

Relevant variables

Axis parameter (AxisName.Pr.)

Variable/Structure name	Name	Details
ForcedStop_Signal	Forced Stop Signal	Sets a signal using the each axis forced stop. Forced Stop Signal (ForcedStop_Signal) is the structure of SIGNAL_SELECT type. The fetch timing of the setting 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 381 External Signal Selection The specific setting and the operation of this signal are shown below. ■I/O Number (StartIO) Ignores the input value. ■Target (Target) When the specification is not set, signal disabled is determined and the signal undetection status is always set. Only [VAR] , [DEV] , and [CONST] can be specified for the data type. *: When unavailable data is set, "Out of Parameter Range (Axis) (error code: 1D80H)" occurs. ■Signal Detection Method (Detection) The specification is allowed at the following level detection only. • 0: Detection at TRUE (HighLevel) • 1: Detection at FALSE (LowLevel) *: When edgedetection specification is set, "Out of Parameter Range (Axis) (error code: 1D80H)" occurs. ■Compensation Time (CompensationTime) Ignores the input value. ■Filter Time (FilterTime) Setting range of the filter time is 0.0 to +5.0. *: When a value outside the range is set, "Out of Filter Time Setting Range of Each Axis Signal Warning (warning code: 0D24H)" occurs. and the axis operates with 0.0 of the filter time.

Axis monitor data (AxesGroupName.Md.)

Variable/Structure name	Name	Details
ForcedStop_Released	Forced Stop Cancelling	Displays the forced stop cancel status. • FALSE: Forced stop input TRUE (Forced stop) • TRUE: Forced stop input FALSE (Forced stop release)
ForcedStop_Signal	Forced Stop Signal	Displays the input status of axis forced stop signal. Forced Stop Signal (ForcedStop_Signal) is the structure of SIGNAL_SELECT type. For details on SIGNAL_SELECT type, refer to the following. ☞ Page 381 External Signal Selection The specific output of this signal is shown below. ■I/O Number (StartIO) "0" is always displayed. ■Target (Target) The fetch result of the parameter is displayed. ■Signal Detection Method (Detection) The fetch result of the parameter is displayed. ■Compensation Time (CompensationTime) "0.0" is always desplayed. ■Filter Time (FilterTime) The fetch result of the parameter is displayed.

Axes group parameter (AxesGroupName.Pr.)

Variable/Structure name	Name	Details
StopMode_ErrorInGroup	Configuration Axes Operation Selection at Axis Stop Cause Occurrence	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: Immediate Stop (ImmediateStop) (Default value)

System parameter (System.Pr.)

Variable/Structure name	Name	Details
ForcedStop_Signal	All Axes Forced Stop Signal	Sets a signal using the all axes forced stop. All Axes Forced Stop Signal (ForcedStop_Signal) is the structure of SIGNAL_SELECT type. The fetch timing of the setting is at Ready ON, the detection timing of the signal is the first operation cycle, and the timing for starting the stop processing is the axis operation cycle. For details on SIGNAL_SELECT type, refer to the following. ☞ Page 381 External Signal Selection The specific setting and the operation of this signal are shown below. ■I/O Number (StartIO) "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. Only [VAR] , [DEV] , and [CONST] can be specified for the data type. *: When unavailable data is set, "Out of Parameter Range(System) (error code: 1D82H)" occurs. ■Signal Detection Method (Detection) The specification is allowed at the following level detection only. • 0: Detection at TRUE (HighLevel) • 1: Detection at FALSE (LowLevel) *: When edge detection specification is set, "Out of Parameter Range (System) (error code: 1D82H)" occurs. ■Compensation Time (CompensationTime) Ignores the input value. ■Filter Time (FilterTime) 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 TimeSetting Range of System Signal Warning (warning code: 0F0FH)" is output occurs and the axis operates with 0.0 of the filter time.

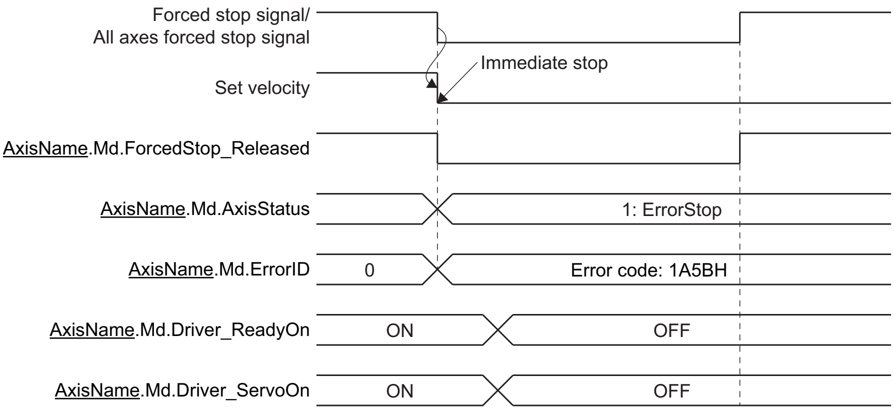
System monitor data (System.Md.)

Variable/Structure name	Name	Details
ForcedStop_Released	Forced Stop Cancelling	Displays the forced stop cancel status. • 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. All Axes Forced Stop Signal (ForcedStop_Signal) is the structure of SIGNAL_SELECT type. For details on SIGNAL_SELECT type, refer to the following. ☞ Page 381 External Signal Selection The specific output of this signal is shown below. ■I/O Number (StartIO) "0" is always displayed. ■Target (Target) The fetch result of the parameter is displayed. ■Signal Detection Method (Detection) The fetch result of the parameter is displayed. ■Compensation Time (CompensationTime) "0.0" is always displayed. ■Filter Time (FilterTime) The fetch result of the parameter is displayed.

Operation at the forced stop occurrence

This section describes the operation when a forced stop has occurred.
An example of a device station which servo OFF is executed when "Quick Stop" is issued is connected is shown below.

During axis operation



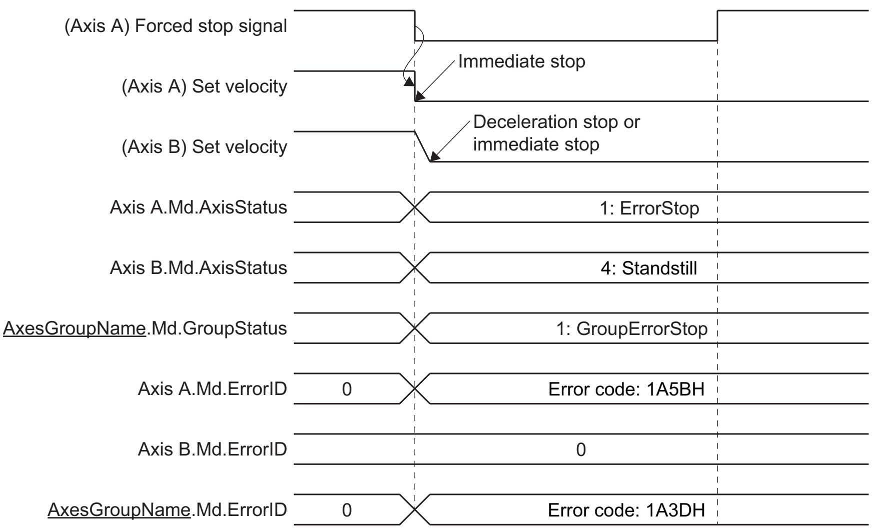
- In the axis that a forced stop occurred, MC_Power is in the error status and the servo OFF is executed. The servo ON will not be executed until the forced stop is released. The error is not output when the axis is not operated.
- In positioning or stopping, "Forced stop detection" is registered in the event history. "Forced Stop Status (error code: 1A5BH)" is also registered in positioning.
- When the axis is in the forced stop status at start, the command will not be issued, and it will be in the same "1: Stopping on error (ErrorStop)" status as the forced stop that is in operation.

During stop

When the axis the forced stop occurred in is during stop, "Forced Stop Status (error code: 1A5BH)" will not occur. 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.



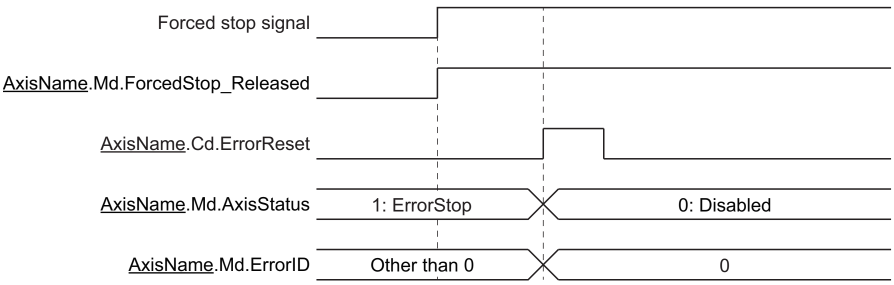
The stop method of the axis B can be set in Configuration Axes Operation Selection at Axis Stop Cause Occurrence (AxesGroupName.Pr.StopMode_ForcedStopInGroup). For the operation of each setting, refer to the following.

☞ Page 180 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.

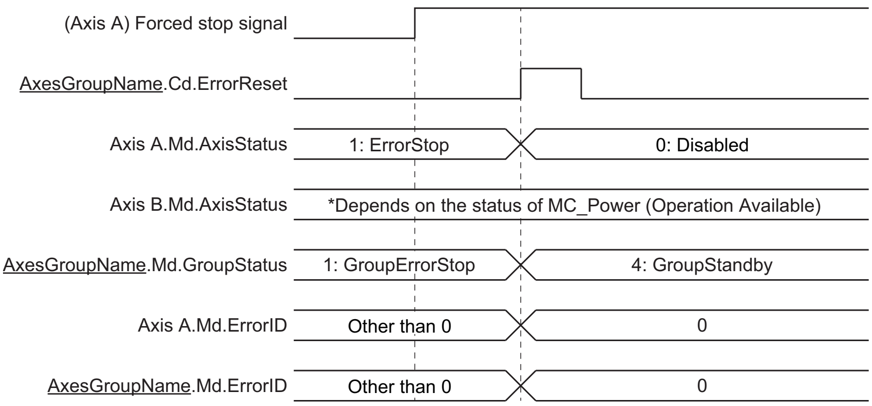
- The forced stop input changes to "1".
- Axis Status (AxisName.Md.AxisStatus) is still "1: Stopping on error (ErrorStop)".

2. Execute the axis error reset.

- Axis Status (AxisName.Md.AxisStatus) changes to "0: Axis disabled (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".
- Axes Group Status (AxesGroupName.Md.GroupStatus) is still "1: Stopping on error (GroupErrorStop)".

2. Execute the axes group error reset.

- Axes Group Status (AxesGroupName.Md.GroupStatus) changes to "4: Standby (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 (Operation Available) continues.

Check method

Whether the forced stop input is in the TRUE/FALSE status can be checked with Forced Stop Cancelling (AxisName.Md.ForcedStop_Released).

Precautions

Relevant add-ons

The following add-ons are required to use this function.

- Axis
- MotionEngine
- ExternalSignal

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.
The address information stored in the motion system cannot be guaranteed while the Homing Request (AxisName.Md.Homing_Request) is TRUE.
Homing Request (AxisName.Md.Homing_Request) is FALSE and Homing Completed (AxisName.Md.Homing_Complete) is TRUE if the homing is executed and is completed normally.

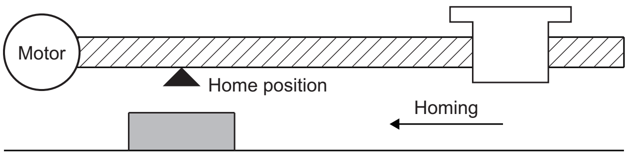
⚠WARNING

- When using an absolute position system, execute a homing 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.

5

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

Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
Homing_Status	Homing Operation Status	Stores the homing status of the driver.*1
Homing_Request	Homing Request	Becomes TRUE when homing is required and becomes FALSE when it is completed. For details, refer to the following. ☞ Page 194 Homing request
Homing_Complete	Homing Completed	Becomes TRUE when homing is normally completed, and it becomes FALSE when operation starts and homing is required.

*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

Relevant FBs

For details on Motion control FBs, refer to the following.

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

Motion control FB	Name	Description
MC_Home	OPR	Executes homing of the specified axis.

Homing request

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 Homing Request (AxisName.Md.Homing_Request) is TRUE is recorded in event history.

For causes of which Homing Request (AxisName.Md.Homing_Request) becomes TRUE, refer to the following.

☞ Page 97 Homing request

Homing Request (AxisName.Md.Homing_Request) becomes FALSE at homing completion.

When homing is not required

Execute a homing request clear in systems that do not require homing. Homing Request (AxisName.Md.Homing_Request) becomes FALSE by performing the homing request clear. For details, refer to the following.

☞ Page 200 Operation Setting for Incompletion of Homing

Precautions

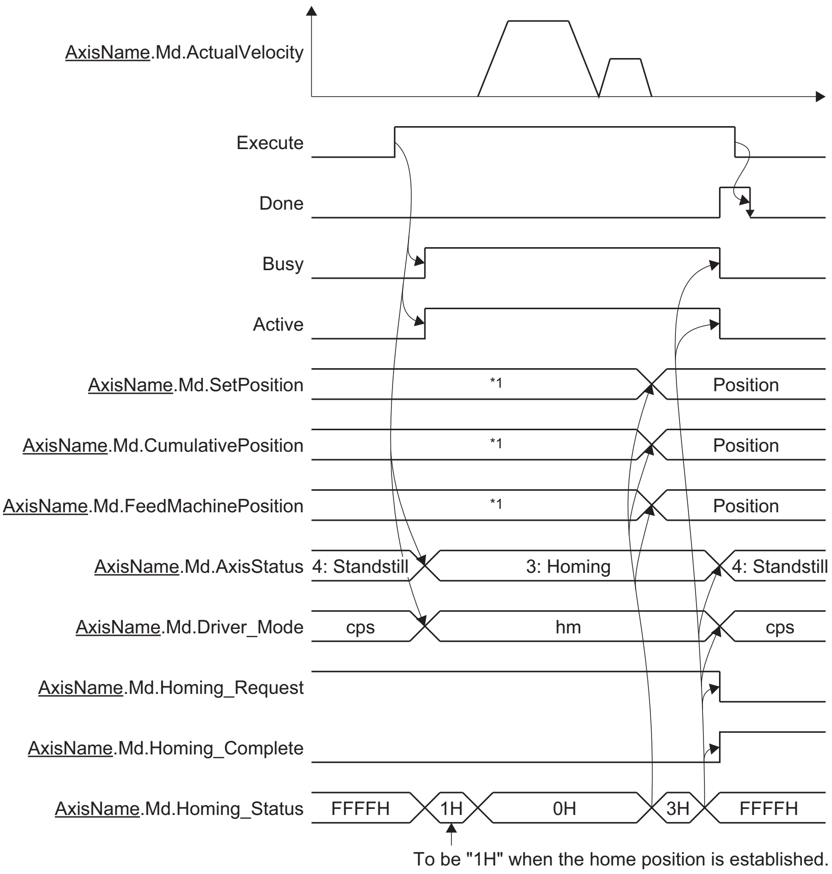
Directly rewriting Homing Request (AxisName.Md.Homing_Request) to FALSE with a program, etc. will not clear the homing request in the Motion system. Be sure to use the homing request clear to set Homing Request (AxisName.Md.Homing_Request) to FALSE.

Driver homing method

The driver is switched to the Homing mode, and the homing is executed based on the positioning pattern set on the driver side. Change the homing data of the driver with MC_WriteParameter (Parameter Write) to change the homing method or each 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



*1 When "0: Follow-up disabled" is selected in Follow-up enabled/disabled selection (Options (Options): Bit 16), it will not be updated during homing. When "0: Follow-up enabled" is selected, it will be updated at being changed the current position during homing.

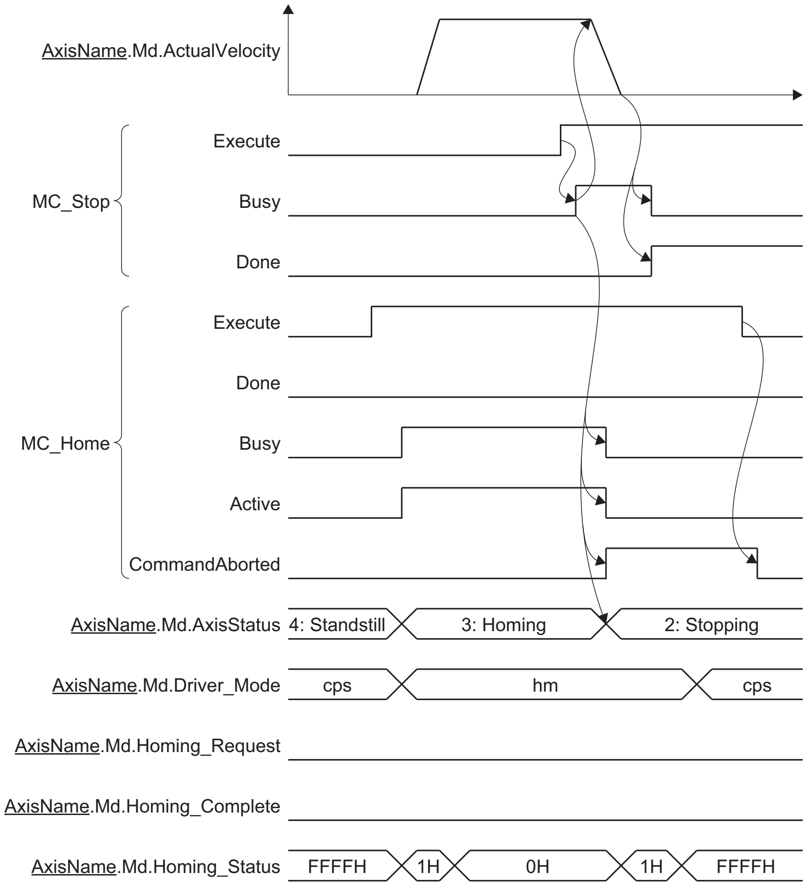
■ When an error occurs

For details when an error occurs, refer to "Basic operation of Execute command (Execute) type Motion control FBs" in the following.

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

5

■ When MC_Stop (Forced Stop) is executed



- When MC_Stop (Forced Stop) is set to 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 device specifications. Therefore, Deceleration (Deceleration) and Jerk (Jerk) of MC_Stop (Forced 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 Home position switch (AbsSwitch)) are assigned, the external signals are transmitted to the driver. However, when the position accuracy is required in the proximity dog based (not Z-phase based) method, using the driver built-in DI is recommended.
- When "0: Follow-up disabled" is selected in Follow-up enabled/disabled selection (Options (Options): Bit 16), the follow up of set position is not performed during driver homing method.
  In this case, do not perform synchronous control with the setting the axis during homing as the master axis. In the slave axis, "Velocity Range Over during Controlling (error code: 1AE8H)" may occur.
- When "1: Follow-up enabled" is selected in Follow-up enabled/disabled selection (Options (Options): Bit 16) and synchronous control is performed with the setting the axis during homing as the master axis, the slave axis operation is as follows depending on Master axis data source selection (MasterValueSource).

Master axis data source selection (MasterValueSource)	Slave axis operation during synchronous control
1: Set Value (mcSetValue) 101: Latest Set Value (mcLatestSetValue)	Continues synchronous control after homing completion.
2: Actual Value (mcActualValue) 102: Latest Actual Value (mcLatestActualValue)	Although the slave axis synchronizes to the master axis until homing completion, synchronous control may not be performed and the servo alarm [AL. 031.1_Servo motor speed error] or the servo alarm [AL. 035.1_Command frequency error] may occur at completion.

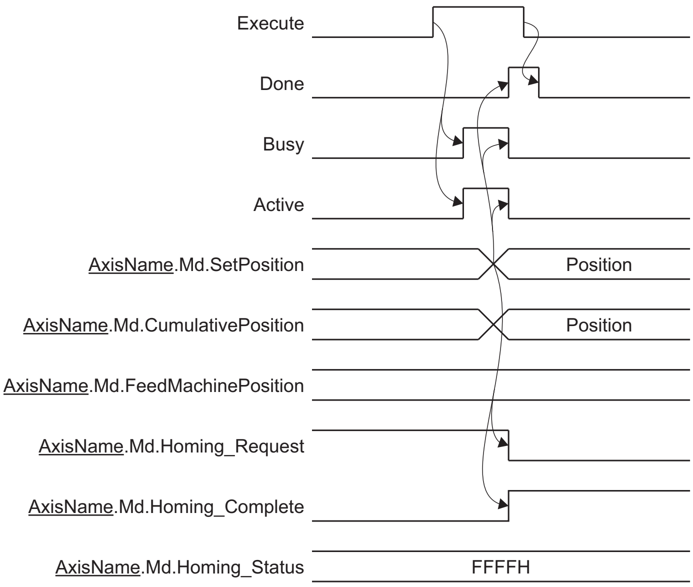
- If the stop cause occurs when Homing Operation Status (AxisName.Md.Homing_Status) is set to "0002H: Homing completed (target is not reached to the target position)", the set position executed follow-up changed significantly. Therefore, the servo alarm [AL. 031.1_Servo motor speed error] or the servo alarm [AL. 035.1_Command frequency error] may occur in the slave axis even if "1: Set Value (mcSetValue)" or "101: Latest Set Value (mcLatestSetValue)" is selected in Master axis data source selection (MasterValueSource).
- 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 838 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 Hardware Stroke Limit Override (AxisName.Cd.HwStrokeLimit_Override) to DISABLE (Check disabled).
- If ONLY_INSIDE (Check disabled for movement returning to the range) is set in Hardware Stroke Limit Override (AxisName.Cd.HwStrokeLimit_Override) at homing start, "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 device station side. It is completed in the motion system, and external signals, etc. are not used. Target position (Position) (home position address) at homing is registered in the motion system as the home position, and Set Position (AxisName.Md.SetPosition) and Cumulative Current Position (AxisName.Md.CumulativePosition) are rewritten to Target position (Position) (home position address).

Timing chart

■ Normal operation



■ When an error occurs

For details when an error occurs, refer to "Basic operation of Execute command (Execute) type Motion control FBs" in the following.

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

Precautions

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

Home position return method	Description
Driver homing method	The software stroke limit check is not performed.
Data set homing method	The software stroke limit check is performed.

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

Relevant add-ons

The following add-ons are required to use this function.

- Axis
- MotionEngine
- MotionControl_General
- NetworkDriver_CCIETSN*1
- ServoDriver_CANopen*1
- SignalIO*2

*1 When using the real axis

*2 When using the Home position switch (AbsSwitch)

Error

Check that Axis Status (AxisName.Md.AxisStatus) is "4: Standby (Standstill)" at MC_Home (Homing) execution. If it is other than "4: Standby (Standstill)", an error will occur and the control during operation will stop.

5

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, 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 Start Permission at Homing Uncompleted (AxisName.Pr.StartableAtUnhomed).

○: Axis start possible, ×: Axis start impossible

FB	Start Permission at Homing Uncompleted (AxisName.Pr.StartableAtUnhomed)	
	"TRUE: Allow" and the homing request (AxisName.Md.Homing_Request) is TRUE	"FALSE: Not allowed" and the homing request (AxisName.Md.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.
For MR-J5(W)-G: MR-J5 User's Manual (Function)

Relevant variables

Axis parameter (AxisName.Pr.)

Variable/Structure name	Name	Details
StartableAtUnhomed	Start Permission at Homing Uncompleted	Sets whether axis start is allowed or not when homing is incompletion is incomplete. • FALSE: Disabled • TRUE: Enabled

Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
Homing_Request	Homing Request	Becomes TRUE when homing is required and becomes FALSE when it is completed. For details, refer to the following. Page 194 Homing request

Axis control data (AxisName.Cd.)

Variable/Structure name	Name	Details
Homing_ClearRequest	Homing Request Clear	Sets the homing request to FALSE forcibly. After the homing request is set to FALSE, FALSE is set in this variable automatically. • FALSE: Not executed • TRUE: Executes the homing request clear

FBs that can be started when homing is incompletion

The following FBs can start the axis regardless of Start Permission at Homing Uncompleted (AxisName.Pr.StartableAtUnhomed) and Homing Request (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 Start Permission at Homing Uncompleted (AxisName.Pr.StartableAtUnhomed) is FALSE and Homing Request (AxisName.Md.Homing_Request) is TRUE, and "Start at Homing Incomplete (error code: 1A22H)" occurs.

5

Homing request clear

When Homing Request (AxisName.Md.Homing_Request) is TRUE, Homing Request (AxisName.Md.Homing_Request) can be changed to FALSE forcibly by setting Homing Request Clear (AxisName.Cd.Homing_ClearRequest) to TRUE.

Precautions

- In Homing Request Clear (AxisName.Cd.Homing_ClearRequest), when turning Homing Request (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 Homing Request Clear (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

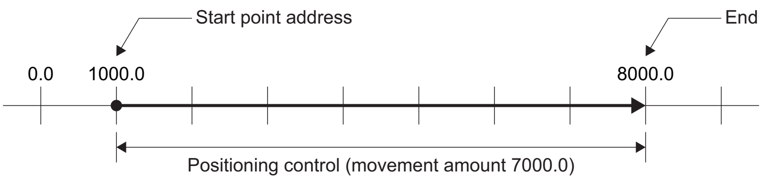
For details on Motion control FBs, refer to the following.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

Motion control FB	Name	Description
MC_MoveAbsolute	Absolute value positioning	Executes positioning after specifying the target position of the absolute position.
MC_MoveRelative	Relative value positioning	Executes positioning after specifying the movement amount of the relative position.

Absolute positioning control

In MC_MoveAbsolute (Absolute Value Positioning), Target position (Position), Velocity (Velocity), Acceleration (Acceleration), Deceleration (Deceleration), Jerk (Jerk), Direction selection (Direction), Buffer mode (BufferMode), and Options (Options) can be set, and positioning is executed from the current position at start (start point address) to the address set in Target position (Position) (end point address).

Ex.
When the start point address is "1000.0" and Target position (Position) is "8000.0", positioning is carried out in the positive direction for a movement amount of "7000.0 (8000.0 - 1000.0)".



Necessary slave object

When using MC_MoveAbsolute (Absolute Value Positioning), set the following slave object to the axis.
- Target position (607AH)

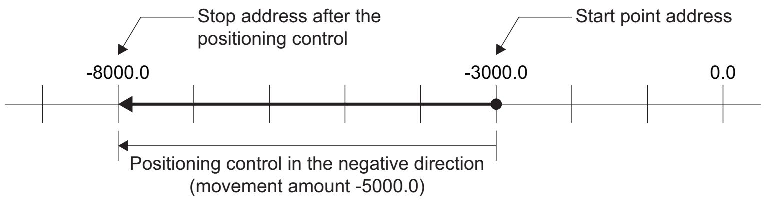
If the above slave object is not set, "Necessary Slave Object Unset (error code: 1AA8H)" will occur, and the axis will not start.
For details on the slave object setting, refer to the following.
☞ Page 56 Axis Assignment

Relative positioning control

In MC_MoveRelative (Relative Value Positioning), Movement amount (Distance), Velocity (Velocity), Acceleration (Acceleration), Deceleration (Deceleration), Jerk (Jerk), and Buffer mode (BufferMode) can be set, and positioning of movement amount set in Movement amount (Distance) is executed from the current position at start (start point address). The movement direction is determined by the sign of the movement amount. Axis Status (AxisName.Md.AxisStatus) becomes "5: During positioning operation (DiscreteMotion)".

Ex.

When the start point address is "-3000.0", and the movement amount is "-5000.0", positioning is carried out to "-8000.0".



Necessary slave object

When using MC_MoveRelative (Relative Value Positioning), set the following slave object to the axis.

- Target position (607AH)

If the above slave object is not set, "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 275 Velocity Control

Relevant FB

For details on Motion control FBs, refer to the following.

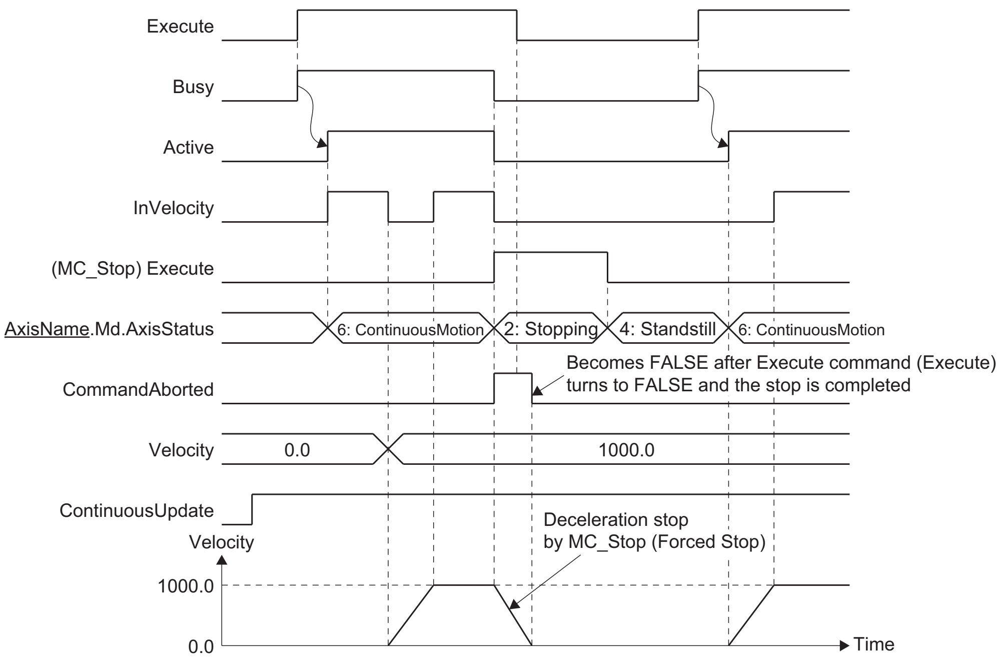
📖MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

Motion control FB	Name	Description
MCv SpeedControl	Speed control (including position loop)	Executes the speed control including position loop.

Control details

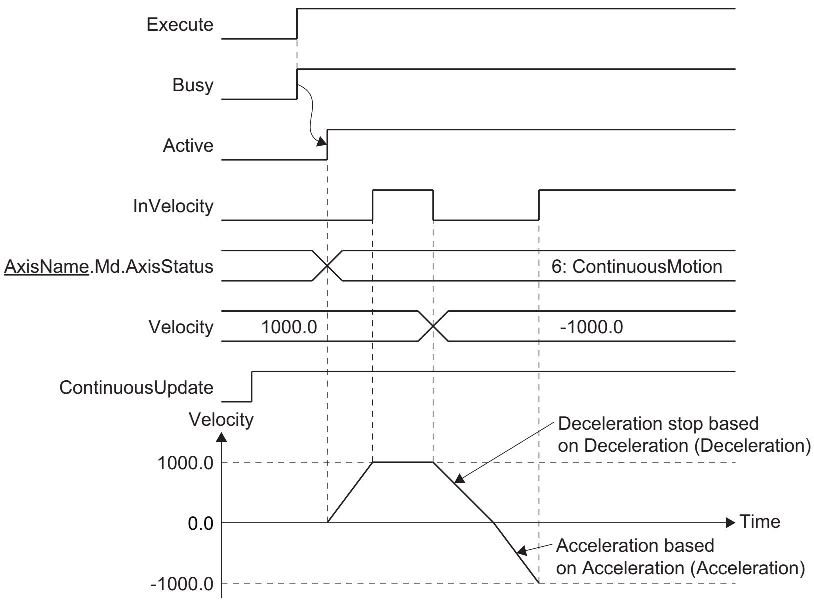
Set the control mode of the driver side to csp and execute the speed control for the specified axis with the specified speed. MCv_SpeedControl (Speed Control (Including Position Loop)) is used to execute. To stop the axis, use MC_Stop (Forced Stop) or start another motion FB.
To perform the speed control with the control mode of the driver set to csv, use MC_MoveVelocity (Speed Control), etc.

Timing chart at start and the stop cause occurrence



Timing chart when the operation direction is changed

If the sign of Velocity (Velocity) is reversed and the operation direction is changed when Continuous update (ContinuousUpdate) is TRUE, the axis accelerates toward the target velocity once after the deceleration stop.



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 133 Buffer mode type

Necessary slave object

When using Speed Control (Including Position Loop) for a single axis, set the following slave objects for the axis.

- Target position (607AH)

If the above slave object is not set, "Necessary Slave Object Unset (error code: 1AA8H)" occurs, and the velocity control of the axis does not start.
For details of slave object settings, refer to the following.

☞ Page 56 Axis Assignment

6.3 Single Axis Manual Control

This section describes the single axis manual control.

Relevant variables

Axis control data (AxisName.Cd.)

Variable/Structure name	Name	Details
SwStrokeLimit_Override	Software Stroke Limit Override	Temporarily switches valid/invalid of the software stroke limit check. • DISABLE: Check disabled • ONLY_INSIDE: Disables the check only for movement toward the setting range. • Other than the above: Check enabled (no disable request)
HwStrokeLimit_Override	Hardware Stroke Limit Override	Temporarily switches valid/invalid of the hardware stroke limit check. • DISABLE: Check disabled • ONLY_INSIDE: Disables the check only for movement toward the setting range. • Other than the above: Check enabled (no disable request)

Relevant FB

For details on Motion control FBs, refer to the following.

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

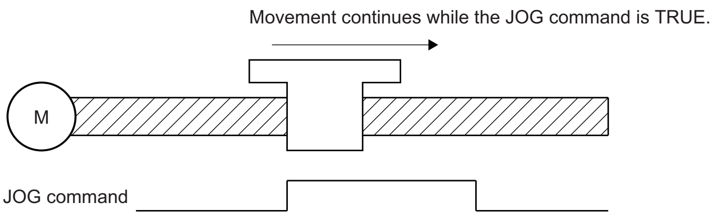
Motion control FB	Name	Description
MCv Jog	JOG operation	Executes JOG operation according to the commanded velocity.

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

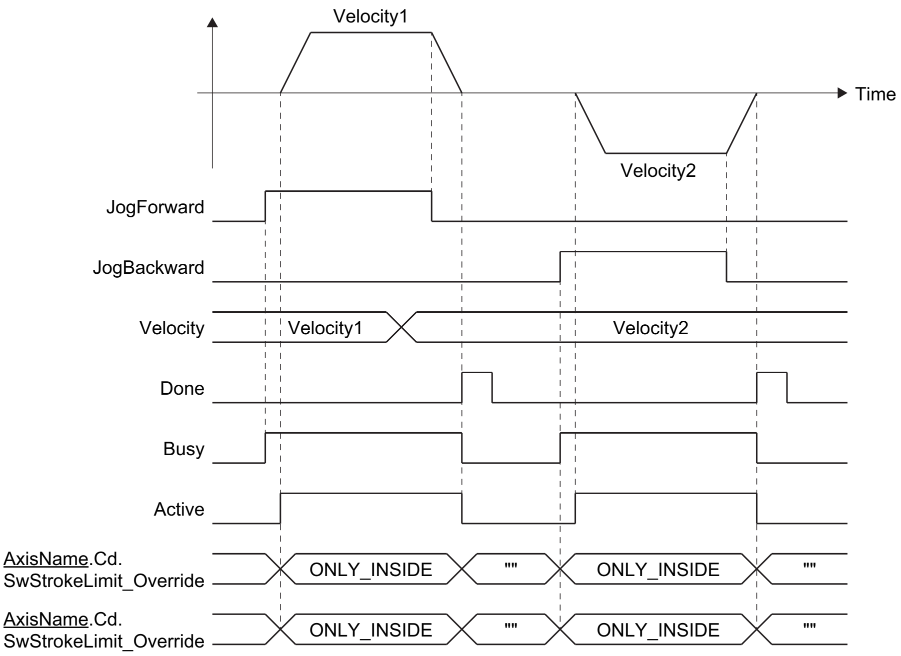
In JOG operation, the command is output from the motion system to the axis while the positive/reverse rotation JOG command is input by using MCv_Jog (JOG Operation), and then the axis operates to the specified direction.

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

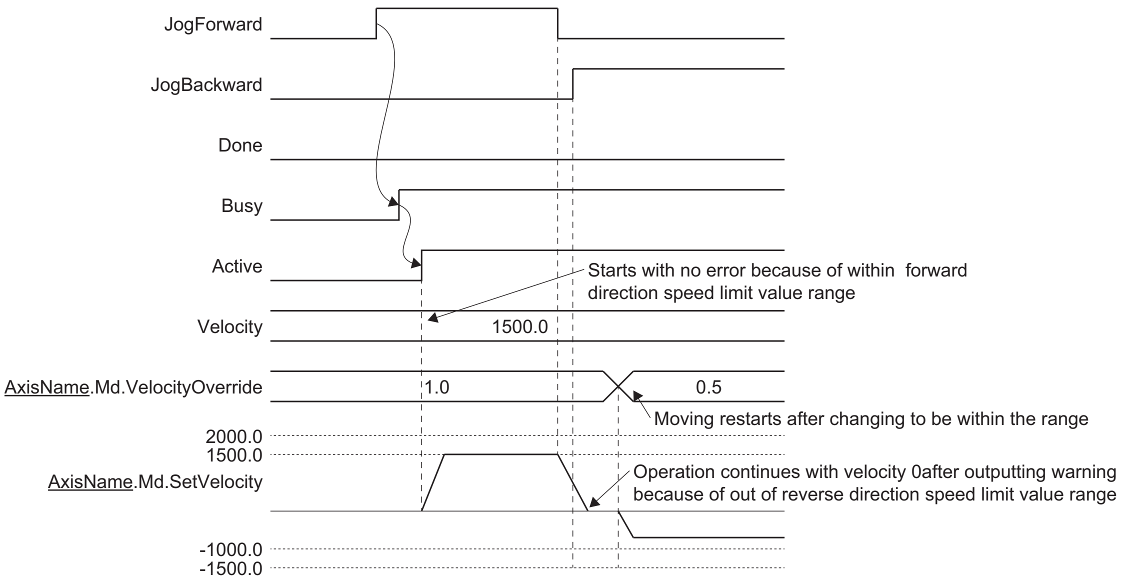
- JOG operation is carried out with MCv_Jog (JOG Operation).
- The target axis is moved to the specified direction during TRUE, by setting Positive rotation JOG command (JogForward) or Reverse rotation JOG command (JogBackward) to TRUE.
- Axis Status (AxisName.Md.AxisStatus) during JOG operation is "6: During continuous operation (ContinuousMotion)".
- A deceleration stop is executed by setting Positive rotation JOG command (JogForward) or Reverse rotation JOG command (JogBackward) to FALSE.
  Axis Status (AxisName.Md.AxisStatus) changes to "4: Standby (Standstill)" after the deceleration stop is completed.
- When Error (Error) becomes TRUE during the deceleration by setting Positive rotation JOG command (JogForward) or Reverse rotation JOG command (JogBackward) to FALSE, Error (Error) becomes TRUE until Positive rotation JOG command (JogForward) or Reverse rotation JOG command (JogBackward) is changed TRUE.
- When another operation instruction is started during a JOG operation, the operation is carried out based on the Buffer mode (BufferMode) setting.
- When a JOG operation is started during another operation instruction, the start request will be ignored and "Start during Operation Warning (warning code: 0D01H)" will occur. Start a JOG operation when Axis Status (AxisName.Md.AxisStatus) is "4: Standby (Standstill)".
- Use velocity change by override function to change the velocity during JOG operation. For details, refer to the following.
  ☞ Page 349 Override Function

Precautions

- For safety, set a small value in Velocity (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 Software Stroke Limit Override (AxisName.Cd.SwStrokeLimit_Override) is other than DISABLE (Check disabled) or ONLY_INSIDE (Check disabled for movement returning to the range), rewrites Software Stroke Limit Override (AxisName.Cd.SwStrokeLimit_Override) into ONLY_INSIDE (Check disabled for movement returning to the range) during MCv_Jog (JOG Operation) execution, and rewrites it into "" during JOG operation completion.
- When software stroke limit function is valid and Hardware Stroke Limit Override (AxisName.Cd.HwStrokeLimit_Override) is other than DISABLE (Check disabled) or ONLY_INSIDE (Check disabled for movement returning to the range), rewrites Hardware Stroke Limit Override (AxisName.Cd.HwStrokeLimit_Override) into ONLY_INSIDE (Check disabled for movement returning to the range) during MCv_Jog (JOG Operation) execution, and rewrites it into "" during JOG operation completion.
- Do not change Software Stroke Limit Override (AxisName.Cd.SwStrokeLimit_Override) and Hardware Stroke Limit Override (AxisName.Cd. HwStrokeLimit_Override) during MCv_Jog (JOG Operation) execution.
- Note that the value of Software Stroke Limit Override (AxisName.Cd.SwStrokeLimit_Override) and Hardware Stroke Limit Override (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, "Velocity Limit Value Over Warning on Direction Change (warning code: 0D20H)" occurs. Also, when the acceleration time exceeds 8400.0 [s] "Acceleration Time Over Warning on Direction Change (warning code: 0D32H)" occurs. (The operation continues with velocity 0.) The moving starts after changing control to remove the cause.

Ex.

When setting Positive Direction Velocity Limit Value (AxisName.Pr.VelocityLimit_Positive) to "2000.0", and Negative Direction Velocity Limit Value (AxisName.Pr.VelocityLimit_Negative) to "1000.0"



Necessary slave object

When using JOG operation, set the following slave objects for the axis.

- Target position (607AH)

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

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

☞ Page 56 Axis Assignment

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

Axes group information (AxesGroupName.AxesGroupRef.)

Variable/Structure name	Name	Details
GroupNo	Axes Group No.	Indicates axes group No. • 0: No setting • 1 to 10000: Set axes group No.

Axes group parameter (AxesGroupName.Pr.)

Variable/Structure name	Name	Details
Axis[1..16]	Configuration Axis	Sets Axis No. (AxisNo) of Axis Information (AxisName.AxisRef) configurating the axes group.
VelocityLimit	Velocity Limit Value	Specifies the velocity.

Axes group monitor data (AxesGroupName.Md.)

Variable/Structure name	Name	Details
NumberOfAxes	Number of Configuration Axes	Displays the number of configuration axes of the axes group.
InterpolationAxes	Interpolation Axes	Displays the configuration axis that is executing the interpolation control of the axes group.
SetVelocity	Set Velocity	Displays 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 Interpolation Axes (AxesGroupName.Md.InterpolationAxes).

Relevant FBs

For details on Motion control FBs, refer to the following.
📖MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

Motion control FB	Name	Description
MCv_MoveLinearInterpolateAbsolute	Absolute value linear interpolation control	Specifies the target position of the absolute position of specified axes group, and executes positioning by linear interpolation control.
MCv_MoveLinearInterpolateRelative	Relative value linear interpolation control	Specifies the movement amount of the relative position of specified axes group, and executes positioning by linear interpolation control.
MCv_MoveCircularInterpolateAbsolute	Absolute value circular interpolation control	Executes positioning by 2-axis circular interpolation by using configuration axes of set axes group, and by specifying the end point and the sub point of the absolute position.
MCv_MoveCircularInterpolateRelative	Relative value circular interpolation control	Executes positioning by 2-axis circular interpolation by using configuration axes of set axes group, and by specifying the relative position from the current position at start to the end point and the sub point.

Types of multiple axes positioning control

- 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

- The FBs that execute multiple axes positioning control and the multiple axes positioning controls that can be executed are shown below.

Motion FB classification	Description	Motion control FB	Executable multiple axes positioning control
Multiple axes positioning control FB	A motion FB that executes one multiple axes positioning control for one FB instance. The input to the multiple axes position control is set using FB input variables.	MCv_MoveLinearInterpolateAbsolute	Absolute value linear interpolation control
		MCv_MoveLinearInterpolateRelative	Relative value linear interpolation control
		MCv_MoveCircularInterpolateAbsolute	Absolute value circular interpolation control
		MCv_MoveCircularInterpolateRelative	Relative value circular interpolation control
Multiple axes positioning data operation FB*1	A motion FB that continuously executes axes positioning controls within one FB instance. The FB I/O variable has an operation profile ID structure, and inputs to the multiple axes positioning control are set using operation profile data.	MCv_MovePositioningData	Absolute value linear interpolation control Relative value linear interpolation control Absolute value circular interpolation control Relative value circular interpolation control

*1 For details on multiple axes positioning data operation, refer to the following.
☞ Page 238 Multiple Axes Positioning Data Operation

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 72 Axes Group Assignment
- In multiple axes positioning control, specify axes (interpolation axes) to be used for interpolation control in the configuration axis of the axes group. Up to 4 axes can be set.

Positioning speed in multiple axes positioning control

Positioning speed at multiple axes positioning control can be set in Velocity (Velocity) 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 velocity specification
- Long axis velocity specification
- Reference axis velocity specification

The motion system control methods for each set method are described in the following.

■ Vector velocity specification

The positioning speed of each interpolation axis is calculated from the movement amount of each interpolation axis by the motion system based on the positioning speed of the set control target.

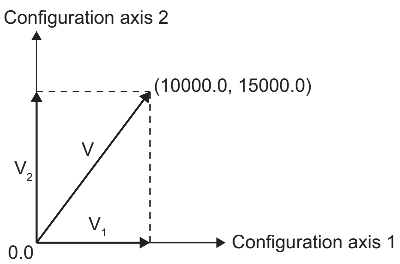
The positioning speed of the control target is called the vector velocity.

Set each inputs in the motion control FB: Linear interpolation axes (LinearAxes) as the interpolation axes, Target position (Position) or Movement amount (Distance) as the movement amount of each interpolation axis, and Velocity (Velocity) as the vector velocity.

6

Ex.

For 2-axis linear interpolation control.



Setting item	Setting value
Linear interpolation axes	Configuration axis 1, Configuration axis 2
Configuration axis 1 movement amount (D_1)	10000.0 [pulses]
Configuration axis 2 movement amount (D_2)	15000.0 [pulses]
Velocity unit of axes group	[s]
Vector velocity (V)	7000.0 [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
Configuration axis 1 positioning speed	$V_1 = V \times D_1 / \sqrt{(D_1^2 + D_2^2)}$
Configuration axis 2 positioning speed	$V_2 = V \times D_2 / \sqrt{(D_1^2 + D_2^2)}$

Point

For the vector velocity specification, the set velocity limit value is valid to the Velocity (Velocity) which is as the vector velocity. For details, refer to the following.

☞ Page 345 Velocity Limit

■ Long axis velocity specification

The control is carried out based on the positioning speed (the long axis velocity) of interpolation axes whose movement amount is the largest in the addresses set in each interpolation axis.

The positioning speed of the other interpolation axes is calculated by the motion system from the movement amount of each interpolation axis.

Set each inputs in the motion control FB: Linear interpolation axes (LinearAxes) as the interpolation axes, Target position (Position) or Movement amount (Distance) as the movement amount of each interpolation axis, and Velocity (Velocity) as the long axis velocity.

Ex.

For 4-axis linear interpolation control.

Setting item	Setting value
Linear	Configuration axis 1, Configuration axis 2, Configuration axis 3, Configuration axis 4
Configuration axis 1 movement amount (D_1)	10000.0 [pulses]
Configuration axis 2 movement amount (D_2)	15000.0 [pulses]
Configuration axis 3 movement amount (D_3)	5000.0 [pulses]
Configuration axis 4 movement amount (D_4)	20000.0 [pulses]
Configuration axis 4 velocity unit	[s]
Long axis velocity (V)	7000.0 [pulses/s]

In the above case, the long axis is the configuration axis 4 whose movement amount is the largest, and the control is carried out by long axis velocity to the configuration axis 4. The positioning speed of the other interpolation axes is calculated by the motion system with the following calculation formulas.

Setting item	Setting value
Configuration axis 1 positioning speed	$V_1 = D_1 / D_4 \times V$
Configuration axis 2 positioning speed	$V_2 = D_2 / D_4 \times V$
Configuration axis 3 positioning speed	$V_3 = D_3 / D_4 \times V$

Point

- For the long axis velocity specification, set velocity limit value is valid for the Velocity (Velocity) which is as the long axis velocity. For details, refer to the following.
☞ Page 345 Velocity Limit
- Note that the vector velocity when specifying the long axis velocity may be larger than the velocity limit value. When setting a value like the following in the 2-axis linear interpolation, the vector velocity exceeds the velocity limit value.

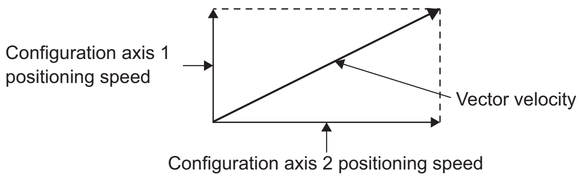
<Example>

When the setting items and the setting values are as follows.

- Linear interpolation axes: Configuration axis 1, Configuration axis 2
- Configuration axis 1 movement amount: 100 [pulses]
- Configuration axis 2 movement amount: 200 [pulses]
- Long axis velocity: 50 [pulses/s]
- Velocity limit value of configuration axis 2: 55 [pulses/s]

In the above case, the reference axis is configuration axis 2 whose movement amount is the largest, and the control is carried out based on the positioning speed specified to the configuration axis 2. The positioning speed and the vector velocity of each axis are shown below.

- Configuration axis 1 positioning speed: 100 / 200 × 50 = 25 [pulses/s]
- Configuration axis 2 positioning speed: 50 [pulses/s]
- Vector velocity: $\sqrt{(25^2 + 50^2)}$ = 55.9 [pulses/s]



The vector velocity becomes a value that exceeds the velocity limit value 55 of configuration axis 2.

■ Reference axis velocity specification

The motion system controls the positioning speed of other interpolation axes calculating from each interpolation axis movement amount based on the positioning speed of the set reference axis (the reference axis velocity), and carries out the control.

In motion control FB, set Linear interpolation axes (LinearAxes) to the reference axis as a interpolation axes to the first element of the array in Linear interpolation axes (LinearAxes). Also, set Target position (Position) or Movement amount (Distance) as the movement amount of each axis and set the Velocity (Velocity) as the reference axis speed.

- If "0.0" is set to the axis movement amount of configuration axes specified as the reference axis, "Reference Axis Movement Amount 0 (error code: 1AABH)" occurs.

Ex.

For 4-axis linear interpolation control.

Setting item	Setting value
Linear interpolation axes	Configuration axis 4, Configuration axis 1, Configuration axis 2, Configuration axis 3
Configuration axis 1 movement amount (D_1)	10000.0 [pulses]
Configuration axis 2 movement amount (D_2)	15000.0 [pulses]
Configuration axis 3 movement amount (D_3)	5000.0 [pulses]
Configuration axis 4 movement amount (D_4)	20000.0 [pulses]
Configuration axis 4 velocity unit	[s]
Reference axis velocity (V)	7000.0 [pulses/s]

In the above case, the reference axis is the configuration axis 4, and control is carried out at the positioning speed specified to the configuration axis 4.

The positioning speed of the other interpolation axes is calculated by the motion system with the following calculation formula.

Setting item	Setting value
Configuration axis 1 positioning speed	$V_1 = D_1 / D_4 \times V$
Configuration axis 2 positioning speed	$V_2 = D_2 / D_4 \times V$
Configuration axis 3 positioning speed	$V_3 = D_3 / D_4 \times V$

Point

- For the reference axis velocity specification, set velocity limit value is valid to the Velocity (Velocity) which is as the reference axis velocity. For details, refer to the following.
  ☞ Page 345 Velocity 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 velocity.

<Example>

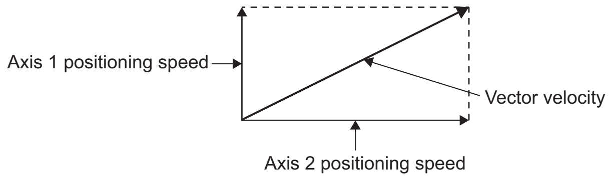
When setting a value like the following in the 2-axis linear interpolation, the positioning speed and the vector velocity of the configuration axis 2 exceed the velocity limit value.

When the setting items and the setting values are as follows.

- Linear interpolation axes: Configuration axis 1, Configuration axis 2
- Configuration axis 1 movement amount: 100 [pulses]
- Configuration axis 2 movement amount: 200 [pulses]
- Reference axis velocity: 50 [pulses/s]
- Configuration axis 1 velocity limit value: 55 [pulses/s]

In the above case, the reference axis is the configuration axis 1, and control is carried out with the velocity limit value set to the configuration axis 1. The positioning speed and the vector velocity of each configuration axis are shown below.

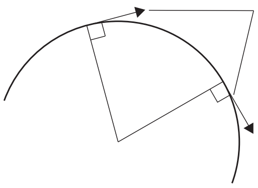
- Configuration axis 1 positioning speed: 50 [pulses/s]
- Configuration axis 2 positioning speed: 200 / 100 × 50 = 100 [pulses/s]
- Vector velocity: $\sqrt{(50^2 + 100^2)}$ = 111.8 [pulses/s]



The positioning speed and the vector velocity of the configuration axis 2 become a value that exceeds the velocity limit value "55" of configuration axis 1.

Circular interpolation control

For the circular interpolation, the control is performed so that the positioning speed on the circular path becomes the specified speed.



Blending of the multiple axes positioning control

The basic operation is based on Blending during multiple axes operation. For details on Blending during multiple axes operation, refer to the following.

☞ Page 137 Multiple start during single axis operation

This section explains operation when selecting BlendingLow, BlendingPrevious, BlendingNext, and BlendingHigh in Buffer mode (BufferMode) of the Motion control FB and setting different speed mode to the Velocity mode (VelocityMode) of the first FB (FB1) and the second FB (FB2) in the multiple axes positioning control.

- It can multiple start even when Velocity mode (VelocityMode) is different between FB1 and FB2. However, during Blending, "Multiple Start Velocity Mode Specified Mismatch Warning (warning code: 0D11H)" occurs.
- The control is carried out using the FB1 Velocity mode (VelocityMode) in FB1 and using the FB2 Velocity mode (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.
- Note that the speed of each axis and the vector velocity may rapidly change when Velocity mode (VelocityMode) of FB1 and FB2 is different during Blending.

■ Operation for BlendingPrevious

Blending from the reference axis velocity specification to the vector velocity specification is as shown below.

It operates so that the FB1 reference axis velocity becomes the speed set to Velocity (Velocity) of FB1 (the reference axis velocity) at the FB1 target position. The reference axis velocity when reaching the FB1 target position is read as the current vector velocity value of FB2 without unit conversion, and it accelerates/decelerates the vector velocity set to Velocity (Velocity) of FB2.

Ex.

When FB1 and FB2 are set to the following

- Axis setting, Axes group setting

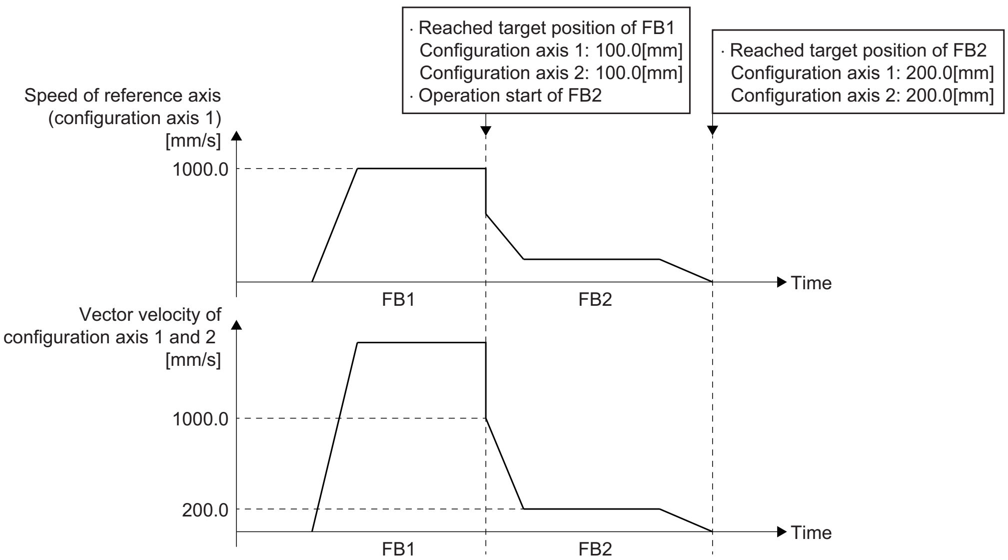
Setting item	Setting value
Position and velocity command unit of the configuration axis 1	mm/s
Position and velocity command unit of the configuration axis 2	mm/s
Position and velocity command unit of the axes group	mm/s

- FB1 input (MCv_MoveLinearInterpolateAbsolute (Absolute Value Linear Interpolation Control)) (Command unit: mm/s)

Setting item	Setting value
Linear interpolation axes (LinearAxes[0])	1
Linear interpolation axes (LinearAxes[1])	2
Linear interpolation axes (LinearAxes[2])	0
Target position (Position[0])	100.0
Target position (Position[1])	100.0
Velocity (Velocity)	1000.0
Velocity mode (VelocityMode)	2: Reference Axis Velocity (ReferenceAxisSpeed)

- FB2 input (MCv_MoveLinearInterpolateAbsolute (Absolute Value Circular Interpolation Control)) (Command unit: mm/s)

Setting item	Setting value
Linear interpolation axes (LinearAxes[0])	1
Linear interpolation axes (LinearAxes[1])	2
Linear interpolation axes (LinearAxes[2])	0
Target position (Position[0])	200.0
Target position (Position[1])	200.0
Velocity (Velocity)	200.0
Velocity mode (VelocityMode)	0: Vector Velocity (VectorSpeed)
Buffer mode (BufferMode)	3: BlendingPrevious (mcBlendingPrevious)



- Reaches the target position of FB1 at the reference axis velocity of 1000.0[mm/s] of the configuration axis 1 that is the reference axis, and switches to FB2.
- FB2 starts operation at the vector velocity of 1000.0[mm/s].
- Decelerates to the vector velocity of 200.0[mm/s].
- Performs positioning to the target position of FB2.

6

■ Operation for BlendingNext

FB1: The reference axis velocity → FB2: The vector velocity

It operates so that the FB1 reference axis velocity becomes the speed set to Velocity (Velocity) of FB2 (the vector velocity) at the FB1 target position. The reference axis velocity when reaching the FB1 target position is read as the current vector velocity value of FB2 without unit conversion, and it accelerates/decelerates the vector velocity set to Velocity (Velocity) of FB2.

Ex.

When FB1 and FB2 are set to the following

- Axis setting, Axes group setting

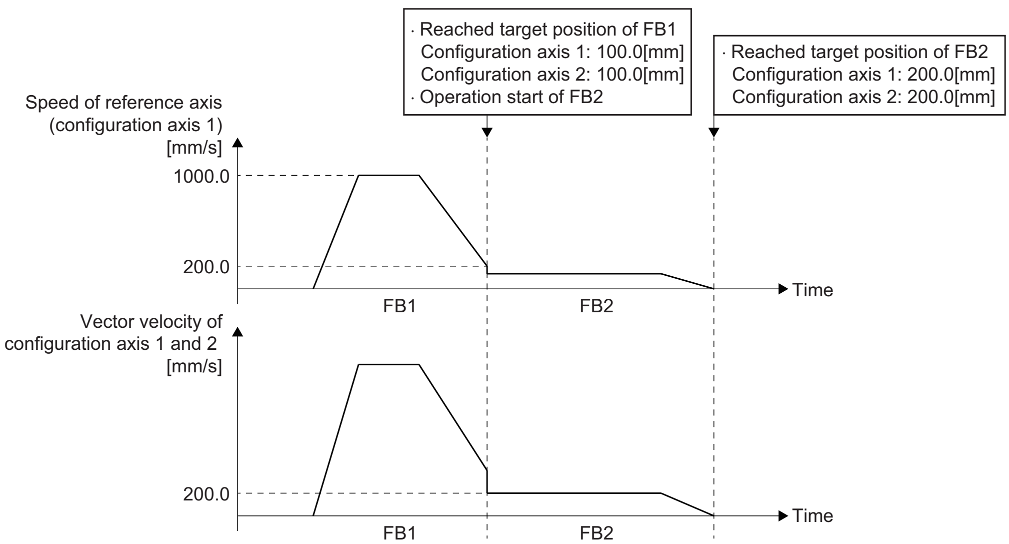
Setting item	Setting value
Position and velocity command unit of the configuration axis 1	mm/s
Position and velocity command unit of the configuration axis 2	mm/s
Position and velocity command unit of the axes group	mm/s

- FB1 input (MCv_MoveLinearInterpolateAbsolute (Absolute Value Circular Interpolation Control)) (Command unit: mm/s)

Setting item	Setting value
Linear interpolation axes (LinearAxes[0])	1
Linear interpolation axes (LinearAxes[1])	2
Linear interpolation axes (LinearAxes[2])	0
Target position (Position[0])	100.0
Target position (Position[1])	100.0
Velocity (Velocity)	1000.0
Velocity mode (VelocityMode)	2: Reference Axis Velocity (ReferenceAxisSpeed)

- FB2 input (MCv_MoveLinearInterpolateAbsolute (Absolute Value Circular Interpolation Control)) (Command unit: mm/s)

Setting item	Setting value
Linear interpolation axes (LinearAxes[0])	1
Linear interpolation axes (LinearAxes[1])	2
Linear interpolation axes (LinearAxes[2])	0
Target position (Position[0])	200.0
Target position (Position[1])	200.0
Velocity (Velocity)	200.0
Velocity mode (VelocityMode)	0: Vector Velocity (VectorSpeed)
Buffer mode (BufferMode)	4: BlendingNext (mcBlendingNext)



- Operates to the middle of the positioning to the target position of FB1 at the reference axis velocity 1000.0 [mm/s] of the configuration axis 1.
- The configuration axis 1 as a reference axis reaches the target position of FB1 at the reference axis velocity of 200.0 [mm/s], and switches to FB2.
- FB2 starts operation at the vector velocity 200.0 [mm/s].
- Performs positioning to the target position of FB2.

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 Velocity mode (VelocityMode) of the Motion control FB. For details and precautions, refer to the following.

☞ Page 74 Technical Units
☞ Page 137 Multiple start during single axis operation

Blending of the multiple axes positioning control

The basic operation is based on Blending during multiple axes operation. For details on Blending during multiple axes operation, refer to the following.

☞ Page 144 Multiple start during multiple axes operation

This section explains operation when selecting BlendingLow, BlendingPrevious, BlendingNext, and BlendingHigh in Buffer mode (BufferMode) of the Motion control FB, and the command unit is different between the first FB (FB1) and the second FB (FB2) in the multiple axes positioning control.

- It can multiple start even when the command unit is different between FB1 and FB2. However, during Blending, "Velocity Command Unit Mismatch Warning (warning code: 0D1EH)" occurs.
- 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.

■ Operation for BlendingPrevious

During Blending, it can multiple 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.

Ex.

When FB1 and FB2 are set to the following

- Axis setting

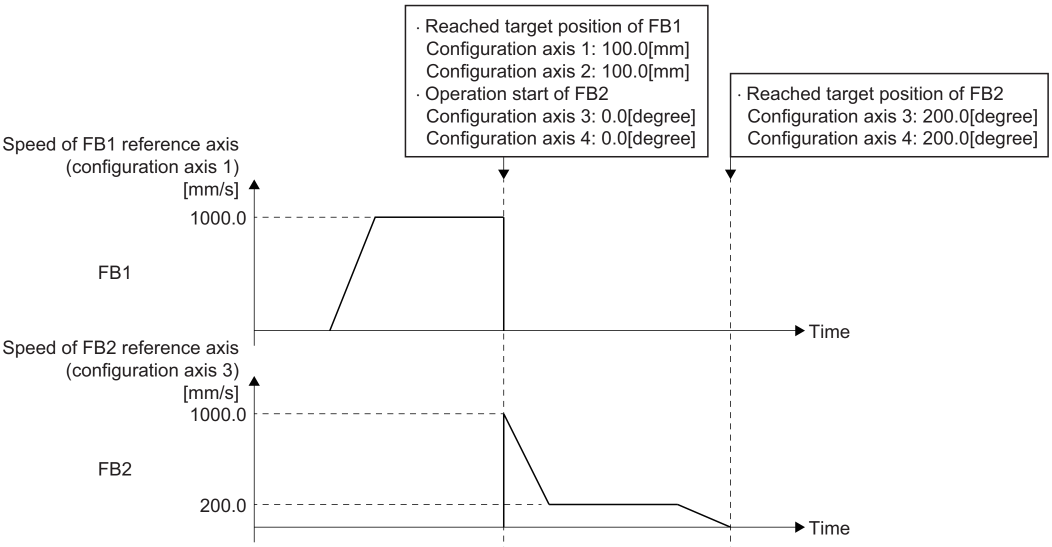
Setting item	Setting value
Position and velocity command unit of the configuration axis 1	mm/s
Position and velocity command unit of the configuration axis 2	mm/s
Position and velocity command unit of the configuration axis 3	degree/min
Position and velocity command unit of the configuration axis 4	degree/min

- FB1 input (MCv_MoveLinearInterpolateAbsolute (Absolute Value Circular Interpolation Control)) (Command unit: mm/s from the command unit of the configuration axis 1)

Setting item	Setting value
Linear interpolation axes (LinearAxes[0])	1
Linear interpolation axes (LinearAxes[1])	2
Linear interpolation axes (LinearAxes[2])	0
Target position (Position[0])	100.0
Target position (Position[1])	100.0
Velocity (Velocity)	1000.0
Velocity mode (VelocityMode)	2: Reference Axis Velocity (ReferenceAxisSpeed)

- FB2 input (MCv_MoveLinearInterpolateAbsolute (Absolute Value Circular Interpolation Control)) (Command unit: degree/min from the command unit of the configuration axis 3)

Setting item	Setting value
Linear interpolation axes (LinearAxes[0])	3
Linear interpolation axes (LinearAxes[1])	4
Linear interpolation axes (LinearAxes[2])	0
Target position (Position[2])	200.0
Target position (Position[3])	200.0
Velocity (Velocity)	200.0
Velocity mode (VelocityMode)	2: Reference Axis Velocity (ReferenceAxisSpeed)
Buffer mode (BufferMode)	3: BlendingPrevious (mcBlendingPrevious)



- Configuration axis 1 as a reference axis reaches the target position of FB1 at the velocity of 1000.0 [mm/s], and switches to FB2.
- Configuration axis 3 as a reference axis starts operation from the current position of FB2 of [0 degree 0 degree] at the velocity of 1000.0 [degree/min].
- Configuration axis 3 as a reference axis decelerates to the velocity of 200.0 [degree/min].
- Performs positioning to the target position of FB2.

■ Operation for BlendingNext

During Blending, it can multiple 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.

Ex.

When FB1 and FB2 are set to the following

- Axis setting

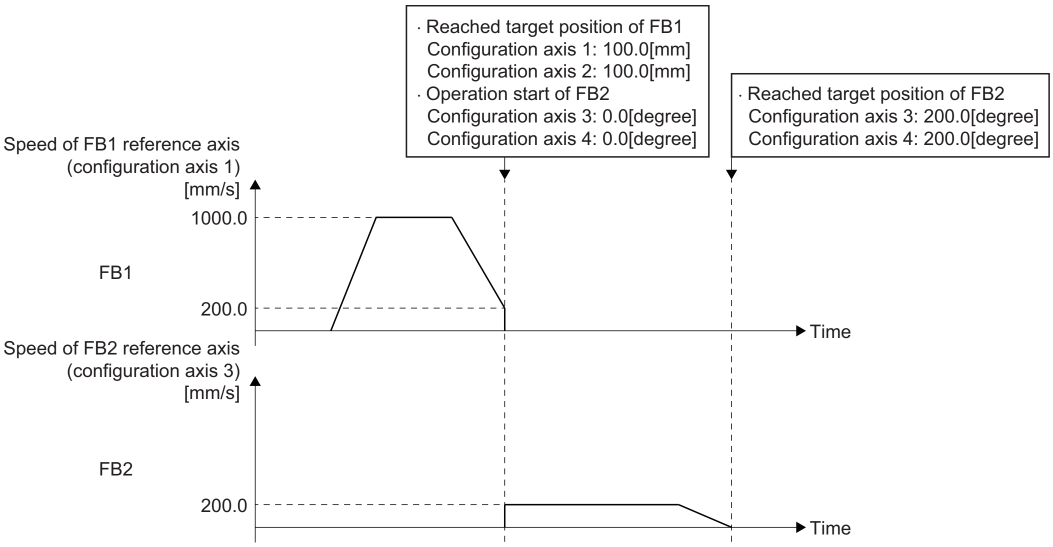
Setting item	Setting value
Position and velocity command unit of the configuration axis 1	mm/s
Position and velocity command unit of the configuration axis 2	mm/s
Position and velocity command unit of the configuration axis 3	degree/min
Position and velocity command unit of the configuration axis 4	degree/min

- FB1 input (MCv_MoveLinearInterpolateAbsolute (Absolute Value Circular Interpolation Control)) (Command unit: mm/s)

Setting item	Setting value
Linear interpolation axes (LinearAxes[0])	1
Linear interpolation axes (LinearAxes[1])	2
Linear interpolation axes (LinearAxes[2])	0
Target position (Position[0])	100.0
Target position (Position[1])	100.0
Velocity (Velocity)	1000.0
Buffer mode (VelocityMode)	2: Reference Axis Velocity (ReferenceAxisSpeed)

- FB2 input (MCv_MoveLinearInterpolateAbsolute (Absolute Value Circular Interpolation Control)) (Command unit: degree/min)

Setting item	Setting value
Linear interpolation axes (LinearAxes[0])	3
Linear interpolation axes (LinearAxes[1])	4
Linear interpolation axes (LinearAxes[2])	0
Target position (Position[2])	200.0
Target position (Position[3])	200.0
Velocity (Velocity)	200.0
Velocity mode (VelocityMode)	2: Reference Axis Velocity (ReferenceAxisSpeed)
Buffer mode (VelocityMode)	4: BlendingNext (mcBlendingNext)



- Configuration axis 1 as a reference axis operates to the middle of the target position of FB1 at the velocity of 1000.0 [mm/s].
- Configuration axis 1 as a reference axis reaches the target position of FB1 at the speed of 200.0 [mm/s], and switches to FB2.
- FB2 with a configuration axis 3, as a reference axis, starts operation at the velocity of 200.0 [degree/min].
- Operates positioning to the target position of FB2.

Acceleration/deceleration processing

The basic operation for acceleration/deceleration processing is based on the acceleration/deceleration processing function. For details on each acceleration/deceleration processing function, refer to the following.

☞ Page 317 Acceleration/deceleration Processing Function

This section explains the operations depending on the Velocity mode (VelocityMode) setting of the Motion control FB. The relationship between the positioning speed and the acceleration/deceleration processing depending on the Velocity mode (VelocityMode) is shown below.

- 0: Vector Velocity (VectorSpeed)
  The acceleration/deceleration processing is applied to the vector velocity.
- 1: Long Axis Velocity (LongAxisSpeed)
  The acceleration/deceleration processing is applied to the long axis velocity.
- 2: Reference Axis Velocity (ReferenceAxisSpeed)
  The acceleration/deceleration processing is applied to the reference axis velocity.

State transition in the multiple axes positioning control

Axes Group Status (AxesGroupName.Md.GroupStatus) of the axes group in the positioning control is "5: Operating (GroupMoving)". For Axes Group Status (AxesGroupName.Md.GroupStatus), refer to the following.

☞ Page 67 Axes group 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 a maximum of 4 axes is performed.

Linear interpolation control includes the following two control methods.

For the Motion control FBs that execute each control, refer to the following.

☞ Page 210 Types of multiple axes positioning control

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 following shows examples of interpolation control for each number of interpolation axes.

1-axis linear interpolation control

■ 1-axis linear interpolation control (absolute value)

1-axis linear interpolation is executed from the current position at start (start point address) to the address set to Target position (Position) (end point address).

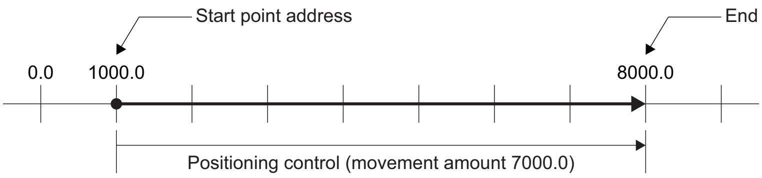
Ex.

When start point address of configuration axis 1 is "1000.0" and Target position (Position) is set to "8000.0"

LinearAxes[0]: 1

(LinearAxes[1]: 0)

Position[0]: 8000.0



■ 1-axis linear interpolation control (relative value)

1-axis linear interpolation is executed by the movement amount set to Movement amount (Distance) from the current position at start (start point address).

The movement direction is determined by a sign (+/-) of the movement amount.

Movement amount	Description
positive (+)	Positioning to the positive direction (address increasing direction)
negative (-)	Positioning to the negative direction (address decreasing direction)

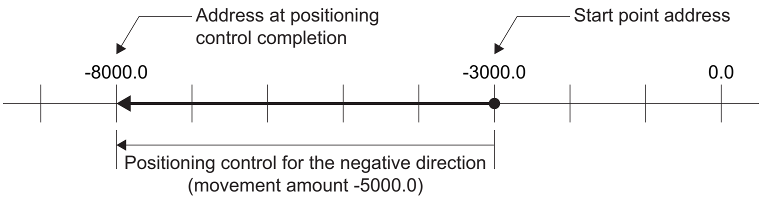
Ex.

When start point address of configuration axis 1 is "-3000.0" and Movement amount (Distance) is set to "-5000.0"

LinearAxes[0]: 1

(LinearAxes[1]: 0)

Distance[0]: -5000.0

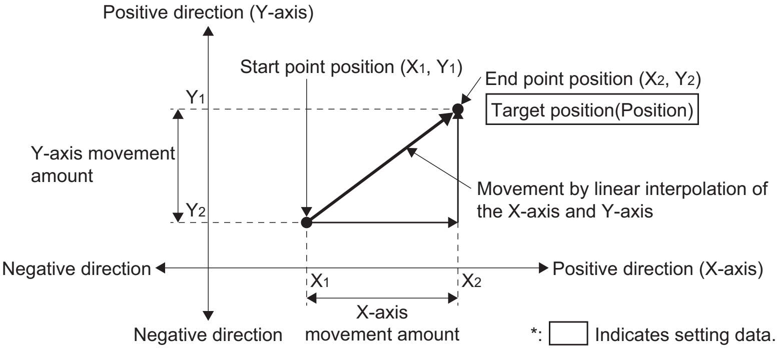


2-axis linear interpolation control

■ 2-axis linear interpolation control (absolute value)

2-axis linear interpolation is executed from the current position at start (start point address) to the address set to Target position (Position) (end point address).

The movement direction is determined by the current position at start point address and the end point address of each axis.



Ex.

When the start point addresses are "configuration axis 1: 1000.0, configuration axis 2: 1000.0" and Target position (Position) settings are "configuration axis 1: 10000.0, configuration axis 2: 4000.0"

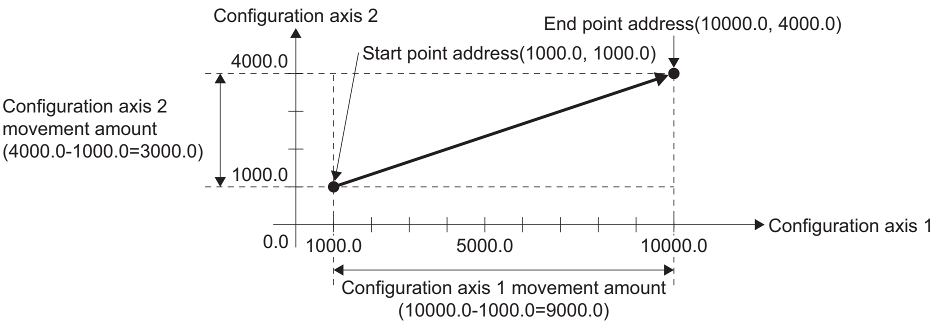
LinearAxes[0]: 1

LinearAxes[1]: 2

(LinearAxes[2]: 0)

Position[0]: 10000.0

Position[1]: 4000.0

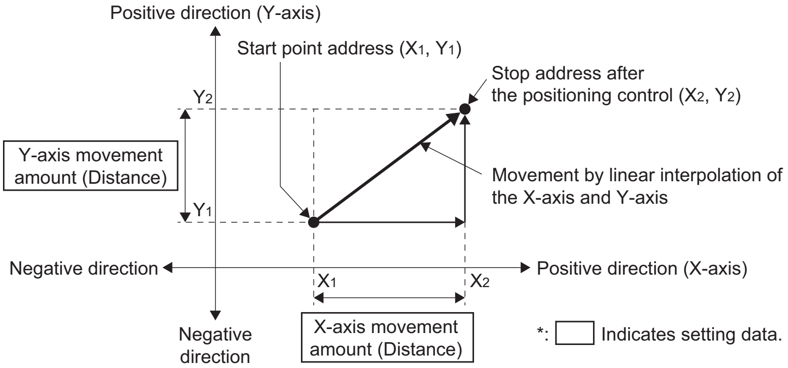


■ 2-axis linear interpolation control (relative value)

2-axis linear interpolation is executed by the movement amount set to Movement amount (Distance) from the current position at start (start point address).

The movement direction is determined by a sign (+/-) of the movement amount.

Movement amount	Description
positive (+)	Positioning to the positive direction (address increasing direction)
negative (-)	Positioning to the negative direction (address decreasing direction)



Ex.

When the start point addresses are "configuration axis 1: 1000.0, configuration axis 2: 4000.0" and Movement amount (Distance) settings are "configuration axis 1: 9000.0, configuration axis 2: -3000.0"

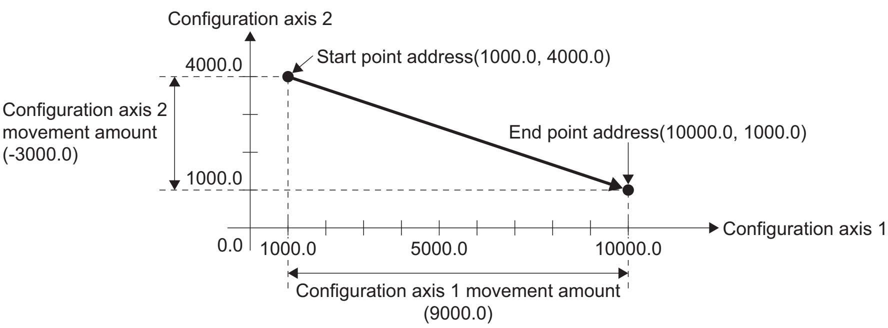
LinearAxes[0]: 1

LinearAxes[1]: 2

(LinearAxes[2]: 0)

Distance[0]: 9000.0

Distance[1]: -3000.0

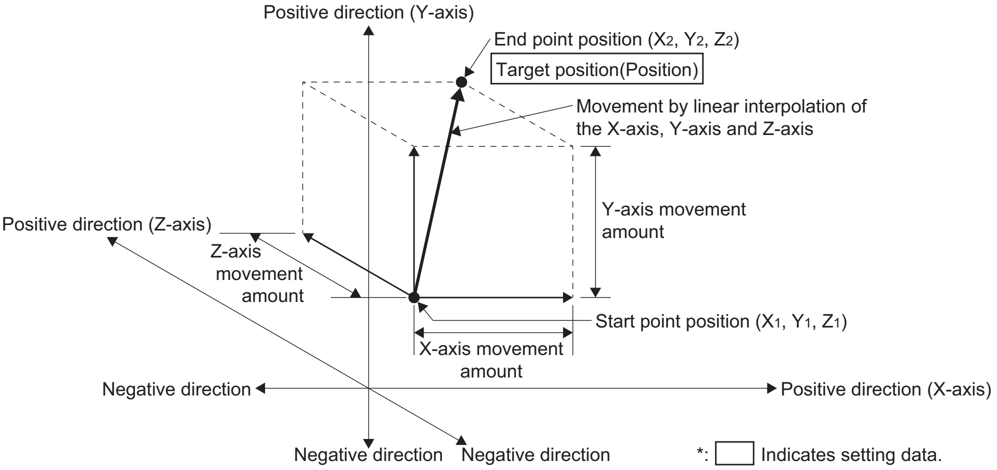


3-axis linear interpolation control

■ 3-axis linear interpolation control (absolute value)

3-axis linear interpolation is executed from the current position at start (start point address) to the commanded position set to Target position (Position).

The movement direction is determined by the current position at start point address and the end point address of each axis.



Ex.

When the start point addresses are "configuration axis 1: 1000.0, configuration axis 2: 2000.0, configuration axis 3: 1000.0" and Target position (Position) settings are "configuration axis 1: 7000.0, configuration axis 2: 8000.0, configuration axis 3: 4000.0"

LinearAxes[0]: 1

LinearAxes[1]: 2

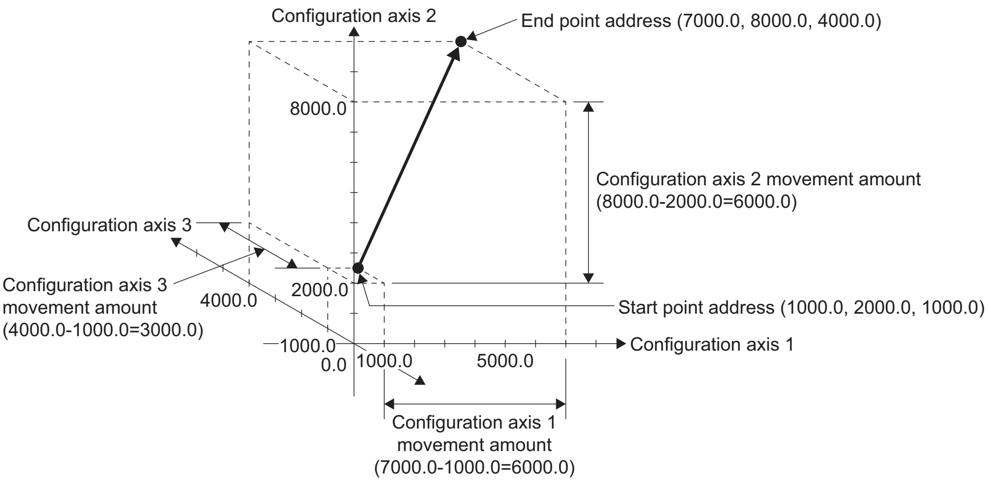
LinearAxes[2]: 3

(LinearAxes[3]: 0)

Position[0]: 7000.0

Position[1]: 8000.0

Position[2]: 4000.0

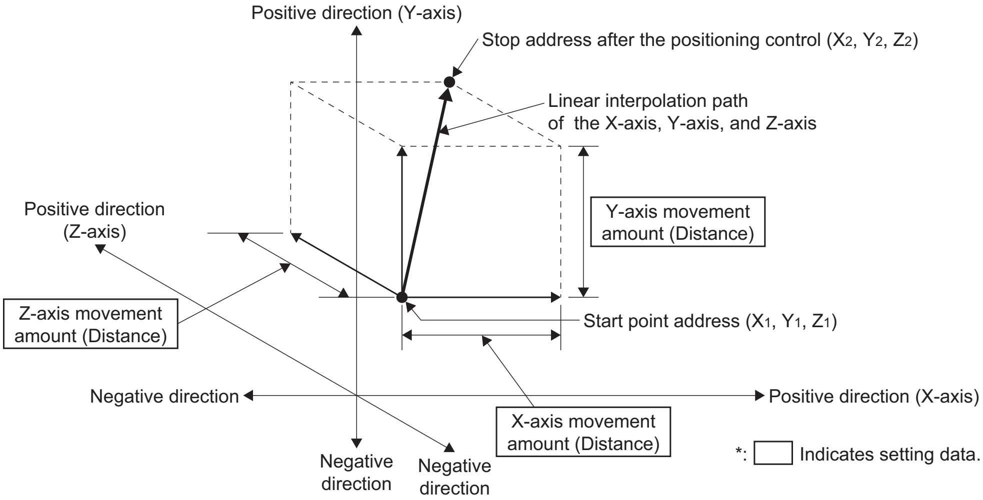


■ 3-axis linear interpolation control (relative value)

3-axis linear interpolation is executed by the movement amount set to Movement amount (Distance) from the current position at start (start point address).

The movement direction is determined by a sign (+/-) of the movement amount.

Movement amount	Description
positive (+)	Positioning to the positive direction (address increasing direction)
negative (-)	Positioning to the negative direction (address decreasing direction)



Ex.

When the start point addresses are "configuration axis 1: 2000.0, configuration axis 2: 1000.0, configuration axis 3: 1000.0" and Movement amount (Distance) settings are "configuration axis 1: 10000.0, configuration axis 2: 5000.0, configuration axis 3: 6000.0".

LinearAxes[0]: 1

LinearAxes[1]: 2

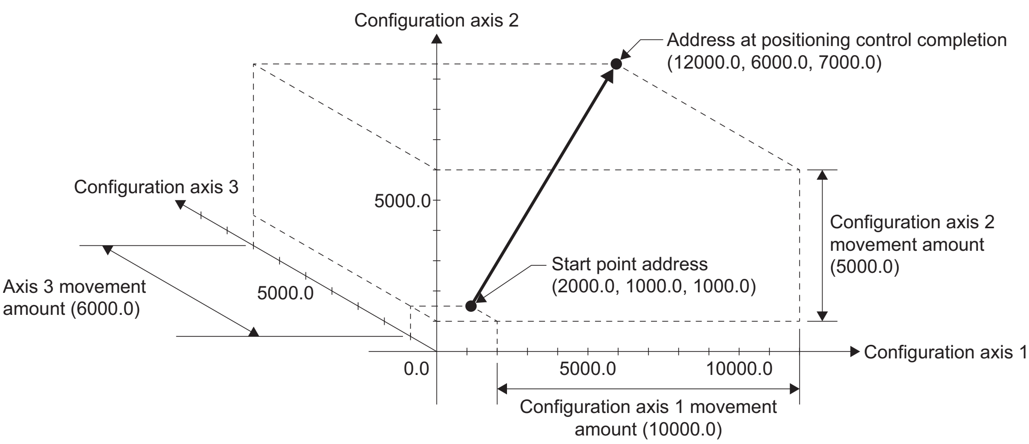
LinearAxes[2]: 3

(LinearAxes[3]: 0)

Distance[0]: 10000.0

Distance[1]: 5000.0

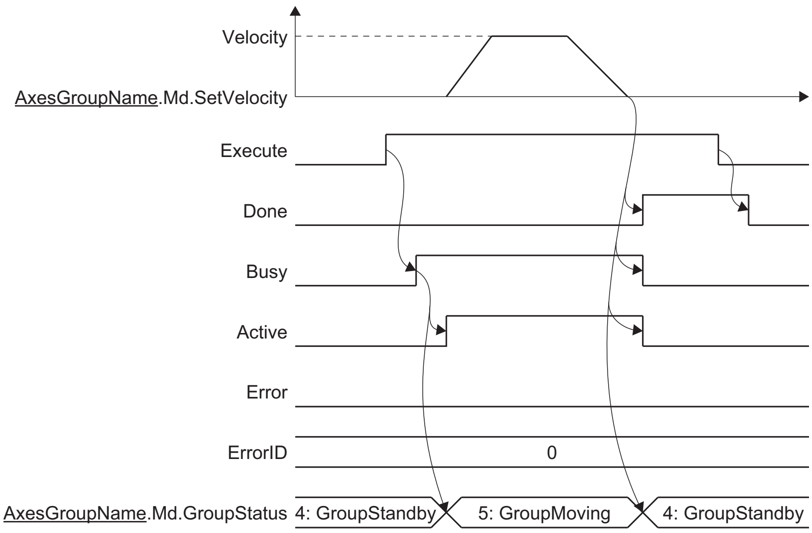
Distance[2]: 6000.0



6

Timing chart

■ When the FB is normally completed



■ When an error occurs

For details when an error occurs, refer to "Basic operation of Execute command (Execute) type Motion control FBs" in the following.

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

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.
Circular interpolation control has the following two control methods.
For the Motion control FBs that execute each control, refer to the following.
☞ Page 210 Types of multiple axes positioning control

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 border point specification, the central point specification, and the radius specification, can be specified with Circular interpolation mode (CircMode) for circular interpolation control.

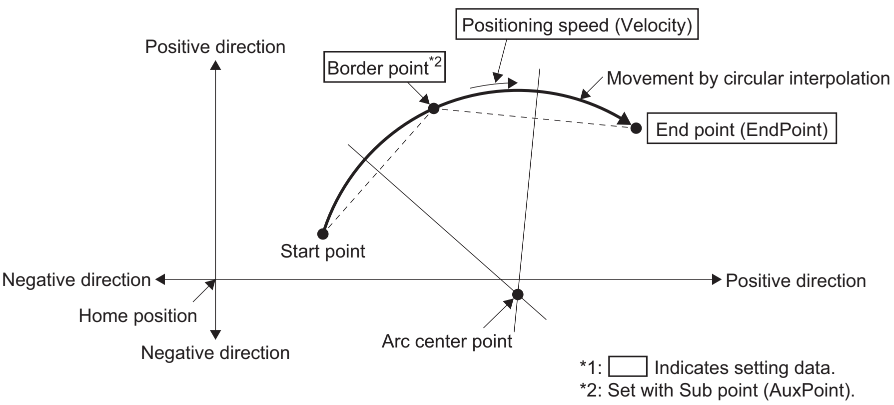
Circular interpolation method	Description
Border point specification	Positioning control is carried out in an arc path passing through the specified border point by using two configuration axes specified to Circular interpolation mode (CircMode).
Central point specification	Two configuration axes specified in Circular interpolation axes (CircAxes) are used to carry out positioning control in an arc path having the specified center point as a center point.
Radius specification	Two configuration axes specified in Circular interpolation axes (CircAxes) are used to carry out positioning control in an arc path having the specified radius.

The operations of circular interpolation control with each circular interpolation mode are shown below.

Circular interpolation control with border point specification

■ Border point specification circular interpolation control (absolute value)

Absolute value circular interpolation control operation with "0: Border Point Specification (mcBorder)"
In absolute value circular interpolation control with border point specification, positioning is carried out from the current position at start (start point address) to the address set to End point (EndPoint) (end point address) in an arc path that passes through the address (border point address) set to Sub point (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 specification. (End point (EndPoint), Positioning speed (Velocity), and Border point shown in the following figure indicate the data to be set.)

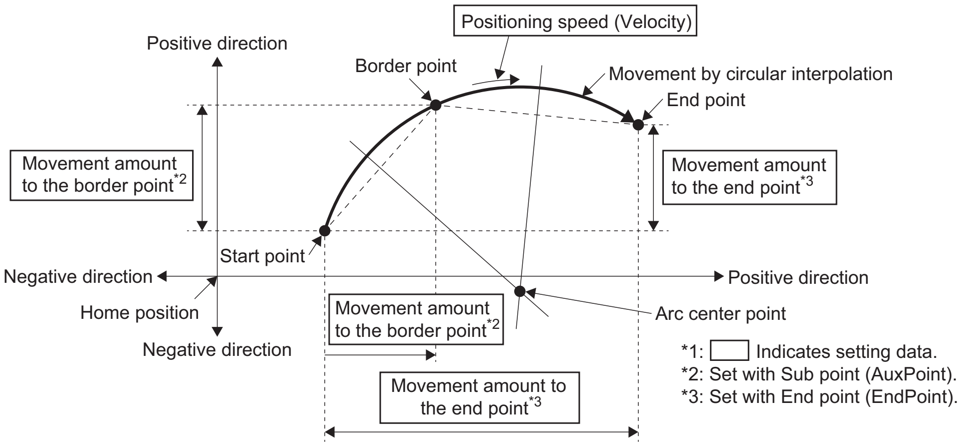


■ Border point specification circular interpolation control (relative value)

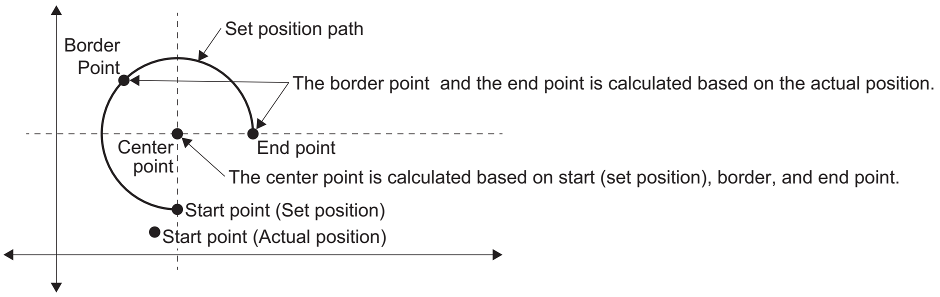
Relative value circular interpolation control operation with "0: Border Point Specification (mcBorder)"

In relative value circular interpolation control with border point specification, positioning is carried out from the current position at start (start point address) to the address of the movement amount set to End point (EndPoint) in an arc path (end point address) that passes through the address of the movement amount set to Sub point (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.



- When "actual position" is selected in position selection during buffer mode (Options (Options): Bit 3) of relative value circular interpolation control, the border point and the end point are calculated based on the actual position. The center point is determined from the start point (set position), the border point, and the end point.



Restriction

2-axis circular interpolation control cannot be set in the following cases.

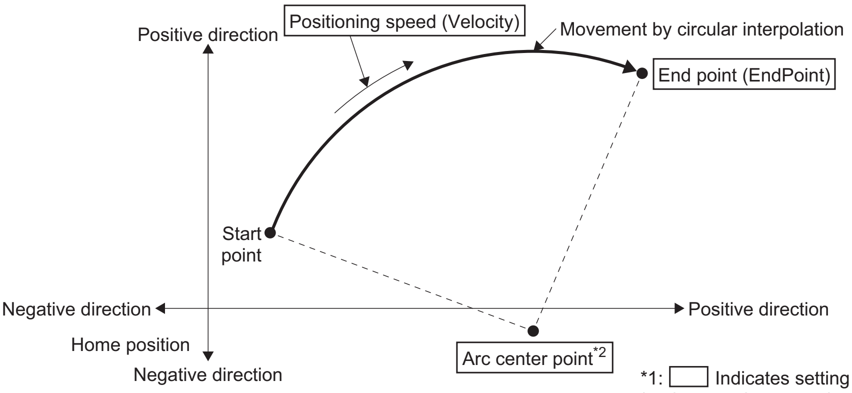
- When the radius exceeds 2147483647.0 (the maximum radius for which 2-axis circular interpolation control is possible is 2147483647.0): The error "Out of Radius Range (error code: 1A6CH)" will occur at positioning start.
- When the border point address exceeds positioning range: The error "Out of Boundary Point Address Range (error code: 1A64H)" will occur.
- When the end point address exceeds positioning range: The error "Out of End Point Address Range (errorcode: 1A6DH)" will occur.
- When the start point address, border point address, and end point address are in the same straight line: Theerror "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 central point specification

■ Central point specification circular interpolation control (absolute value)

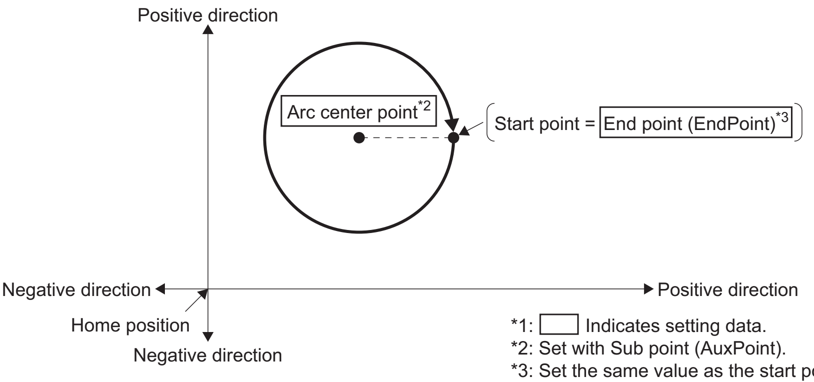
In absolute value circular interpolation control operation with center point specification, positioning is carried out from the current position at start (start point address) to the end position set in End point (EndPoint) in an arc path having the center point set in Sub point (AuxPoint) as its center. For absolute value circular interpolation control with center point specification, the arc path is set using Path selection (PathChoice). For details, refer to the following.

☞ Page 233 Path of central point specification circular interpolation control



*1: ☐ Indicates setting data.
*2: Set with Sub point (AuxPoint).

- To perform positioning for a perfect circle with a radius from the start point address to the arc center point, set the End point (EndPoint) to the same address as the start point. Setting the end point in End point (EndPoint) to the same value as the start point causes the end point and the start point to be identical.



*1: ☐ Indicates setting data.
*2: Set with Sub point (AuxPoint).
*3: Set the same value as the start point.

Ex.

When End point (EndPoint) is set to the same value as the start point

```
AxesGroup001.Pr.Axis[1]:= Axis0001.AxisRef;
AxesGroup001.Pr.Axis[2]:= Axis0002.AxisRef;
EndPoint[0]:= Axis0001.Md.SetPosition;
EndPoint[1]:= Axis0002.Md.SetPosition;
```

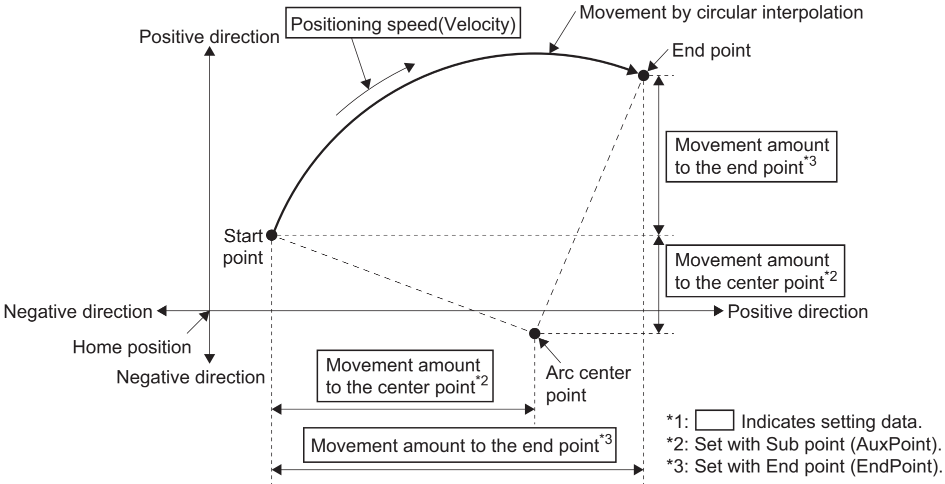
Specify the set position of the configuration axes to be used in circular interpolation to End point (EndPoint) as shown above.

■ Central point specification circular interpolation control (relative value)

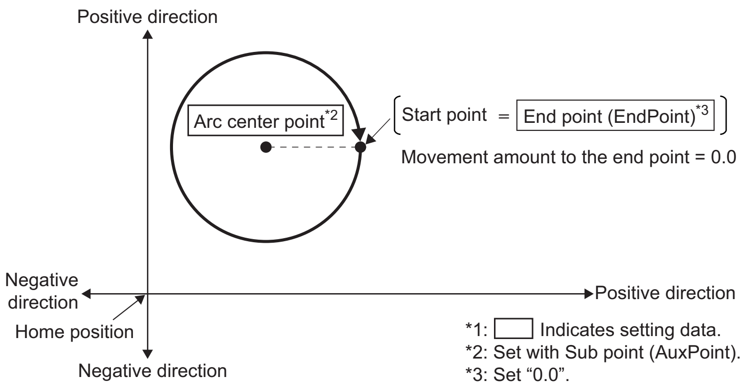
In relative value circular interpolation control operation with center point specification, positioning is carried out from the current position at start (start point address) to the movement amount address set in End point (EndPoint) in an arc path having the movement amount address (center point address) set in Sub point (AuxPoint) as its center.

For relative value circular interpolation control with center point specification, the arc path is set using Path selection (PathChoice). For details, refer to the following.

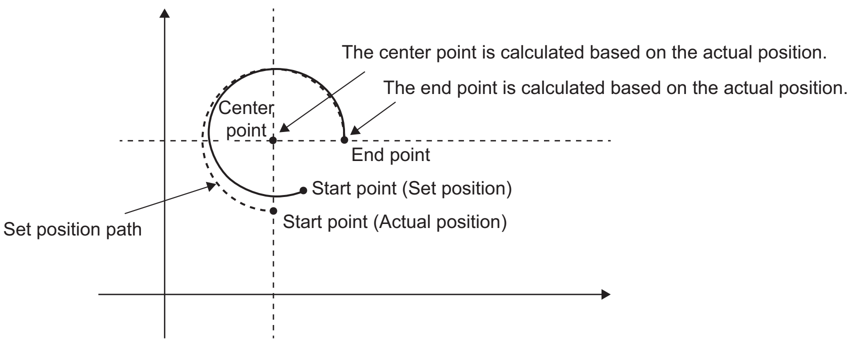
☞ Page 233 Path of central point specification circular interpolation control



- Setting the movement amount to the end point as "0.0" in End point (EndPoint) causes the end point and the start point to be identical and performs positioning for a perfect circle with a radius from the start point address to the arc center point.



- When "actual position" is selected in position selection during buffer mode (Options (Options): Bit 3) of relative value circular interpolation control, the center point and the end point are calculated based on the actual position. The circular path may be ellipse.

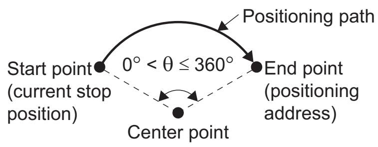
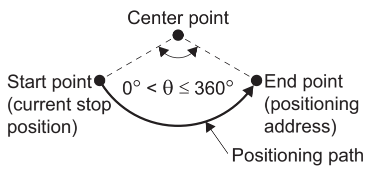
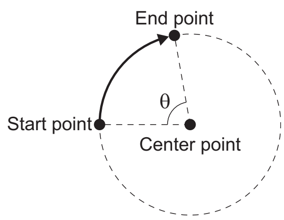
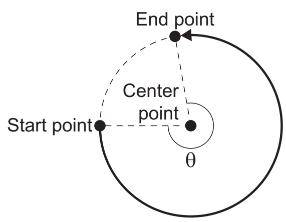


Restriction

2-axis circular interpolation control with center point specification cannot be set in the following cases.
- When the radius exceeds "2147483647.0", the maximum radius for which 2-axis circular interpolation control is possible. ("Out of Radius Range (error code: 1A6CH)")
- When the center point address exceeds positioning range. ("Out of Center Point Address Range (errorcode: 1A6BH)")
- When the start point address is the same as the center point address. ("Start Point - Center Point AddressSame Value (error code: 1A65H)")
- When the end point address is the same as the center point address. ("End Point - Center Point AddressSame Value (error code: 1A67H)")

■ Path of central point specification circular interpolation control

The path in the circular interpolation control with the central point specification can be set by Path selection (PathChoice). The setting value, setting description, arc center angle that can be controlled, and path of Path selection (PathChoice) are shown below.

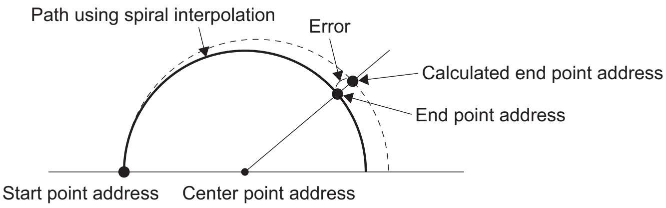
Setting value	Setting description	Arc center angle that can be controlled	Path
0: CW (mcCW)	CW	0° < θ ≤ 360°	
1: CCW (mcCCW)	CCW	0° < θ ≤ 360°	
2: Shortcut (mcShortWay)	Shortcut • When the start point, center point, and end point are on one straight line, becomes CW. (A semicircle is drawn in the CW direction.) • When the start point is same as the end point, "Path Choice Setting Incorrect (error code: 1AB7H)" occurs.	0° < θ ≤ 180°	
3: Detour (mcLongWay)	Detour • When the start point, center point, and end point are on one straight line, becomes CW. (A semicircle is drawn in the CW direction.) • When the start point is same as the end point, becomes CW. (A perfect circle is drawn in CW direction.)	180° < θ ≤ 360°	

■ Circular interpolation error tolerance

In central point specification circular interpolation control, the arc path calculated from the start position and center point position may deviate from the end position set in End point (EndPoint).

Circular interpolation error tolerance (CircularErrorTolerance) sets allowable range of an error between the calculated arc path and the end position.

- Calculated error ≤ Circular interpolation error tolerance (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 > Circular interpolation error tolerance (CircularErrorTolerance)
  At the positioning start, "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.

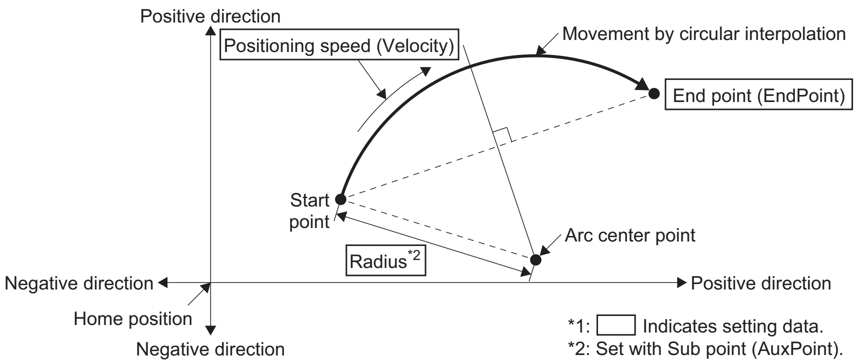
Radius specification circular interpolation control

■ Radius specification circular interpolation control (absolute value)

In absolute value circular interpolation control operation with the radius specification, positioning is carried out from the current position at start (start point address) to the address set to End point (EndPoint) (end point address) in an arc path having the radius set in Sub point (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.

For absolute value circular interpolation control with radius specification, the arc path is set using Path selection (PathChoice). For details, refer to the following.

☞ Page 236 Path of radius specification circular interpolation control

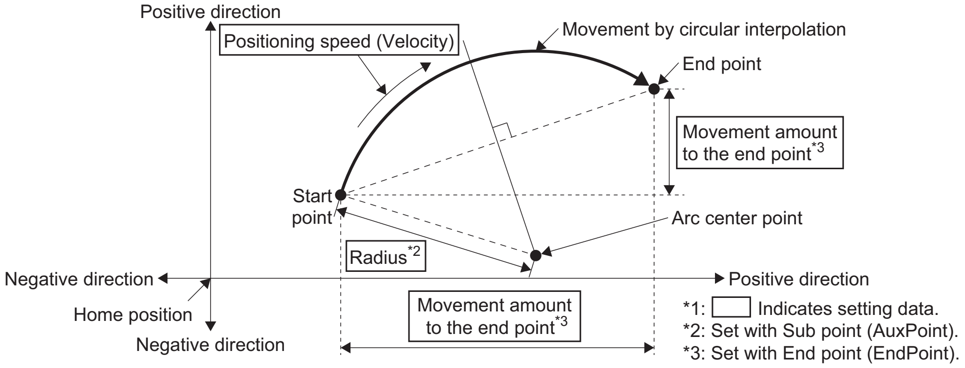


■ Radius specification circular interpolation control (relative value)

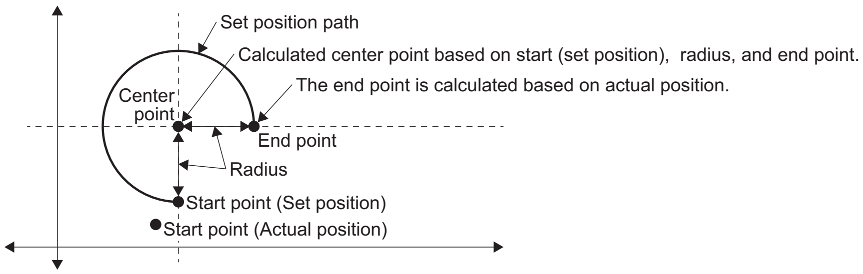
In relative value circular interpolation control operation with radius specification, positioning is carried out from the current position at start (start point address) to the movement amount address (end point address) set in End point (EndPoint) in an arc path having the radius set in Sub point (AuxPoint). The resulting control path is an arc having as its center the intersection point between the perpendicular bisectors of the straight line between the current position at start and the end point address and the set radius.

For relative value circular interpolation control with radius specification, the arc path is set using Path selection (PathChoice). For details, refer to the following.

☞ Page 236 Path of radius specification circular interpolation control



- When "actual position" is selected in position selection during buffer mode (Options (Options): Bit 3) of relative value circular interpolation control, the center point and the end point are calculated based on the actual position. The center point is determined by the start point (set position), radius, and end point.



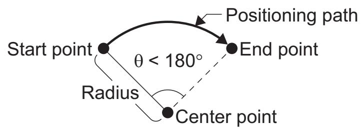
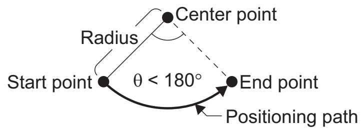
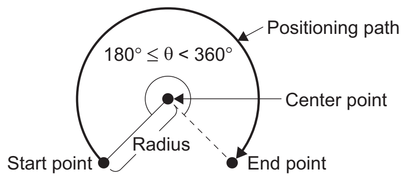
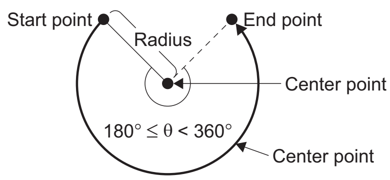
Restriction

2-axis circular interpolation control with center point specification cannot be set in the following cases.

- When the radius exceeds "2147483647.0", the maximum radius for which 2-axis circular interpolation control is possible. ("Out of Radius Range (error code: 1A6CH)")
- When the end point address exceeds positioning range. ("Out of End Point Address Range (error code: 1A6DH)")
- When the start point address is the same as the end point address. ("Start Point - End Point Address Same Value (error code: 1A66H)")
- When the distance between the start point address and the end point address is larger than the radius. ("Radius Setting Error (error code: 1A6EH)")

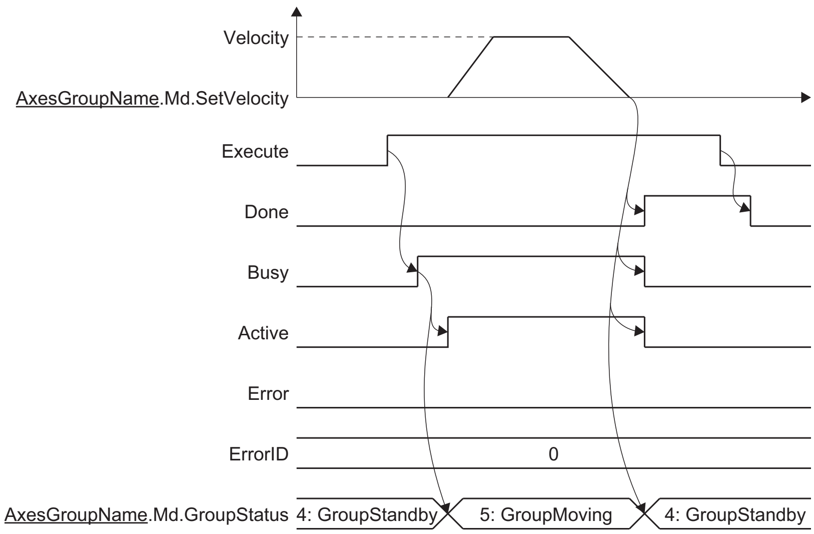
■ Path of radius specification circular interpolation control

The path in the circular interpolation control with the radius specification can be set by Path selection (PathChoice). The setting value, setting description, arc center angle that can be controlled, and path of Path selection (PathChoice) are shown below.

Setting value	Setting description	Arc center angle that can be controlled	Path
0: mcCW	CW • "Path Choice Setting Incorrect (error code: 1AB7H)" will occur and a semicircle can not be drawn.	0° < θ < 180°	
1: mcCCW	CCW • "Path Choice Setting Incorrect (error code: 1AB7H)" will occur and a semicircle can not be drawn.	0° < θ < 180°	
4: mcCWLongWay	CW Detour • The start point, end point, and center point are on one straight line, a semicircle is drawn	180° ≤ θ < 360°	
5: mcCCWLongWay	CCW Detour • The start point, end point, and center point are on one straight line, a semicircle is drawn	180° ≤ θ < 360°	

Timing chart

■ When the operation completes normally



■ When an error occurs

For details when an error occurs, refer to "Basic operation of Execute command (Execute) type Motion control FBs" in the following manual.

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

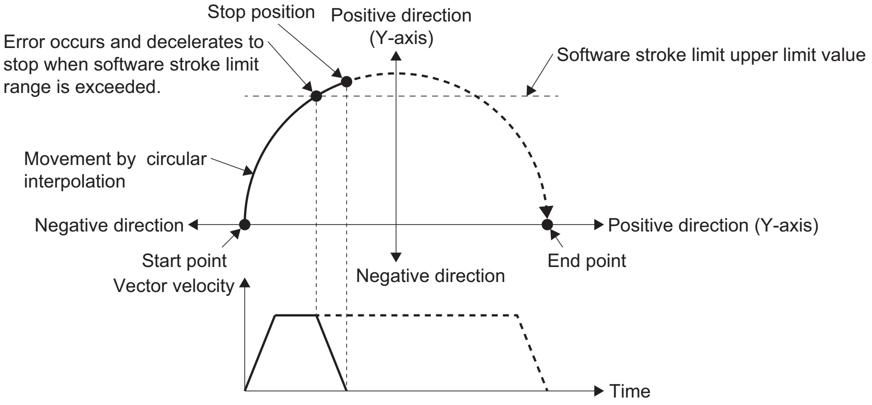
Precautions

- Circular interpolation including an axis whose stroke limit is set to invalid cannot be carried out. "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, "Software Stroke Limit Over (Forward Direction) (error code: 1A03H)" or "Software Stroke Limit Over (Reverse Direction) (error code: 1A04H)" will occur, and the operation will stop.

Ex.

When the upper limit of the software stroke limit is exceeded in the positive direction of the Y-axis

For an immediate stop, stops at the same time that the error occurs. For a deceleration stop, decelerates to a stop along the arc of the circle as shown below.



Necessary slave object

When using multiple axes positioning control, set the following slave objects for all configuration axes in the specified axes group.

- Target position (607AH)

If there is a configuration axis present that does not have the slave object set, "Necessary Slave Object Unset (error code: 1AA8H)" occurs and the axis will not start.

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

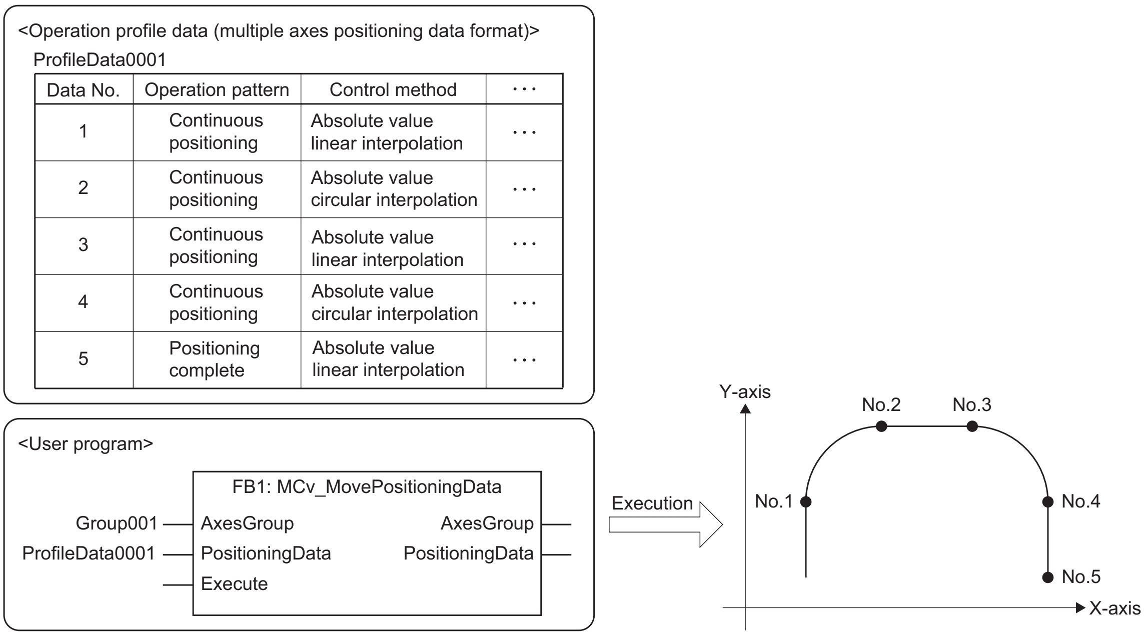
☞ Page 56 Axis Assignment

6.5 Multiple Axes Positioning Data Operation

Multiple axes positioning data operation performs positioning for an axes group with a specified positioning method or velocity up to a predetermined pass point in one start. The positioning method and velocity can be changed for each pass point. Interpolation operation is possible for 1 to 4 configuration axes of the axes group.
The following parameters are set in the operation profile data (multiple axes positioning data format). For details on operation profile data, refer to the following.
☞ Page 460 Multiple axes positioning data
- Command position address of the pass point
- Control method from an arbitrary pass point to the next pass point
- Velocity from an arbitrary pass point to the next pass point

Point

By using the multiple axes positioning data operation, a continuous multiple axes positioning operation is possible without writing a user program to multiple start positioning control FBs.



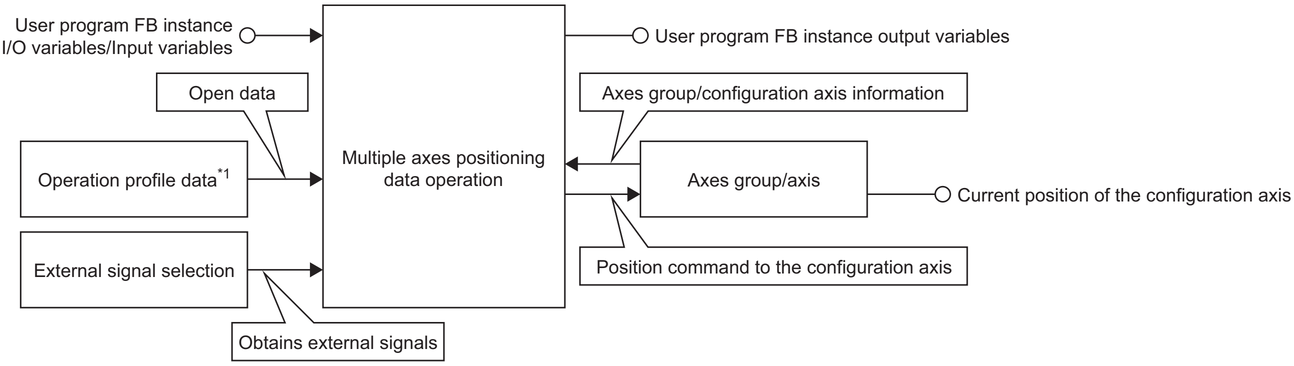
Operation of this function for each system status

○: Possible, ×: Not possible

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

Overall block diagram

The relationship between data used in the multiple axes positioning operation function and other functions is shown below.



*1 For relevancy with the operation profile data, refer to the following.

☞ Page 444 Overall block diagram

Relevant variables

MC_POSITIONING_DATA_ID

Variable/Structure name	Name	Details
ProfileID	Profile ID	Sets the operation profile ID. The Profile ID (ProfileID) is a PROFILE_ID structure.

Relevant FBs

For details on Motion control FBs, refer to the following.

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

Motion control FB	Name	Description
MCv_MovePositioningData	Multiple axes positioning data operation	Operates the axes group by following the specified positioning data.

Control details

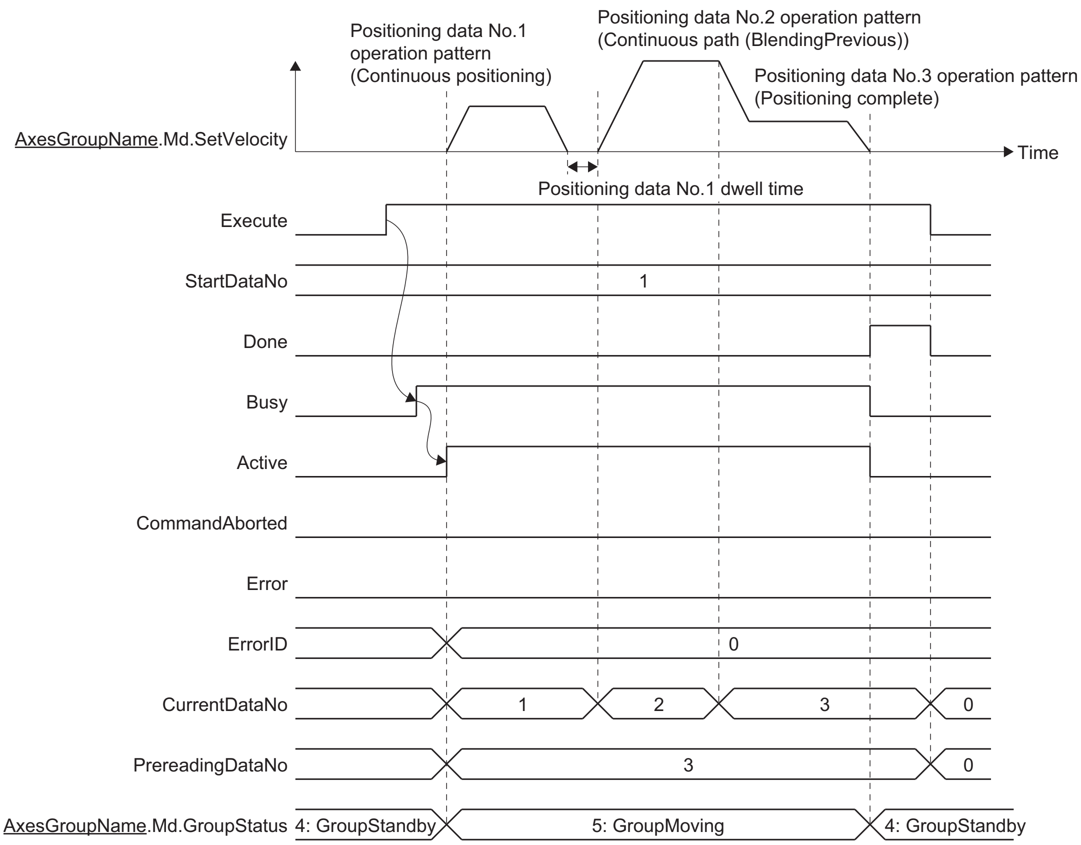
- This function performs the continuous positioning operation specified in the operation profile data (multiple axes positioning data format) for the axes group.
- This function is executed by MCv_MovePositioningData (Multiple Axes Positioning Data Operation)
- Before using this function, it is required to set and enable the axes group. For details, refer to the following.
  ☞ Page 65 Axes Group
  ☞ Page 72 Axes Group Assignment
- Before using this function, it is necessary to make the axes positioning data format operation profile data controllable by opening it in the open area. For the operation profile data format and details on operating the operation profile data, refer to the following.
  ☞ Page 444 Operation Profile Data
- The multiple axes positioning data operation reads the positioning data from the Start positioning data No. (StartDataNo) in ascending order, and continuously executes the multiple axes positioning operation internally. For details, refer to the following.
  ☞ Page 241 Positioning data setting

Timing chart

The Axes group status (AxesGroupName.Md.GroupStatus) of the axes group in control becomes "5: Operating (GroupMoving)". For axes group statuses, refer to the following.

☞ Page 67 Axes group state transition

■ When the operation completes normally



■ When an error occurs

For details when an error occurs, refer to "Basic operation of Execute command (Execute) type Motion control FBs" in the following manual.

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

Positioning data setting

This section describes the positioning operation of the multiple axes positioning data operation based on the positioning data settings.

Positioning data is set in the operation profile data (multiple axes positioning data format). For details on the operation profile data (multiple axes positioning data format), refer to the following.

☞ Page 460 Multiple axes positioning data

Setting positioning data

The positioning data set differs depending on the Control method (ControlMethod). The following table shows the positioning data that can be set.

◎: Always set, ○: Set as required ("—" when not required),

—: Setting not required (Use the initial value or a value within the setting range.)

Variable name	Name	Control method (ControlMethod)					
		Linear interpolation	Circular interpolation	NOP	JUMP	LOOP	LEND
DataNo	Positioning data No.	◎	◎	◎	◎	◎	◎
OperationPattern	Operation pattern	◎	◎	—	—	—	—
ControlMethod	Control method	◎	◎	◎	◎	◎	◎
InterpolationAxis1 to InterpolationAxis4	Interpolation axes 1 to Interpolation axes 4	◎*1	◎*2	—	—	—	—
CircMode	Circular interpolation mode	—	◎	—	—	—	—
CircPathChoice	Path selection	—	◎	—	—	—	—
Position1 to Position4	Target position/movement amount/end point 1 to Target position/movement amount/end point 4	◎*1	◎*2	—	—	—	—
Direction1 to Direction4	Direction selection 1 to Direction selection 4	○	—	—	—	—	—
CircAuxPoint1 to CircAuxPoint4	Sub point 1 to Sub point 4	—	◎*2	—	—	—	—
VelocityMode	Velocity mode	○	—	—	—	—	—
Velocity	Velocity	○	○	—	—	—	—
Acceleration	Acceleration	○	○	—	—	—	—
Deceleration	Deceleration	○	○	—	—	—	—
Jerk	Jerk	○	○	—	—	—	—
CircErrorTolerance	Circular interpolation error tolerance	—	○	—	—	—	—
DataOptions	Positioning data options	○	○	—	—	—	—
DwellTime	Dwell time	○	○	—	—	—	—
Mcode	M code	○	○	—	—	—	—
McodeOutput_Override	M code output timing override	○	○	—	—	—	—
JumpDestinationDataNo	JUMP destination positioning data No.	—	—	—	◎	—	—
ConditionSignalNo	Condition signal No.	—	—	—	○	—	—
LoopCount	LOOP to LEND loop count	—	—	—	—	◎	—
SkipSignalNo	Skip signal No.	—	—	—	—	—	—
Pitch	Pitch	—	—	—	—	—	—

6

*1 The settings are as follows depending on the number of configuration axes set.
◎: Always set, —: Setting not required

Variable name	Name	1 axis	2 axes	3 axes	4 axes
InterpolationAxis1	Interpolation axes 1	◎	◎	◎	◎
InterpolationAxis2	Interpolation axes 2	—	◎	◎	◎
InterpolationAxis3	Interpolation axes 3	—	—	◎	◎
InterpolationAxis4	Interpolation axes 4	—	—	—	◎
Position1	Target position/movement amount/end point 1	◎	◎	◎	◎
Position2	Target position/movement amount/end point 2	—	◎	◎	◎
Position3	Target position/movement amount/end point 3	—	—	◎	◎
Position4	Target position/movement amount/end point 4	—	—	—	◎

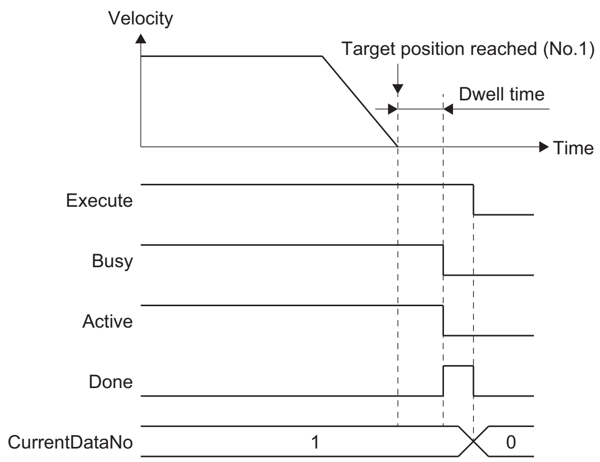
*2 Interpolation axes 3 (InterpolationAxis3), Interpolation axes 4 (InterpolationAxis4), Target position/movement amount/end point 3 (Position3), Target position/movement amount/end point 4 (Position4), Sub point 3 (CircAuxPoint3) and Sub point 4 (CircAuxPoint4) do not require setting "—: setting not required".

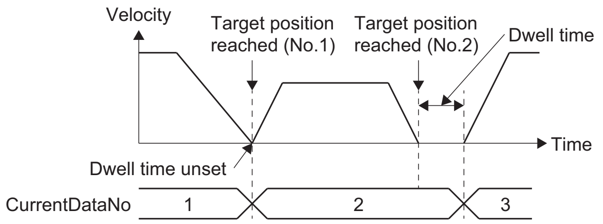
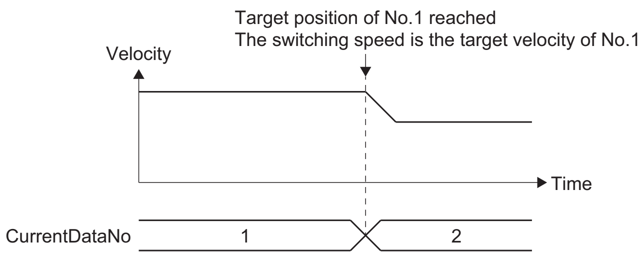
■ Positioning data No. (DataNo)

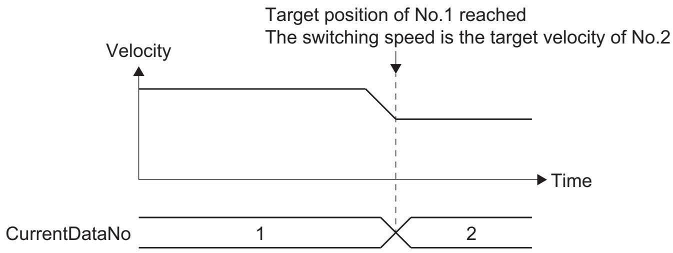
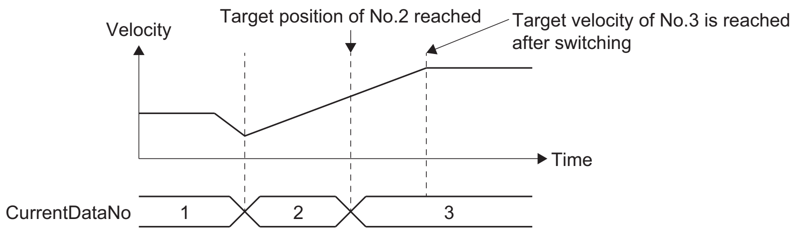
- The positioning data No. sets the index No. of the positioning data.
- MCv_MovePositioningData (Multiple Axes Positioning Data Operation) executes the positioning data in Positioning data No. (DataNo) order. The positioning data is set with the Positioning data No. (DataNo) in ascending order.
- The multiple axes positioning data operation reads the positioning data from the Start positioning data No. (StartDataNo) in ascending order, and continuously executes the multiple axes interpolation operation internally.
- Sets whether to continue executing the next positioning data or not in the operation pattern. (☞ Page 243 Operation pattern (OperationPattern))
- The positioning data can be set with the positioning data No. value blank. However, a blank (unset) positioning data No. cannot be used for the multiple axes positioning data operation. If an unset positioning data No. is executed for the multiple axes positioning data operation, "Operation Profile Data Control Profile ID Setting Incorrect (error code: 1AFAH)" occurs when the corresponding positioning data is read.
- The setting range is "1 to Total number of positioning data".

■ Operation pattern (OperationPattern)

- Set whether to continue executing the positioning data after the set data or not.
- For positioning continue data, the positioning data of the next data No. is automatically executed. In order to complete multiple axes positioning data operation, set the Operation pattern (OperationPattern) of the positioning data of the last operation to "0: Positioning complete (PositioningComplete)".
- Positioning continue data will continue operation until it reaches positioning complete data. Therefore, if there is no positioning complete data, the operation is performed up to the positioning data No. of the total number of positioning data and then the operation is performed again from the No.1 positioning data.
- The types of Operation pattern (OperationPattern) are shown below. The types that can be set differ depending on Control method (ControlMethod).

Positioning continue/ complete	Setting value	Description
Complete	0: Positioning complete (PositioningComplete)	• Executes positioning of the specified 1 data, and completes the multiple axes positioning data operation when the target position is reached. • When the dwell time is set, control will complete after the set time is passed. 

Positioning continue/ complete	Setting value	Description
Continue	1: Continuous positioning (ContinuousPositioning)	• Automatically executes the positioning of the next positioning data No. • The next positioning will be executed in order after the target position is reached. When the target position is reached the operation will automatically decelerate, and the next positioning will execute after the set velocity of the axes group becomes zero. • When the dwell time is set, the next positioning will execute after the set time is passed. • When M code is set, the next positioning will execute after M code reset. For details, refer to the following. ☞ Page 268 M code output function 
Continue	2: Continuous path (BlendingLow)(ContinuousBlendingLow)	• Automatically executes the positioning of the next positioning data No. • The next positioning will be executed in order after the target position is reached. The operation will not stop between the positioning being executed and the next positioning. • When the dwell time is set, the operation will be the same as continuous positioning. • The switching speed is the lower target velocity of either the positioning being executed or the next positioning. • The operation is the same operation as Continuous path (BlendingPrevious) when the target velocity of the positioning being executed is smaller, and Continuous path (BlendingNext) when the target velocity of the next positioning is smaller.
Continue	3: Continuous path (BlendingPrevious)(ContinuousBlendingPrevious)	• Automatically executes the positioning of the next positioning data No. • The next positioning will be executed in order after the target position is reached. The operation will not stop between the positioning being executed and the next positioning. • When the dwell time is set, the operation will be the same as continuous positioning. • This will operate using the target velocity of the positioning being executed until the target position is reached. • When the positioning data switches, the velocity will change until the target velocity of the next positioning. 

Positioning continue/ complete	Setting value	Description
Continue	4: Continuous path (BlendingNext)(ContinuousBlendingNext)	• Automatically executes the positioning of the next positioning data No. • The next positioning will be executed in order after the target position is reached. The operation will not stop between the positioning being executed and the next positioning. • When the dwell time is set, the operation will be the same as continuous positioning. • This is operated so that the set velocity becomes the target velocity of the next positioning.  • Depending on the movement amount of the positioning being executed, the movement amount to change velocity to the target velocity of the next positioning may be insufficient at the target position of the positioning being executed. In this case, velocity change is executed immediately, but the velocity cannot reach the target velocity of the next positioning at the target position of the positioning being executed. 
Continue	5: Continuous path (BlendingHigh)(ContinuousBlendingHigh)	• Automatically executes the positioning of the next positioning data No. • The next positioning will be executed in order after the target position is reached. The operation will not stop between the positioning being executed and the next positioning. • When the dwell time is set, the operation will be the same as continuous positioning. • The switching speed is the higher target velocity of either the positioning being executed or the next positioning. • The operation is the same operation as Continuous path (BlendingPrevious) when the target velocity of the positioning being executed is higher, and Continuous path (BlendingNext) when the target velocity of the next positioning is higher.
Continue	— (no setting)	When the operation pattern setting is not required for the control method (NOP, JUMP, LOOP, and LEND), positioning will continue, but the positioning data No. executed next will differ depending on the control method. For details, refer to the following. ☞ Page 246 Control method (ControlMethod)

6

■ Control method (ControlMethod)

- Set the control method of positioning data.
- When the control method of the positioning data specified in the Positioning data No. (DataNo) is other (NOP, JUMP, LOOP, and LEND) when executing the multiple axes positioning data operation, the operation will read in order from the Start Positioning Data No. (StartDataNo) until the positioning data of the positioning operation (linear interpolation, circular interpolation) is read.
  The positioning data of the positioning operation read first, will be the first executed positioning operation. For reading after the positioning data No. that executes the first positioning operation, refer to the following.
  ☞ Page 257 Pre-reading
- The types of control methods are shown below.

Positioning operation	Setting value	Description
Linear interpolation	0101H: Absolute value linear interpolation (LinearAbsolute)	Specifies the target position of the absolute position, and performs a linear interpolation operation so that the path from the start point (move start point) to the end point is linear. For control details of linear interpolation, refer to the following.*1 ☞ Page 223 Linear interpolation control
	0102H: Relative value linear interpolation (LinearRelative)	Specifies the relative movement amount from the current position, and performs linear a interpolation operation so that the path from the start point (move start point) to the end point is linear. For control details of linear interpolation, refer to the following.*1 ☞ Page 223 Linear interpolation control
Circular interpolation	0103H: Absolute value circular interpolation (CircularAbsolute)	Specifies the absolute position end point and the sub point, and performs 2 axes circular interpolation operation so that the path from the start point (move start point) to the end point forms an arc for 2 linear axes that is arranged orthogonally and mechanically. For control details of circular interpolation, refer to the following.*1 ☞ Page 229 Circular interpolation control
	0104H: Relative value circular interpolation (CircularRelative)	Specifies the movement amount of end point and the sub point from the current position, and performs 2 axes circular interpolation so that the path from the start point (move start point) to the end point forms an arc for 2 linear axes that is arranged orthogonally and mechanically. For control details of circular interpolation, refer to the following.*1 ☞ Page 229 Circular interpolation control

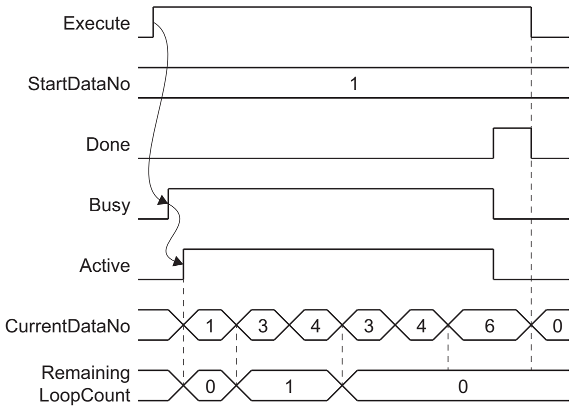
Positioning operation	Setting value	Description
Other	0080H: NOP(NOP)	Moves to the next positioning data No. without any processing. ■NOP use example When a speed switch or temporary stop (automatic deceleration) may be performed between the positioning of two points, NOP can be used to reserve the data. The positioning operation can then be changed just by changing the control method.
	0082H: JUMP(JUMP)	JUMP positioning data jumps to the specified positioning data No. The types of JUMP are shown below. ■Unconditional JUMP • Execution conditions are not set for JUMP. • Jumps to the positioning data No. set in JUMP destination positioning data No. (JumpDestinationDataNo) of the positioning data. • The Condition signal No. (ConditionSignalNo) of the positioning data is set to "0", or the setting is omitted. ■Conditional JUMP • Execution conditions are set for JUMP. • The condition signals set in Condition signal No. (ConditionSignalNo) of the positioning data are used as execution conditions. • Condition signal 1 to Condition signal 10 (ConditionSignal1 to ConditionSignal10) of the operation profile data (multiple axes positioning data format) are used for execution conditions. Set the condition signal to external signal (bit input signal). For the setting details, refer to the following. ☞ Page 266 Condition signal setting • From the condition signals, set the numbers used for each positioning data to Condition signal No. (ConditionSignalNo). For example, when using Condition signal 2 (ConditionSignal2), Condition signal No. (ConditionSignalNo) is set to "2". • If the condition signal set in Condition Signal No. (ConditionSignalNo) is in the signal input status, the execution conditions will be judged as established. If it is not in the signal input status, the execution conditions will be judged as not established. • If the execution conditions are established, the operation jumps to the positioning data No. set in JUMP destination positioning data No. (JumpDestinationDataNo) of the positioning data. If the execution conditions are not established, the next positioning data No. is executed without jumping. • The execution conditions are judged when pre-reading the applicable positioning data whose Control method (ControlMethod) is "0082H: JUMP (JUMP)".
	0083H: LOOP(LOOP)	Loop control by repetition (LOOP to LEND).[*2] Repeats the positioning data between the LOOP to LEND loop for the number of times set in Number of LOOP to LEND repetitions (LoopCount).
	0084H: LEND(LEND)	Returns the operation to the top of the repeating loop (LOOP to LEND).[*2] The loop ends after executing the number of loops set in Number of LOOP to LEND repetitions (LoopCount). After that, the positioning data of the next positioning data No. is executed. Set as shown below when control is completed after repeating execution for number of times set in Number of LOOP to LEND repetitions (LoopCount). • Set positioning data with an Operation pattern (OperationPattern) that is "1: Continuous Positioning (ContinuousPositioning)" to the positioning data No. before LEND. • Set a dummy positioning data (for example, relative value linear interpolation with movement amount 0) to the positioning data No. after LEND.

*1 Positioning operations (linear interpolation and circular interpolation) have the following differences.

Function	Description
Control change function	There is no control change function by retrigger/continuous update. Use the override function when changing the target velocity or acceleration/deceleration and acceleration/deceleration time. For details on the override function, refer to the following. ☞ Page 265 Override function ☞ Page 349 Override Function
Positioning data continuous execution	There is no multiple start relating to continuous execution of operation data. To perform a continuous execution of operation data, use the Operation pattern (OperationPattern).

*2 The operation of the repeating loop with the set LOOP, LEND is shown below.

Positioning data No. (DataNo)	Operation pattern (OperationPattern)	Control method (ControlMethod)	Number of LOOP to LEND repetitions (LoopCount)	Operation
1	3: Continuous path (BlendingPrevious) (ContinuousBlendingPrevious)	0101H: Absolute value linear interpolation (LinearAbsolute)	—	Positioning data executes in the order of No.1 → 2 → 3 → 4 → 5 → 2 → 3 → 4 → 5 → 6. • Timing chart
2	—	0083H: LOOP(LOOP)	2	
3	3: Continuous path (BlendingPrevious) (ContinuousBlendingPrevious)	0101H: Absolute value linear interpolation (LinearAbsolute)	—	
4	1: Continuous positioning (ContinuousPositioning)	0101H: Absolute value linear interpolation (LinearAbsolute)	—	
5	—	0084H: LEND(LEND)	—	
6	0: Positioning complete (PositioningComplete)	0101H: Absolute value linear interpolation (LinearAbsolute)	—	



Precautions

- Set a positioning data No. of positioning data with a setting other than JUMP to JUMP destination positioning data No. (JumpDestinationDataNo). When the jump destination Control method (ControlMethod) is "0083H: JUMP (JUMP)", "Operation Profile Data Control Profile ID Setting Incorrect (error code: 1AFAH)" occurs when executing an FB or during control.
- When using conditional JUMP, execution conditions (signal status of the condition signal) are judged when pre-reading the applicable positioning data whose Control method (ControlMethod) is "0082H: JUMP (JUMP)". Establish the execution conditions before the Pre-reading positioning data No. (PrereadingDataNo) passes JUMP. If the conditions are not established before passing JUMP, the execution conditions will be processed as not established.

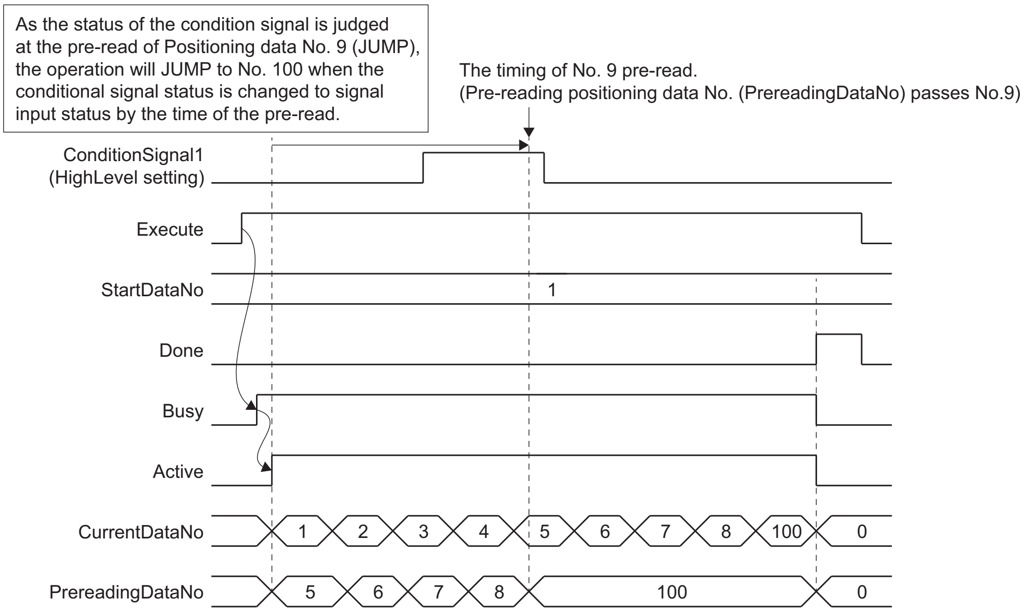
Ex.

When the following positioning data are set

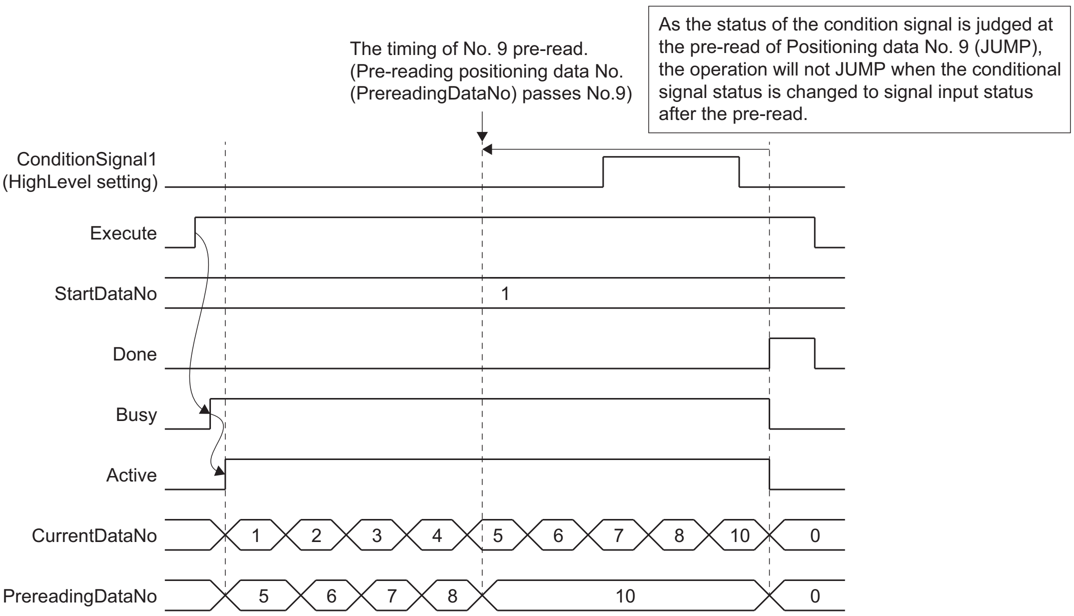
Positioning data No. (DataNo)	Operation pattern (OperationPattern)	Control method (ControlMethod)	JUMP destination positioning data No. (JumpDestinationDataNo)	Condition signal No. (ConditionSignalNo)
1	1: Continuous positioning (ContinuousPositioning)	0101H: Absolute value linear interpolation (LinearAbsolute)	—	—
2	1: Continuous positioning (ContinuousPositioning)	0101H: Absolute value linear interpolation (LinearAbsolute)	—	—
3	1: Continuous positioning (ContinuousPositioning)	0101H: Absolute value linear interpolation (LinearAbsolute)	—	—
4	1: Continuous positioning (ContinuousPositioning)	0101H: Absolute value linear interpolation (LinearAbsolute)	—	—
5	1: Continuous positioning (ContinuousPositioning)	0101H: Absolute value linear interpolation (LinearAbsolute)	—	—
6	1: Continuous positioning (ContinuousPositioning)	0101H: Absolute value linear interpolation (LinearAbsolute)	—	—
7	1: Continuous positioning (ContinuousPositioning)	0101H: Absolute value linear interpolation (LinearAbsolute)	—	—

Positioning data No. (DataNo)	Operation pattern (OperationPattern)	Control method (ControlMethod)	JUMP destination positioning data No. (JumpDestinationDataNo)	Condition signal No. (ConditionSignalNo)
8	1: Continuous positioning (ContinuousPositioning)	0101H: Absolute value linear interpolation (LinearAbsolute)	—	—
9	—	0082H: JUMP(JUMP)	100	1
10	0: Positioning complete (PositioningComplete)	0101H: Absolute value linear interpolation (LinearAbsolute)	—	—
⋮	⋮	⋮	⋮	⋮
100	0: Positioning complete (PositioningComplete)	0101H: Absolute value linear interpolation (LinearAbsolute)	—	—

■When the condition signal is set to signal input status before the conditional JUMP pre-read



■When the condition signal is set to signal input status after the conditional JUMP pre-read



- An error does not occur if there is no LEND after LOOP, however LOOP control will not be executed.
- Any LEND before LOOP is executed will be ignored.
- When there is positioning complete data between LOOP to LEND, the multiple axes positioning data operation will complete after that positioning data has been executed, and loop control will not be executed.

- Nesting of LOOP to LEND is not possible. If set, loop control for the inner level LOOP to LEND only will be executed.

Ex.

When loop control is executed for the inner level LOOP to LEND only

Positioning data No. (DataNo)	Operation pattern (OperationPattern)	Control method (ControlMethod)	Number of LOOP to LEND repetitions (LoopCount)	Operation
1	1: Continuous positioning (ContinuousPositioning)	0101H: Absolute value linear interpolation (LinearAbsolute)	—	• Positioning data executes in the order of No.1 → 2 → 3 → 4 → 5 → 6 → 7 → 4 → 5 → 6 → 7 → 8 → 9. • The settings for positioning data No.2 LOOP and positioning data No.8 LEND are ignored.
2	—	0083H: LOOP (LOOP)	3	
3	1: Continuous positioning (ContinuousPositioning)	0101H: Absolute value linear interpolation (LinearAbsolute)	—	
4	—	0083H: LOOP (LOOP)	2	
5	1: Continuous positioning (ContinuousPositioning)	0101H: Absolute value linear interpolation (LinearAbsolute)	—	
6	1: Continuous positioning (ContinuousPositioning)	0101H: Absolute value linear interpolation (LinearAbsolute)	—	
7	—	0084H: LEND (LEND)	—	
8	—	0084H: LEND (LEND)	—	
9	1: Continuous positioning (ContinuousPositioning)	0101H: Absolute value linear interpolation (LinearAbsolute)	—	

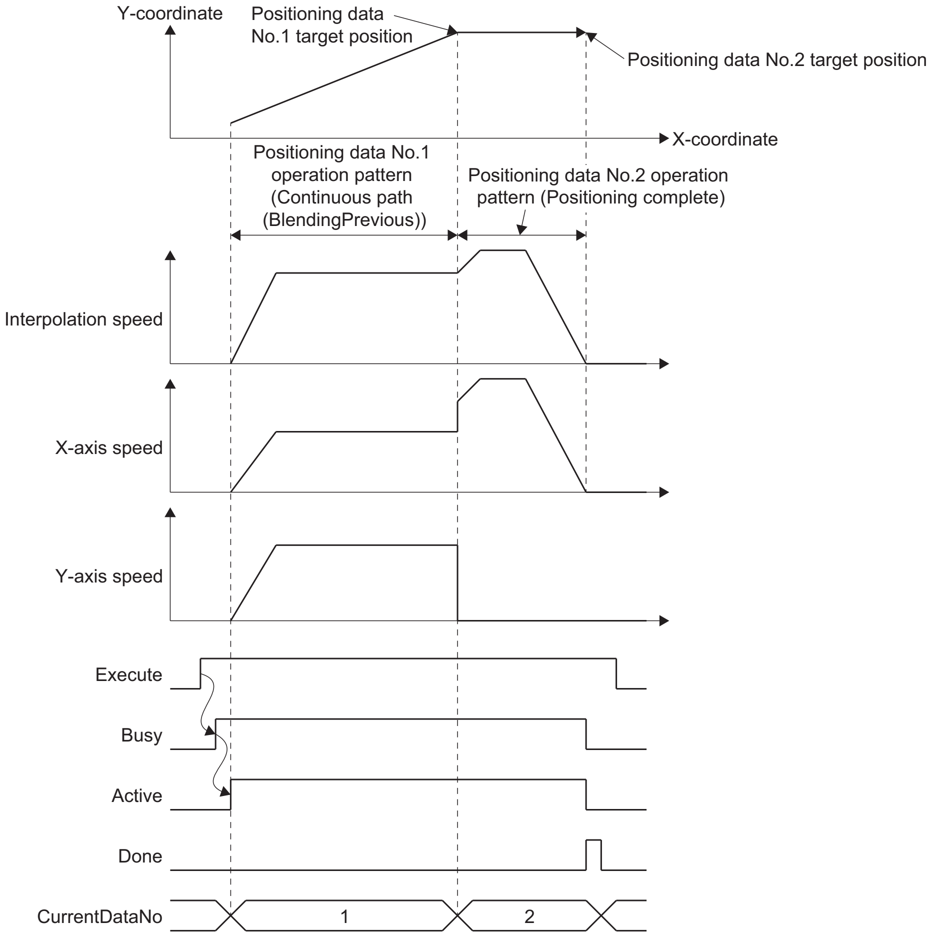
- When JUMP is set after LOOP, loop control will still be executed if LEND is set after the positioning data No. of the JUMP destination.

Ex.

When LOOP to LEND and JUMP (unconditional JUMP) are used together

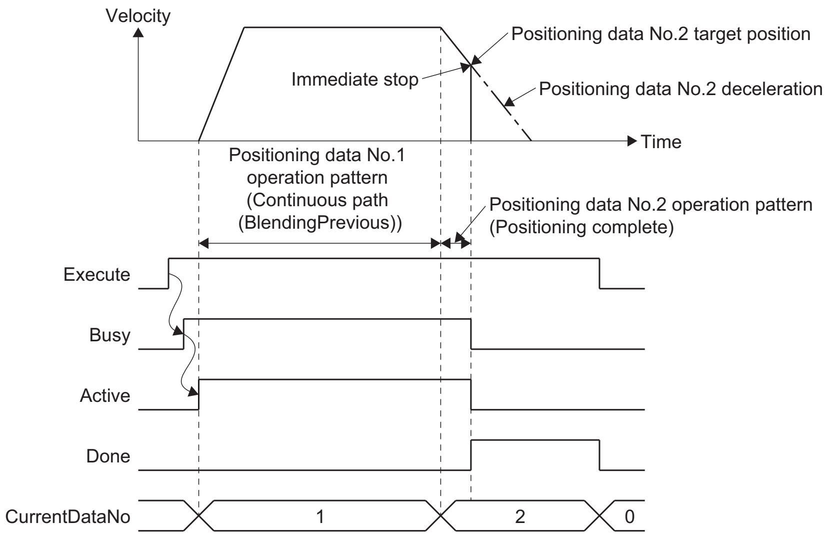
Positioning data No. (DataNo)	Operation pattern (OperationPattern)	Control method (ControlMethod)	Positioning data No. (DataNo)	Jump destination positioning data No. (JumpDestinationDataNo)	Operation
1	1: Continuous positioning (ContinuousPositioning)	0101H:Absolute value linear interpolation (LinearAbsolute)	—	—	• Positioning data executes in the order of No.1 → 2 → 3 → 4 → 101 → 102 → 2 → 3 → 4 → 101 → 102 → 103. • When there is a LOOP or LEND in positioning data No.5 to 100, these settings will be ignored as they are not in the execution order.
2	—	0083H:LOOP (LOOP)	2	—	
3	1: Continuous positioning (ContinuousPositioning)	0101H:Absolute value linear interpolation (LinearAbsolute)	—	—	
4	—	0082H: JUMP (JUMP)	—	101	
⋮	⋮	⋮	⋮	⋮	
101	1: Continuous positioning (ContinuousPositioning)	0101H:Absolute value linear interpolation (LinearAbsolute)	—	—	
102	—	0084H:LEND (LEND)	—	—	
103	1: Continuous positioning (ContinuousPositioning)	0101H:Absolute value linear interpolation (LinearAbsolute)	—	—	

- When switching from positioning data that is multiple axes interpolation data and continuous path data to the next positioning data, the interpolation speed at switching is used as the switching speed. Therefore, a rapid speed change without performing acceleration/deceleration may occur depending on the movement amount and direction of each axis in the next positioning data.



- When switching from continuous path data to positioning complete data or continuous positioning data, the operation may not be able to secure the appropriate distance for deceleration after switching with the switching speed, leading to the target position being overrun. In this case, the automatic deceleration operation will be based on the Operation setting at overrun (AxesGroupName.Pr.OverrunOperation) setting.

Setting value	Operation
1: Immediate stop (ImmediateStop)	Deceleration starts immediately. Upon reaching the target position operation is immediately stopped and an overrun warning is output.



Point

When switching to continuous positioning data and decelerating due to an insufficient movement amount, the positioning of the next positioning data No. is executed after the axis stops.

- There is an upper limit to the number of continuous executions for other (NOP, JUMP, LOOP, and LEND) positioning data. Set positioning data so that it does not exceed the number of continuous executions. When the upper limit is exceeded when executing an FB or during control, "Operation Profile Data Control Profile ID Setting Incorrect (error code: 1AFAH)" occurs.

Upper limit of the number of continuous executions
100 times

Ex.

When NOP is used for positioning data No.2 to No.102

In the following case, an error can be avoided by setting the Control method (ControlMethod) of No.2 to "0082H: JUMP (JUMP)", and JUMP destination positioning data No. (JumpDestinationDataNo) to "103".

Positioning data No. (DataNo)	Operation pattern (OperationPattern)	Control method (ControlMethod)	JUMP destination positioning data No. (JumpDestinationDataNo)	Operation
1	1: Continuous positioning (ContinuousPositioning)	0101H:Absolute value linear interpolation (LinearAbsolute)	—	• The upper limit for the number of continuous executions is exceeded with positioning data No.2 → ... → 102, and an error occurred.
2	—	0080H: NOP (NOP)	—	
⋮	⋮	⋮	⋮	
102	—	0080H: NOP (NOP)	—	
103	1: Continuous positioning (ContinuousPositioning)	0101H:Absolute value linear interpolation (LinearAbsolute)	—	

Ex.

When looping a wait operation until the conditions are established using only the conditional JUMP

When performing a wait operation before FB execution, configure a user program that interlocks FB start.

When performing a wait operation with intermediate positioning data positioning data, use the M code output function. For details on the M code output function, refer to the following.

☞ Page 268 M code output function

Positioning data No. (DataNo)	Operation pattern (OperationPattern)	Control method (ControlMethod)	JUMP destination positioning data No. (JumpDestinationDataNo)	Operation
1	—	0080H: NOP (NOP)	—	• The upper limit for the number of continuous executions is exceeded by repeating NOP and JUMP as positioning data No.1 → 2 → 1 → ..., and an error occurred.
2	—	0082H: JUMP (JUMP)	1	
3	1: Continuous positioning (ContinuousPositioning)	0101H: Absolute value linear interpolation (LinearAbsolute)	—	

■ Interpolation axes 1 to Interpolation axes 4 (InterpolationAxis1 to InterpolationAxis4)

- Sets the interpolation axes from the configuration axes of the axes group.
- The setting contents differ depending on the Control method (ControlMethod) setting.

Control method (ControlMethod)	Description
0101H: Absolute value linear interpolation (LinearAbsolute) 0102H: Relative value linear interpolation (LinearRelative)	Set the Linear interpolation axes (LinearAxes) of linear interpolation. For details, refer to the following. ☞ Page 223 Linear interpolation control Note that Interpolation axes 1 to Interpolation axes 4 (InterpolationAxis1 to InterpolationAxis4), should each be interpreted as Linear interpolation axes (LinearAxes[0] to LinearAxes[3]).
0103H: Absolute value circular interpolation (CircularAbsolute) 0104H: Relative value circular interpolation (CircularRelative)	Set the Circular interpolation axes (CircAxes"0", CircAxes"1") of circular interpolation to Interpolation axes 1 (InterpolationAxis1) and Interpolation axes 2 (InterpolationAxis2). For details, refer to the following. ☞ Page 229 Circular interpolation control Interpolation axes 3 (InterpolationAxis3) and Interpolation axes 4 (InterpolationAxis4) are not used. The settings are ignored.

- The configuration axes specified as interpolation axes can be set differently for each positioning data. A different number of interpolation axes can also be set. However, if the specified configuration axes and number of interpolation axes is different between continuous path data and the next positioning data, note that the speed of each axis can rapidly change when switching positioning data. For details on positioning speed, refer to the following.
  ☞ Page 259 Positioning speed
  ☞ Page 211 Positioning speed in multiple axes positioning control

■ Circular interpolation mode (CircMode)

Sets the circular interpolation method (border point specification, center point specification, and radius specification) used to perform circular interpolation control.

For details on Circular interpolation mode (CircMode), refer to the following.

☞ Page 229 Circular interpolation control with border point specification
☞ Page 231 Circular interpolation control with central point specification
☞ Page 234 Radius specification circular interpolation control

■ Path selection (CircPathChoice)

Sets the rotation direction (CW, CCW, Shortcut, Detour, CW Detour, CCW Detour) of the circular interpolation control.

The Path selection (CircPathChoice) that can be set differs depending on Circular interpolation mode (CircMode). For details, refer to the following.

☞ Page 233 Path of central point specification circular interpolation control
☞ Page 236 Path of radius specification circular interpolation control

■ Target position/movement amount/end point 1 to target position/movement amount/end point 4 (Position1 to Position4)

The setting contents differ depending on the Control Method (ControlMethod) settings.

Control method (ControlMethod)	Description
0101H: Absolute value linear interpolation (LinearAbsolute)	• Sets the target position of the absolute position. • Target position/movement amount/end point 1 to Target position/movement amount/end point 4 (Position1 to Position4) indicate the target position of the configuration axes set in Interpolation axes 1 to Interpolation axes 4 (InterpolationAxis1 to InterpolationAxis4). • Target position/movement amount/end point 1 to target position/movement amount/end point 4 (Position1 to Position4) corresponding to "0" or omitted Interpolation axes 1 to interpolation axes 4 (InterpolationAxis1 to InterpolationAxis4) can be omitted. • For setting values, refer to "Target position (Position)" under "MC_MoveAbsolute (Absolute Value Positioning)" in the following manual. MELSEC iQ-R Programming Manual (Motion Control Function Blocks)
0102H: Relative value linear interpolation (LinearRelative)	• Sets the movement amount from the current position when positioning data is executed. • Target position/movement amount/end point 1 to Target position/movement amount/end point 4 (Position1 to Position4) indicate the movement amount of the configuration axes set in Interpolation axes 1 to Interpolation axes 4 (InterpolationAxis1 to InterpolationAxis4). • Target position/movement amount/end point 1 to Target position/movement amount/end point 4 (Position1 to Position4) corresponding to "0" or omitted Interpolation axes 1 to Interpolation axes 4 (InterpolationAxis1 to InterpolationAxis4) can be omitted. • The setting value is "-10000000000.0 to 10000000000.0".
0103H: Absolute value circular interpolation (CircularAbsolute)	• Sets the end point position of the absolute position. • Target position/movement amount/end point 1 (Position1) and Target position/movement amount/end point 2 (Position2) indicate the target position of the configuration axes set in Interpolation axes 1 (InterpolationAxis1) and Interpolation axes 2 (InterpolationAxis2). • Target position/movement amount/end point 3 (Position3) and Target position/movement amount/end point 4 (Position4) can be omitted. (The settings are ignored.) • The setting value is "-10000000000.0 ≤ Setting value < 10000000000.0". *: When the ring counter is enabled, it will be the ring counter range.
0104H: Relative value circular interpolation (CircularRelative)	• Sets the movement amount from the current position to the end point when positioning data is executed. • Target position/movement amount/end point 1 (position1) and Target position/movement amount/end point 2 (position2) indicate the movement amount of the configuration axes set in Interpolation axes 1 (InterpolationAxis1) and Interpolation axes 2 (InterpolationAxis2). • Target position/movement amount/end point 3 (Position3) and Target position/movement amount/end point 4 (Position4) can be omitted. (The settings are ignored.) • The setting value is "-10000000000.0 to 10000000000.0".

■ Direction selection 1 to Direction selection 4 (Direction1 to Direction4)

Sets the direction (positive direction, negative direction, and shortest path) used to move using linear interpolation control.
The setting contents differ depending on the Control method (ControlMethod) settings.

Control method (ControlMethod)	Description
0101H: Absolute value linear interpolation (LinearAbsolute)	• Sets the direction (positive direction, negative direction, shortest path) used to move from the current position to the target position. • Direction selection 1 to Direction selection 4 (Direction1 to Direction4) indicate the direction selection of the configuration axes set in Interpolation axes 1 to Interpolation axes 4 (InterpolationAxis1 to InterpolationAxis4). • A linear axis with a valid software stroke limit ignores this setting. • Direction selection 1 to Direction selection 4 (Direction1 to Direction4) corresponding to "0" or omitted Interpolation axes 1 to Interpolation axes 4 (InterpolationAxis1 to InterpolationAxis4) can be omitted. • For details on each direction selection, refer to "Direction selection (Direction)" under "MC_MoveAbsolute (Absolute Value Positioning)" in the following manual. MELSEC iQ-R Programming Manual (Motion Control Function Blocks)
0102H: Relative value linear interpolation (LinearRelative)	Setting is not required. The values are ignored.

■ Sub point 1 to sub point 4 (CircAuxPoint1 to CircAuxPoint4)

Sets the position of the sub point (border point, center point, and radius) that executes circular interpolation control.

The setting contents differ depending on the Circular interpolation mode (CircMode) settings.

Circular interpolation mode (CircMode)	Description
0: Border point specification (mcBorder) 1: Center point specification (mcCenter)	• Sub point 1 (CircAuxPoint1) and Sub point 2 (CircAuxPoint2) indicate the border point/center point of the configuration axes set in Interpolation axes 1 (InterpolationAxis1) and Interpolation axes 2 (InterpolationAxis2). It is set with an absolute value or a relative value. • Sub point 3 (CircAuxPoint3) and Sub point 4 (CircAuxPoint4) can be omitted. (The settings are ignored.) • The setting range depending on the set Control method (ControlMethod) is as follows. ■0103H: Absolute value circular interpolation (CircularAbsolute) -10000000000.0 ≤ Setting value < 10000000000.0 *: When ring counter is enabled, this becomes the ring counter range. ■0104H: Relative value circular interpolation (CircularRelative) -10000000000.0 to 10000000000.0
2: Radius specification (mcRadius)	• Sub point 1 (CircAuxPoint1) indicates the radius of the arc. • Sub point 2 to Sub point 4 (CircAuxPoint2 to CircAuxPoint4) can be omitted. (The settings are ignored.) • The setting range is "0.000001 to 2147483647.0".

■ Velocity mode (VelocityMode)

Sets the velocity mode (vector velocity specification, long axis velocity specification, and reference axis velocity specification) of linear interpolation control.

For details on the Velocity mode (VelocityMode), refer to the following.

☞ Page 211 Vector velocity specification
☞ Page 212 Long axis velocity specification
☞ Page 214 Reference axis velocity specification

■ Velocity (Velocity)

Sets the positioning speed of the multiple axes interpolation.

For details, refer to the following.

☞ Page 211 Positioning speed in multiple axes positioning control

The setting range is "0, 0.0001 to 2500000000.0".

■ Acceleration (Acceleration)

Sets the acceleration for multiple axes interpolation.

The setting contents differ depending on the acceleration/deceleration method setting (bit 0 to 2) of Positioning data options (DataOptions). For details on the acceleration/deceleration method, refer to the following.

☞ Page 317 Acceleration/deceleration Processing Function

■ Deceleration (Deceleration)

Sets the deceleration for the multiple axes interpolation.

The setting contents differ depending on the acceleration/deceleration method setting (bit 0 to 2) of Positioning data options (DataOptions). For details on the acceleration/deceleration method, refer to the following.

☞ Page 317 Acceleration/deceleration Processing Function

■ Jerk (Jerk)

Sets the jerk for the multiple axes positioning control.

The setting contents differ depending on the acceleration/deceleration method setting (bit 0 to 2) of Positioning data options (DataOptions). For details on the acceleration/deceleration method, refer to the following.

☞ Page 317 Acceleration/deceleration Processing Function

■ Circular interpolation error tolerance (CircErrorTolerance)

Sets the circular interpolation error tolerance when executing circular interpolation control with the center point specification.

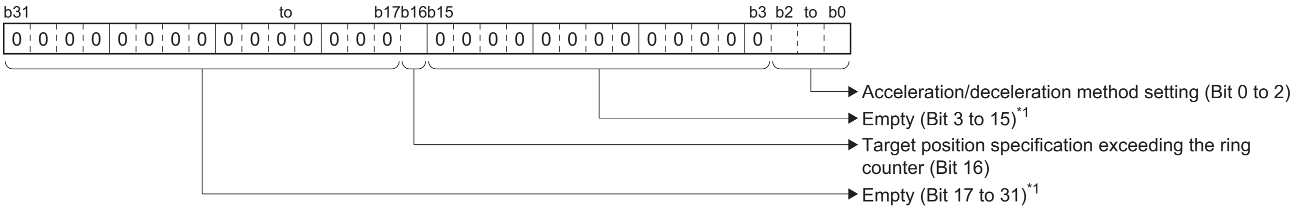
For details on Circular interpolation error tolerance (CircErrorTolerance), refer to the following.

☞ Page 234 Circular interpolation error tolerance

The setting range is "0.000001 to 100000.0".

■ Positioning data options (DataOptions)

Sets the function options for linear interpolation control and circular interpolation control by specifying the bit.



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

Bit	Name	Description
0 to 2	Acceleration/deceleration method setting	Sets the acceleration/deceleration method for executing control. • 0: Acceleration/deceleration specification method (mcAccDec) • 1: Acceleration/deceleration time-fixed method (mcFixedTime) *: For details on the acceleration/deceleration method, refer to the following. ☞ Page 322 Acceleration/deceleration method
16	Target position specification exceeding the ring counter	Sets to allow or not allow the target position to exceed the ring counter upper limit value and ring counter lower limit value when software stroke limit is disabled. Set when "0101H: Absolute Value Linear Interpolation (LinearAbsolute)" is set in the Control method (ControlMethod). • 0: Do not allow • 1: Allow

■ Dwell time (DwellTime)

Sets the dwell time.

The setting range is shown below.

- 0.0 [s]: Dwell function disabled
- 0.000001 [s] to 8400.0 [s]: Dwell function enabled

*1 When set to a positive number less than 0.000001, it will be read as "0.0 [s]".

■ M code (Mcode)

Sets the M code for each positioning data.

- Set "0 (default value)" to not output the M code.
- The setting range is "1 to 65535".

■ M code output timing override (McodeOutput_Override)

Sets the M code output timing for each positioning data.

For details on the M code output timing, refer to the following.

☞ Page 268 M code output function

Setting value	Description
0: Use FB options (UseFbOptions)	Uses the M code output timing (Options (Options): Bit 16) setting value of MCv_MovePositioningData (Multiple Axes Positioning Data Operation).
1: WITH mode (WithMode)	Changes M strobe (Mstrobe) to TRUE when positioning of the positioning data starts, and stores the positioning data M code (Mcode) in Valid M code (ValidMcode).
2: AFTER mode (AfterMode)	Changes M strobe (Mstrobe) to TRUE when the target position is reached (after the setting time has passed when the dwell time has been set), and stores the positioning data M code in M code (Mcode).

■ JUMP destination data No. (JumpDestinationDataNo)

Sets the position data No. of the jump destination in JUMP.

Sets the Positioning data No. (DataNo) of a positioning data whose Control method (ControlMethod) is other than "0082H: JUMP (JUMP)".

The setting range is "1 to Total number of positioning data".

■ Condition signal No. (ConditionSignalNo)

Sets the Condition signal No.

- Set "0 (default value)" when not using the condition signal.
- The setting range is "1 to 10".

■ Number of LOOP to LEND repetitions (LoopCount)

Sets the number of repetitions for loop control

The setting range is "1 to 65535".

■ Skip signal No (SkipSignalNo.)

Set this to "0".

■ Pitch (Pitch)

Set this to "0".

Pre-reading

- The Motion system imports positioning data by pre-reading the positioning data ahead of the positioning data being executed in MCv_MovePositioningData (Multiple Axes Positioning Data Operation) from the operation profile data open area.
- When executing an FB, up to 4 multiple axes interpolation data ahead of the Start positioning data No. (StartDataNo) are pre-read. When positioning continues after the pre-read multiple axes interpolation data, pre-reading continues while the FB is being executed. Pre-reading occurs when switching positioning data so that the number of pre-read multiple axes interpolation data stays at the maximum of 4.
- During FB execution, the last positioning data No. of the pre-read multiple axis interpolation data is stored in the Pre-reading positioning data No. (PrereadingDataNo).

6

Precautions

- When positioning data with short movement amounts are executed continuously where positioning data switches in several operation cycles, the operation may wait for pre-read to occur in the middle of positioning data execution.
  When waiting for for pre-read to occur in continuous path data, the set velocity will suddenly change to "0" in one operation cycle when the target position is reached. After that, when pre-read occurs and the positioning data switches to the next positioning data, the set velocity will suddenly change to switching speed in one operation cycle. To avoid the sudden change in velocity due to pre-read waiting, increase the movement amount of each positioning data, or decrease the target velocity.

Near pass

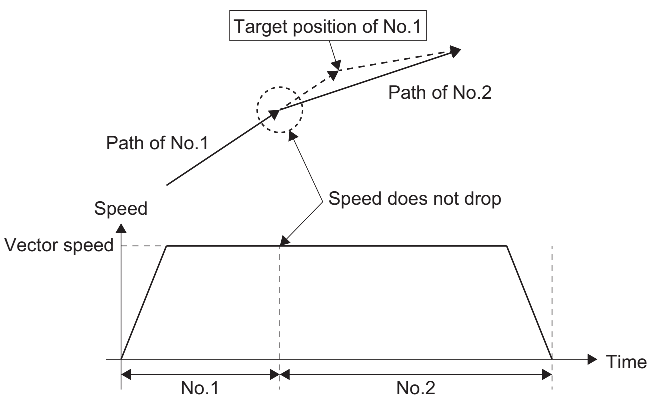
When executing continuous path data, near pass is performed to suppress machine vibration that occurs when switching to the next positioning data.

The near pass carries over the remaining movement amount at the end of positioning to the next positioning. Speed reductions are eliminated by not aligning positions at each positioning operation, which suppresses machine vibrations caused by speed changes.

Since position alignment is not performed for each positioning, the operation is controlled in a path that passes near the target position of the positioning data.

Ex.

For executing 2-axes linear interpolation with continuous path (No.1 operation pattern set to continuous path and continuing to No.2)



Precautions

- When using the multiple axes interpolation operation with continuous path data, if the movement amount of the multiple axes interpolation operation is too small, the set velocity may not reach the target velocity.

Combination with other functions

This section describes when positioning data is executed with the operation pattern of continuous positioning.

■ Software stroke limit

The basic operation is based on the specifications of the software stroke limit relevant to operation FBs. For details, refer to the following.

☞ Page 304 Software Stroke Limit

- Returning from outside the software stroke limit range
  While the software stroke limit is disabled by Software Stroke Limit Override (AxisName.Cd.SwStrokeLimit_Override) of the axes group configuration axes, in addition to when a start, or multiple start occurs, if positioning data is pre-read, "Start when SW Stroke Limit is Disabled (event code: 07F5)" is recorded in the event history.
- Checking the software stroke limit enabled/disabled status
  - The software stroke limit enabled/disabled status can be checked with the Software stroke limit override (AxisName.Md.SwStrokeLimit_Override) of each axes group configuration axis.
  - In addition to when a start or multiple start occurs, the Software stroke limit override (AxisName.Cd.SwStrokeLimit_Override) of the axes group configuration axis is imported when positioning data is pre-read, and the Software stroke limit override (AxisName.Cd.SwStrokeLimit_Override) is updated. For details of imported values and updated values, refer to the following.
    ☞ Page 315 Checking the hardware stroke limit enabled/disabled
- Software stroke limit check specifications
  For MCv_MovePositioningData (Multiple Axes Positioning Data Operation), an additional check is performed for each positioning data.

Check timing	Check details	Processing when an error occurs
When FB is executing, or during control (When positioning data is pre-read)	An error occurs when the target position of positioning data is outside the software stroke limit range.	"Software Stroke Limit Over (Target Position) (error code: 1A00H)" occurs, and the axis is not controlled.
	An error occurs when the position at positioning data execution is outside the software stroke limit range.	"Software Stroke Limit Over (Start Position) (error code: 1A01H)" occurs, and the axis is not controlled.
During control	An error occurs when the current position is outside the software stroke limit range during control.	"Software Stroke Limit Over (Forward Direction) (error code: 1A03H)" or "Software Stroke Limit Over (Reverse Direction) (error code: 1A04H)" occurs and stop processing is executed.*1

*1 The error occurs when detecting a command to outside the software stroke limit range, and stop processing is executed.
To stop within the software stroke limit range, set to "immediate stop".

■ Hardware stroke limit check

The basic operation is based on the specifications of the hardware stroke limit relevant to operation FBs. For details, refer to the following.

☞ Page 311 Hardware Stroke Limit

- Returning from outside the hardware stroke limit range
  While the hardware stroke limit is disabled by Hardware stroke limit override (AxisName.Cd.HwStrokeLimit_Override) of the axes group configuration axes, in addition to when a start, or multiple start occurs, if positioning data is pre-read, "Start when HW Stroke Limit is Disabled (event code: 07F6)" is recorded in the event history.
- Checking the hardware stroke limit enabled/disabled status
  - The hardware stroke limit check enabled/disabled status can be checked with the Hardware stroke limit override (AxisName.Md.HwStrokeLimit_Override) of each axes group configuration axis.
  - In addition to when a start or multiple start occurs, the Hardware stroke limit override (AxisName.Cd.HwStrokeLimit_Override) of the axes group configuration axis is imported when positioning data is pre-read, and the Hardware stroke limit override (AxisName.Cd.SwStrokeLimit_Override) is updated. For details of imported values and updated values, refer to the following.
    ☞ Page 315 Checking the hardware stroke limit enabled/disabled

■ Positioning speed

The basic operation is based on the specifications of the positioning speed when using multiple axes positioning control. For details, refer to the following.

☞ Page 211 Positioning speed in multiple axes positioning control

- Operation when specifying speed 0
  - When the Velocity (Velocity) of the first executed positioning data in positioning is set to "0.0", and MCv_MovePositioningData (Multiple Axes Positioning Data Operation) is started, "Out of Velocity Range (error code:1A4DH)" occurs. When starting positioning with the target velocity as "0.0", and executing speed change during control, use the override function. For details on the override function, refer to the following.
    ☞ Page 349 Override Function
    For details on the override function in multiple axes positioning data operation, refer to the following.
    ☞ Page 265 Override function
  - When the Velocity (Velocity) of the first executed positioning data in positioning is set to "0.0", and MCv_MovePositioningData (Multiple Axes Positioning Data Operation) is multiple started, depending on the set velocity of the previous FB, the following occurs.
    - 0.0: "Out of Velocity Range (error code: 1A4DH)" occurs at multiple start.
    - Other than 0.0: The specified velocity of the previous FB is used.
  - When the Velocity (Velocity) of a positioning data in the middle of the operation is set to "0.0", the specified speed of the previous positioning data is used.
- During continuous path

  The following explains the operation in the multiple axes positioning data operation when "2: Continuous path (BlendingLow) (ContinuousBlendingLow)", "3: Continuous path (BlendingPrevious) (ContinuousBlendingPrevious)", "4: Continuous path (BlendingNext) (ContinuousBlendingNext)", or "5: Continuous path (BlendingHigh) (ContinuousBlendingHigh)" is selected in Operation pattern (OperationPattern) of the first positioning data (No.1) and different velocity modes are set to the Velocity mode (VelocityMode) of the first positioning data (No.1) and the second positioning data (No.2).
  - Positioning can continue for positioning data (No.1) and positioning data (No.2) when the Velocity mode (VelocityMode) is different for positioning data (No.1) and positioning data (No.2). However, when positioning data (No.2) is pre-read, "Multiple Start Velocity Mode Specified Mismatch Warning (warning code: 0D11H)" occurs.
  - Positioning data (No.1) uses the positioning data (No.1) Velocity mode (VelocityMode), and positioning data (No.2) uses positioning data (No.2) Velocity mode (VelocityMode) for the positioning operation.
  - When switching between positioning data (No.1) and positioning data (No.2), the velocity is not converted using the Velocity mode (Velocity). The value of the positioning velocity in the velocity mode of the positioning data (No. 1) is switched as it is to the positioning speed in the velocity mode of the positioning data (No.2).
  - When Velocity mode (VelocityMode) is different for positioning data (No.1) and positioning data (No.2), note that the speed of each axis and the vector velocity may rapidly change when switching positioning data.

Ex.

When "3: Continuous path (BlendingPrevious) (ContinuousBlendingPrevious)" is selected

The operation when switching from positioning data with reference axis velocity specification to positioning data with vector velocity specification is as follows.

It operates so that the No.1 axis reference velocity becomes the velocity (reference axis velocity) set in Velocity (Velocity) of No. 1 at the No.1 target position.

The reference axis velocity when reaching the No.1 target position is read as the current vector velocity value of No.2 without unit conversion, and it accelerates/decelerates the vector velocity set in Velocity (Velocity) of No.2.

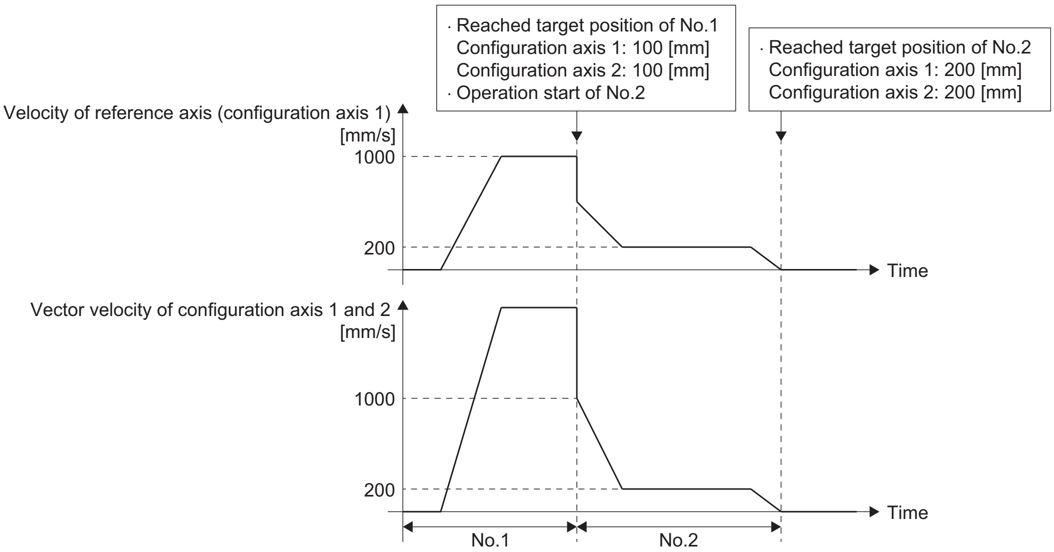
- Axis setting, axes group setting

Setting item	Setting value
Position and velocity command unit of configuration axis 1	mm/s
Position and velocity command unit of configuration axis 2	mm/s
Position and velocity command unit of the axes group	mm/s

- No.1 (command unit: mm/s), No.2 (command unit: mm/s) settings

Setting item	Setting value	
	No.1[*1]	No.2[*1]
Operation pattern (OperationPattern)	3: Continuous path (BlendingPrevious) (ContinuousBlendingPrevious)	1: Continuous positioning (ContinuousPositioning)
Control method (ControlMethod)	0101H:Absolute value linear interpolation (LinearAbsolute)	0101H: Absolute value linear interpolation (LinearAbsolute)
Interpolation axes 1 (InterpolationAxis1)	1	1
Interpolation axes 2 (InterpolationAxis2)	2	2
Interpolation axes 3 (InterpolationAxis3)	0	0
Target position/movement amount/end point 1 (Position1)	100	200
Target position/movement amount/end point 2 (Position2)	100	200
Velocity mode (VelocityMode)	2: Reference axis velocity specification (ReferenceAxisSpeed)	0: Vector velocity specification (VectorSpeed)
Velocity (Velocity)	1000	200

*1 The command unit for No.1 and No.2 is "mm/s".



■Operation example

(1) The operation reaches the target position of No.1 at the reference axis velocity 1000 [mm/s] of configuration axis 1 (the reference axis), and switches to No.2.

(2) No.2 starts operation at vector velocity 1000 [mm/s].

(3) Decelerates until the vector velocity 200 [mm/s].

(4) Positioning is performed to the target position of No.2.

■ Command unit

The basic operation is based on the specifications of the command unit when using multiple axes positioning control. For details, refer to the following.

☞ Page 219 Command unit in the multiple axes positioning control

- Position command unit

  When the position command unit of the axes group and the interpolation axes set in positioning data are different, the warning "Position Command Unit Mismatch Warning (warning code: 0D08H)" occurs at start or pre-read.
- Velocity command unit during continuous path

  The operation in the multiple axes positioning data operation when "2: Continuous path (BlendingLow) (ContinuousBlendingLow)", "3: Continuous path (BlendingPrevious) (ContinuousBlendingPrevious)", "4: Continuous path (BlendingNext) (ContinuousBlendingNext)", or "5: Continuous path (BlendingHigh) (ContinuousBlendingHigh)" is selected in Operation pattern (OperationPattern) of the first positioning data (No.1) and different velocity command units are used for the first positioning data (No.1) and the second positioning data (No.2) is as follows.
  - Positioning can continue for positioning data (No.1) and positioning data (No.2) when the command unit is different for positioning data (No.1) and positioning data (No.2). However, when positioning data (No.2) is pre-read, " Position Command Unit Mismatch Warning (warning code: 0D08H)" occurs.
  - Positioning data (No.1) uses the positioning data (No.1) command unit, and positioning data (No.2) uses positioning data (No.2) command unit for the positioning operation.
  - When switching between positioning data (No.1) and positioning data (No.2), the velocity is not converted using the position command mode. The value of the positioning velocity in the command unit of the positioning data (No.1) is switched as it is to the positioning speed in the command unit of the positioning data (No.2).

6

Ex.

When "3: Continuous path (BlendingPrevious) (ContinuousBlendingPrevious)" is selected

When the command units of No.1 and No.2 are different, positioning can continue with No.2 when the operation pattern of No.1 is set to continuous path.

The velocity value for No.1 will be taken as the velocity for No.2 without unit conversion.

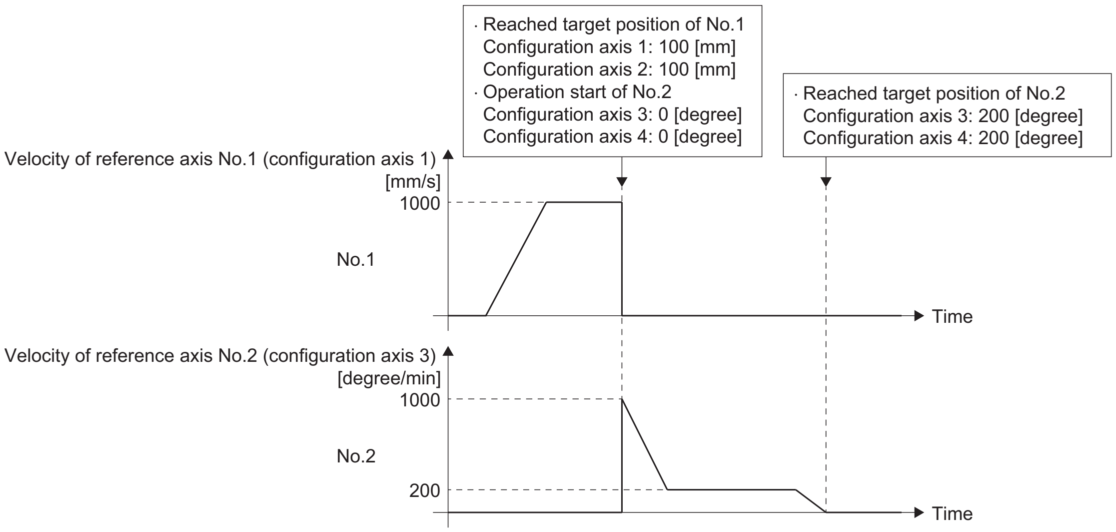
- Axis setting

Setting item	Setting value
Position and velocity command unit of configuration axis 1	mm/s
Position and velocity command unit of configuration axis 2	mm/s
Position and velocity command unit of configuration axis 3	degree/min
Position and velocity command unit of configuration axis 4	degree/min

- No.1 and No.2 settings

Setting item	Setting value	
	No.1*1	No.2*2
Operation pattern (OperationPattern)	3: Continuous path (BlendingPrevious) (ContinuousBlendingPrevious)	1: Continuous positioning (ContinuousPositioning)
Control method (ControlMethod)	0101H:Absolute value linear interpolation (LinearAbsolute)	0101H:Absolute value linear interpolation (LinearAbsolute)
Interpolation axes 1 (InterpolationAxis1)	1	3
Interpolation axes 2 (InterpolationAxis2)	2	4
Interpolation axes 3 (InterpolationAxis3)	0	0
Target position/movement amount /end point 1 (Position1)	100	200
Target position/movement amount/end point 2 (Position2)	100	200
Velocity mode (VelocityMode)	2: Reference axis velocity specification (ReferenceAxisSpeed)	2: Reference axis velocity specification (ReferenceAxisSpeed)
Velocity (Velocity)	1000	200

*1 The command unit for No.1, and command unit for configuration axis 1 is "mm/s".
*2 The command unit for No.2, and command unit for configuration axis 3 is "degree/min".



■Operation example

(1) The operation reaches the target position of No.1 at the reference axis velocity 1000 [mm/s] of configuration axis 1 (the reference axis), and switches to No.2.

(2) No.2 starts operation at vector velocity 1000 [mm/s].

(3) Decelerates until the vector velocity 200 [mm/s].

(4) Positioning is performed to the target position of No.2.

■ Acceleration/deceleration processing function

The basic operation is based on the specifications of the acceleration/deceleration processing function. For details, refer to the following.

☞ Page 317 Acceleration/deceleration Processing Function

- Acceleration/deceleration method setting

  The acceleration/deceleration method is set in Positioning data options (DataOptions) in positioning data.

  Do not use Options (Options) in MCv_MovePositioningData (Multiple Axes Positioning Data Operation).
- Operation when specifying acceleration/deceleration 0

  When the acceleration or deceleration of a positioning data in the middle of the operation is set to "0.0", the specified acceleration or specified deceleration of the previous positioning data is used.
- Acceleration limit/deceleration limit

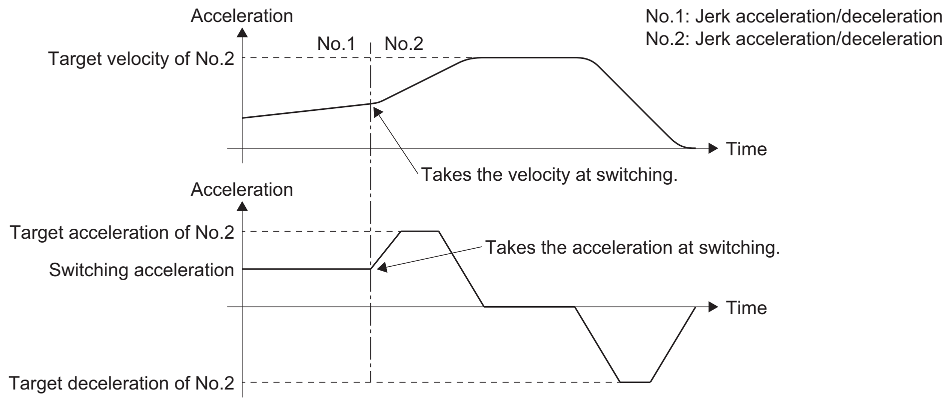
  The acceleration limit and deceleration limit are executed for each positioning data. When the target acceleration, or target deceleration of a positioning data in the middle of the operation exceeds the acceleration limit value, or deceleration limit value, "Acceleration Limit Value Over (error code: 1A38H)", or "Deceleration Limit Value Over (error code: 1A39H)" occurs. The timing when the error is output is at the start, or when pre-reading during control.
- Jerk limit

  The jerk limit is executed for each positioning data. When the jerk set in a positioning data in the middle of the operation exceeds the jerk limit value, "Jerk Limit Value Over (error code: 1A3AH)" occurs. The timing when the error is output is at the start, or when pre-reading during control.
- Acceleration/deceleration method and operation pattern

  ■Switching "from jerk acceleration/deceleration to another jerk acceleration/deceleration"

  The axis accelerates/decelerates taking over the velocity and the acceleration (deceleration) when switching positioning data. However, when the jerk setting values are different between the positioning data being executed and the next positioning data, the jerk of the next positioning data will be the same as the jerk of the positioning data being executed.

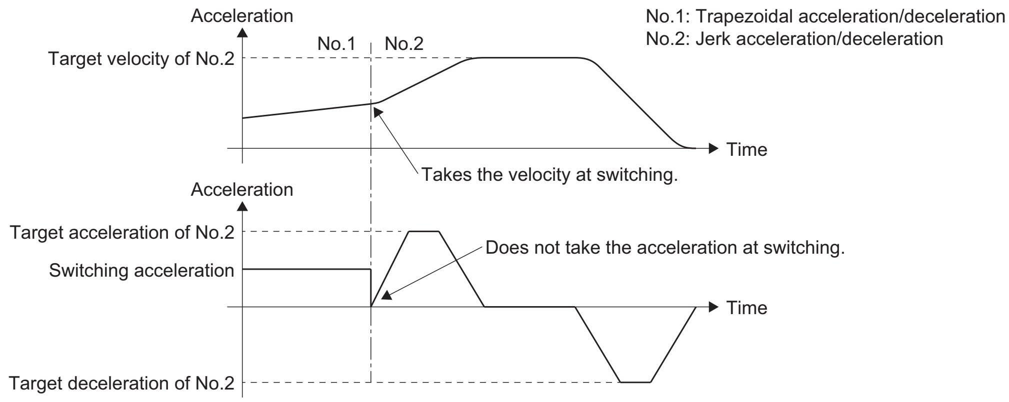
  Also, when switching during acceleration (deceleration), the on-going velocity waveform continues in case all settings of the target velocity, the acceleration, the deceleration, and the jerk are the same between the positioning data being executed and the next positioning data. If there is a different setting in any of those, the velocity change is performed to the target velocity of the next positioning data.



  ■Switching from other than "jerk acceleration/deceleration to another jerk acceleration/deceleration"

  In the following cases, the acceleration (deceleration) is not taken over when switching positioning data.

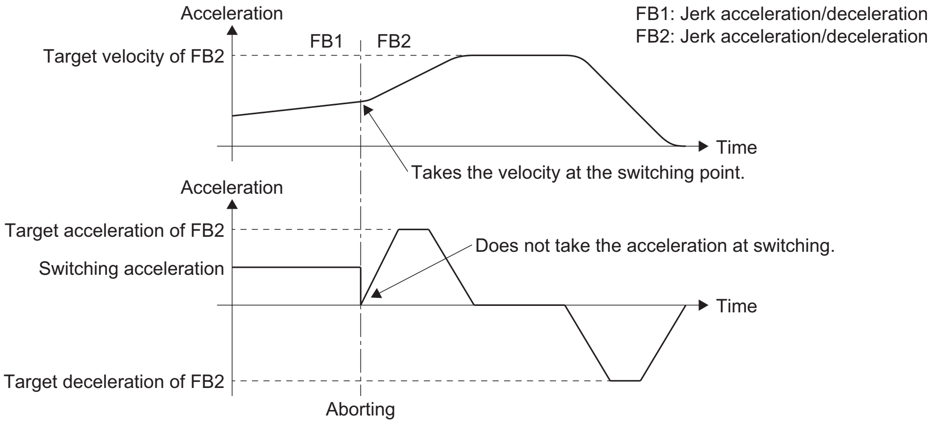
  - Switching from the jerk acceleration/deceleration to the trapezoidal acceleration/deceleration
  - Switching from the trapezoidal acceleration/deceleration to another trapezoidal acceleration/deceleration
  - Switching from the trapezoidal acceleration/deceleration to the jerk acceleration/deceleration



• Acceleration/deceleration method and multiple start

The specific acceleration/deceleration operation for MCv_MovePositioningData (Multiple Axes Positioning Data Operation) is shown below.

When MCv_MovePositioningData (Multiple Axes Positioning Data Operation) is included in either the under-control FB or the next FB, acceleration (deceleration) is not taken over at FB switching.



Even when switching from "jerk acceleration/deceleration to another jerk acceleration/deceleration", acceleration/deceleration may change rapidly if Aborting is executed during accelerating/decelerating of the under-control FB.

(Acceleration/deceleration is taken over only when multiple start occurs for the group force stop FB during the control of MCv_MovePositioningData (Multiple Axes Positioning Data Operation)).

To avoid rapid change in acceleration/deceleration, execute Aborting when the under-control FB is at a fixed speed.

■ Velocity limit

The basic operation is based on the specifications of the acceleration/deceleration processing function. For details, refer to the following.

☞ Page 345 Velocity Limit

• The velocity limit is executed for each positioning data. If the target velocity of a positioning data in the middle of the operation is exceeding the velocity limit value, it will be the same operation as at start. The timing when "Velocity Limit Value Over (error code: 1A07H)" is output is at start, or when pre-reading during control.

■ Override function

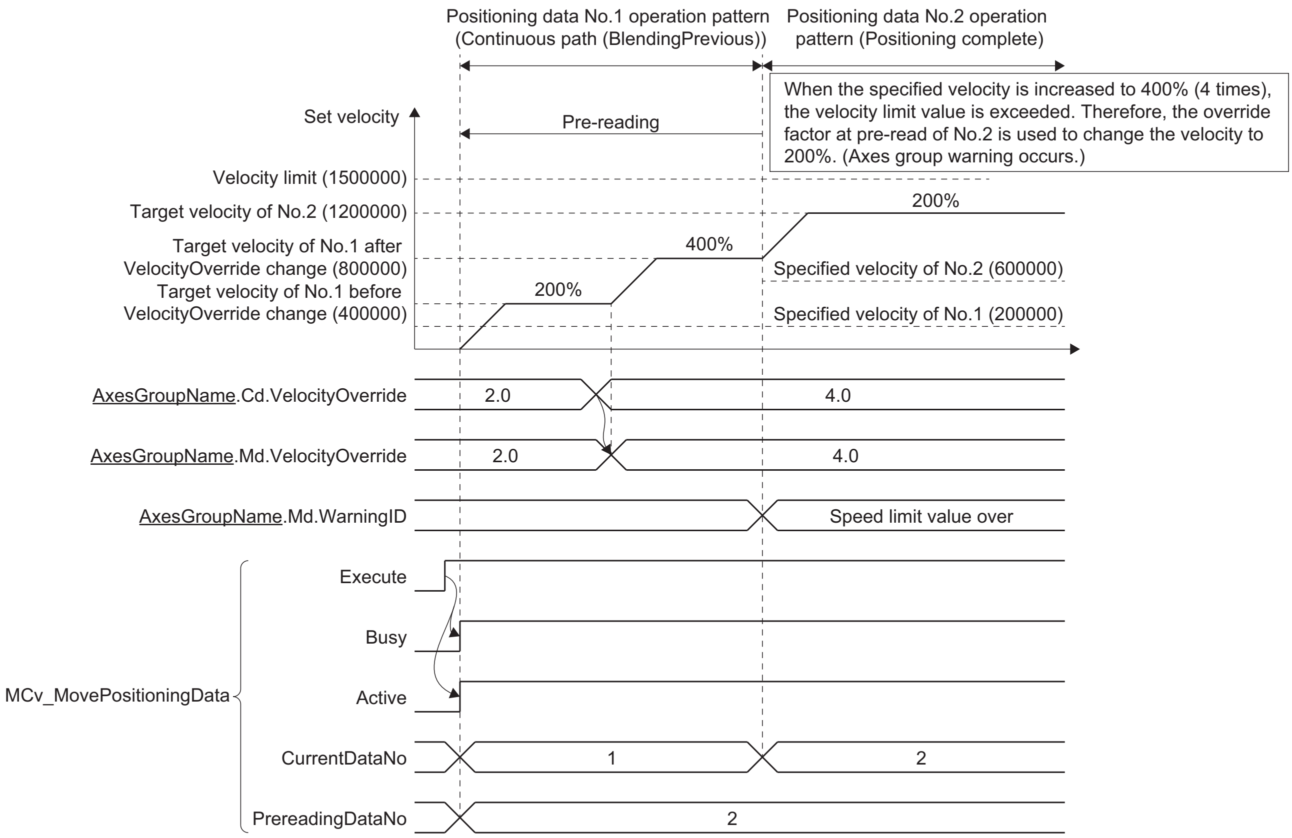
The basic operation is based on the specifications of the override function. For details, refer to the following.

☞ Page 349 Override Function

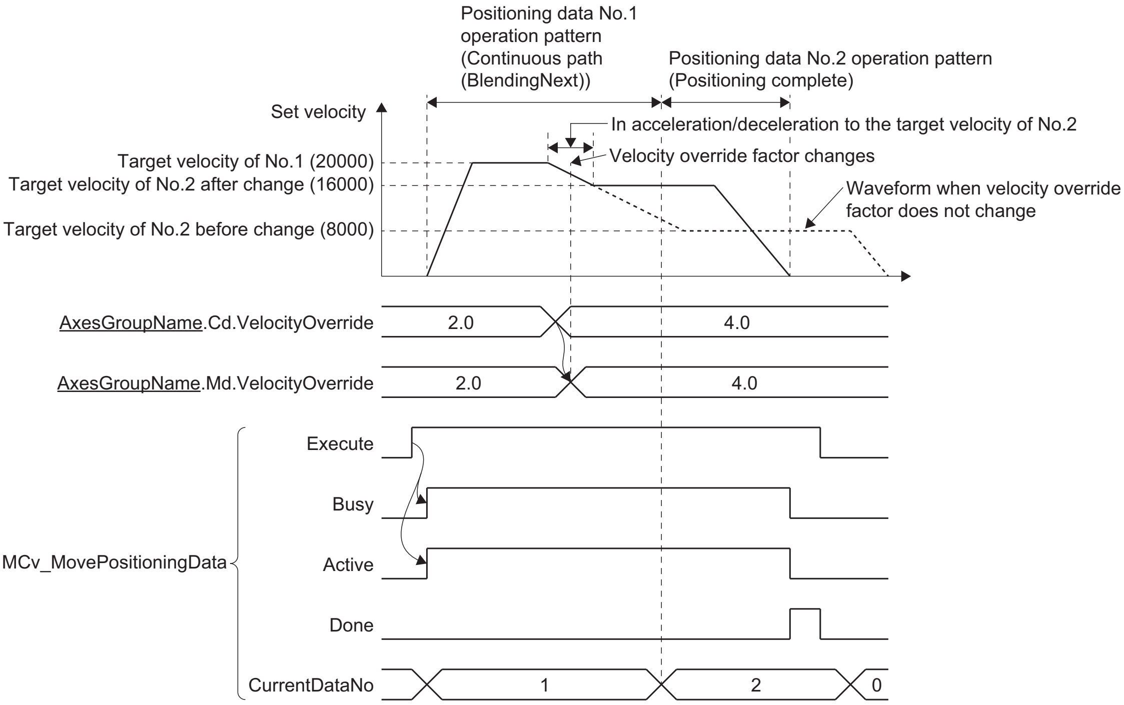
- Operation at velocity/acceleration/deceleration limit value over by override

  The operation when velocity/acceleration/deceleration limit value over occurs at positioning data switching is shown below.

  - The next positioning data control change is executed at switching.
  - When the control is changed at switching, velocity/acceleration/deceleration limit value over may occur. When velocity/acceleration/deceleration limit value over occurs, the override factor at pre-read is used to control.
  - The override factor is also imported when velocity/acceleration/deceleration limit value over occurs at switching. When the limit value is not exceeded at the next control change, the override factor after change will be used for operation.



• Target velocity change during acceleration/deceleration by "4: Continuous path (BlendingNext) (ContinuousBlendingNext)"
The operation when the target velocity has changed during acceleration/deceleration to the target velocity of the next positioning while executing positioning data with the operation pattern set to "4: Continuous path (BlendingNext) (ContinuousBlendingNext)" is shown below.



Condition signal setting

This section explains the setting items of Condition signal 1 to Condition signal 10 (ConditionSignal1 to ConditionSignal10) when used in Multiple Axes Positioning Data Operation.

• The condition signal is used for setting the execution conditions for JUMP in the positioning data in which the Control method (ControlMethod) is "0082H: JUMP (JUMP)". For details, refer to the following.
☞ Page 246 Control method (ControlMethod)

• The following shows the setting range and description of each item.

Setting item	Setting range	Description
I/O Number (StartIO)	—	Not used. The setting is ignored.
Target (Target)	• ""*1 ■Data type • [OBJ]*2 • [VAR] • [DEV] ■Type • BOOL type data *: For details on the character string format to be set, refer to the following. ☞ Page 385 TARGET_REF structure	Set the signal to be used.
Signal detection method (Detection)	0: Detection at TRUE (HighLevel) 1: Detection at FALSE (LowLevel)	Set the signal logic.
Compensation time (CompensationTime)	0.0	Set to "0.0".
Filter time (FilterTime)	0.0	Set to "0.0".

*1 When Target (Target) is "" (a character string of length 0), the corresponding condition signal is treated as not set.
*2 When [OBJ] is set as the data type, the target modification cannot be omitted. If the setting is omitted, "Operation Profile Data Control Profile ID Setting Incorrect (error code: 1AFAH)" will occur.

• When a value outside the range is set, "Operation Profile Data Control Profile ID Setting Incorrect (error code: 1AFAH)" will occur at FB execution.

Sub functions

Dwell function

The dwell function sets the time to wait until completing positioning data after positioning stops.

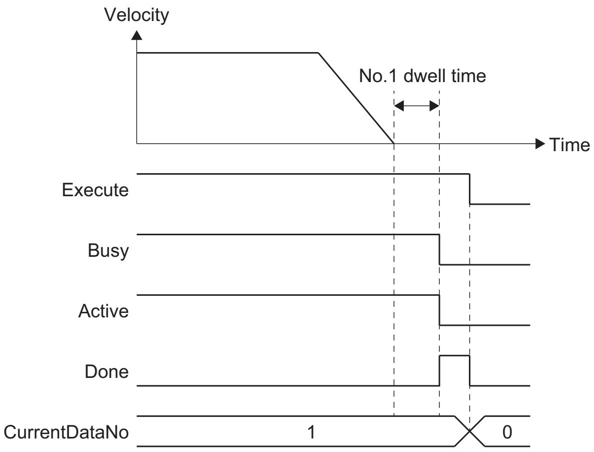
When dwell time is set to a value other than "0.0 [s]", the dwell function is enabled.

The dwell time is set to Dwell time (DwellTime) of the positioning data.

The operations depending on the operation pattern are shown below.

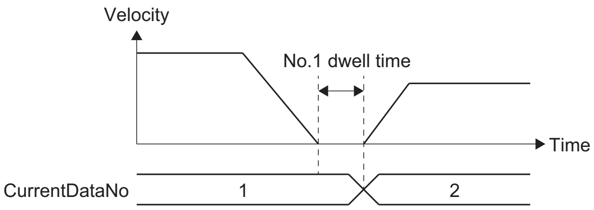
■ When the operation pattern is positioning complete

When the dwell time has passed after stopping positioning, the positioning data operation is completed.



■ When the operation pattern is continuous positioning

When the dwell time has passed after stopping positioning, the operation transitions to the positioning operation of the next positioning data.



■ When the operation pattern is continuous path

The operation is the same as when the operation pattern is "continuous positioning".

When not waiting for the dwell time to elapse or executing a continuous path operation, specify "0.0 [s] (disabled)" to the dwell time.

Precautions

- Setting the dwell time with the control cycle interval (the value of the control cycle multiplied by an integer) of the axes group configuration axes is recommended.
  When the dwell time is not the control cycle interval, the operation when waiting for the dwell time to pass depending on the dwell time setting value and length of one control cycle is shown in the table below.

Dwell time (DwellTime) setting value	Operation for waiting for dwell time to pass
Shorter than one control cycle	The operation waits for dwell time to pass only for one control cycle.*1
Equal to or longer than one control cycle	The operation waits for dwell time to pass only for a time scaled with the control cycle, and with any fractional parts truncated.

*1 To disable waiting for dwell time to pass, set Dwell time (DwellTime) to "0.0 [s]".

Ex.

For the following cases

Example	Description
1	■When the dwell time is shorter than one control cycle When the control cycle is "0.001 [s] (1.0 [ms])", if Dwell time (DwellTime) is set to "0.0005 (0.5 [ms])", the operation waits for dwell time to pass only for "0.001 (1.0 [ms])".
2	■When the dwell time is exceeds one control cycle When the control cycle is "0.001 [s] (1.0 [ms])", if Dwell time (DwellTime) is set to "0.0025 (2.5 [ms])", the operation waits for dwell time to pass only for "0.002 (2.0 [ms])".

M code output function

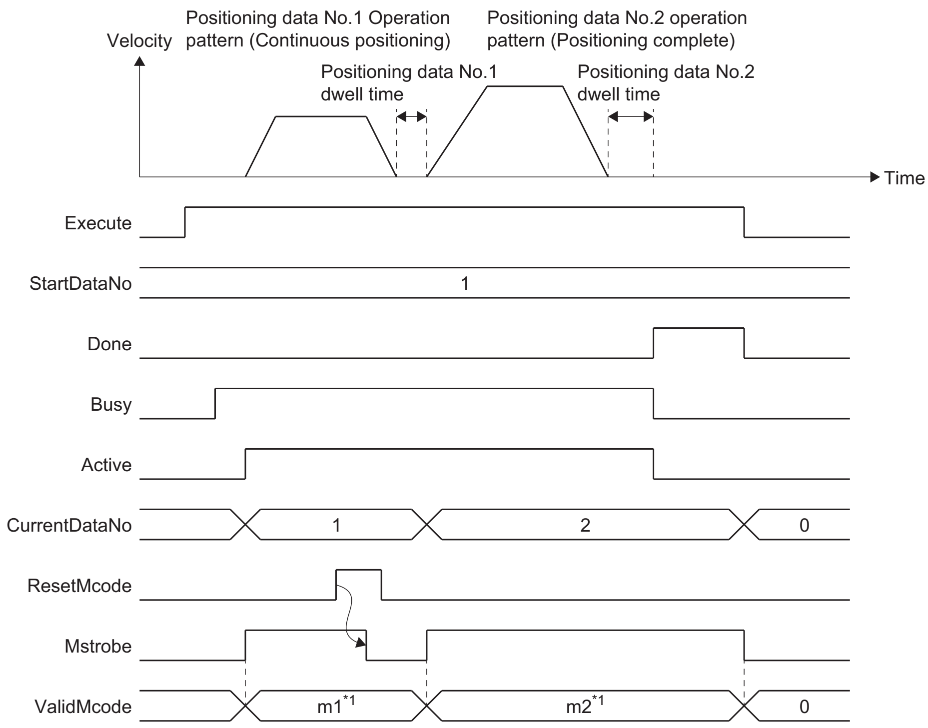
This function is used to command sub functions (such as clamping, drill rotation, and tool replacement) related to the positioning data being executed.

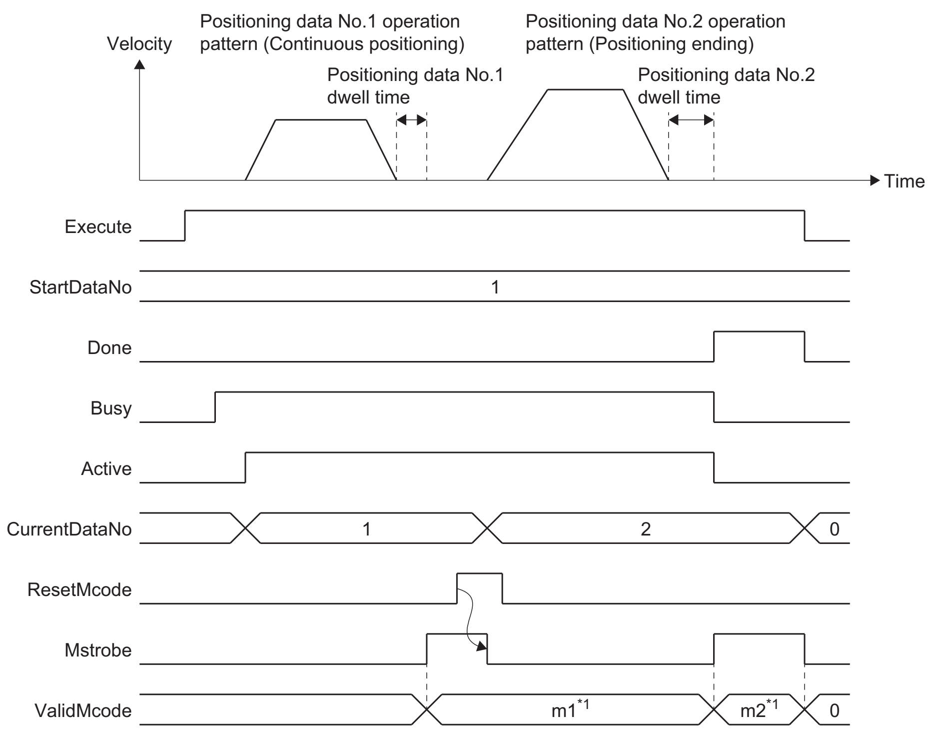
M code can be set to each positioning data. The timing of the M code output (storage) can also be set.

- This function is enabled when a value other than "0" is set to M code.
- M code is set in M code (Mcode) of the positioning data.
- When executing each positioning data of multiple axes positioning data operation, M strobe (Mstrobe) is changed to TRUE at the set output timing, and the M code is stored in Valid M code (ValidMcode).

■ M code output timing

- The output timing of M code is set in M code output timing (Options (Options): Bit 16) of MCv_MovePositioningData (Multiple Axes Positioning Data Operation). The output timing of M code for each positioning data can also be set with M code output timing override (McodeOutput_Override) of the positioning data.
- There is "WITH mode" and "AFTER mode" for the output timing of M code.

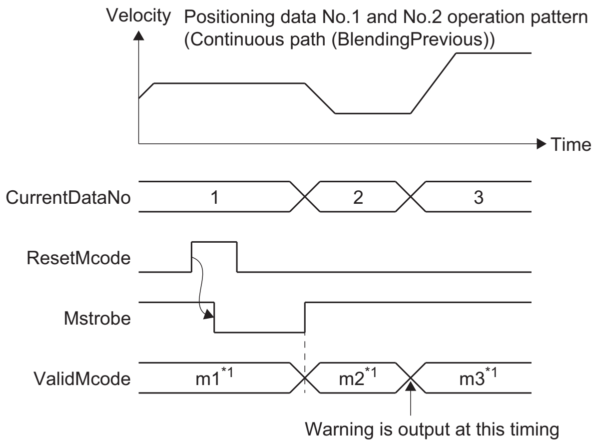
<table>
<tr><th>Output timing</th><th>Description</th></tr>
<tr><td>WITH mode</td><td>Changes M strobe (Mstrobe) to TRUE when positioning of the positioning data starts, and stores the positioning data M code (Mcode) in Valid M code (ValidMcode).


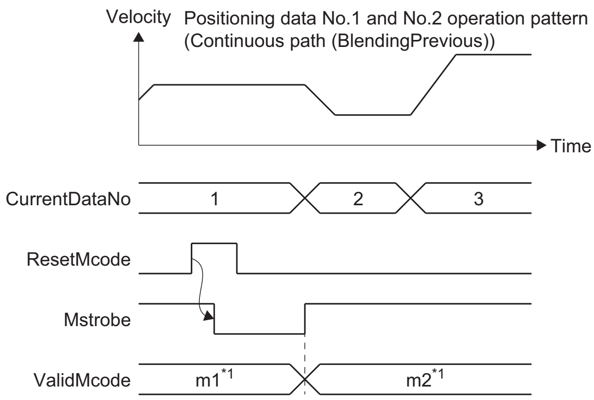
*1: m1 and m2 indicate the M code (Mcode) set to each positioning data No.</td></tr>
<tr><td>AFTER mode</td><td>Changes M strobe (Mstrobe) to TRUE when the target position is reached (after the setting time has passed when the dwell time has been set), and stores the positioning data M code in M code (Mcode).


*1: m1 and m2 indicate the M code (Mcode) set to each positioning data No.</td></tr>
</table>

■ M code reset wait function

- During M code output (while M strobe (Mstrobe) is TRUE), the operation waits to complete positioning data processing until M code reset is executed. While waiting, multiple axes positioning data operation control is continued.
- M code is reset when M code reset (ResetMcode) turns ON. M strobe (Mstrobe) becomes FALSE when M code is reset.
- The following are the operations depending on the operation pattern.

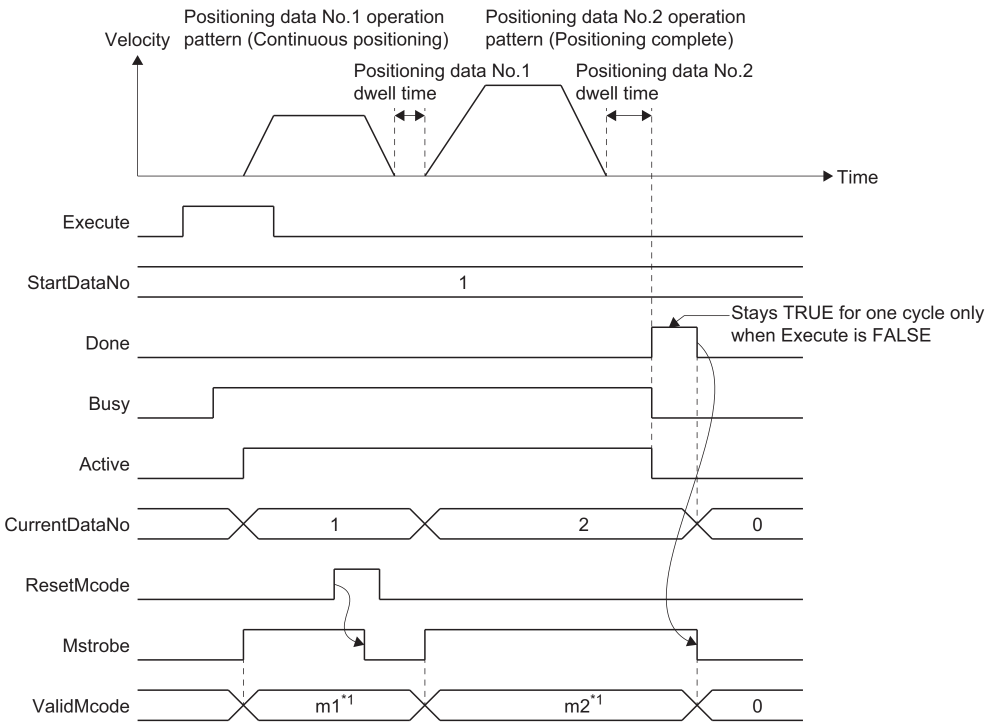
<table>
<tr><th>Operation pattern</th><th>Description</th></tr>
<tr><td>Positioning complete</td><td>The operation does not wait for M code reset.
The FB completes even when M strobe (Mstrobe) is TRUE.</td></tr>
<tr><td>Continuous positioning</td><td>While M strobe (Mstrobe) is TRUE, the operation waits to transit to the next positioning data.</td></tr>
<tr><td>Continuous path</td><td>The operation does not wait for M code reset.
The next positioning operation is executed even when M strobe (Mstrobe) is TRUE.
■When M code is set in the next positioning data
"Operation Profile Data Control Profile ID Setting Incorrect Warning (warning code: 0D48H)" occurs.


*1: m1, m2, and m3 indicate the M code (Mcode) set to each positioning data No.
■When M code is not set in the next positioning data
A warning is not output, but M code outputting continues.


*1: m1 and m2 indicate the M code (Mcode) set to each positioning data No. (The M code (Mcode) of positioning data No.3 is "0 (no setting)")</td></tr>
</table>

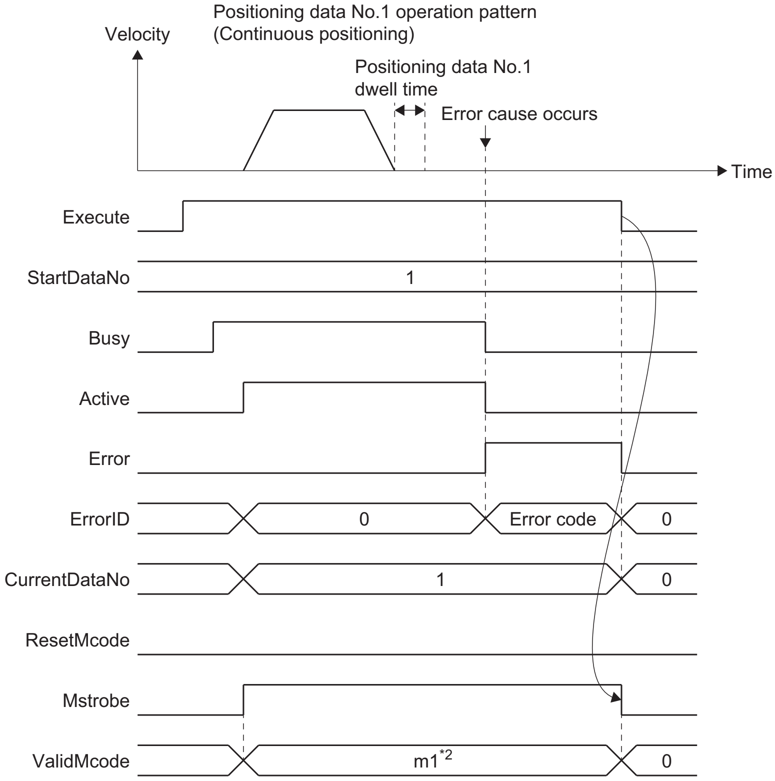
Precautions

- If M code (Mcode) is set to "0", M code is not output. (M strobe (Mstrobe) does not become TRUE at the M code output timing, and Valid M code (ValidMcode) is not updated.)
- M code is reset when M code reset (ResetMcode) turns ON. M code will not reset when M code reset (ResetMcode) is changed to TRUE in advance, before M strobe (Mstrobe) becomes TRUE. Perform mutual exclusion in the user program to prevent M code reset (ResetMcode) from turning ON while M strobe (Mstrobe) is FALSE.
- For continuous path, if positioning operation time is short, there may not be enough time to reset M code and "Operation Profile Data Control Profile ID Setting Incorrect Warning (warning code: 0D48H)" may occur. In this case, setting "0" to the M code (Mcode) of the positioning data for which the warning occurs allow the warning to be avoided by not outputting M code.
- Valid M code (ValidMcode) and M strobe (Mstrobe) are reset at completion when Execute command (Execute) is FALSE at completion. When Execute command (Execute) is TRUE at completion, they are reset when Execute command (Execute) changes to FALSE.



*1: m1 and m2 indicate the M code (Mcode) set to each positioning data No.

• If an error occurs or execution is canceled while waiting for M code reset, waiting for reset status will be canceled with M strobe (Mstrobe) remaining TRUE.



*2: m1 indicates the M code (Mcode) set to each positioning data No.

Precautions

- In MCv_MovePositioningData (Multiple Axes Positioning Data Operation), the time taken for start and multiple start (Aborting) may be extended due to operation profile data analysis such as pre-reading and external signal settings.
- When an error is detected during multiple axis positioning data operation control, "MCv_MovePositioningData Instruction Error (error code: 34B2H)" is output together with an error indicating the abnormality details. Detailed information such as the detected positioning data No. is recorded in the event history as the information for "MCv_MovePositioningData Instruction Error (error code: 34B2H)".

Required slave object

When using MCv_MovePositioningData (Multiple Axes Positioning Data Operation), set the following slave object to all of the configuration axes of the specified axes group.

- Target position (607AH)

When there is a configuration axis which does not have the above slave object set, "Necessary Slave Object Unset (error code: 1AA8H)" occurs and the axis does not start.

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

☞ Page 56 Axis Assignment

6

Relevant add-ons

The following add-ons are required to use this function.

- MotionEngine
- MotionControl_General
- ProfileControl
- SignalIO*1
- ServoDriver_CANopen*2

*1 For setting any of Condition signal 1 to Condition signal 10 (ConditionSignal1 to ConditionSignal10) of the operation profile data (multiple axes positioning data format)

*2 For setting the signal type [OBJ] in any of Condition signal 1 to Condition signal 10 (ConditionSignal1 to ConditionSignal10) of the operation profile data (multiple axes positioning data format)

Combination of the version

The versions of the engineering tool which are compatible with this function are as follows.

Function	Engineering tool	Version
Multiple Axes Positioning Data Operation (MCv_MovePositioningData) compatible	Motion control setting function	"1.042U" or later
Multiple Axes Positioning Data Operation (MCv_MovePositioningData) conditional JUMP compatible		"1.050C" or later

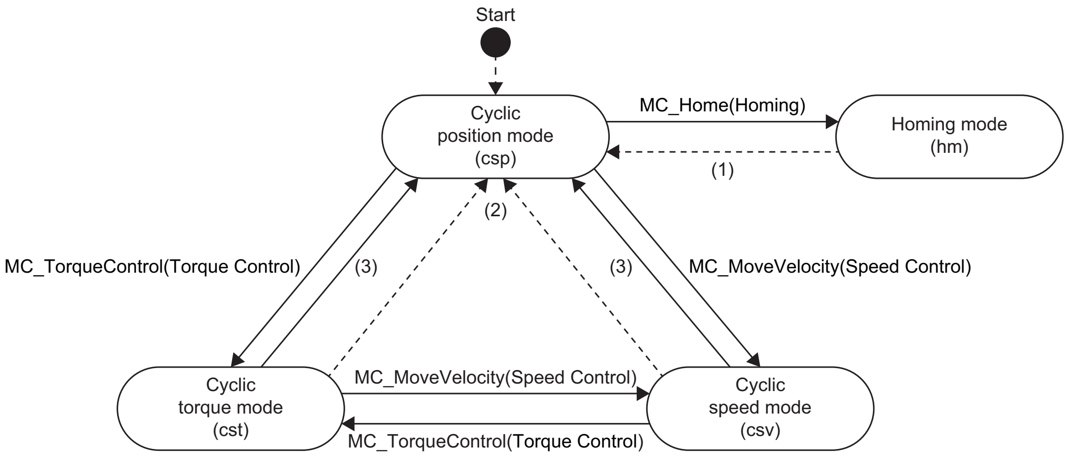
7 DIRECT CONTROL

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 (Speed Control (Including Position Loop))	This function outputs the commanded position based on the set velocity for each control cycle.
	Cyclic synchronous velocity mode (csv)	MC_MoveVelocity (Speed Control)	This function outputs the set velocity for each control cycle. (Excluding the position loop.)
Torque control	Cyclic synchronous torque mode (cst)	MC_TorqueControl (Torque Control)	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 Motion control FB is executed. The following shows the state transition.



No.	State transition description
(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 a Motion control FB other than MC_MoveVelocity (Speed Control) and MC_TorqueControl (Torque Control).

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 851 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

Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
SetVelocity	Set Velocity	Stores the velocity calculated from the difference of the set position updated by the follow up.
SetAcceleration	Set Acceleration	Stores a value calculated from the difference of the set velocity.
TargetVelocity	Target Velocity	Stores the actual set velocity whose override and the velocity limit value are considered to Velocity (Velocity).
ActualVelocity	Actual Velocity	Stores the actual velocity (same as positioning control).
Io_TargetVelocity	Object Data_TargetVelocity	Displays the value of object data TargetVelocity. (The velocity command to send to device stations)
Io_VelActualValue	Object Data_VelActualValue	Displays the value of object data VelActualValue. (Feedback of the velocity which is received from device stations)

Relevant FB

For details on Motion control FBs, refer to the following.

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

Motion control FB	Name	Description
MC_MoveVelocity	Speed control	The driver is switched to the csv, and the velocity control is executed based on the specified velocity.

Control details

MC_MoveVelocity (Speed Control) 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 (Acceleration), Deceleration (Deceleration), and Jerk (Jerk). To finish this FB, start MC_Stop (Forced Stop).

Velocity initial value selection

Set the speed initial value when switching control mode from csp (cyclic synchronous position mode) to csv (cyclic synchronous velocity mode) to Velocity initial value selection (Options (Options): Bit 16, 17).

- The csp (cyclic synchronous position mode)→The csv (cyclic synchronous velocity mode)

Setting value	Description
0: Set velocity	The velocity during command (Set Velocity (AxisName.Md.SetVelocity)) is used as the velocity commanded to the drive unit immediately after switching.
1: Actual velocity	Motor speed (Object Data_VelActualValue (AxisName.Md.Io_VelActualValue)) received from the drive unit at switching is used.*1
2: Automatic selection	The lower velocity between "0: Set velocity" and "1: Actual velocity" is used as the velocity commanded to the drive unit immediately after switching.*2

*1 When slave object "Velocity actual value (606CH)" is not mapped, motor speed received from the drive unit at switching is not used. (Speed initial value selection becomes "0.0".)

*2 When slave object "Velocity actual value (606CH)" is not mapped, velocity commanded to the drive unit immediately after switching becomes "0: Set velocity".

7

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: Stopping on error (ErrorStop)	Immediate stop*1
2	Drive unit power supply is FALSE Drive unit network disconnection detection Drive unit Forced stop input to the drive unit Enable (Enable) of MC_Power (Operation Available) is FALSE Servo ON request (ServoON) of MC_Power (Operation Available) is FALSE (When Process Selection at Servo OFF Command During Operation (AxisName.Pr.StopMode_ServoOff) is "4: Servo OFF After Immediate Stop (ServoOffAfterImmediateStop)")*2	—	1: Stopping on error (ErrorStop)	Immediate stop*1
3	Hardware stroke limit upper/lower limit error occurrence	1: Stopping on error (ErrorStop)	1: Stopping on error (ErrorStop)	Deceleration stop/Immediate stop (Follows Stop Selection at Hardware Stroke Limit Error Occurrence (AxisName.Pr.StopMode_HwStrokeLimit)) After stopping, switch to the csp.
4	CPU module error occurrence PLC READY is OFF Moderate error or major error of the motion system occurs Cycle over error occurrence	1: Stopping on error (ErrorStop)	1: Stopping on error (ErrorStop)	Deceleration stop/Immediate stop (Follows Stop Selection at All Axes Stop Cause Occurrence (System.Pr.StopMode_All)) After stopping, switch to the csp.
5	Software stroke limit upper/lower limit error occurrence	1: Stopping on error (ErrorStop)	1: Stopping on error (ErrorStop)	Deceleration stop/Immediate stop (Follows Stop Selection at Software Stroke Limit Error Occurrence (AxisName.Pr.StopMode_SwStrokeLimit)) After stopping, switch to the csp.
6	Axis error detection*3	1: Stopping on error (ErrorStop)	1: Stopping on error (ErrorStop)	Deceleration stop/Immediate stop (Follows Stop Selection at Stop Cause Occurrence (AxisName.Pr.StopMode_General)) After stopping, switch to the csp.
7	Execute command (Execute) of MC_Stop (Forced Stop) is TRUE	2: Decelerating to stop (Stopping)	2: Decelerating to stop (Stopping)*4	Deceleration stop (Follows the deceleration speed set in the FB) After stopping, switch to the csp.
8	"Stop signal (STOP)" of external input signal is TRUE	No change	4: Standby (Standstill)	Deceleration stop/Immediate stop (Follows Stop Selection at Stop Cause Occurrence (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 Process Selection at Servo OFF Command During Operation (AxisName.Pr.StopMode_ServoOff). For details, refer to the following.
☞ Page 162 Relevant variables

*3 If an axis error (the error that transits the state into the "1: Stopping on error (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 command (Execute) of MC_Stop (Forced Stop) is FALSE at stop completion, Axis Status (AxisName.Md.AxisStatus) after stop will be "4: Standby (Standstill)".

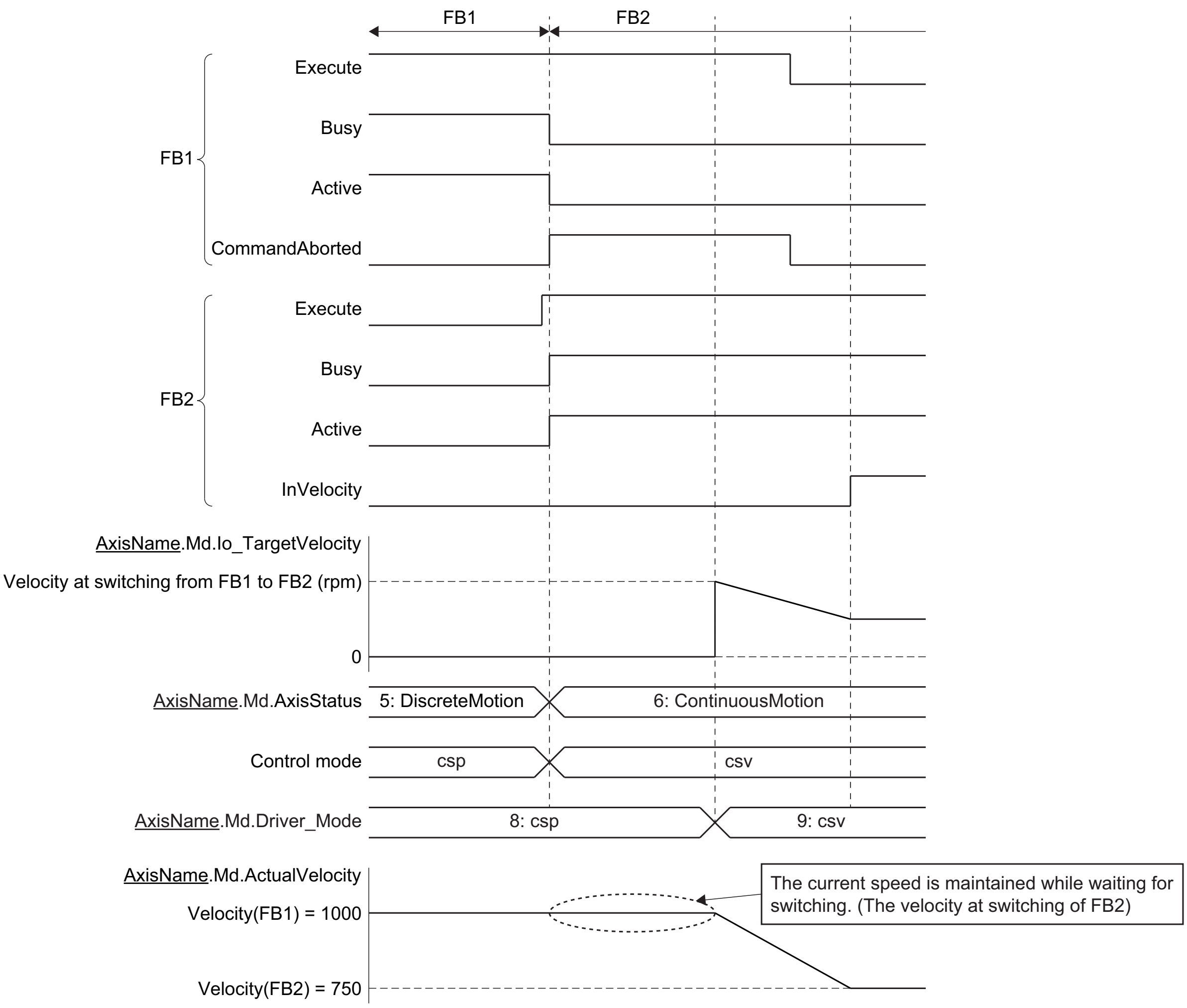
Buffer mode (BufferMode)

When starting this FB during another instruction execution

■ Single axis position control (FB1) → Velocity control (FB2)

- 0: Aborting (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, Target Velocity (TargetVelocity) is obtained based on the actual velocity and the acceleration/deceleration process is started again.

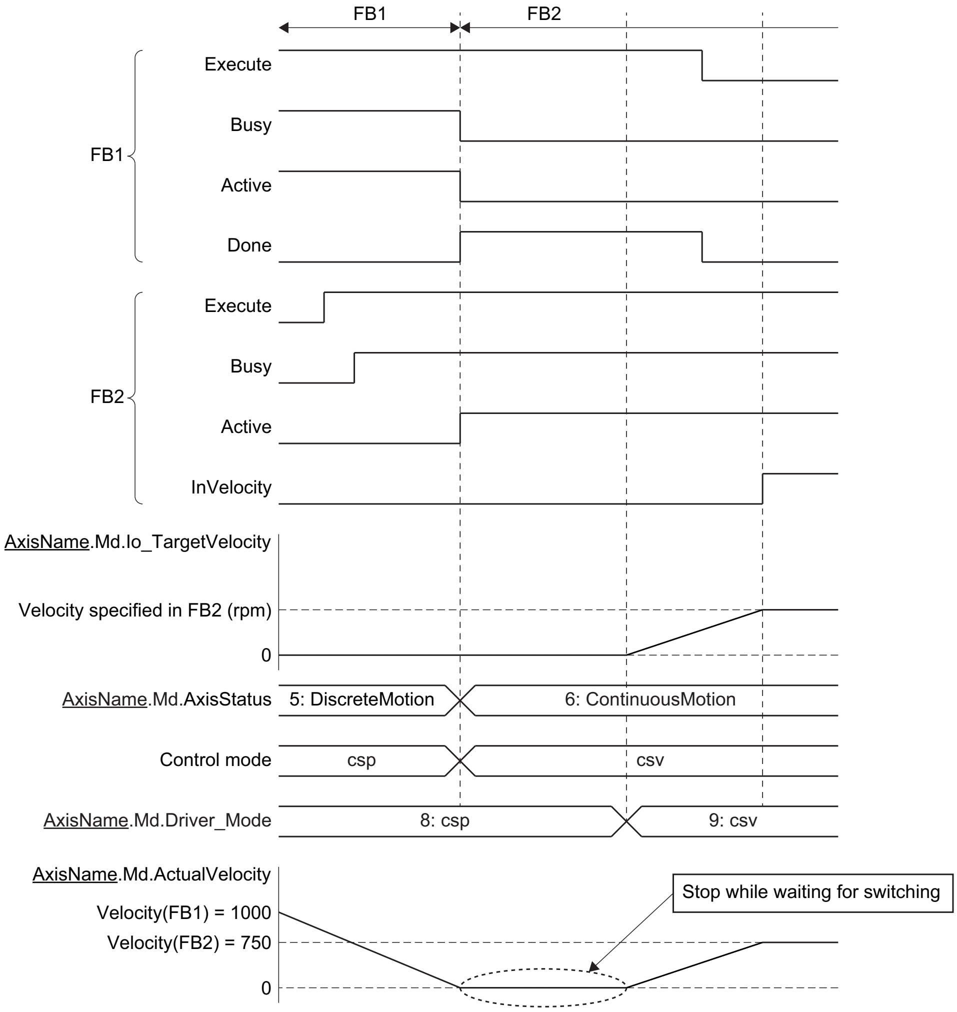


• 1: Buffered (mcBuffered)

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 device specifications. If switching to csp is not completed within 1 [s], "Control Mode Switching Error (error code: 1A1DH)" occurs for the timeout, and the axis stops.)

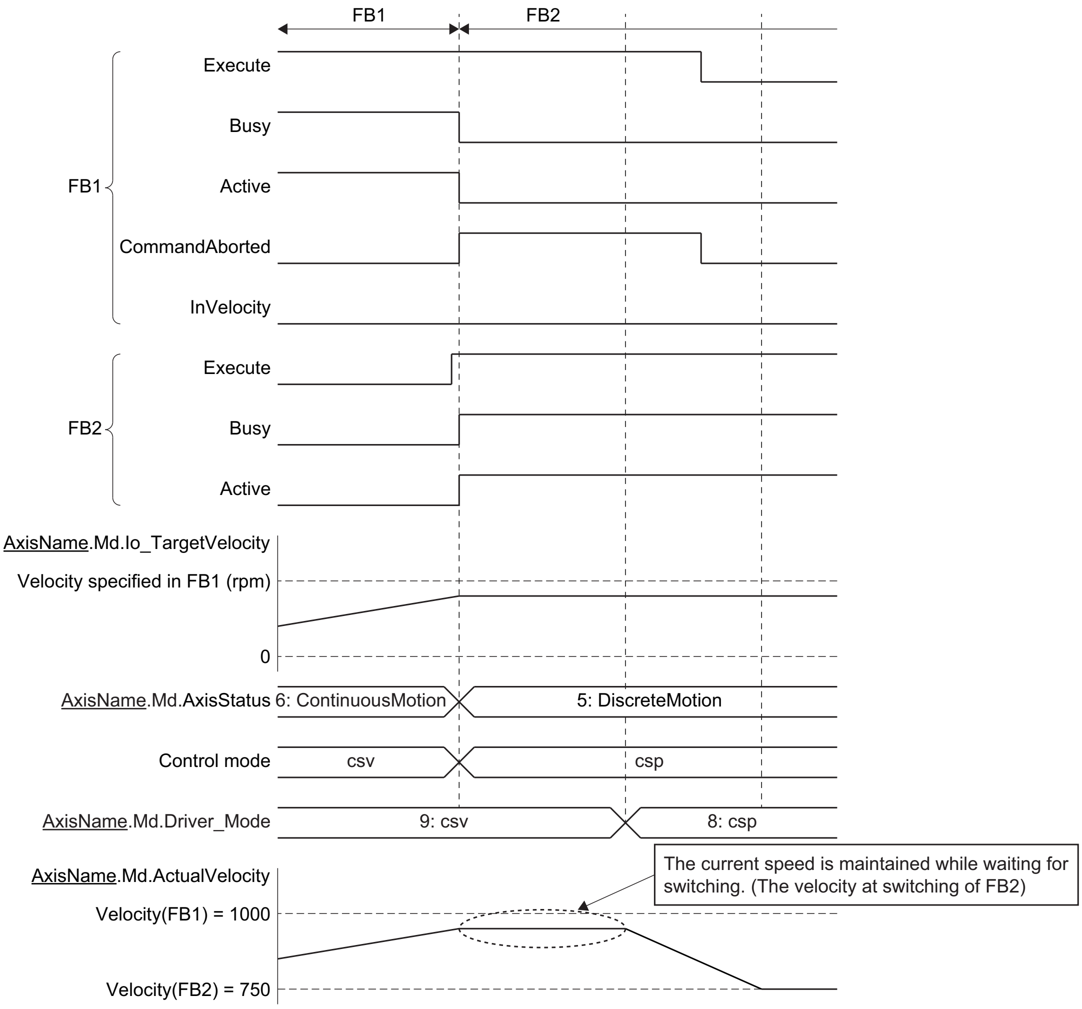
3. While waiting, velocity control is continued with the set velocity when Execute command (Execute) of FB2 is TRUE.

4. When the driver switches to csp, the positioning control starts.

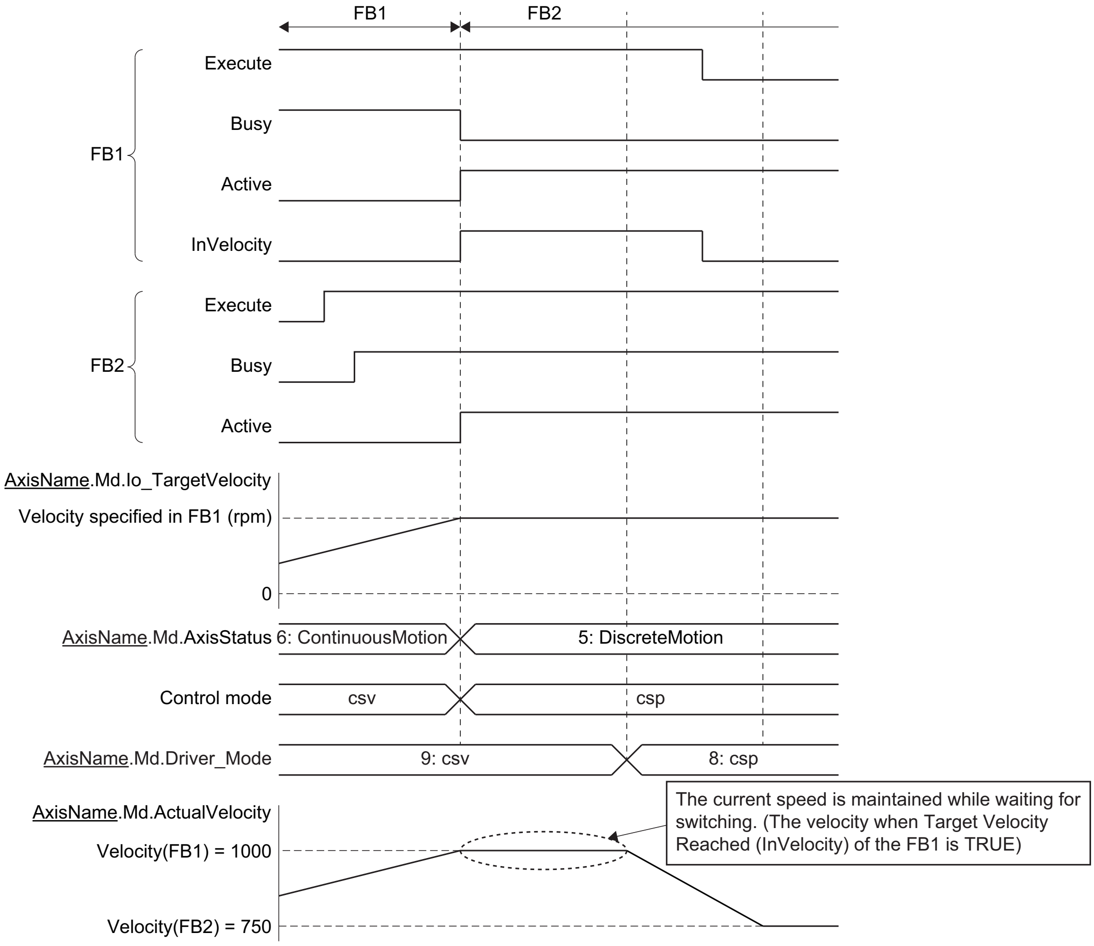
Point

- 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 Operation Setting at Overrun (AxisName.Pr.OverrunOperation).
☞ Page 132 Multiple Start (Buffer Mode)
- For operation at the overrun occurrence, refer to the following.

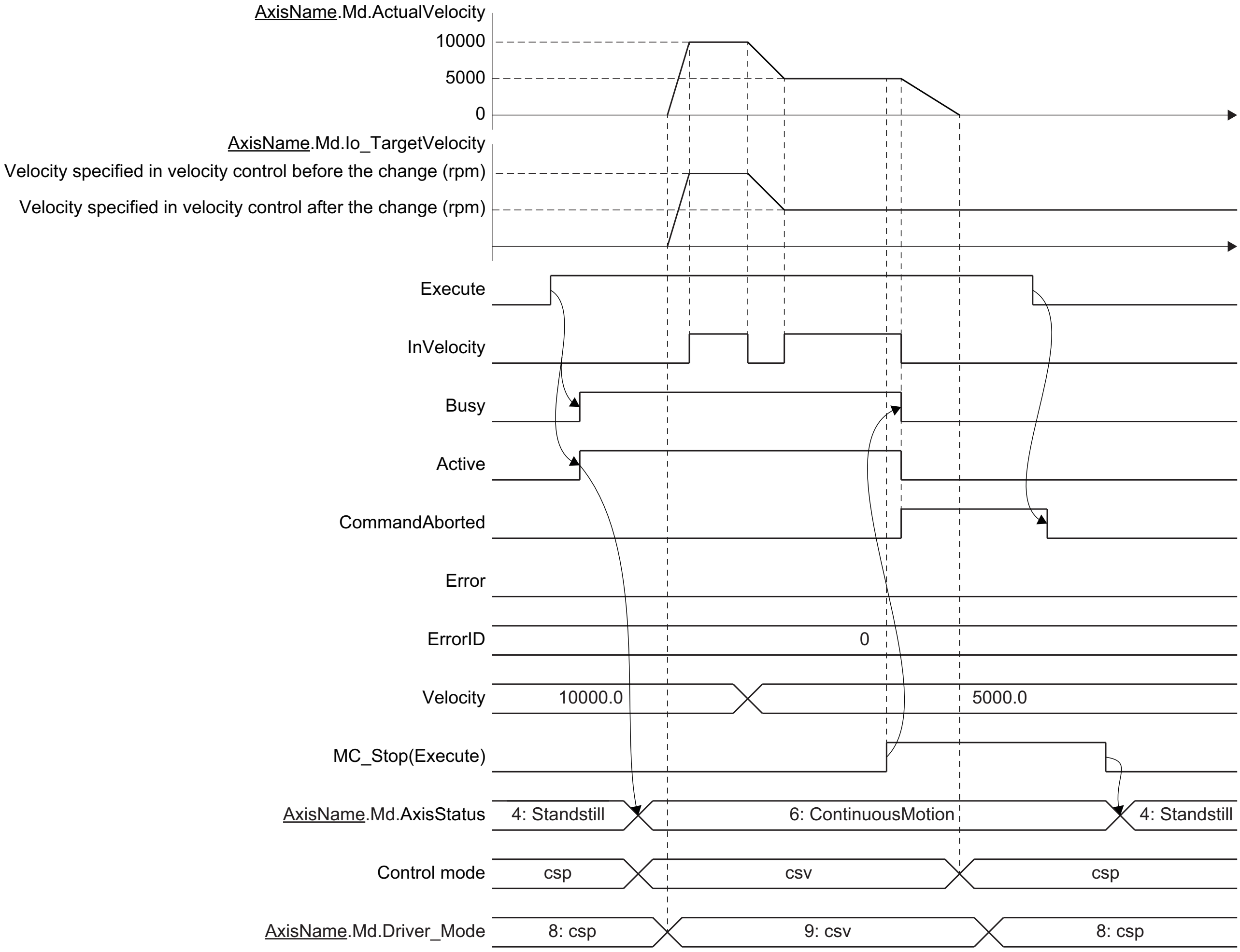
- 0: Aborting (mcAborting)



- 1: Buffered (mcBuffered)



■ Velocity control → Stop instruction (MC_Stop (Forced Stop))

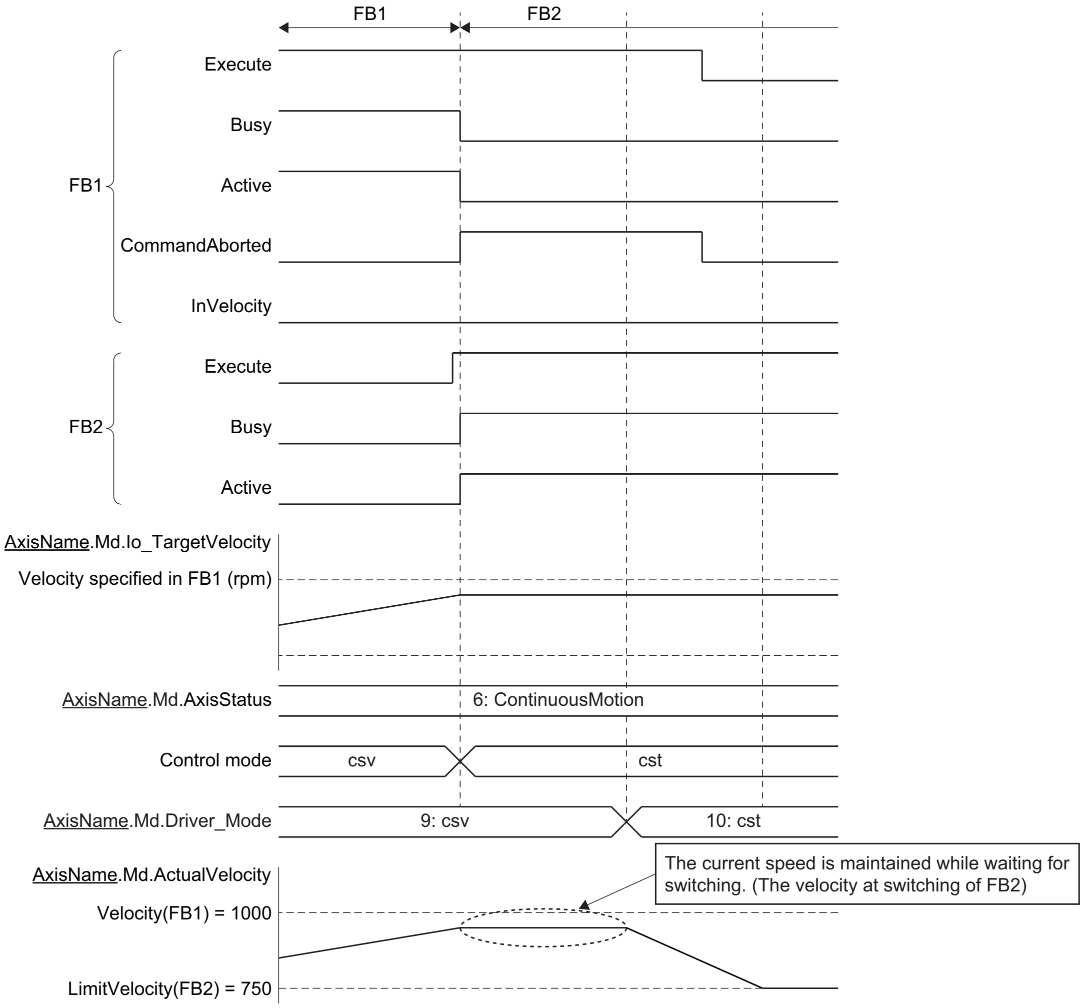


■ Velocity control → Torque control

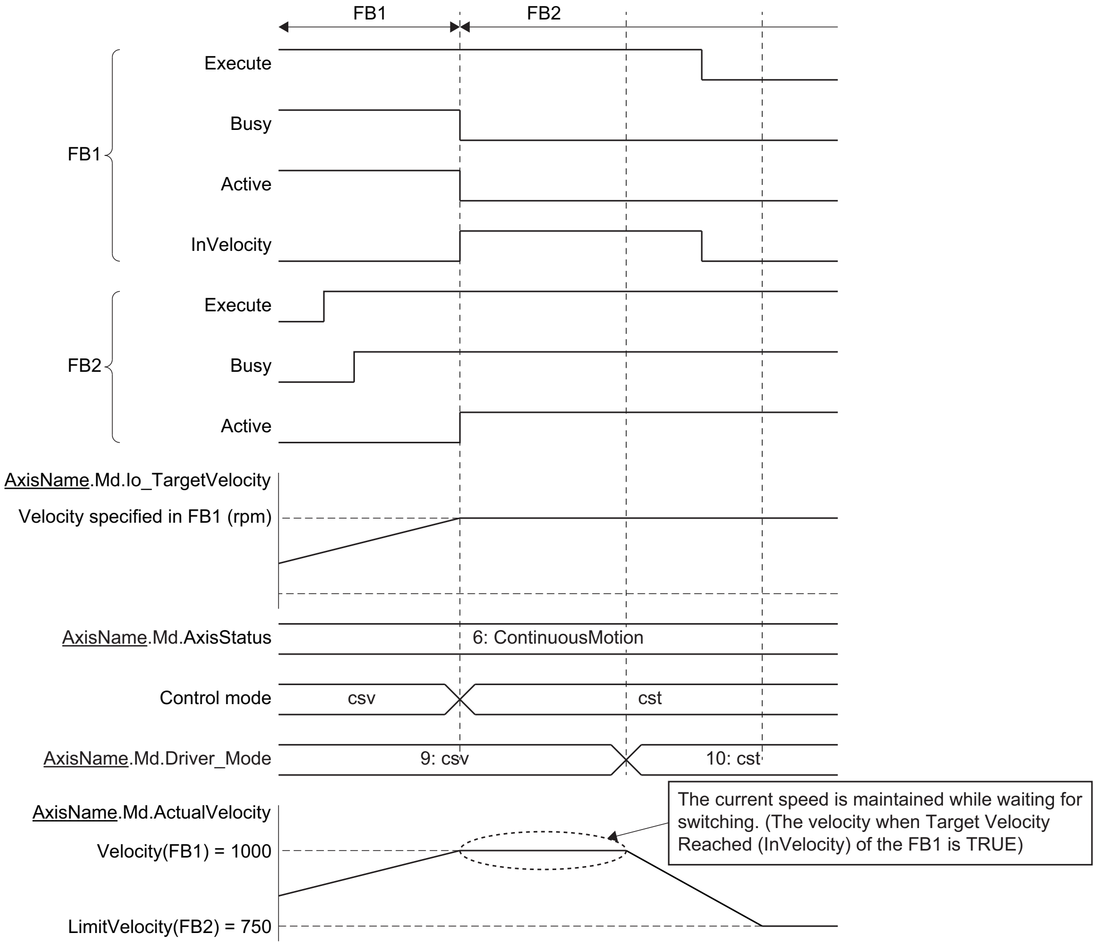
1. Switching request to the cst is executed to the driver.

2. Started torque control is analyzed, and the axis waits until the driver switches to the cst.

- 0: Aborting (mcAborting)



- 1: Buffered (mcBuffered)



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

- Velocity Override Factor (AxisName.Cd.VelocityOverride) and Acceleration Override Factor (AxisName.Cd.AccelerationOverride) are valid.
- The set position and the machine feed value are updated by follow up.
- Switching time to the control mode depends on the specifications of the driver device.
- 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 Driver Control Mode (AxisName.Md.Driver_Mode) is switched to "9: csv".
- When switching from the csp to the csv or from the csv to the csp without waiting for stop of the motor by using the MR-J5(W)-G, 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 834 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

Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	details
SetVelocity	Set Velocity	Stores the velocity calculated from the difference of the set position updated by the follow up.
SetAcceleration	Set Acceleration	Stores a value calculated from the difference of the set velocity.
TargetVelocity	Target Velocity	Stores value set to the limit velocity (LimitVelocity)
ActualVelocity	Actual Velocity	Stores the actual velocity (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_TorqueActualValue	Object Data_TorqueActualValue	Displays the value of object data TorqueActualValue. (The actual torque which is received from device stations)
Io_VelActualValue	Object Data_VelActualValue	Displays the value of object data VelActualValue. (Feedback of the velocity which is received from device stations)

Relevant FB

For details on Motion control FBs, refer to the following.

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

Motion control FB	Name	Description
MC_TorqueControl	Torque control	The driver is switched to the cst, and the torque control is executed based on the specified target torque.

Control details

MC_TorqueControl (Torque Control) 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 Torque positive direction ramp (TorquePositiveRamp) and Torque negative ramp (TorqueNegativeRamp). To finish this FB, start MC_Stop (Forced 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)".

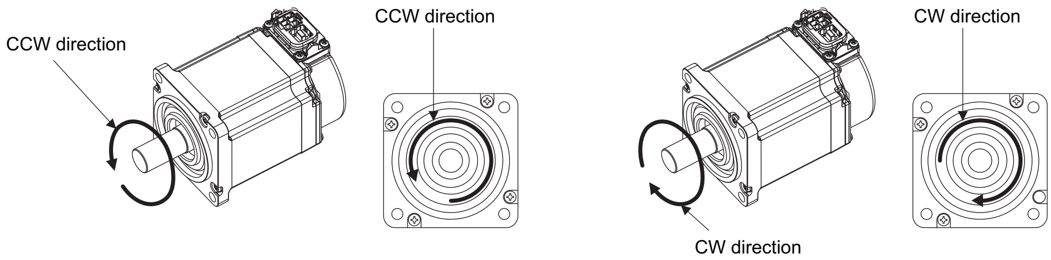
- When the servo parameter (extension setting) "Torque POL reflection selection (PC29.3)" is set to "0: Enabled"

Setting value of "Travel direction selection (PA14)"	Set torque	Torque generation direction of servomotor
0: Positioning address increasing CCW rotation	Positive value (positive direction)	CCW direction[*1]
	Negative value (negative direction)	CW direction[*1]
1: Positioning address increasing CW rotation	Positive value (positive direction)	CW direction[*1]
	Negative value (negative direction	CCW direction[*1]

- When the servo parameter (extension setting) "Torque POL reflection selection (PC29.3)" is set to "1: Disabled" (Initial value)

Setting value of "Travel direction selection (PA14)"	Set torque	Torque generation direction of servomotor
0: Positioning address increasing CCW rotation	Positive value (positive direction)	CCW direction*1
	Negative value (negative direction)	CW direction*1
1: Positioning address increasing CW rotation	Positive value (positive direction)	CCW direction*1
	Negative value (negative direction)	CW direction*1

*1 For details, refer to the following illustration.



Torque ramp function selection (Options (Options): Bit 16, 17)

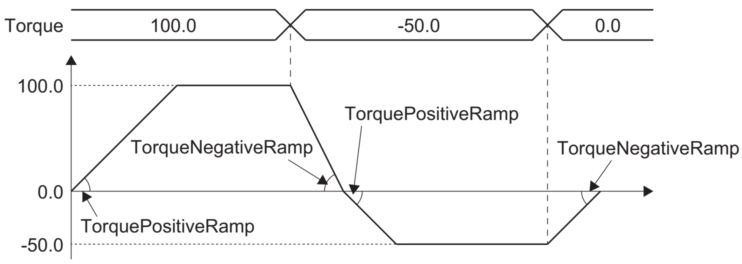
Set the method from set torque until the target torque is reached.

Torque positive direction ramp (TorquePositiveRamp) and Torque negative ramp (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.

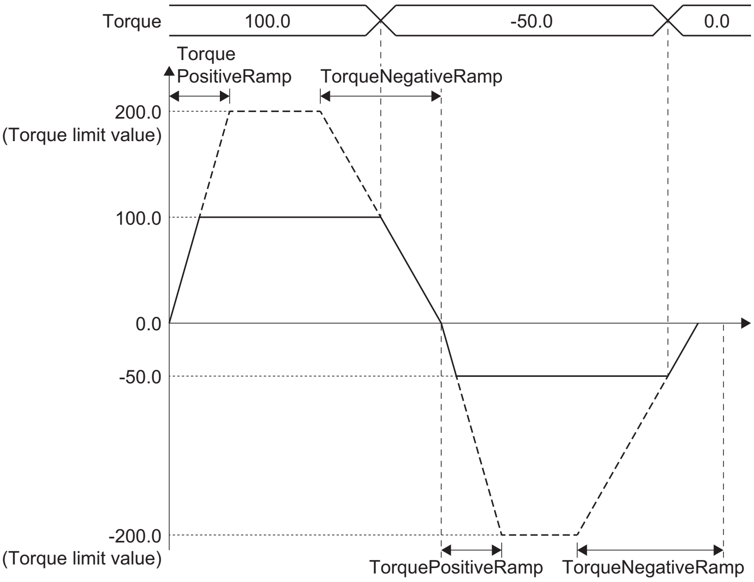
Setting range
0.0 to 1000.0 [%/s]



■ For "1: Time constant method"

Set the time until the set torque reaches the positive/negative torque limit value from "0".

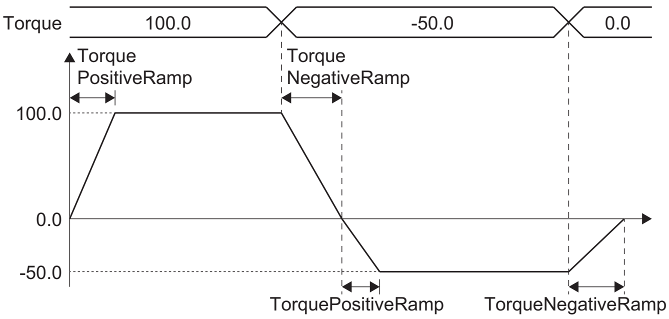
Setting range
0.0 to 8400.0 [s]



■ For "2: Fixed time method"

Set the time from the current set torque until the target torque is reached.

Setting range
0.0 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 0.0, the value reaches the target torque in 1 operation cycle.

Torque initial value selection (Options (Options): Bit 18)

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" to this bit. Set "1: Actual torque" only when switching the control mode immediately after the command to the motor is completed without waiting for the servo motor to stop.

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) After stop	Stop process
1	"Forced stop input" is FALSE	1: Stopping on error (ErrorStop)	Immediate stop*1
2	Drive unit power supply is FALSE	1: Stopping on error (ErrorStop)	Immediate stop*1
	Drive unit network disconnection detection		
	Drive unit		
	Forced stop input to the drive unit		
	Enable (Enable) of MC_Power (Operation Available) is FALSE		
	Servo ON request (ServoON) of MC_Power (Operation Available) is FALSE (When Process Selection at Servo OFF Command During Operation (AxisName.Pr.StopMode_ServoOff) is "4: Servo OFF After Immediate Stop (ServoOffAfterImmediateStop)", "5: Servo OFF After Deceleration to Stop (ServoOffAfterDecelStop)")*2		
3	Hardware stroke limit upper/lower limit error occurrence	1: Stopping on error (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: Stopping on error (ErrorStop)	
	PLC READY is OFF		
	Moderate error or major error of the motion system occurs		
	Cycle over error occurrence		
5	Software stroke limit upper/lower limit error occurrence	1: Stopping on error (ErrorStop)	
6	Axis error detection*3	1: Stopping on error (ErrorStop)	
7	Execute command (Execute) of MC_Stop (Forced Stop) is TRUE	2: Decelerating to stop (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 (Forced Stop))
8	"Stop signal (STOP)" of external input signal is TRUE	4: Standby (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 Process Selection at Servo OFF Command During Operation (AxisName.Pr.StopMode_ServoOff). For details, refer to the following.
☞ Page 162 Relevant variables
*3 If an axis error (the error that transits the state into the "1: Stopping on error (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 command (Execute)" of MC_Stop (Forced Stop) is FALSE at stop completion, Axis Status (AxisName.Md.AxisStatus) after stop will be "4: Standby (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.

Buffer mode (BufferMode)

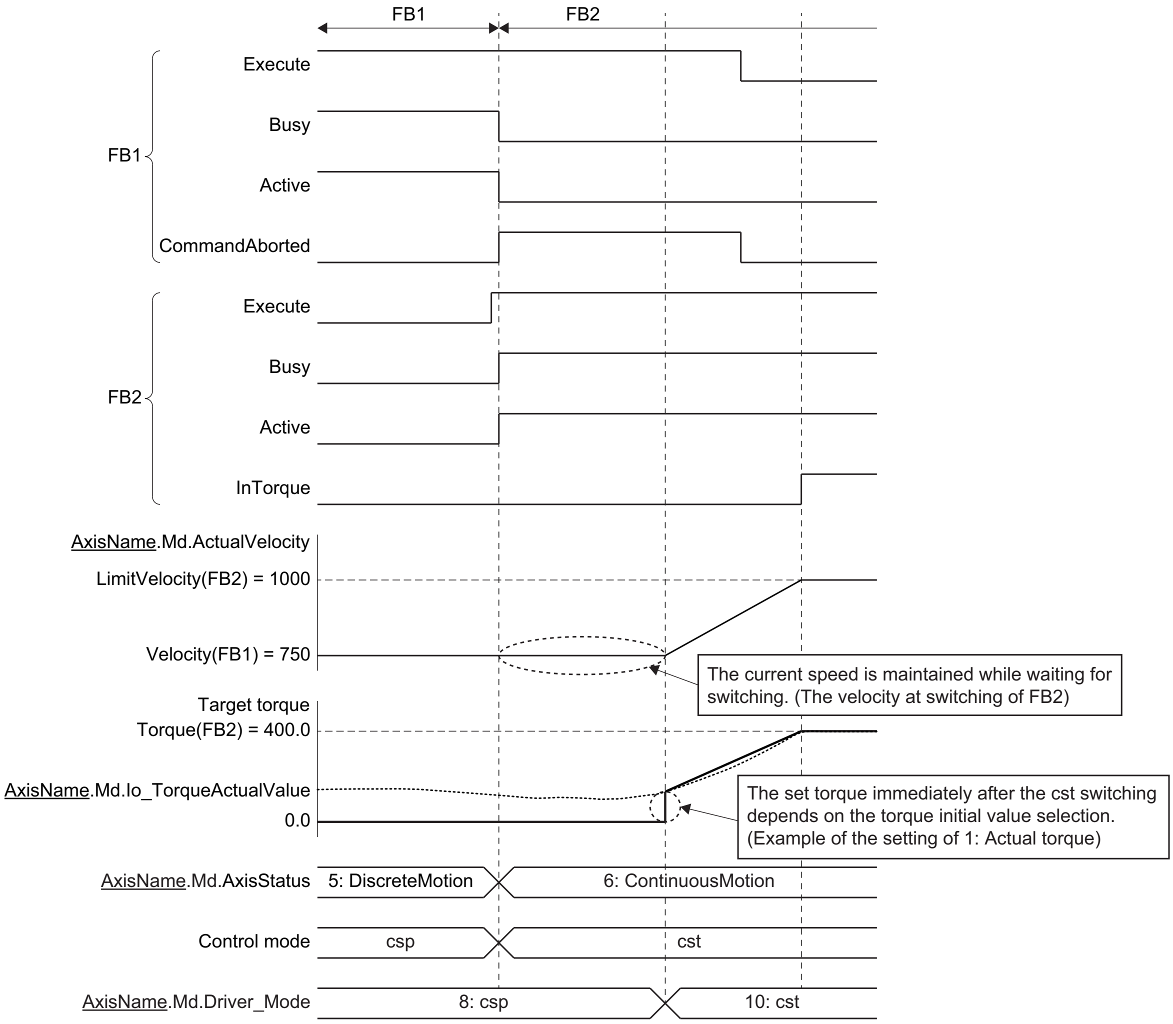
When starting this FB during another instruction execution

While this FB is executed, only Aborting and Buffered are supported.

■ Position control (FB1) → Torque control (FB2)

- 0: Aborting (mcAborting)

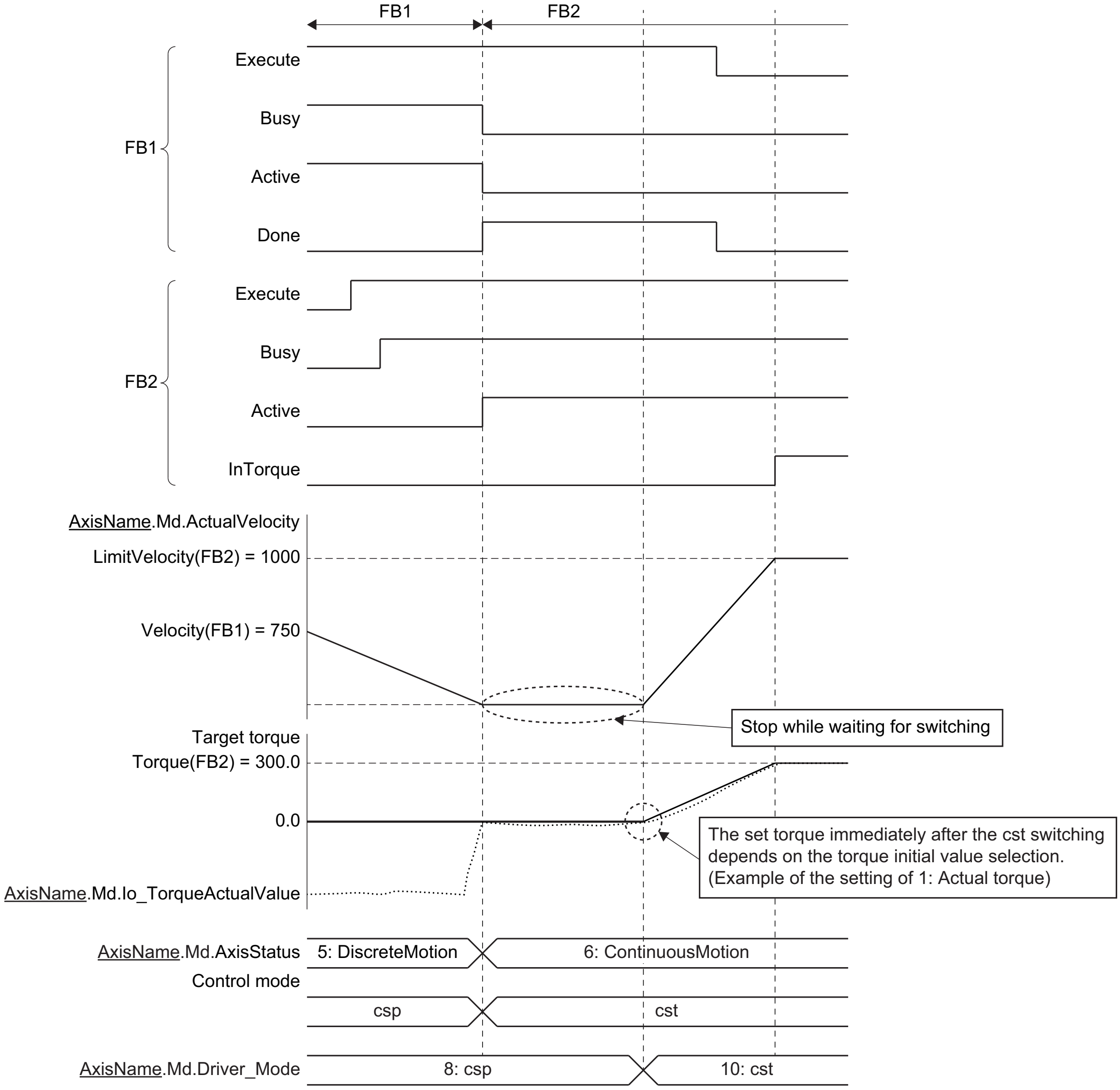
1. A switching request to cst is issued to the driver.
2. Until the driver is switched, the axis is operated at the previous velocity with the velocity control including the position loop.
3. When the driver switches to csv, the torque increase/decrease from the current actual torque starts and the acceleration/deceleration process of velocity limit value from the current set velocity starts.



- 1: Buffered (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.



7

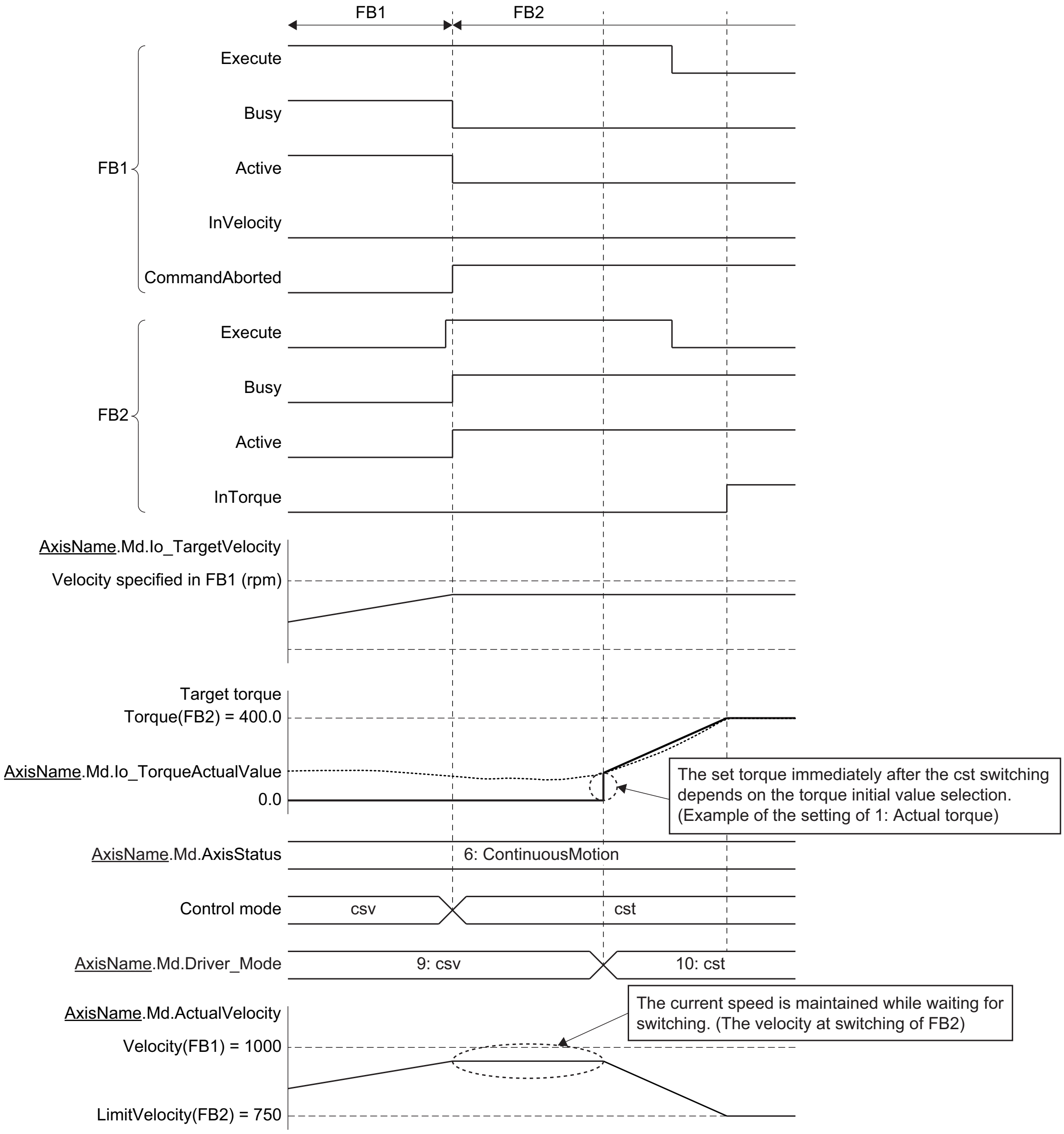
■ Velocity control (FB1) → Torque control (FB2)

1. A cst change request is issued to the driver.

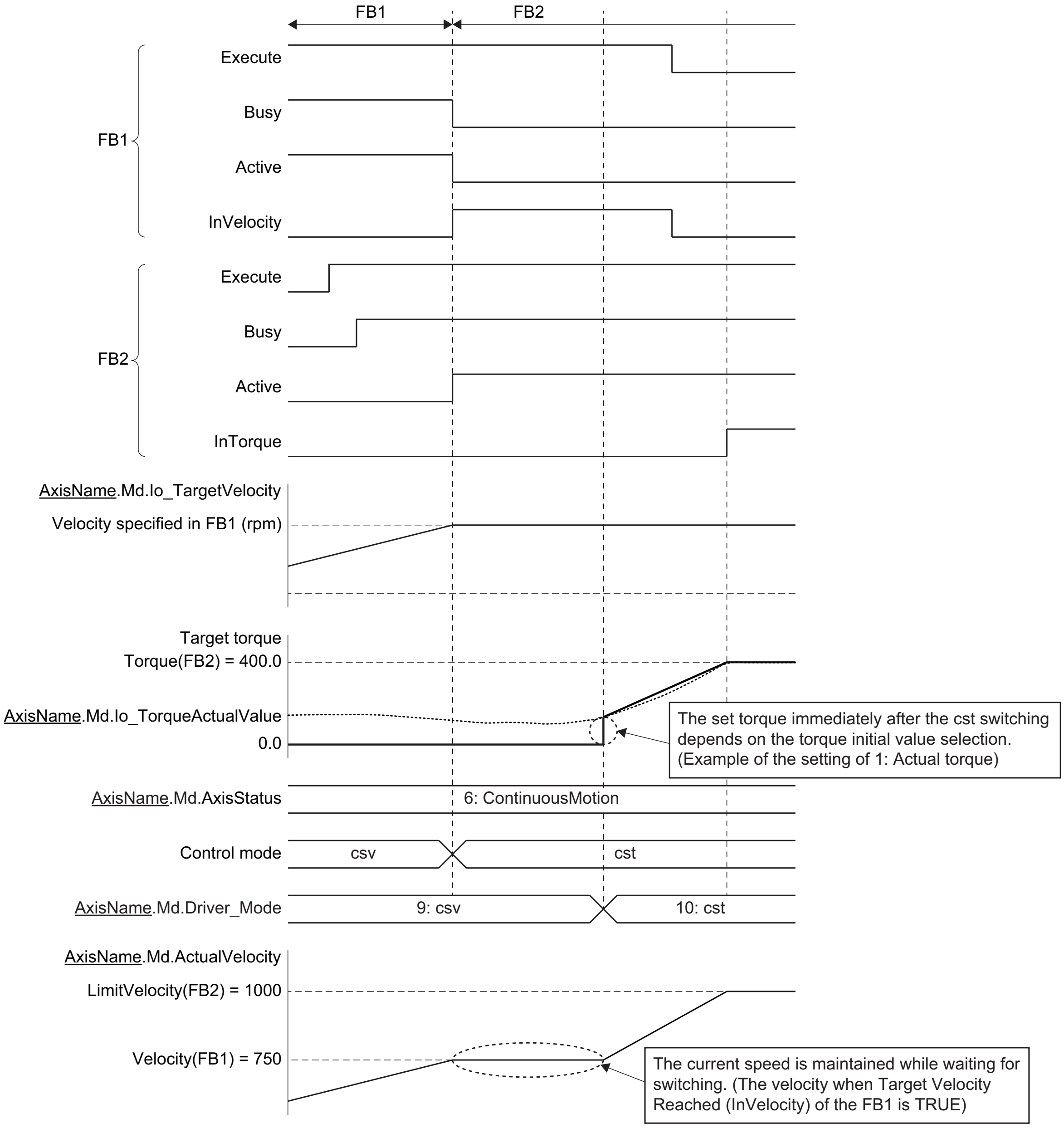
2. While waiting for change, csv is continued.

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.

- 0: Aborting (mcAborting)

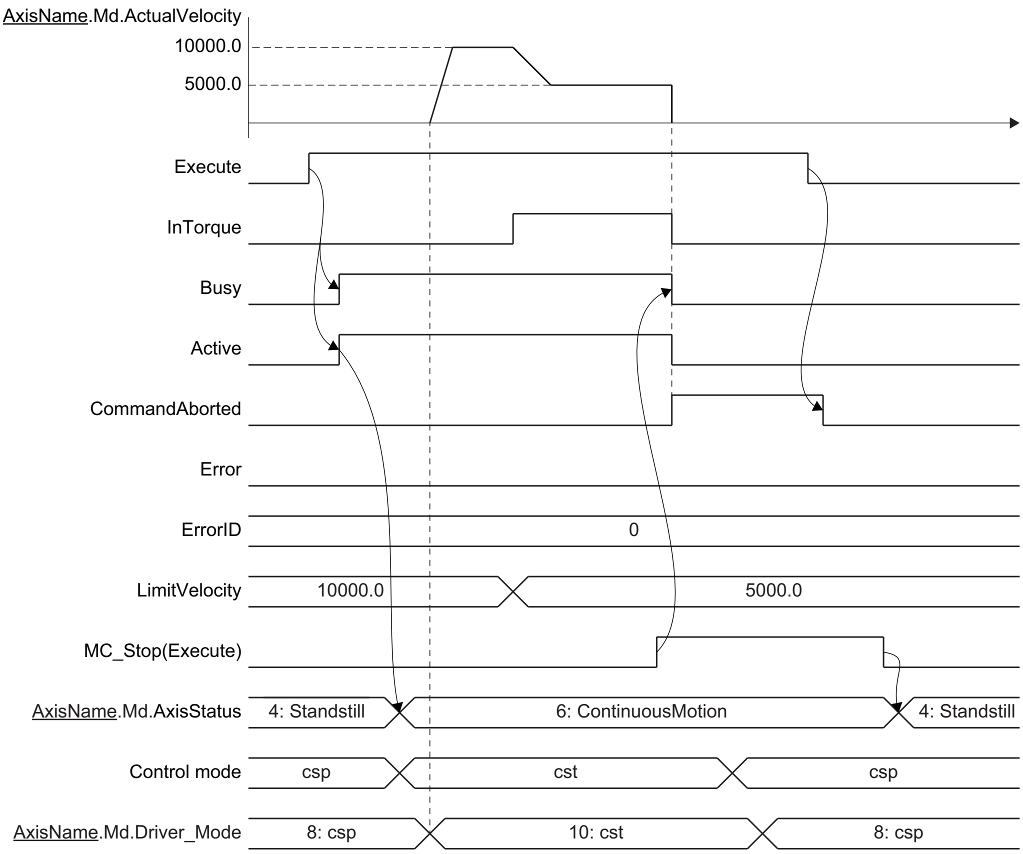


• 1: Buffered (mcBuffered)



When starting another instruction during this FB execution

■ Torque control → Stop instruction (MC_Stop (Forced Stop))



■ Torque control → Positioning (Absolute value/Relative value positioning)

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 device specifications. If switching to the csp is not completed within 1 second [s], "Control Mode Switching Error (error code: 1A1DH)" occurs due to a timeout and the axis stops.)

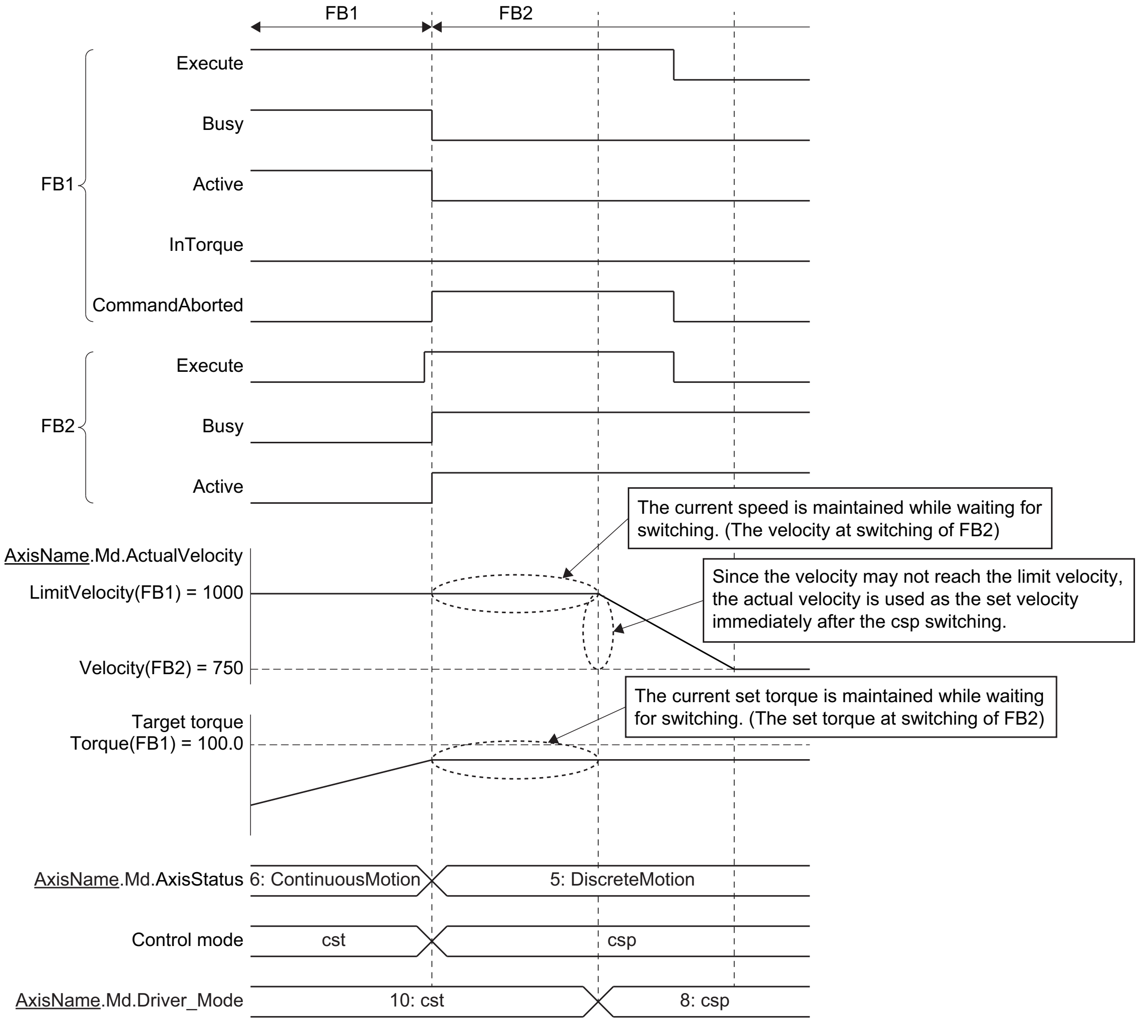
3. While waiting for switching, torque control is continued with the torque and the limit velocity at switching.

4. When the driver switches to the csp, the positioning control starts.

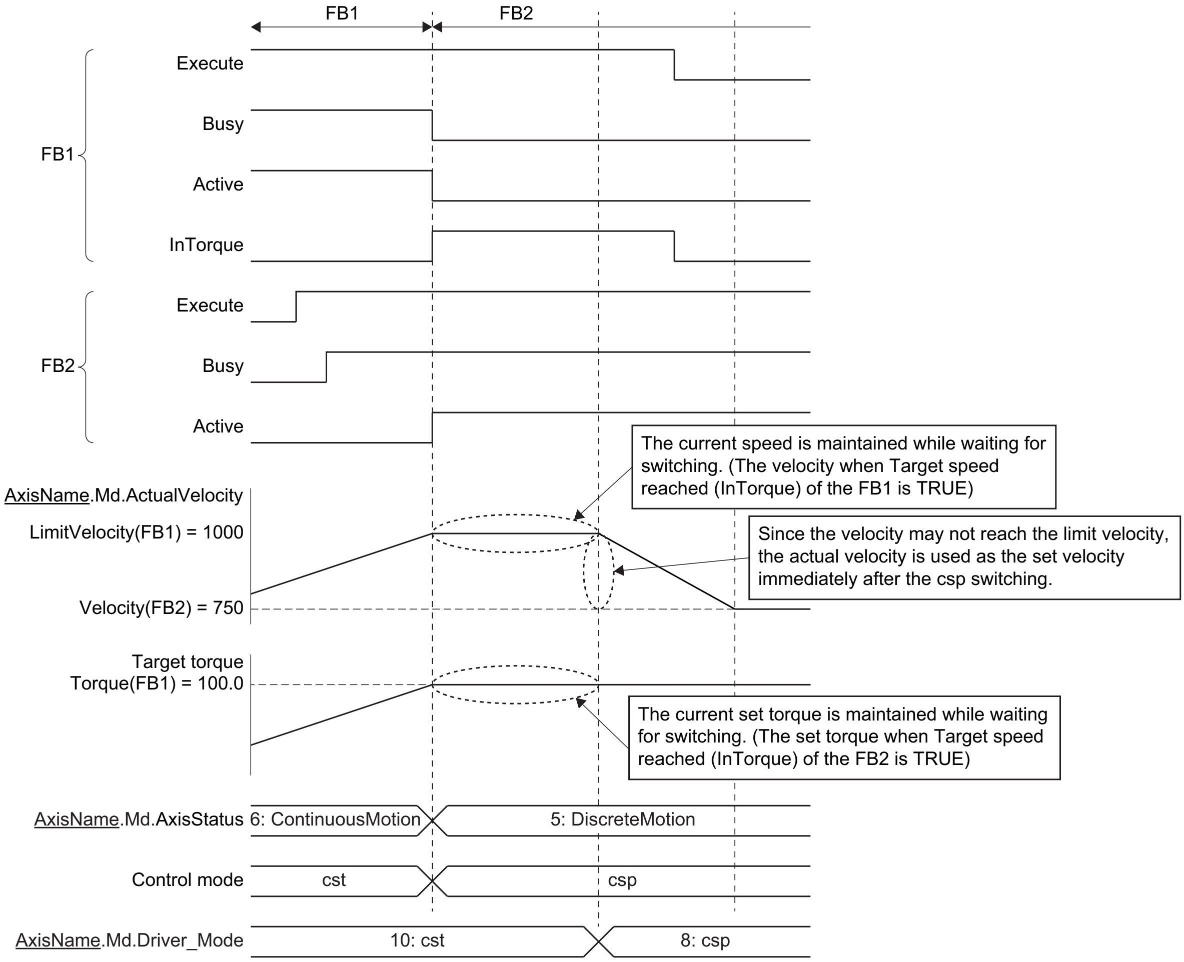
Point

- 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 Operation Setting at Overrun (AxisName.Pr.OverrunOperation).
☞ Page 132 Multiple Start (Buffer Mode)
- For operation at the overrun occurrence, refer to the following.

• 0: Aborting (mcAborting)



• 1: Buffered (mcBuffered)



■ Torque control (FB1) → Velocity control (FB2)

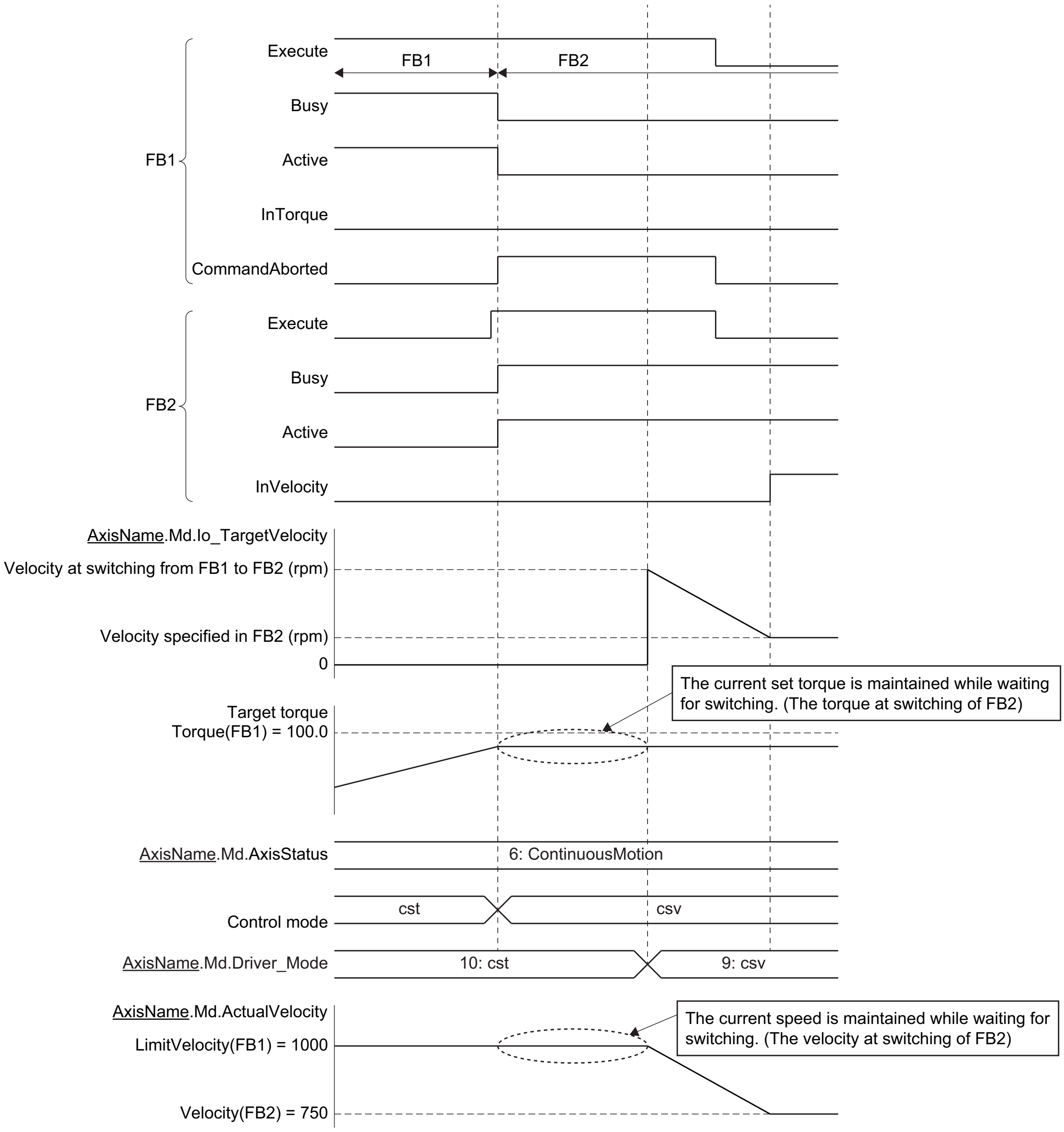
1. A csv switching request is issued to the driver.

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 device specifications. If switching to the csv is not completed within 1 [s], "Control Mode Switching Error (error code: 1A1DH)" occurs due to a timeout, and the axis will stop.)

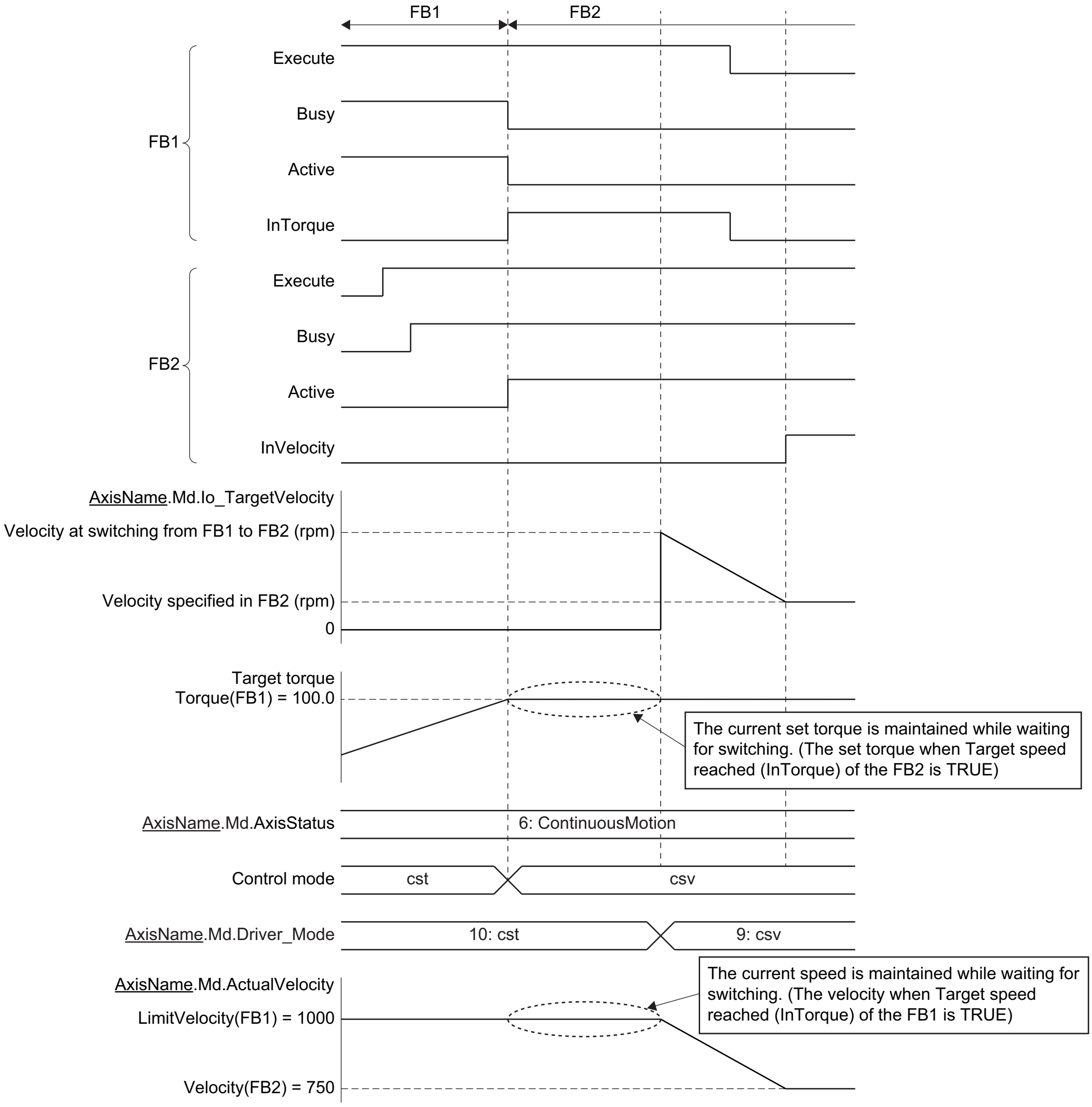
3. While waiting for switching, torque control is continued with the torque and the limit velocity at switching.

4. When the driver switches to the csv, velocity control is started.

- 0: Aborting (mcAborting)



• 1: Buffered (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 "0.0". After that the operation starts to the changed direction.

Precautions

- Velocity Override Factor (AxisName.Cd.VelocityOverride) and Acceleration Override Factor (AxisName.Cd.AccelerationOverride) are valid.
- The set position and the machine feed value are updated by follow up.
- Time until the control mode is switched depends on the specifications of the driver device.
- 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 Driver Control Mode (AxisName.Md.Driver_Mode) is switched to "10: cst".
- When switching from the csp to the cst or from the cst to the csp without waiting for stop of the motor by using the MR-J5(W)-G, 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 834 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 (Positive Direction Torque Limit Value (AxisName.Md.TorqueLimit_Positive), Negative Direction Torque Limit Value (AxisName.Md.TorqueLimit_Negative)), "Torque Limit Value Over Warning (warning code: 0D12H)" will occur and the axis operates at the value before change.

During torque control, if Positive Direction Torque Limit Value (AxisName.Cd.TorqueLimit_Positive) and Negative Direction Torque Limit Value (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 and the cumulative current position to an arbitrary address. The current position change method includes a method of changing the set position at control execution. (MC_SetPosition (Current Position Change)) (The machine feed value 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 Axis Status (AxisName.Md.AxisStatus) is "4: Standby (Standstill)".
- 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 420 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 (Current Position Change)
	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 (Parameter Write). However, it cannot be changed during control.
*2 The maximum value of the limit value can not be changed.

Relevant FBs

For details on Motion control FBs, refer to the following.
📖MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

Motion control FB	Name	Description
MC_SetPosition	Current position change	Changes the current position of the specified axis (set position, actual position).

Current position change

The control details of current position change are shown below.

Execution when the axis stops

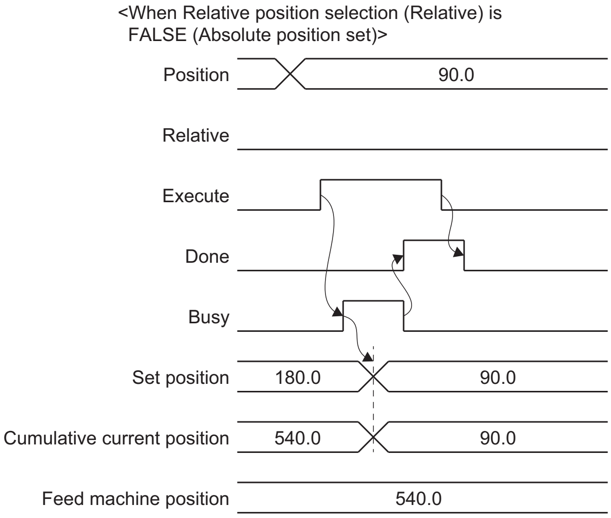
The current position change operation starts at rising edge detection of Execute command (Execute) of MC_SetPosition (Current Position Change).

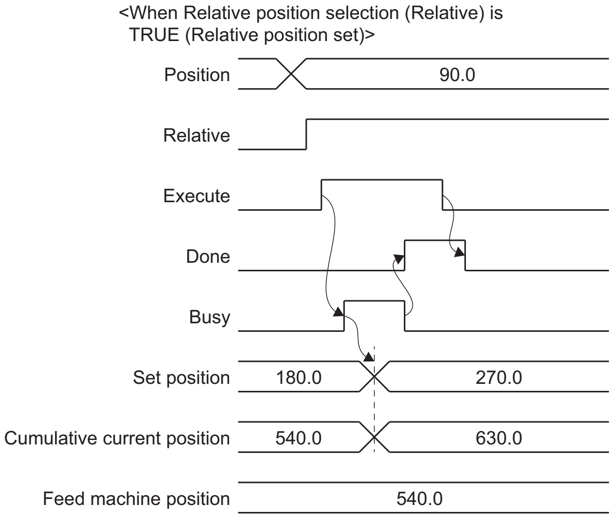
In the current position change, the set position and the cumulative current position are changed to the position specified in Target position (Position).

Target position (Position) is "Relative position set" when Relative position selection (Relative) is TRUE, and it is "Absolute position set" when Relative position selection (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).





8

Execution during axis operation

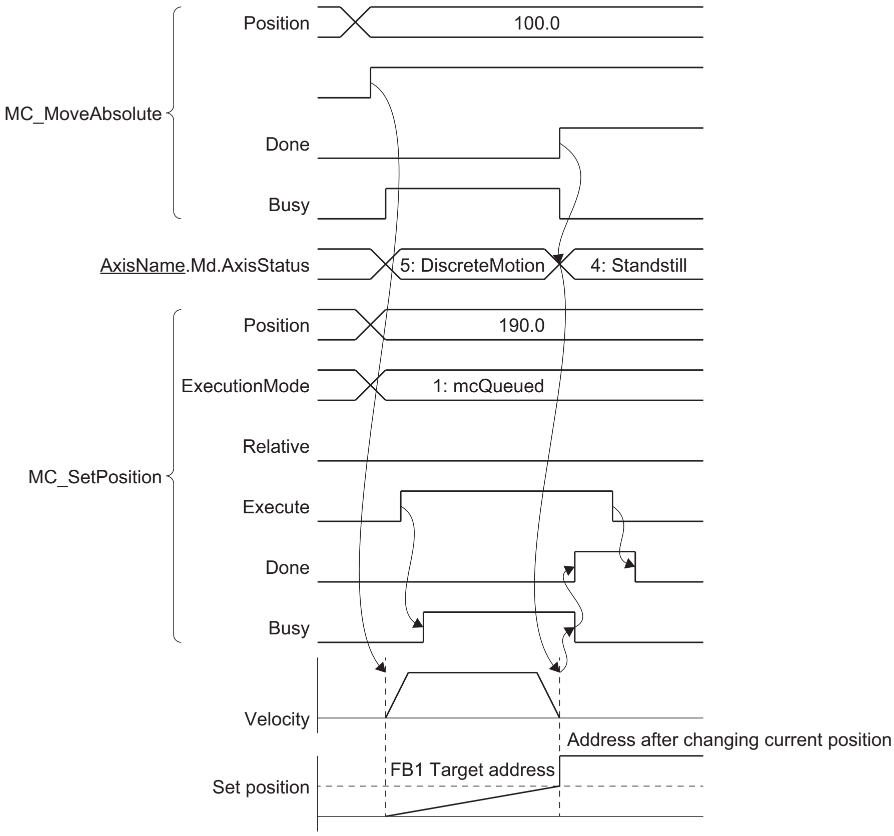
■ Execute at Completion of Previous One

When "1: Execute at Completion of Previous One (mcQueued)" is set in Execution mode (ExecutionMode), this FB is executed after the on-going FB is completed.

Executing (Busy) turns TURE by rising edge detection of Execute command (Execute), and this FB waits until Axis Status (AxisName.Md.AxisStatus) turns "4: Standby (Standstill)".

As soon as Axis Status (AxisName.Md.AxisStatus) turns "4: Standby (Standstill)", the current position change operation will start.

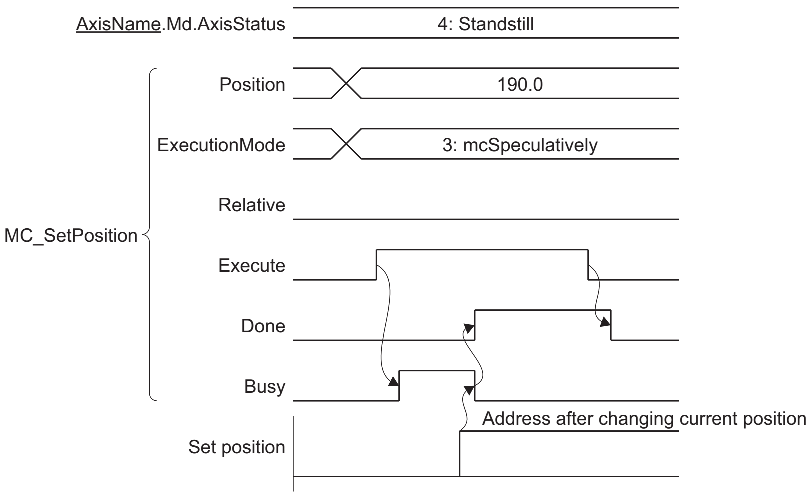
Ex.



■ Speculative execution

When "3: Execute Speculatively (mcSpeculatively)" is specified to Execution mode (ExecutionMode), the current position change can be performed only when Axis Status (AxisName.Md.AxisStatus) is "4: Standby (Standstill)" at the rising edge detection of Execute command (Execute). If Axis status (AxisName.Md.AxisStatus) of relevant axis is not "4: Standby (Standstill)", or FB instruction is being executed, "MC_SetPosition Instruction Error (error code: 344EH)" occurs, and the current position change is not performed.

Ex.



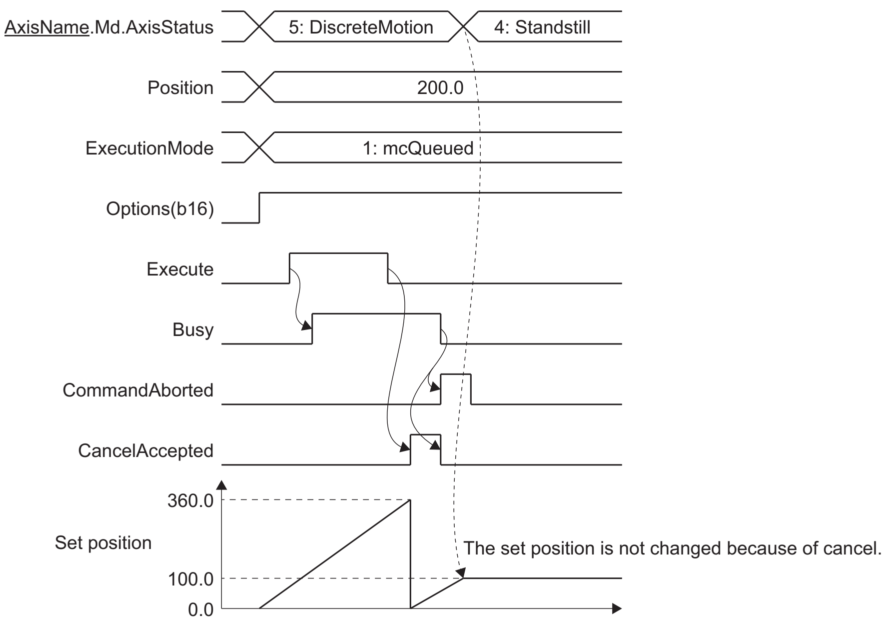
Cancel

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 (Options): Bit 16) set to "1: Allow".
- Cancel is started at falling edge detection of Execute command (Execute).
- Cancel is accepted only when Executing (Busy) of the output pin is TRUE.
- When FB accepts cancel, Cancel acceptance (CancelAccepted) of the output pin becomes TRUE.
- When cancel is completed, Abortion of execution (CommandAborted) of the output pin becomes TRUE.
- When cancel was carried out, the set position is not changed.

Ex.

When canceling while Axis Status (AxisName.Md.AxisStatus) is waiting for switching to "4: Standby (Standstill)" after starting MC_SetPosition (Current Position Change) with "1: Execute at Completion of Previous One (mcQueued)" in Execution mode (ExecutionMode)



8.2 Command In-position

The command in-position function checks the remaining distance to the target position, and changes the command in-position 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 changes Command In-position (AxisName.Md.CmdInPos) to TRUE when the remaining distance to the target position became Command In-position Width (AxisName.Md.CmdInPos_Width) or less.
It changes Command In-position (AxisName.Md.CmdInPos) to FALSE when the axis starts and when the FB is switched by multiple start.

Relevant variables

Axis parameter (AxisName.Pr.)

Variable/Structure name	Name	Details
CmdInPos_Width	Command In-position Width	Sets the command in-position width. • 0.0: Function disabled • 0.000000001 to 10,000,000,000.0: Function enabled*1

*1 If positive numbers less than 0.000000001 are set, it imports 0.0

Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
CmdInPos_Width	Command In-position Width	Displays the command in-position width. • The value imports the setting value of Command In-position Width (AxisName.Pr.CmdInPos_Width). • When Command In-position Width (AxisName.Md.CmdInPos_Width) is 0.0, the command in-position function of the axis is disabled.
CmdInPos	Command In-position	Displays whether the remaining distance to the target position is Command In-position Width (AxisName.Md.CmdInPos_Width) or less. • FALSE: More than Command in-position width • TRUE: Less than Command in-position width

Axes group parameter (AxesGroupName.Pr.)

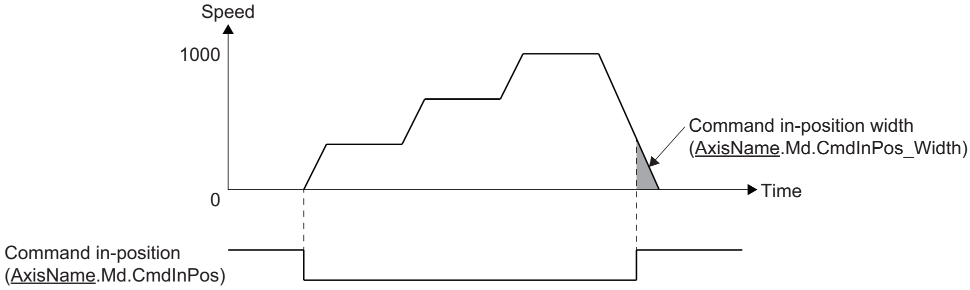
Variable/Structure name	Name	Details
CmdInPos_Width	Command In-position Width	Sets the command in-position width on the composite axis of the axes group. • 0.0: Function disabled • 0.000000001 to 10,000,000,000.0: Function enabled*1

Axes group monitor data (AxesGroupName.Md.)

Variable/Structure name	Name	Details
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 Command In-position Width (AxesGroupName.Pr.CmdInPos_Width). • When Command In-position Width (AxesGroupName.Md.CmdInPos_Width) is 0.0, the command in-position function of the axes group is disabled.
CmdInPos	Command In-position	Displays whether the remaining distance to the target position on the composite axis is Command In-position Width (AxesGroupName.Pr.CmdInPos_Width) or less. • FALSE: More than Command in-position width • TRUE: Less than Command in-position width

Command in-position width check

A command in-position width check is carried out every operation cycle.
However, while the axis stops, Command In-position (AxisName.Md.CmdInPos) is not refreshed.
When Command In-position Width (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 Command In-position Width (AxesGroupName.Md.CmdInPos_Width). When the long axis velocity specification and the reference axis speed specification are specified on the linear interpolation control, the remaining distance on the long axis and the reference axis are checked.
The flag is stored in Command In-position (AxesGroupName.Md.CmdInPos).
During axes group operation, the command in-position function of the configuration axes is disabled. Command In-position (AxisName.Md.CmdInPos) of the configuration axes is not changed to TRUE during axes group operation. (It is changed to FALSE when the axes group is started.)



- (the remaining distance) ≤ (Command in-position width (AxisName.Md.CmdInPos_Width))

8

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 "machine feed value" 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 Software Stroke Limit Target (AxisName.Pr.SwStrokeLimit_Target).

Relevant variables

Axis parameter (AxisName.Pr.)

Variable/Structure name	Name	Details
SwStrokeLimit_Upper	Software Stroke Limit Upper Value	Sets the software stroke limit upper value. • -10000000000.0 to 10000000000.0
SwStrokeLimit_Lower	Software Stroke Limit Lower Value	Sets the software stroke limit lower value. • -10000000000.0 to 10000000000.0
SwStrokeLimit_Target	Software Stroke Limit Target	Sets the software stroke limit target. • -1: Disabled (Invalid) • 1: Set position (SetPosition) • 3: Feed machine position (FeedMachinePosition)

Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
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. • DISABLE: Check disabled • ONLY_INSIDE: Disables the check only for movement toward the setting range. • Other than the above: Check enabled (no disable request)

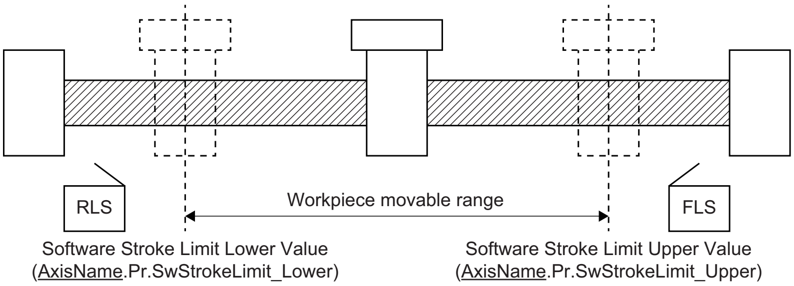
Axis control data (AxisName.Cd.)

Variable/Structure name	Name	Details
SwStrokeLimit_Override	Software Stroke Limit Override	Temporarily switches enabled/disabled of the software stroke limit. • DISABLE: Check disabled • ONLY_INSIDE: Disables the check only for movement toward the setting range. • Other than the above: Check enabled (no disable request)

Definition of movable range

Movable range differences by selection of set position and machine feed value

The following drawing shows the movable range of the workpiece when the software stroke limit function is used.

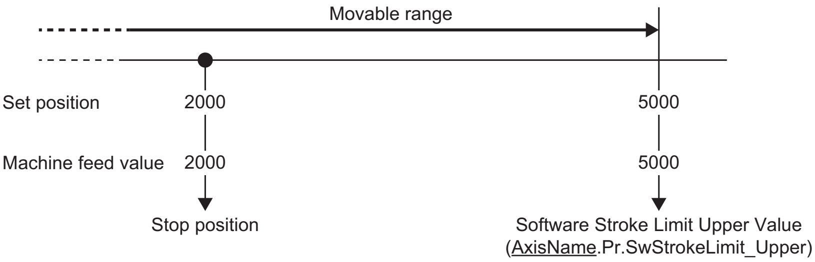


The set position or the machine feed value can be specified in the software stroke limit. (The cumulative current position cannot be specified.)

The following drawing shows the differences in the operation when the set position is used or when the machine feed value is used in the movable range limit check.

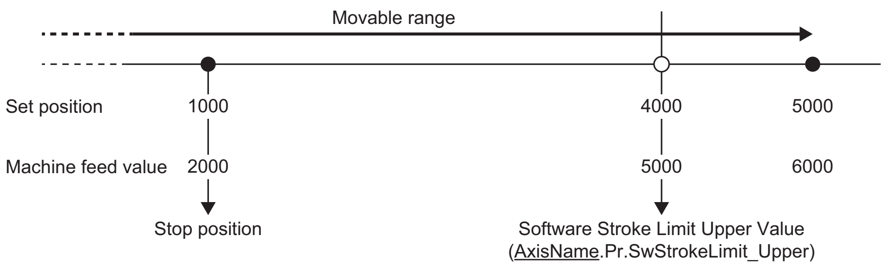
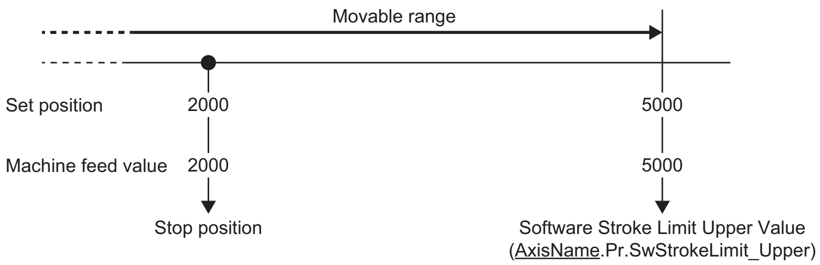
Ex.

When assuming the current value is 2000, and Software Stroke Limit Upper Value (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 machine feed value will stay the same at 2000.

Setting	Description
When the machine feed value is set at the limit	The set position of 4000 (machine feed value: 5000) becomes the upper stroke limit. 
When the set position is set at the limit	The set position of 5000 (machine feed value: 6000) becomes the upper stroke limit. 

Point

When "3: Machine Feed Value (FeedMachinePosition)" is set in Software Stroke Limit Target (AxisName.Pr.SwStrokeLimit_Target), the movable range becomes an absolute range referenced on the home position. When "1: Set Position (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

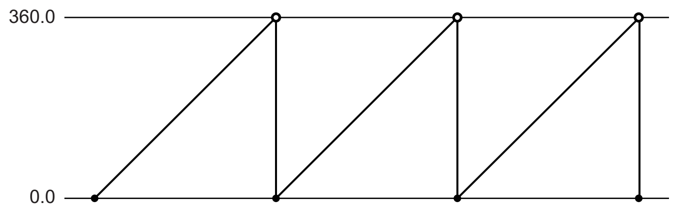
- Set position

  The storing range of the set position is shown below.

Monitor data	Range
Set Position (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

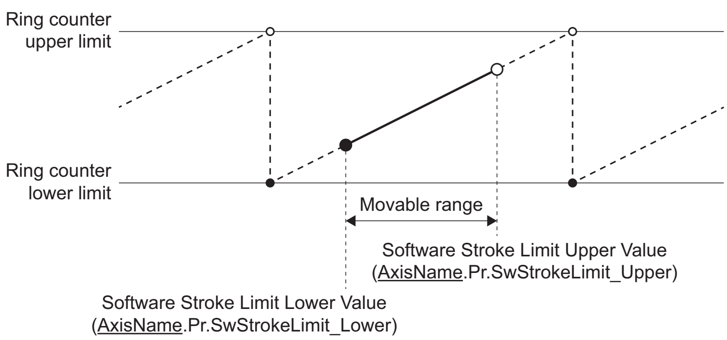
  The setting range of the software stroke limit upper/lower value is shown below.

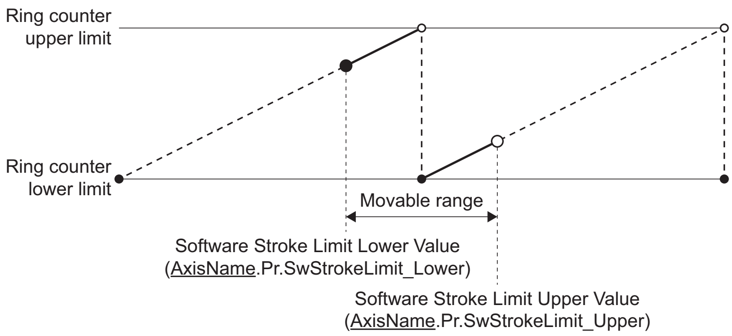
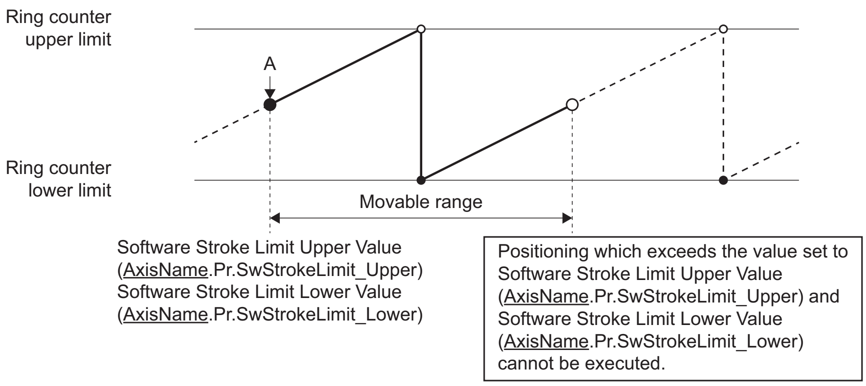
Parameter	Setting range
Software Stroke Limit Upper Value (AxisName.Pr.SwStrokeLimit_Upper)	Ring counter lower value < Setting value ≤ Ring counter upper value
Software Stroke Limit Lower Value (AxisName.Pr.SwStrokeLimit_Lower)	Ring counter lower value ≤ Setting value < Ring counter upper value

Movable range: Software Stroke Limit Lower Value (AxisName.Pr.SwStrokeLimit_Lower) ≤ Set position < Software Stroke Limit Upper Value (AxisName.Pr.SwStrokeLimit_Upper)

- Details of the movable range

  The movable range varies as shown below depending on the relationship between Software Stroke Limit Upper Value (AxisName.Pr.SwStrokeLimit_Upper) and Software Stroke Limit Lower Value (AxisName.Pr.SwStrokeLimit_Lower).

Movable range	Description
Software Stroke Limit Upper Value (AxisName.Pr.SwStrokeLimit_Upper) > Software Stroke Limit Lower Value (AxisName.Pr.SwStrokeLimit_Lower)	

<table>
<tr><th>Movable range</th><th>Description</th></tr>
<tr><td>Software Stroke Limit Upper Value (AxisName.Pr.SwStrokeLimit_Upper) < Software Stroke Limit Lower Value (AxisName.Pr.SwStrokeLimit_Lower)</td><td>
</td></tr>
<tr><td>Software Stroke Limit Upper Value (AxisName.Pr.SwStrokeLimit_Upper) = Software Stroke Limit Lower Value (AxisName.Pr.SwStrokeLimit_Lower)</td><td>

• Positioning that crosses over the values set to Software Stroke Limit Upper Value (AxisName.Pr.SwStrokeLimit_Upper) and Software Stroke Limit Lower Value (AxisName.Pr.SwStrokeLimit_Lower) cannot be executed.</td></tr>
</table>

■ When selecting the set position or the machine feed value when the ring counter is invalid

• Machine feed value

The storing range of the machine feed value is shown below.

Monitor data	Storing range
Machine Feed Value (AxisName.Md.FeedMachinePosition)	-10000000000.0 ≤ Machine feed value < 10000000000.0

• Setting the software stroke limit

The setting range of the software stroke limit upper/lower value is shown below.

Parameter	Setting range
Software Stroke Limit Upper Value (AxisName.Pr.SwStrokeLimit_Upper)	Software Stroke Limit Lower Value (AxisName.Pr.SwStrokeLimit_Lower) < Setting range ≤ 10000000000.0
Software Stroke Limit Lower Value (AxisName.Pr.SwStrokeLimit_Lower)	-10000000000.0 ≤ Setting range < Software Stroke Limit Upper Value (AxisName.Pr.SwStrokeLimit_Upper)

Movable range: Software Stroke Limit Lower Value (AxisName.Pr.SwStrokeLimit_Lower) ≤ Machine feed value < Software Stroke Limit Upper Value (AxisName.Pr.SwStrokeLimit_Upper)

Invalidating the software stroke limit

To invalidate the software stroke limit, set "-1: Invalid" in Software Stroke Limit Target (AxisName.Pr.SwStrokeLimit_Target). By setting "-1: Invalid (Invalid)", the software stroke limit check is not executed in all controls regardless of the setting values of Software Stroke Limit Upper Value (AxisName.Pr.SwStrokeLimit_Upper) and Software Stroke Limit Lower Value (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 (Check disabled for movement returning to the range) or DISABLE (Check disabled) in Software Stroke Limit Override (AxisName.Cd.SwStrokeLimit_Override), execute the axis operation to return to the movable range.
Software Stroke Limit Override (AxisName.Cd.SwStrokeLimit_Override) is case-sensitive

Point

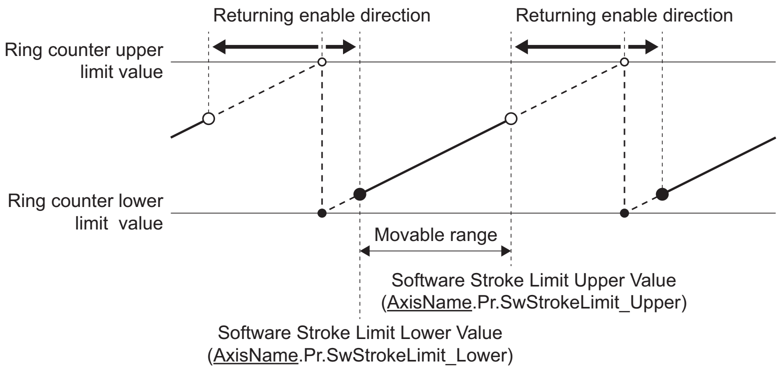
- In Software Stroke Limit Override (AxisName.Cd.SwStrokeLimit_Override), the software stroke limit can be temporarily disabled without changing the settings of Software Stroke Limit Upper Value (AxisName.Pr.SwStrokeLimit_Upper) and Software Stroke Limit Lower Value (AxisName.Pr.SwStrokeLimit_Lower).
- In Software Stroke Limit Override (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 a start or multiple start occurs while the software stroke limit is disabled by Software Stroke Limit Override (AxisName.Cd.SwStrokeLimit_Override), "Start when SW Stroke Limit is Disabled (event code: 07F5)" is recorded in the event history.

■ Returning enable direction to the movable range

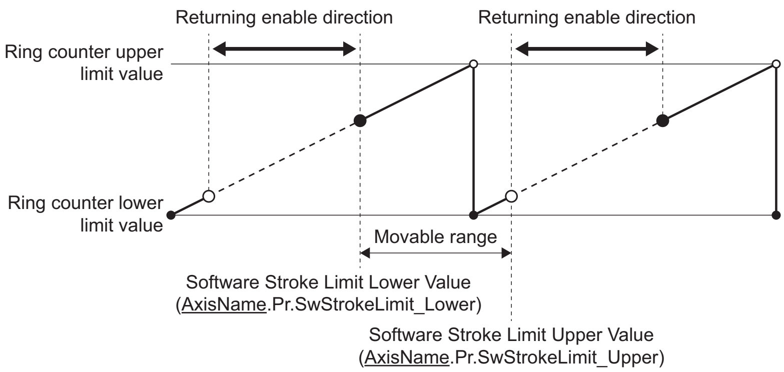
- When setting ONLY_INSIDE (Check disabled for movement returning to the range) in Software Stroke Limit Override (AxisName.Cd.SwStrokeLimit_Override), movement in the returning enable direction from outside the software stroke limit range to the movable range is allowed.
- The returning enable direction 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, "Software Stroke Limit Over (start Position) (error code: 1A01H)" or "Software Stroke Limit Over (Target Position) (error code: 1A00H)" will occur, 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, "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 (Check disabled) in Software Stroke Limit Override (AxisName.Cd.SwStrokeLimit_Override) and carry out a move to the movable range.

■ Returning enable direction when setting ONLY_INSIDE (Check disabled for movement returning to the range) in Software Stroke Limit Override (AxisName.Cd.SwStrokeLimit_Override)

- Software Stroke Limit Upper Value (AxisName.Pr.SwStrokeLimit_Upper) > Software Stroke Limit Lower Value (AxisName.Pr.SwStrokeLimit_Lower)



- Software Stroke Limit Upper Value (AxisName.Pr.SwStrokeLimit_Upper) < Software Stroke Limit Lower Value (AxisName.Pr.SwStrokeLimit_Lower)



Checking the software stroke limit enabled/disabled

- In the monitor Software Stroke Limit Override (AxisName.Md.SwStrokeLimit_Override), the software stroke limit check enabled/disabled status can be checked.
- Software Stroke Limit Override (AxisName.Md.SwHwStrokeLimit_Override) imports setting values of Software Stroke Limit Override (AxisName.Md.SwStrokeLimit_Override) at PLC READY from OFF to ON, at start, and at multiple start, and updates as in the table below.

Software Stroke Limit Target (AxisName.Pr.SwStrokeLimit_Target)	Software Stroke Limit Override (AxisName.Cd.SwStrokeLimit_Override)	Software Stroke Limit Override (AxisName.Md.SwStrokeLimit_Override)
-1: Invalid (Invalid)	—	DISABLE: Check disabled
1: Set position (SetPosition) 3: Machine feed value (FeedMachinePosition)	Other than the following: No disable request DISABLE: Check disabled ONLY_INSIDE: Check disabled for movement returning to the range	Display is blank: No disable request DISABLE: Check disabled ONLY_INSIDE: Check disabled for movement returning to the range

8

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"	"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	"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	"Software Stroke Limit Over (Forward Direction) (error code: 1A03H)" or "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 functions are different from the above.

- Homing control (☞ Page 199 Precautions)
- Multiple axes positioning data operation (☞ Page 258 Software stroke limit)

Precautions

- Homing must be executed beforehand for the software stroke limit function to function properly.
- If Software Stroke Limit Upper Value (AxisName.Pr.SwStrokeLimit_Upper) or Software Stroke Limit Lower Value (AxisName.Pr.SwStrokeLimit_Lower) is outside the setting range, "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 "3: Machine Feed Value (FeedMachinePosition)" is set as the setting value of Software Stroke Limit Target (AxisName.Pr.SwStrokeLimit_Target) to the axis type which does not have Machine Feed Value (AxisName.Md.FeedMachinePosition), "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 381 External Signal Selection

Operation of this function for each system status

○: Possible, △: Possible (restricted)

Status	Operation availability
STOP	○
RUN	○
Moderate error	△*1
Major error	△*1

*1 Depending on the error, it may not be possible to send the FLS/RLS signal.
When a moderate or major error occurs, the motion system notifies the device station that an error has been detected in the CPU operating status. For details on the operation of the device station when an error is detected, refer to each device station manual.

Relevant variables

Axis parameter constant (AxisName.PrConst.)

Variable/Structure name	Name	Details
HwStrokeLimit_FlsSignal	Upper Limit Signal	Sets a signal using the upper limit signal (FLS). Upper limit signal (HwStrokeLimit_FlsSignal) is the structure of SIGNAL_SELECT type. 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 383 SIGNAL_SELECT structure The specific setting and the operation of this signal are shown below. ■I/O Number (StartIO) Ignores the input value. ■Target (Target) When the specification is not set, signal disabled is determined and the signal undetection status is always set. Only [VAR], [DEV], and [CONST] can be specified for the data type. *: When unavailable data is set, "Out of Parameter Range (Axis) (error code: 1D80H)" is output occurs. ■Signal Detection Method (Detection) The specification is allowed at the following level detection only. • 0: Detection at TRUE (HighLevel) • 1: Detection at FALSE (LowLevel) *: When a value outside the range set, "Out of Parameter Range (Axis) (error code: 1D80H)" occurs. ■Compensation Time (CompensationTime) Ignores the input value. ■Filter Time (FilterTime) Setting range is 0.0 to +5.0 [s]. *: When a value outside the range is set, "Out of Filter Time Setting Range of Each Axis Signal Warning (warning code: 0D24H)" occurs, and operates at the filter time 0.0.

Variable/Structure name	Name	Details
HwStrokeLimit_RlsSignal	Lower Limit Signal	Sets a signal using the lower limit signal (RLS). Lower limit signal (HwStrokeLimit_RlsSignal) is the structure of SIGNAL_SELECT type. 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 383 SIGNAL_SELECT structure The specific setting and the operation of this signal are shown below. ■I/O Number (StartIO) Ignores the input value. ■Target (Target) When the specification is not set, signal disabled is determined and the signal undetection status is always set. Only [VAR], [DEV], and [CONST] can be specified for the data type. *: When unavailable data is set, "Out of Parameter Range (Axis) (error code: 1D80H)" occurs. ■Signal Detection Method (Detection) The specification is allowed at the following level detection only. • 0: Detection at TRUE (HighLevel) • 1: Detection at FALSE (LowLevel) *: When edge detection specification is set, "Out of Parameter Range (Axis) (error code: 1D80H)" occurs. ■Compensation Time (CompensationTime) Ignores the input value. ■Filter Time (FilterTime) Setting range is 0.0 to +5.0 [s]. *: When a value outside the range is set, "Out of Filter Time Setting Range of Each Axis Signal Warning (warning code: 0D24H)" occurs, and operates at the filter time 0.0.

Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
HwStrokeLimit_FlsStatus	Upper Limit Signal Status	Displays the detection status of the upper limit signal (FLS). • FALSE: Detecting the upper limit signal (FLS) • TRUE: Not detecting the upper limit signal (FLS) *: Displays the detection 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 lower limit signal (RLS). • FALSE: Detecting the lower limit signal (RLS) • TRUE: Not detecting the lower limit signal (RLS) *: Displays the detection state which reflects the signal detection method or the filter time but not the signal input state.
HwStrokeLimit_Override	Hardware Stroke Limit Override	Displays the acceptance status of the enabled/disabled switching command of the hardware stroke limit. • DISABLE: Check disabled • ONLY_INSIDE: Disables the check only for movement toward the setting range. • Other than the above: Check enabled (no disable request)

Axis control data (AxisName.Cd.)

Variable/Structure name	Name	Details
HwStrokeLimit_Override	Hardware Stroke Limit Override	Switches the enabled/disabled of the hardware stroke limit. • DISABLE: Check disabled • ONLY_INSIDE: Disables the check only for movement toward the setting range. • Other than the above: Check enabled (no disable request)

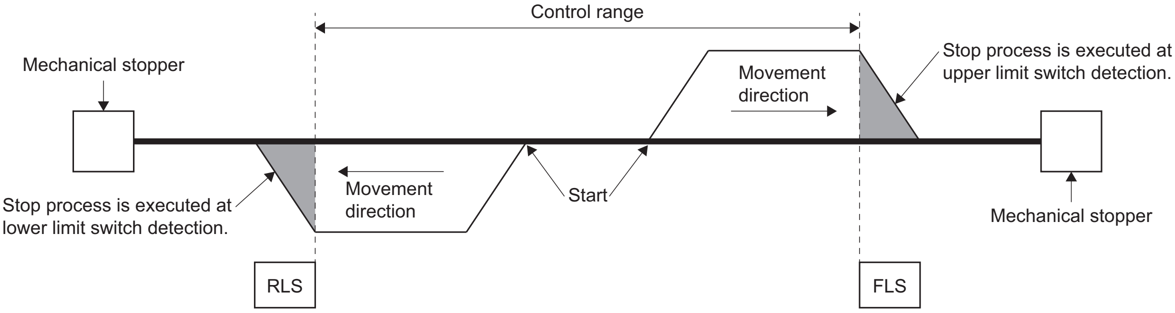
Hardware stroke limit check details

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 upper limit signal (FLS) is detected at start.	At start	"FLS Signal Detection (at Start) (error code: 1A2DH)" occurs and the axis does not start.
Signal input from the lower limit signal (RLS) is detected at start.	At start	"RLS Signal Detection (at Start) (error code: 1A2EH)" occurs and the axis does not start.
Signal input from the upper limit signal (FLS) is detected during control.	During control	"FLS Signal Detection (Controlling) (error code: 1A2FH)" occurs and the axis stops.*1
Signal input from the lower limit signal (RLS) is detected during control	During control	"RLS Signal Detection (Controlling) (error code: 1A30H)" occurs 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.

8

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 (Check disabled for movement returning to the range) or DISABLE (Check disabled) in Hardware Stroke Limit Override (AxisName.Cd.HwStrokeLimit_Override), execute the axis operation to return to the controllable range.

Point

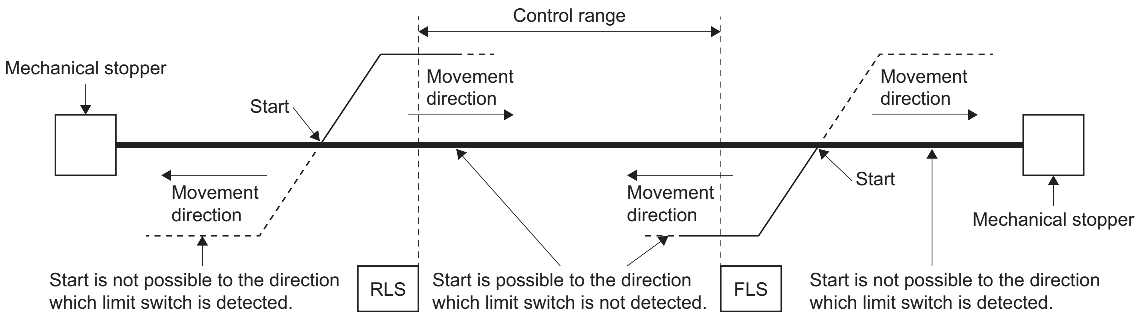
- When setting an item other than ONLY_INSIDE (Check disabled for movement returning to the range) and DISABLE (Check disabled) in Hardware Stroke Limit Override (AxisName.Cd.HwStrokeLimit_Override), the axis operates with "No disable request".
  Hardware Stroke Limit Override (AxisName.Cd.HwStrokeLimit_Override) is case-sensitive.
- If a start or multiple start occurs while the hardware stroke limit is disabled by Hardware Stroke Limit Override (AxisName.Cd.HwStrokeLimit_Override), "Start when HW Stroke Limit is Disabled (event code: 07F6)" is recorded in the event history.

Precautions

Hardware Stroke Limit Override (AxisName.Cd.HwStrokeLimit_Override) is valid only in the hardware stroke limit processing on the Motion module side. The stroke limit processing on the drive module side is not affected by Hardware Stroke Limit Override (AxisName.Cd.HwStrokeLimit_Override).

When setting ONLY_INSIDE (Check disabled for movement returning to the range)

The axis can start to the direction in which the limit switch has not been detected.



When setting DISABLE (Check disabled)

The axis can start regardless of the limit switch detection status.

Checking the hardware stroke limit enabled/disabled

- In Hardware Stroke Limit Override (AxisName.Md.HwStrokeLimit_Override), the hardware stroke limit check enabled/disabled status can be checked.
- Hardware Stroke Limit Override (AxisName.Md.HwStrokeLimit_Override) imports the setting value of Hardware Stroke Limit Override (AxisName.Cd.HwStrokeLimit_Override) at start and at multiple start, and updates as in the table below.

Hardware Stroke Limit Override (AxisName.Cd.HwStrokeLimit_Override)	Hardware Stroke Limit Override (AxisName.Md.HwStrokeLimit_Override)
DISABLE (Check disabled)	DISABLE (Disabling hardware stroke limit check)
ONLY_INSIDE (Check disabled for movement returning to the range)	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 upper limit signal (FLS) in the direction in which the set position increases, and install the lower limit signal (RLS) in the direction in which the set position decreases. If inverting the install positions of the upper limit signal (FLS)/lower limit signal (RLS), hardware stroke limit function cannot be operated properly.
- The upper limit signal (FLS)/lower limit signal (RLS) 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 838 Using methods

Point

- 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.

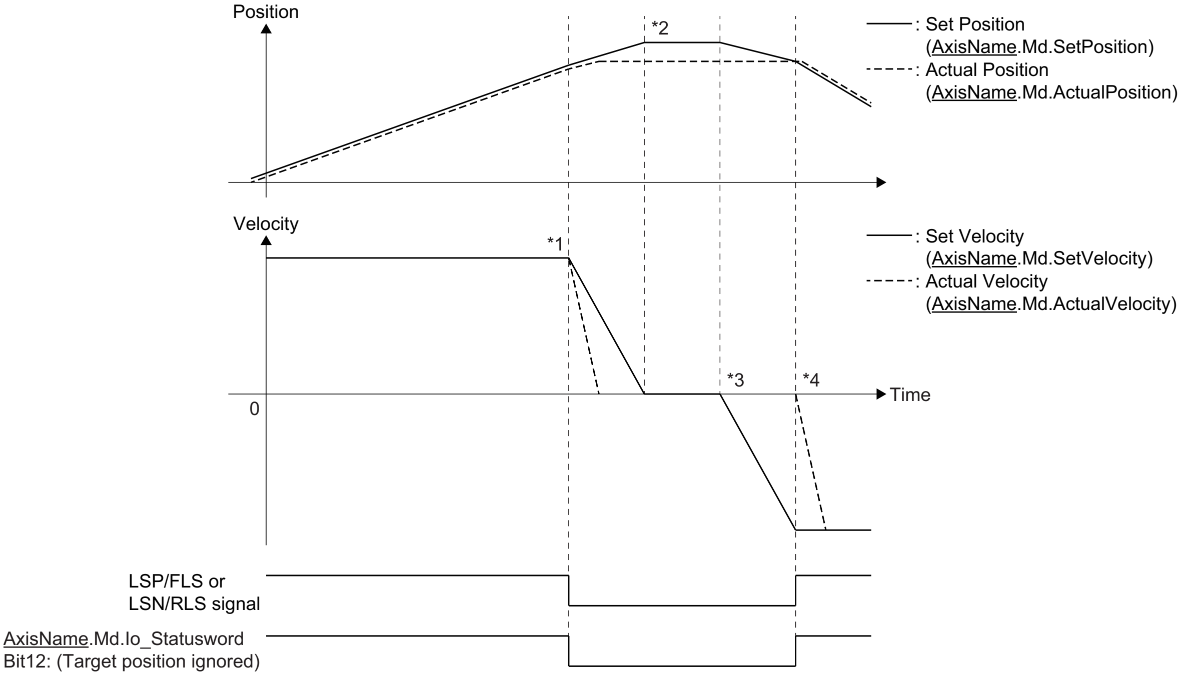
- The LSP/LSN signal input to the servo amplifier when connecting to MR-J5(W)-G can be set as the hardware stroke limit signal. For details on the setting method, refer to the following.
  ☞ Page 838 Using methods
- Depending on the setting of "Limit switch enabled status selection" (PD41.2) of the servo parameter "Function selection D-4(PD41)" when connecting to MR-J5(W)-G, the operation differs at LSP/FLS or LSN/RLS signal detection.

Setting	Operation
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 Driver Command Discard Detection Setting (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 to set "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 Driver Command Discard Detection Setting (AxisName.Pr.StopOption_DriverTargetIgnored).
  The following shows an operation example.

Ex.

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 Driver Command Discard Detection Setting (AxisName.Pr.StopOption_DriverTargetIgnored) is enabled, the motion system will detect "Driver Command Discard Detection (error code: 1AE6H)" and execute the command stop. For details on the setting, refer to the following.

☞ Page 161 Stop

*2 After completion of the stop process on the motion system side, it stops with the status where Actual Position (AxisName.Md.ActualPosition) deviates from Set Position (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 machine feed value) 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.

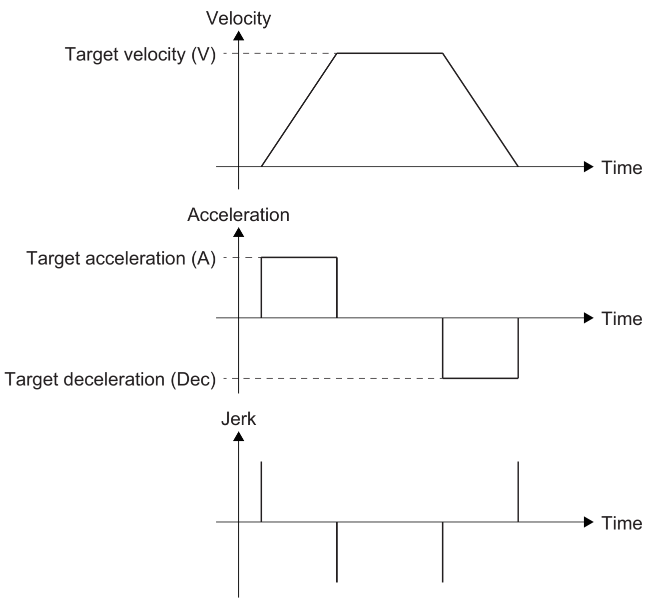
Acceleration/deceleration method

The following methods can be selected for the acceleration/deceleration method. When a value outside the range is set in the acceleration/deceleration method, "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 Acceleration (Acceleration), Deceleration (Deceleration), and Jerk (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 velocity.

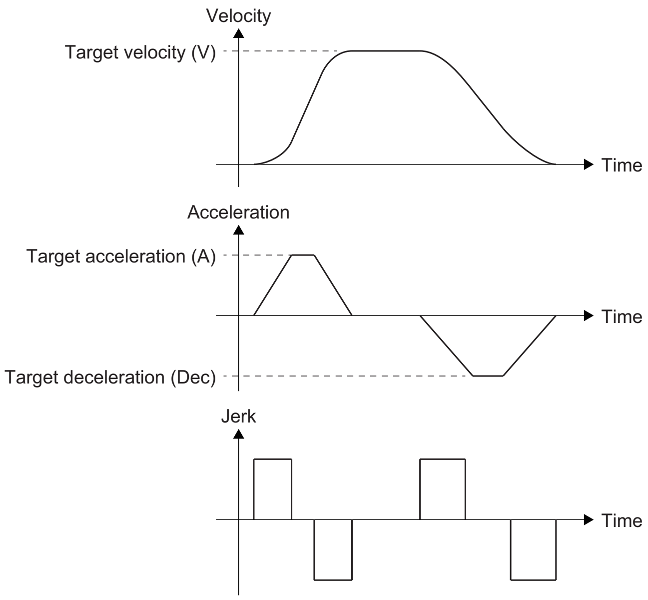
■ Trapezoidal acceleration/deceleration

When "0.0" is set in the jerk, the curve is called the trapezoidal acceleration/deceleration. The velocity shows a trapezoidal waveform.



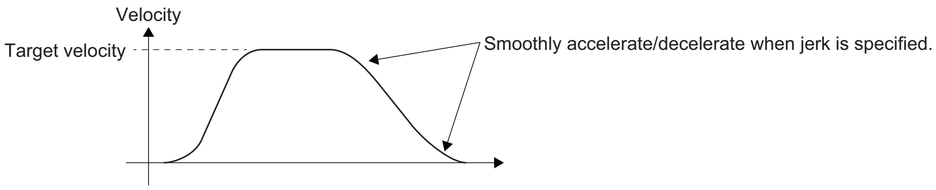
■ Jerk acceleration/deceleration

When a value other than "0.0" is set in the jerk, the curve is called the jerk acceleration/deceleration. The velocity 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 velocity 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

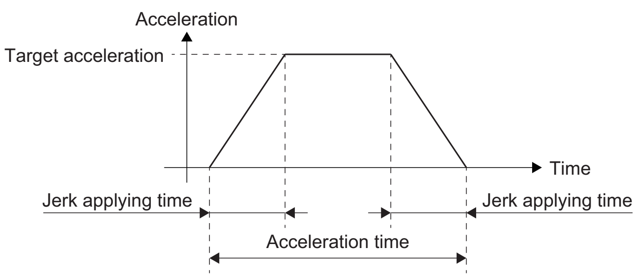
The sum of the time required to reach the target acceleration after startup and the time from the target acceleration to the acceleration 0 at the end of acceleration is called the jerk applying time. The ratio of jerk applying time to acceleration (deceleration) time is called the jerk applying ratio.

The jerk applying time and the jerk applying ratio can be calculated with the following formula.

- Jerk applying time

  Jerk applying time = 2 × Target acceleration ÷ Jerk
- Jerk applying ratio

  Jerk applying ratio = (Jerk applying time ÷ Acceleration time) × 100 [%]



Relevant variables

Axis parameter (AxisName.Pr.)

Variable/Structure name	Name	Details
VelocityLimit_Positive	Positive Direction Velocity Limit Value	Sets the positive direction velocity limit value. For details, refer to the following. ☞ Page 345 Velocity Limit
VelocityLimit_Negative	Negative Direction Velocity Limit Value	Sets the negative direction velocity limit value. For details, refer to the following. ☞ Page 345 Velocity Limit
VelocityLimit_OverOperation	Operation Setting at Velocity Limit Value Exceeded	Sets the operation at velocity limit value exceeded. For details, refer to the following. ☞ Page 345 Velocity Limit
AccelerationLimit	Acceleration Limit Value	Sets the acceleration limit value. • 0.0000, positive numbers from 0.0001 to 2147483647.0 U/s^2*1 *: When "1: Acceleration/deceleration time-fixed method (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. • 0.0000, positive numbers from 0.0001 to 2147483647.0 U/s^2*1 *: When "1: Acceleration/deceleration time-fixed method (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. • 0.0000, positive numbers from 0.0001 to 2147483647.0 U/s^3*1 *: 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: Error (Not Started) (ACCError) • 1: Maximum Acceleration/Deceleration (MaximumAcceleration)
OverrunOperation	Operation Setting at Overrun	Sets the operation performed when the target position is exceeded during positioning operation. • 1: Immediate Stop (ImmediateStop) • 2: Keep Current Acceleration/Deceleration (KeepCurrentAcc)

*1 When a real number less than 0.0001 is set, the value is imported as "0.0000".

Axis monitor data (AxisName.Md)

Variable/Structure name	Name	Details
CommandedAcceleration	Commanded Acceleration	Stores the acceleration specified by users. (Unit: U/s^2) When "1: Acceleration/deceleration time-fixed method (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 Commanded Acceleration (AxisName.Md.CommandedAcceleration) of the configuration axis.
CommandedDeceleration	Commanded Deceleration	Stores the deceleration specified by users. (Unit: U/s^2) When "1: Acceleration/deceleration time-fixed method (mcFixedTime)" is set in the acceleration/ deceleration method, it is an illegal value. During axes group operation, stores "0.0" in Commanded Deceleration (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 Target Acceleration (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 Target Deceleration (AxisName.Md.TargetDeceleration) of the configuration axis.
SetAcceleration	Set Acceleration	Stores the command output acceleration. It is the acceleration/deceleration calculated from the difference of set velocity. (Unit: U/s^2)*1 It indicates accelerating or decelerating by a sign. "0.0" is stored when the target velocity is reached. During axes group operation, stores the set acceleration of the configuration axis. • 0.0: During stop or fixed-speed. • Plus sign: During acceleration • Minus sign: During deceleration

Variable/Structure name	Name	Details
CommandedJerk	Commanded Jerk	Stores the jerk specified by users. (Unit: U/s^3) During axes group operation, stores "0.0" in Commanded Jerk (AxisName.Md.CommandedJerk) of the configuration axis.
InVelocity	Target Velocity Reached	Displays whether the target velocity has been reached or not. During axes group operation, stores FALSE in Target Velocity Reached (AxisName.Md.InVelocity) of the configuration axis. • FALSE: Not reached • TRUE: Reached
AutoDeceleration	Automatically Decelerating	Displays the status of auto deceleration processing. TRUE is stored while the auto deceleration processing is executed. When the multiple 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 Automatically Decelerating (AxisName.Md.AutoDeceleration) of the configuration axis. • FALSE: Not during auto deceleration • TRUE: During auto deceleration

*1 The Values to be stored include the error, because the floating point type error occurs.

Axes group parameter (AxesGroupName.Pr)

Variable/Structure name	Name	Details
VelocityLimit	Velocity Limit Value	Sets the velocity limit value. For details, refer to the following. ☞ Page 345 Velocity Limit
AccelerationLimit	Acceleration Limit Value	Sets the acceleration limit value. • 0.0000, positive numbers from 0.0001 to 2147483647.0 U/s^2*1 *: When "1: Acceleration/deceleration time-fixed method (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. • 0.0000, positive numbers from 0.0001 to 2147483647.0 U/s^2*1 *: When "1: Acceleration/deceleration time-fixed method (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. • 0.0000, positive numbers from 0.0001 to 2147483647.0 U/s^3*1 • When "0.0000" is set, it does not limit the jerk.
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: Error (Not Started) (ACCError) • 1: Maximum Acceleration/Deceleration (MaximumAcceleration)
OverrunOperation	Operation Setting at Overrun	Sets the operation performed when the target position is exceeded during operation. • 1: Immediate Stop (ImmediateStop)

*1 When a real number less than 0.0001 is set, the value is imported as "0.0000".

Axes group monitor data (AxesGroupName.Md.)

Variable/Structure name	Name	Details
CommandedAcceleration	Commanded Acceleration	Stores the acceleration specified by users. (Unit: U/s^2) When "1: Acceleration/deceleration time-fixed method (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: Acceleration/deceleration time-fixed method (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)
SetAcceleration	Set Acceleration	Stores the command output acceleration. It is the acceleration/deceleration calculated from the difference of set velocity. (Unit: U/s^2)*1 It indicates accelerating or decelerating by a sign. "0.0" is stored when the target velocity is reached. • 0.0: During stop or fixed-speed. • Plus sign: During acceleration • Minus sign: During deceleration
CommandedJerk	Commanded Jerk	Stores the jerk specified by users. (Unit: U/s^3)
InVelocity	Target Velocity Reached	Displays whether the target velocity has been reached or not. • FALSE: Not reached • TRUE: Reached
AutoDeceleration	Automatically Decelerating	Displays the status of auto deceleration processing. "TRUE" is stored while the auto deceleration processing is executed. When the multiple 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. • FALSE: Not automatically decelerating • TRUE: Automatically decelerating

*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 "bit0 to 2" of Options (Options) of the motion FB.

The details of the bits are shown below.

Bit	Function description
0 to 2	Acceleration/deceleration method setting • 0: Acceleration/deceleration specification method (mcAccDec) • 1: Acceleration/deceleration time-fixed method (mcFixedTime) For details, refer to the following. ☞ Page 317 Acceleration/deceleration Processing Function
3 to 31	The function differs depending on the FB.

Whether a method can be selected or not differs depending on the FB. For details, refer to the following.

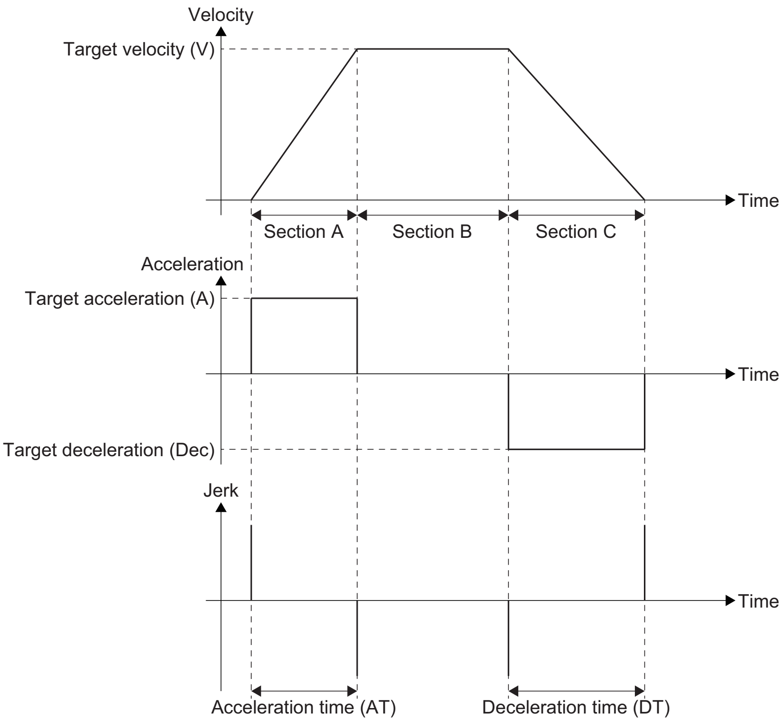
📖MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

Acceleration/deceleration method

Acceleration/deceleration specification method

When "0: Acceleration/deceleration specification method (mcAccDec)" is selected in the Acceleration/deceleration method setting (Options (Options): Bit 0 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.	×	○

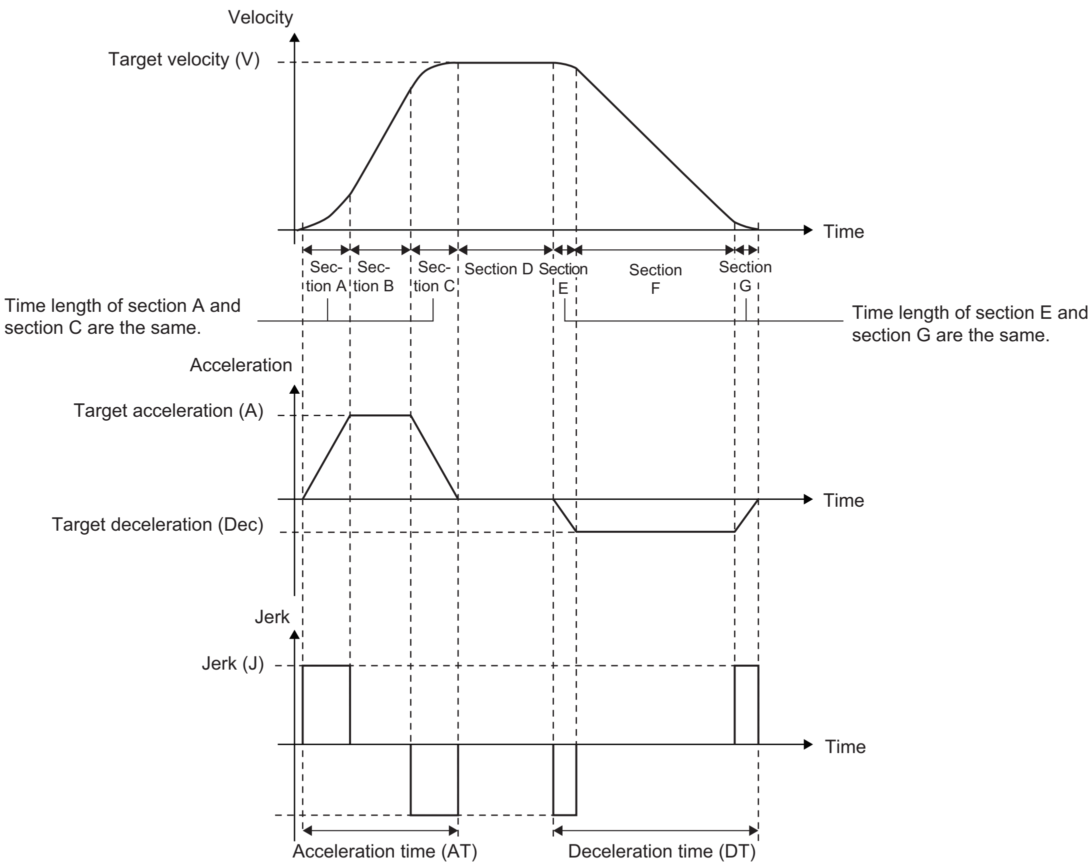
- Acceleration/deceleration time parameter calculation

○: 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 velocity 0.0 to the target velocity	V ÷ A*1	○	×
	Deceleration section time	DT	Time to decelerate from the target velocity to the velocity 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].

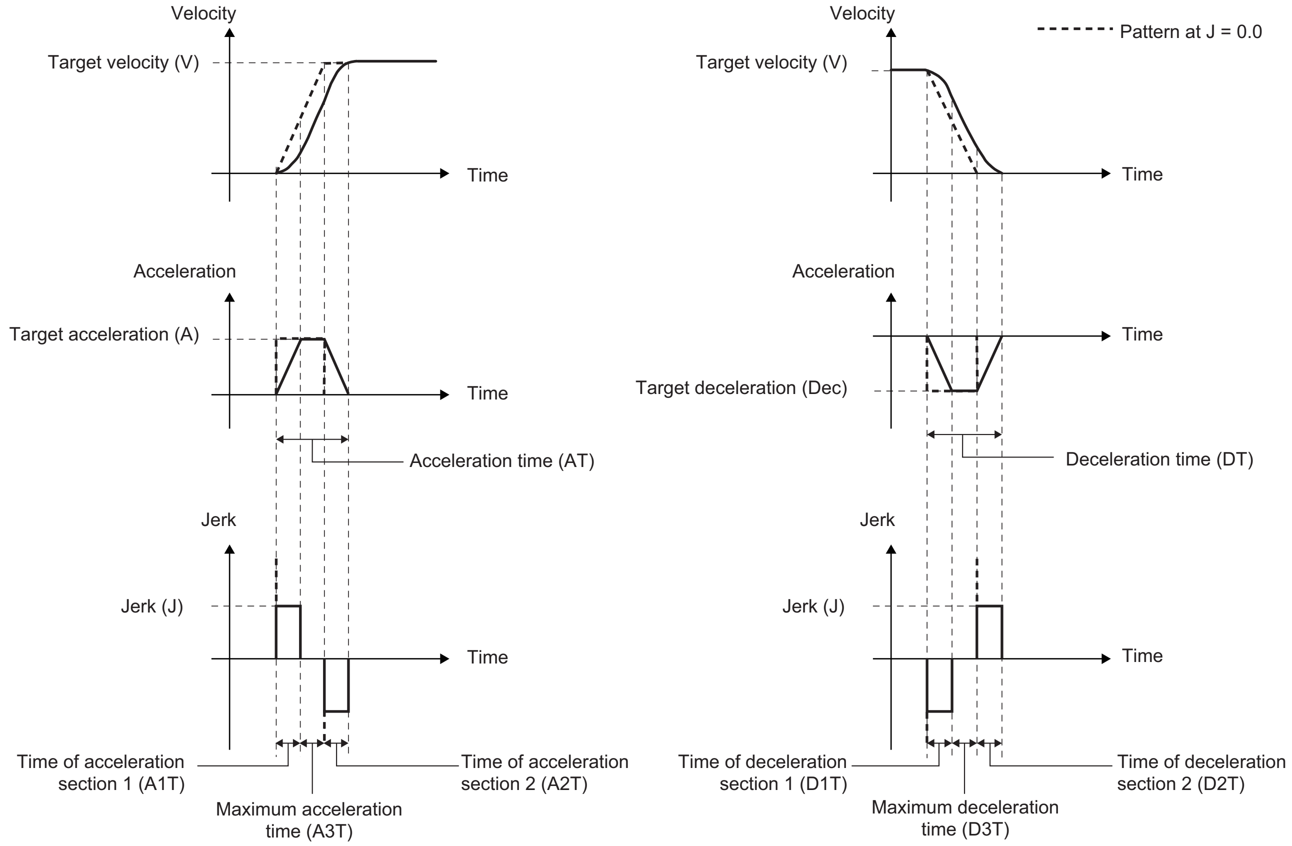
■ Jerk acceleration/deceleration



○: 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*1	A	—	—	○	×
	Target deceleration*1	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)*3	○	×
	Acceleration time*2	AT	Time from velocity 0.0 to the target velocity	A1T + A2T + A3T = (V ÷ A) + (A ÷ J)*3	○	×
	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)*3	×	○
	Deceleration time*2	DT	Time to decelerate from the target velocity to velocity 0.0	D1T + D2T + D3T = (V ÷ Dec) + (Dec ÷ J)*3	×	○

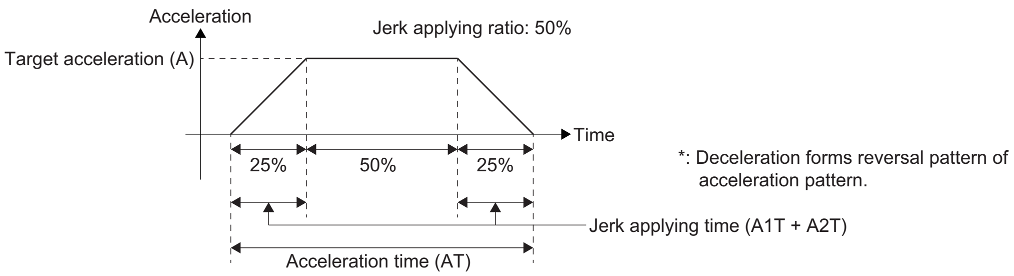
*1 Target acceleration/target deceleration may be adjusted by acceleration/deceleration waveform adjustment. For details, refer to the following.
☞ Page 329 Acceleration/deceleration waveform adjustment when specifying jerk
*2 Acceleration time and deceleration time may be limited. For details, refer to the following.
☞ Page 334 Limit of acceleration time/deceleration time
*3 This is the calculation formula when unit of V is [/s]. When other than [U/s], calculate by converting into [U/s].

- Jerk calculation

  When the velocity 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.

When calculating the jerk with the following values set.

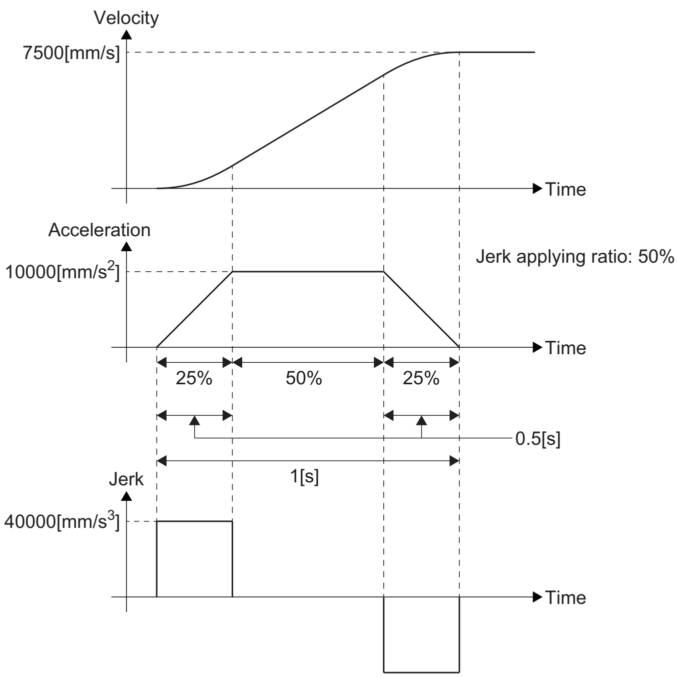
Item	Setting value
Velocity	7500 [mm/s]
Acceleration	10000 [mm/s^2]
Jerk applying ratio	0.5 (50%)

$$\text{Jerk (J)} = \frac{(2 - 0.5) \times 10000^2}{0.5 \times 7500} = 40000[\text{mm/s}^3]$$

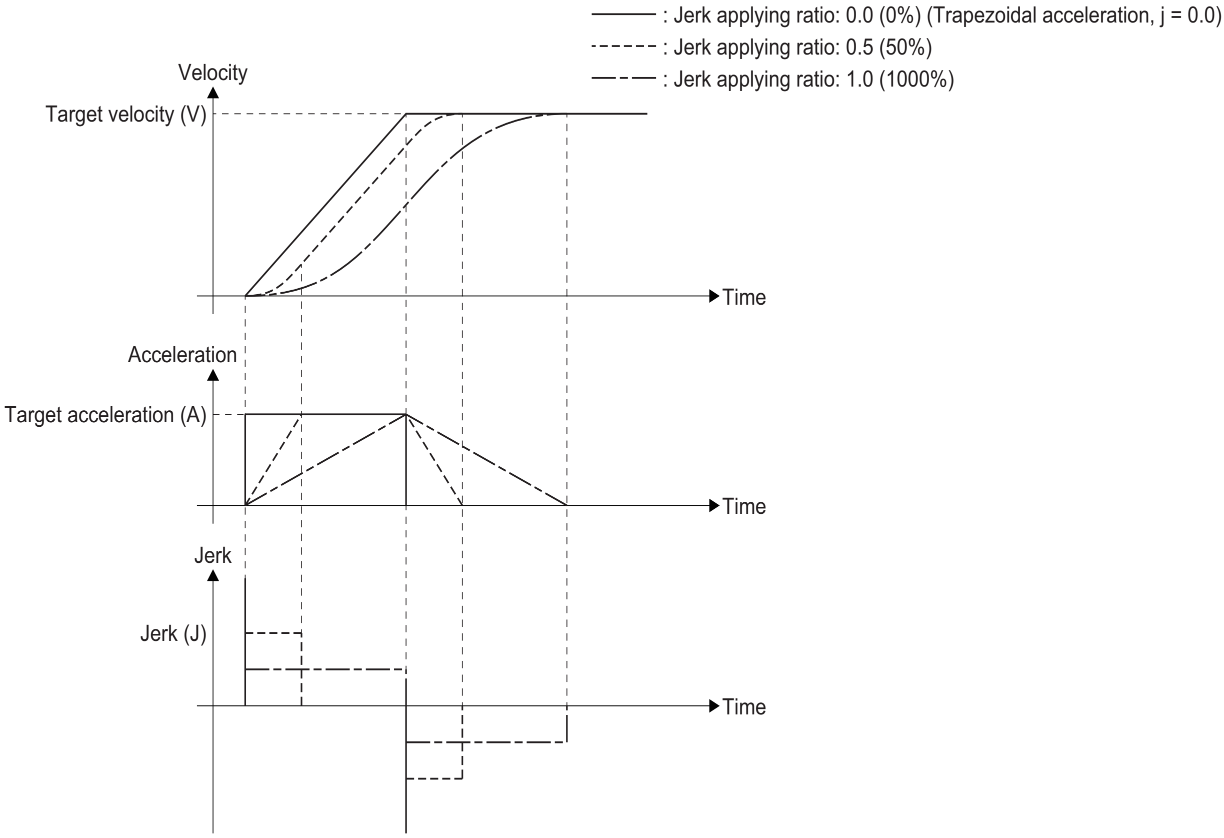
The acceleration time can be calculated when the velocity, the acceleration, and the jerk are fixed. For details, refer to the following.

☞ Page 323 Jerk acceleration/deceleration

$$\text{Acceleration time (AT)} = \frac{\text{Speed (AT)}}{\text{Acceleration (A)}} + \frac{\text{Acceleration (A)}}{\text{Jerk (J)}} = \frac{7500}{10000} + \frac{10000}{40000} = 1[\text{s}]$$



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 velocity 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 velocity may be reached before the target acceleration is reached.

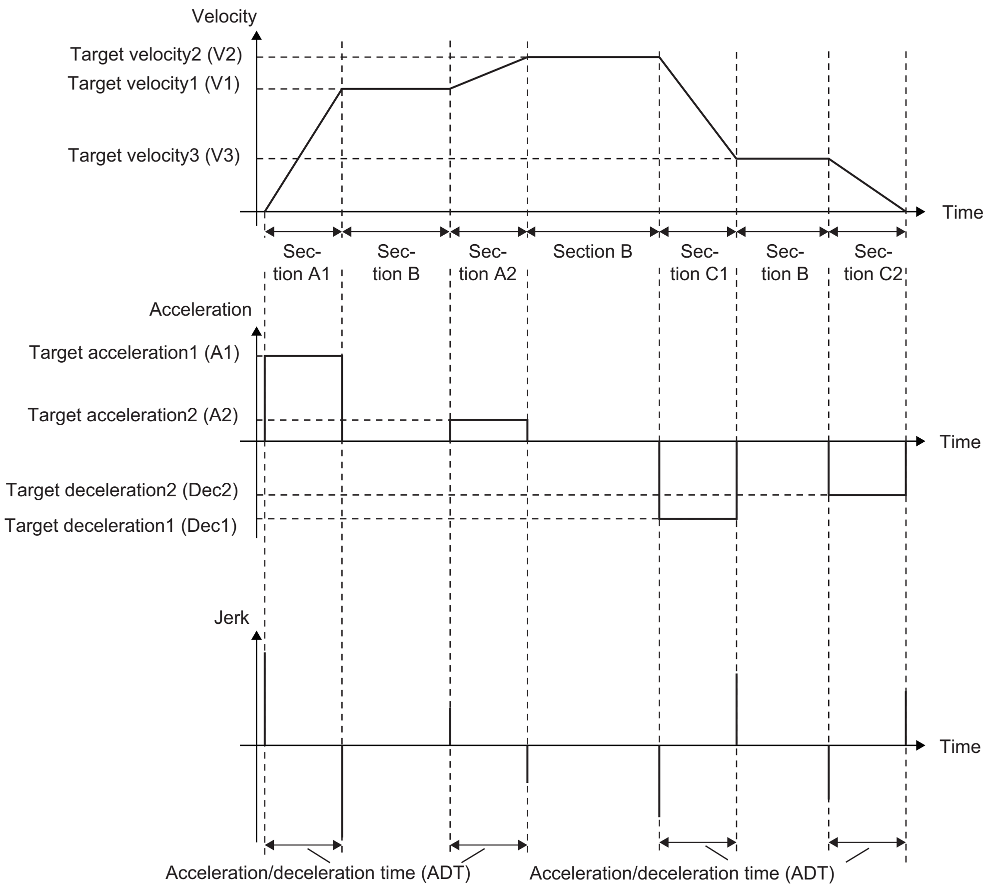
For details, refer to the following.

☞ Page 329 Acceleration/deceleration waveform adjustment when specifying jerk

Acceleration/deceleration time-fixed method

When "1: Acceleration/deceleration time-fixed method (mcFixedTime)" is selected in the Acceleration/deceleration method setting (Options (Options): Bit 0 to 2) of the FB, the acceleration/deceleration time is set in Acceleration (Acceleration), Deceleration (Deceleration), and Jerk (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*1	ADT	—	—	○	○
Calculated value	Current velocity	Vnow	The set velocity at target acceleration and target deceleration calculation	—	○	○
	Target acceleration*2	A	Acceleration from the current velocity to the target velocity*3	(V - Vnow)÷ ADT*4	○	×
	Target deceleration*2	Dec	Deceleration to decelerate from the current velocity to the target velocity*3	(V - Vnow)÷ ADT*4	×	○

*1 The deceleration time at automatic deceleration may be longer than the specified acceleration/deceleration time (ADT) due to the compensation of the remaining distance.

*2 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.

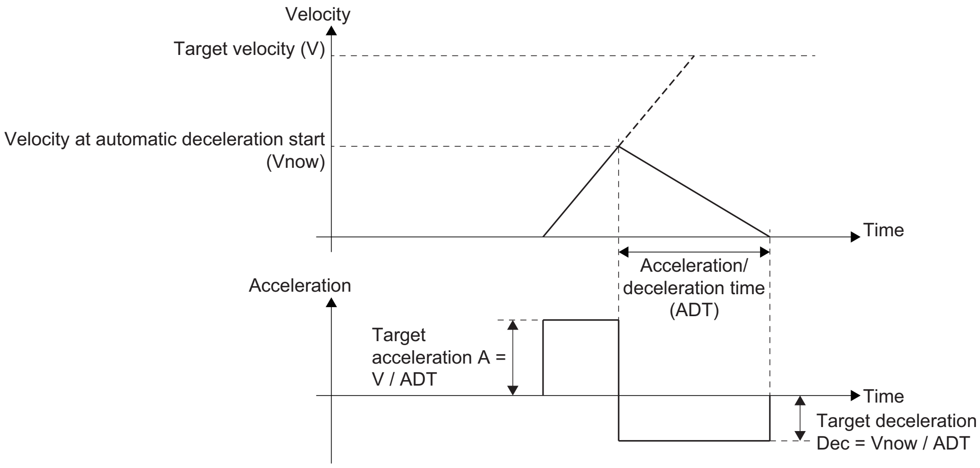
☞ Page 332 Acceleration limit value/deceleration limit value

*3 The target velocity will be "0.0" at stop.

*4 This is the calculation formula when unit of V is [/s]. When other than [U/s], calculate by converting into [U/s].

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.

Acceleration/deceleration	Condition
For acceleration	$V \geq \frac{A^2}{J}$
For deceleration	$V \geq \frac{Dec^2}{J}$

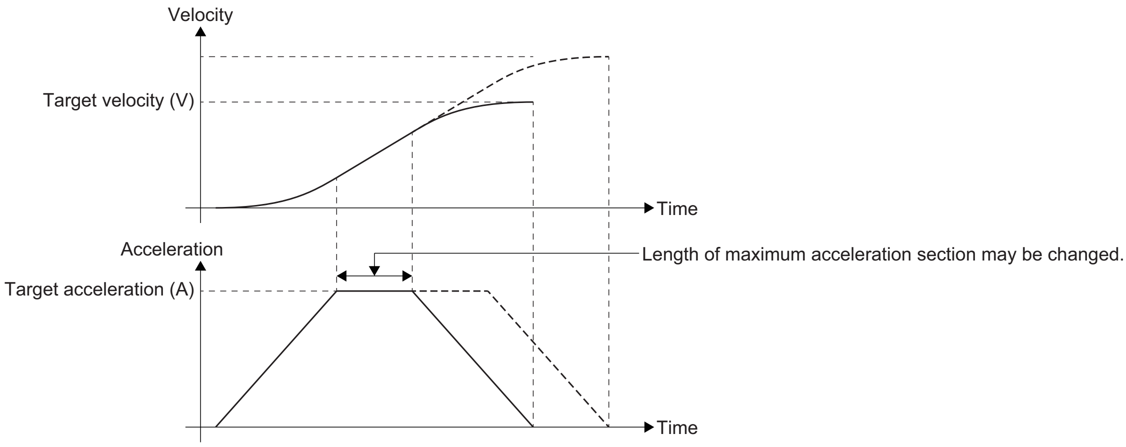
When the condition is satisfied

The target acceleration (deceleration) can be reached.

The maximum acceleration section time and the maximum deceleration section time are adjusted by the velocity, 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 322 Acceleration/deceleration method



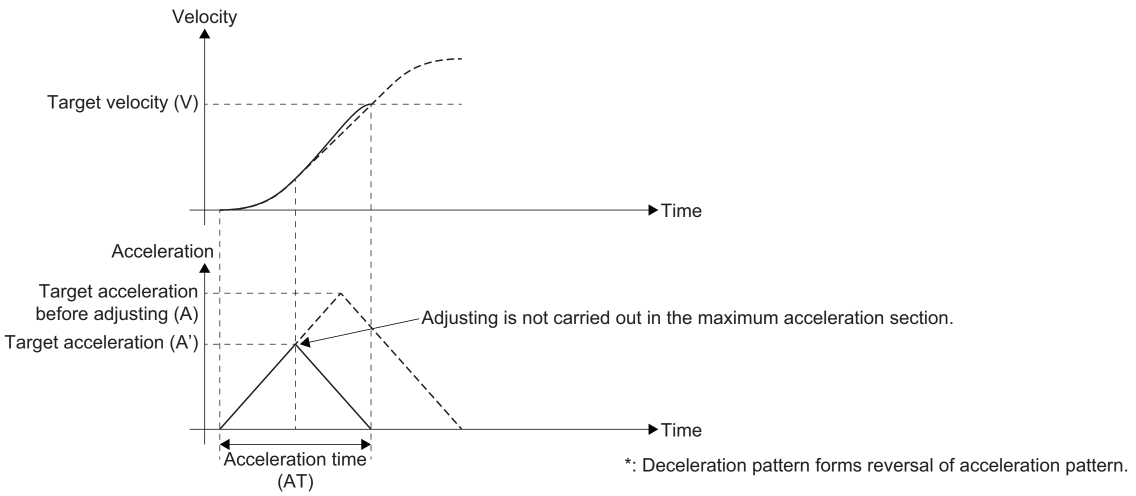
*: Deceleration pattern forms reversal of acceleration pattern.

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: Acceleration/deceleration time-fixed method (mcFixedTime)" is selected in Acceleration/deceleration method setting (Options (Options): Bit 0 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

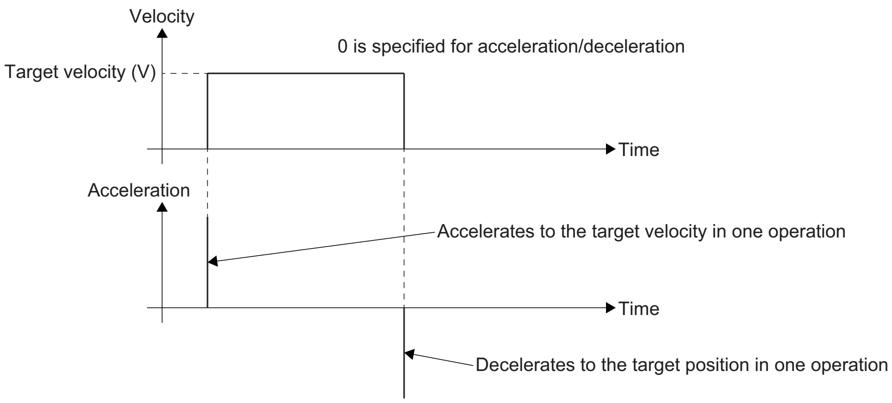
Select the operation when "0.0" is specified for the acceleration and the deceleration in Operation Selection at Start Acceleration/Deceleration 0 (AxisName.Pr.AccelerationZeroBehavior). To operate by the vector velocity at interpolation control, follow Operation Selection at Start Acceleration/Deceleration 0 (AxesGroupName.Pr.AccelerationZeroBehavior.

■ When selecting "-1: Error (Not Started) (ACCError)"

"Acceleration/Deceleration 0 Specified Operation Error at Start (error code: 1A0CH)" occurs, and the axis does not start.

■ When selecting "1: Maximum Acceleration/Deceleration (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: Maximum Acceleration/Deceleration (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 (Forced Stop)/MC_GroupStop (Group Forced Stop) 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: Acceleration/deceleration specification method (mcAccDec)" is selected in Acceleration/deceleration method setting (Options (Options): Bit 0 to 2) of the FB, acceleration/deceleration is limited by the acceleration/deceleration limit value. When "1: Acceleration/deceleration time-fixed method (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.

Specify acceleration limit value and deceleration limit value in Acceleration Limit Value (AxisName.Pr.AccelerationLimit) and Deceleration Limit Value (AxisName.Pr.DecelerationLimit). To operate by the vector velocity at interpolation control, follow Acceleration Limit Value (AxesGroupName.Pr.AccelerationLimit) and Deceleration Limit Value (AxesGroupName.Pr.DecelerationLimit).

Operation

This operation is for the FB which has Acceleration (Acceleration) and Deceleration (Deceleration) of input variables.
Acceleration/deceleration is limited at the following timing.

■ At start

If exceeding the acceleration limit value or the deceleration limit value, "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 exceeding the acceleration limit value or the deceleration limit value by changing the acceleration or the deceleration, "Acceleration Limit Value Over Warning (warning code: 0D06H)" or "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, "Acceleration Limit Value Over (error code: 1A38H)" or "Deceleration Limit Value Over (error code: 1A39H)" will occur, and the under-control FB will be interrupted and stopped. For details, refer to the following.

☞ Page 159 Precautions

■ At stop processing

- When decelerating stop by MC_Stop (Forced Stop) / MC_GroupStop (Group Forced Stop)
  When exceeding the deceleration limit value, "Deceleration Limit Value Over (error code: 1A39H)" occurs and the axis stops according to Stop Selection at Stop Cause Occurrence (AxisName(AxesGroupName).Pr.StopMode_General).
- When decelerating stop by stop cause other than MC_Stop (Forced Stop)/MC_GroupStop (Group Forced Stop)
  When exceeding the deceleration limit value, "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 Jerk limit value (JerkLimit). When "0.0" is specified for the jerk limit value, the jerk is not limited.
Specify the jerk limit value in Jerk Limit Value (AxisName.Pr.JerkLimit). To operate by the vector velocity at interpolation control, follow Jerk Limit Value (AxesGroupName.Pr.JerkLimit).

Operation

This operation is for the FB which has Jerk (Jerk) of input variables.
Jerk is limited at the following timing.

■ At start

If the specified jerk at start exceed the jerk limit value, "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, "Jerk Limit Value Over (error code: 1A3AH)" will occur, and the under-control FB 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 8400.0 [s].

When "0: Acceleration/deceleration specification method (mcAccDec)" is selected in Acceleration/deceleration method setting (Options (Options): Bit 0 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 322 Acceleration/deceleration specification method

Operation

■ At start

The operation at start with Acceleration/deceleration method setting (Options (Options): Bit 0 to 2) is as follows.

- When "0: Acceleration/deceleration specification method (mcAccDec)"
  If the acceleration time or the deceleration time exceeds 8400.0 [s], "Acceleration Time Over at Start (error code: 1A0FH)" or "Deceleration Time Over at Start (error code: 1A10H)" will occur, and the axis will not start.
- When "1: Acceleration/deceleration time-fixed method (mcFixedTime)"
  When acceleration time and deceleration time exceeding 8400.0 [s] are specified, "Out of Acceleration/Deceleration Time Range (error code: 1A0DH)" occurs, and the axis will not start.

■ At control change

Limits time from the issuance of the control change instruction until Acceleration/deceleration method setting (Options (Options): Bit 0 to 2) as follows.

- When "1: Acceleration/deceleration time-fixed method (mcFixedTime)"
  Carrying out a velocity change, an acceleration change, or a deceleration change may exceed 8400.0 [s]. 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 operates as below.

Warning	Operation
Acceleration Time Limit Over Warning (warning code: 0D04H)"	changes to the velocity and deceleration are not accepted. It keeps operating with the previous velocity and deceleration.
Deceleration Time Limit Over Warning (warning code: 0D05H)"	changes to the velocity and deceleration are not accepted. It keeps operating with the previous velocity and deceleration.

- When "1: Acceleration/deceleration time-fixed method (mcFixedTime)"
  When the acceleration/deceleration time change exceeding 8400.0 [s] is performed, "Out of Acceleration/Deceleration Time Range Warning (error code: 0D19H)" occurs, and the axis will not accept the acceleration/deceleration time change. When the velocity change is performed at the same time, the axis operates with previous acceleration time and deceleration time. If the velocity change is not performed at the same time, the axis keeps the current acceleration/deceleration operation during accelerating/decelerating.

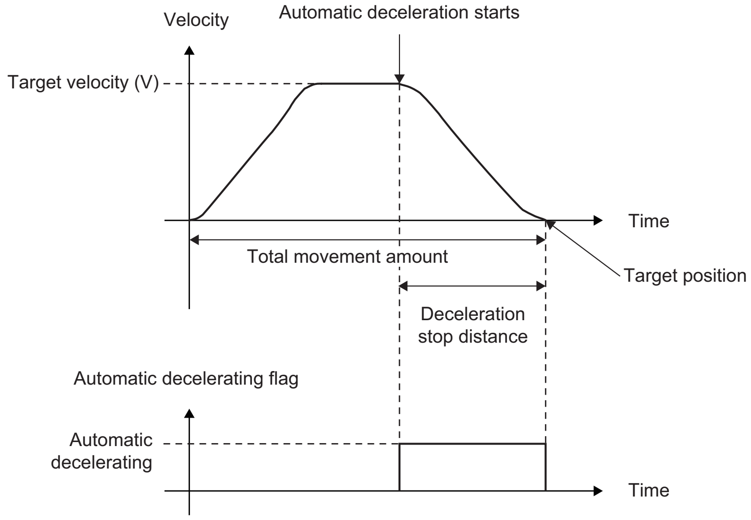
■ When the acceleration time/deceleration time is within one operation cycle

When the acceleration time or the deceleration time is less than one operation cycle due to the specified velocity, acceleration, deceleration, and jerk, the velocity reaches the target velocity within one operation cycle regardless of the jerk.

Automatic deceleration

Decelerating to a velocity of "0.0" to stop at the specified target position is called automatic deceleration.
During automatic deceleration, Automatically Decelerating (AxisName.Md.AutoDeceleration) is TRUE.
When automatic deceleration is performed, the deceleration may go ahead or behind the target deceleration. Therefore, Set Acceleration (AxisName.Md.SetAcceleration) may exceed Target Deceleration (AxisName.Md.TargetDeceleration).
Besides, the current velocity 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, Set Acceleration (AxisName.Md.SetAcceleration) will be "0.0".

Item	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.

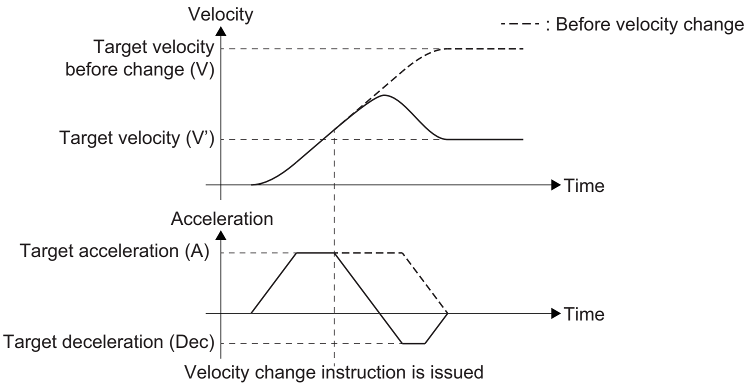
Deceleration stop distance > Total movement amount - Cumulative movement amount - Next Δ movement amount

- 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.



Item	Description
Velocity change	Indicates the following operations • For retrigger/continuous update the velocity input variables • For executing the velocity override factor change
Acceleration change	Indicates the following operations • For retrigger/continuous update the acceleration input variables • For executing the acceleration override factor change
Deceleration change	Indicates the following operations • For retrigger/continuous update the deceleration input variables • For executing the acceleration override factor 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.

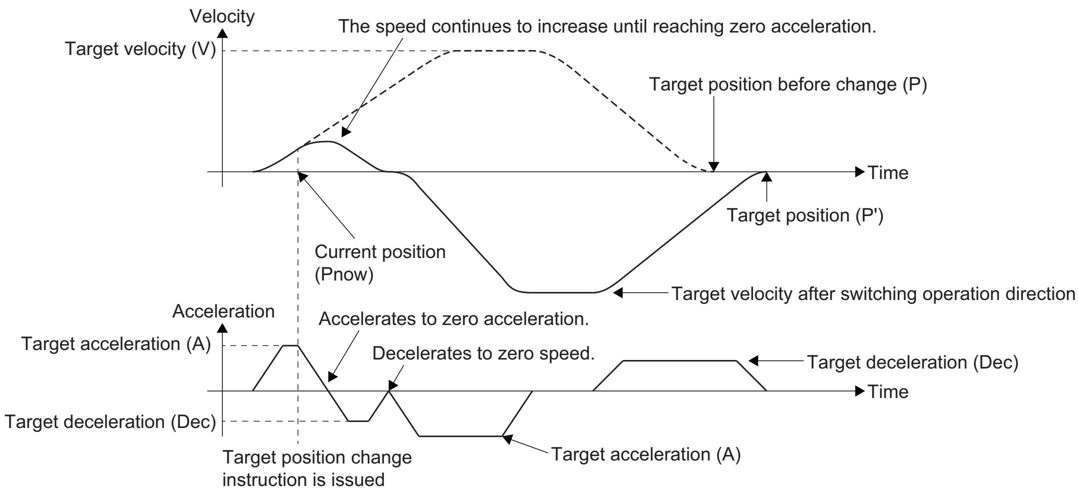
☞ Page 376 Target position/movement distance change

Item	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 velocity 0. When the velocity 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 379 Set velocity change

Item	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.

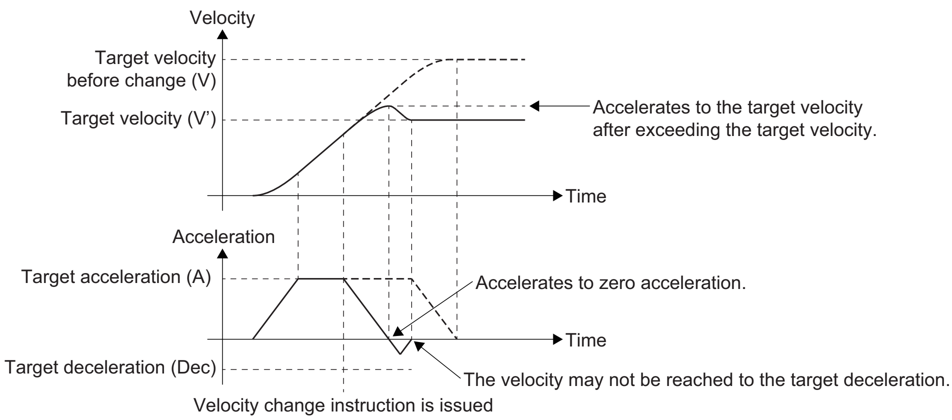
☞ Page 329 Acceleration/deceleration waveform adjustment when specifying jerk

■ When exceeding the target acceleration

If velocity 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

When "0: Acceleration/deceleration specification method (mcAccDec)" is selected in Acceleration/deceleration method setting (Options (Options): Bit 0 to 2) of the FB, acceleration change is performed.

Item	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 329 Acceleration/deceleration waveform adjustment when specifying jerk

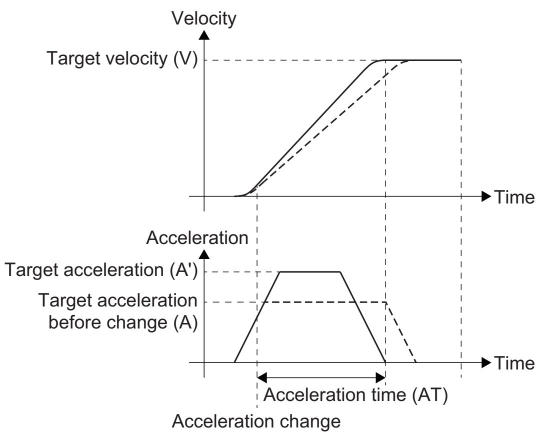
Also, when velocity change and acceleration change are simultaneously performed, the target acceleration may not be reached because the target velocity is given priority.

■ Acceleration change

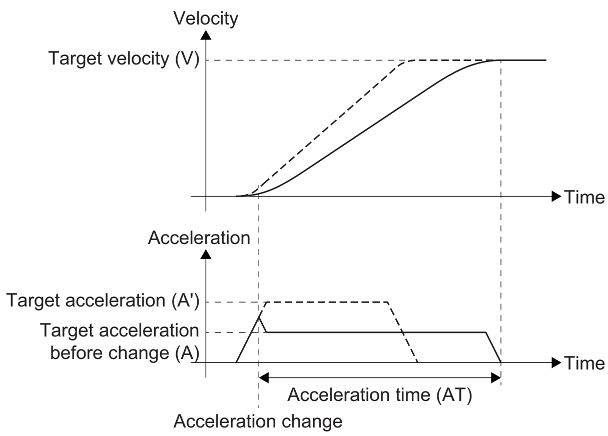
The acceleration change pattern can be classified based on the operation status when acceleration change is performed.

- 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 velocity and the acceleration immediately before the acceleration change.

■When the target acceleration is increased

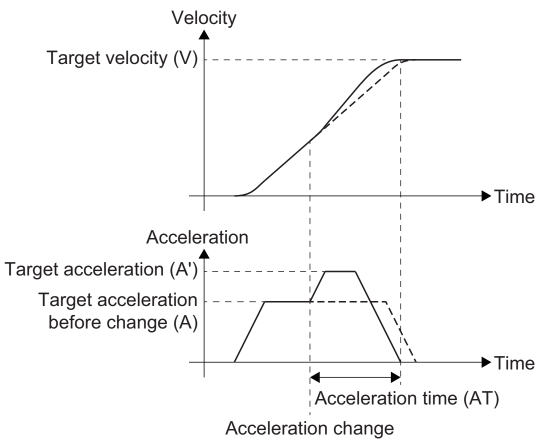


■When the target acceleration is decreased

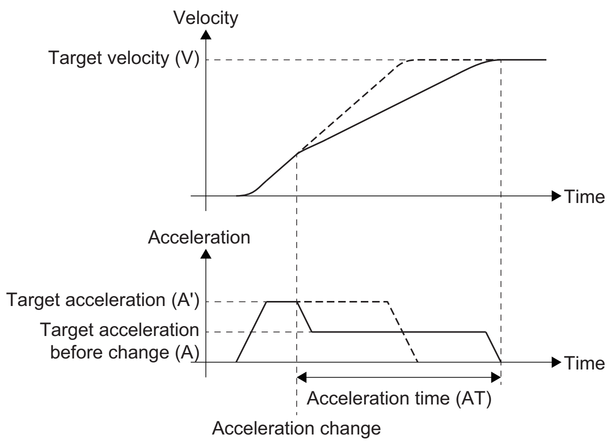


- When executing acceleration change in the maximum acceleration section
  Acceleration is increased or decreased to the target acceleration after change by using jerk, and taking over the velocity and the acceleration immediately before the acceleration change.

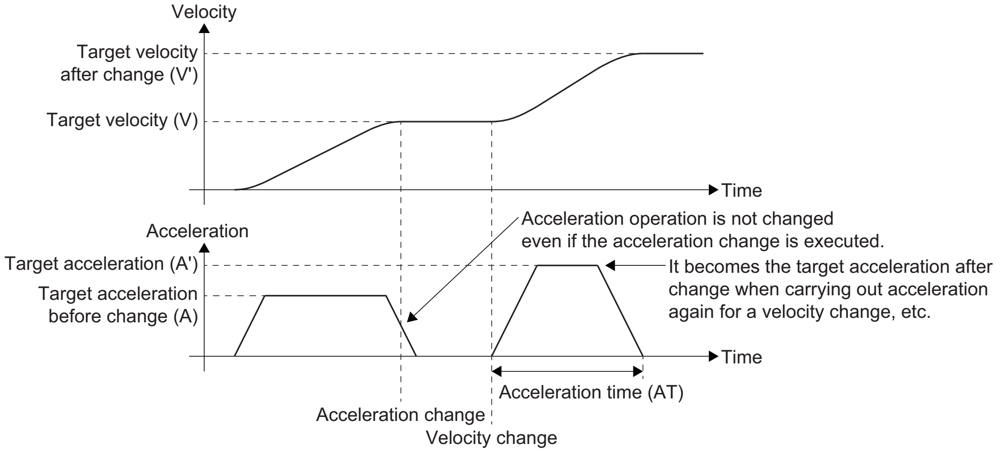
■When the target acceleration is increased



■When the target acceleration is decreased



- 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: Acceleration/deceleration specification method (mcAccDec)" is selected in Acceleration/deceleration method setting (Options (Options): Bit 0 to 2) of the FB, deceleration change can be performed.

Item	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 Operation Setting at Overrun (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. For details, Refer to the following.
  ☞ Page 339 Acceleration 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)

Acceleration/deceleration operation

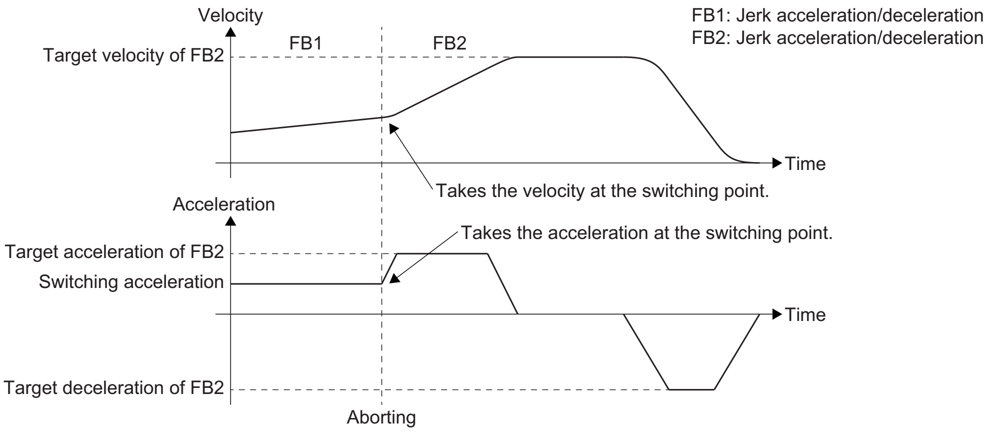
■ When the acceleration/deceleration methods are different between the under-control FB and the next FB

"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 under-control FB and the next FB

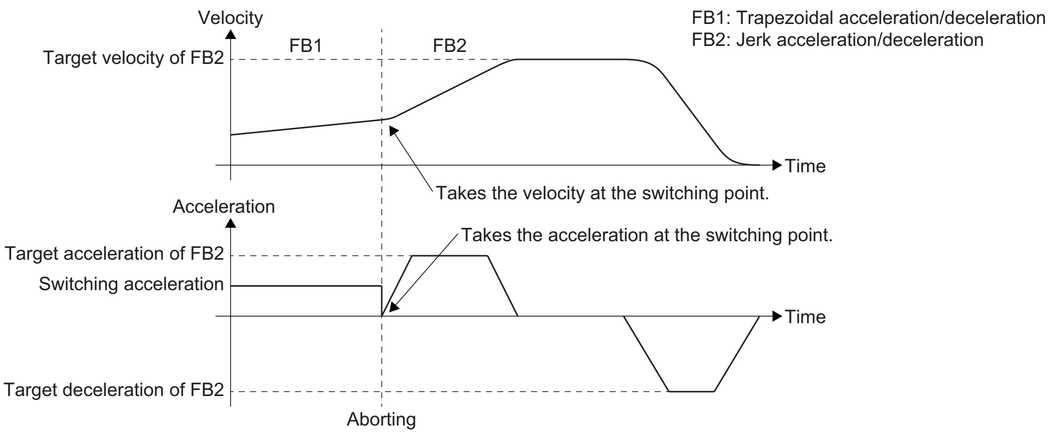
- When switching "from the jerk acceleration/deceleration to the another jerk acceleration/deceleration"

  The axis accelerates/decelerates taking over the velocity and the acceleration (deceleration) at the switching point. However, when the jerk setting values are different between the under-control FB and the next FB, the jerk of the next FB will be the same as the jerk of the under-control FB. 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 under-control 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



Precautions

Precautions when the range is exceeded

Precautions for when the acceleration/deceleration calculated from the acceleration/deceleration and the acceleration time/deceleration time after acceleration override exceeds the range are shown below.

■ When the acceleration calculated from the acceleration and the acceleration time after acceleration override exceeds the range of 0.0000 or 0.0001 to 2147483647.0 [U/s^2]

"Out of Acceleration Range Clamping Warning (warning code: 0D35H)" occurs, and the acceleration will be as follows.

- When "acceleration after acceleration override > 2147483647.0 [U/s^2]"
  The acceleration will be 2147483647.0 [U/s^2].
- When "0.0000 < acceleration after acceleration override < 0.0001"
  The acceleration will be "0.0000". For operation, refer to the following.
  ☞ Page 331 Operation when specifying acceleration/deceleration 0

■ When the deceleration calculated from the deceleration and the deceleration time after acceleration override exceeds the range of 0.0000 or 0.0001 to 2147483647.0 [U/s^2]

"Out of Deceleration Range Clamping Warning (warning code: 0D36H)" occurs, and the deceleration will be as follows.

- When "deceleration after acceleration override > 2147483647.0 [U/s^2]"
  The deceleration clamps at 2147483647.0 [U/s^2].
- When "0.0000 < deceleration after acceleration override < 0.0001"
  The deceleration will be "0.0000". For operation, refer to the following.
  ☞ Page 331 Operation when specifying acceleration/deceleration 0

■ When the acceleration time (deceleration time) after acceleration override exceeds 8400.0 [s]

"Out of Acceleration/Deceleration Time Range Clamping Warning (warning code: 0D37H)" occurs and the acceleration time (deceleration time) will be 8400.0 [s].

Restrictions of commanded acceleration, commanded deceleration, and commanded jerk lower limit values

To perform floating-point operation, the lower limit value of commanded acceleration, commanded deceleration, and commanded jerk are limited by the following restrictions.

■ Lower limit restrictions of acceleration/deceleration

The set velocity may not change as information error occurs if the values of calculation cycle converted velocity and acceleration differ greatly. Therefore, the values of velocity and acceleration need to be set staying within a valid digit (approximately 15 digits) of a number of floating-points (64 bit). When the calculation cycle converted acceleration is smaller than the set velocity, "Out of Operation Cycle Converted Acceleration Range Warning (warning code: 0D38H)" occurs. Similarly, "Out of Operation Cycle Converted Deceleration Range Warning (warning code: 0D39H)" occurs for deceleration. Check the set velocity. If it is not changed, either increasing the acceleration and deceleration or extending the operation cycle is needed.

Ex.

When control change is performed with the following settings during fixed-speed and velocity Command Unit (AxisName(AxesGroupName).Pr.Unit_Velocity) is "HFD000300 [/ms]", operation cycle is "0.125 [ms]" and velocity "2499999999.9"

- Velocity (Velocity): 2500000000.0
- Acceleration (Acceleration): 62.0

*: Acceleration/deceleration specification method (Trapezoidal acceleration/deceleration)

Calculation cycle converted velocity upper limit value = 2500000000.0 [pulse/ms] × 0.125 [ms] = 312500000.0 [pulse]
Calculation cycle converted acceleration = 62.0 [pulse/s^2] × (0.125 [ms] ÷ 1000)^2 = 9.6875E-07 [pulse]

In above case, "Out of Operation Cycle Converted Acceleration Range Warning (warning code: 0D38H)" occurs because the calculation cycle converted acceleration lower limit value is 1E-06 [pulse].

■ Lower limit restriction of jerk

The set velocity or set acceleration may not change as information error occurs if the value of jerk to the values of calculation cycle converted velocity and acceleration differ greatly. Therefore, the value of jerk needs to be set staying within a valid digit (approximately 15 digits) of a number of floating-points (64 bit).

When the calculation cycle converted jerk is smaller than the velocity and the acceleration, "Out of Operation Cycle Converted Jerk Range Warning (warning code: 0D3AH)" occurs. Check the set velocity or set acceleration. If they are not changed, either increasing the jerk or extending the operation cycle is needed.

Ex.

When started with the following settings and Velocity Command Unit (AxisName(AxesGroupName).Pr.Unit_Velocity) is "HFD000300 [/ms]" and operation cycle is "0.125 [ms]"

- Velocity (Velocity): 100000.0
- Acceleration (Acceleration): 30000.0
- Deceleration (Deceleration): 30000.0
- Jerk (Jerk): 20.0

*: The acceleration/deceleration method (jerk acceleration/deceleration)

Calculation cycle converted velocity upper limit value = 100000.0 [pulse/ms] $\times$ 0.125 [ms] = 12500.0 [pulse]
Calculation cycle converted acceleration = 30000.0 [pulse/s^2] $\times$ (0.125 [ms] $\div$ 1000)^2 = 0.00046875 [pulse]
Calculation cycle converted jerk = 20.0 [pulse/s^3] $\times$ (0.125 [ms] $\div$ 1000)^3 = 3.90625E-11 [pulse]

In above case, the calculation cycle converted acceleration lower limit value will be "4.0E-11 [pulse]". Also, "Out of Operation Cycle Converted Jerk Range Warning (warning code: 0D3AH)" occurs because the calculation cycle converted jerk lower limit value will be the same value "4.0E-11 [pulse]".

When the calculation cycle converted jerk is too small, "Jerk Acceleration/Deceleration Disabled Warning (warning code: 0D3BH)" occurs and operates as Trapezoidal acceleration/deceleration.

Ex.

When started with the following settings and Velocity Command Unit (AxisName(AxesGroupName).Pr.Unit_Velocity) is "H00000300 [/s]" and operation cycle is "31.25 [μs]"

- Velocity (Velocity): 0.4
- Acceleration (Acceleration): 0.005
- Deceleration (Deceleration): 0.005

In above case, "Jerk Acceleration/Deceleration Disabled Warning (warning code: 0D3BH)" occurs because the calculation cycle converted jerk is "3.05176E-18".

9.2 Velocity Limit

The "velocity limit function" limits the set velocity to a value within the velocity limit value setting range when the set velocity during control exceeds the velocity limit value. By limiting the set velocity within the preset velocity limit value setting range, even if an incorrect command or unexpected operation occurs, damage to the machine can be prevented.

Relevant variables

Axis parameter (AxisName.Pr.)

Variable/Structure name	Name	Details
VelocityLimit_Positive	Positive Direction Velocity Limit Value	Sets the address increasing direction velocity limit value for each axis.
VelocityLimit_Negative	Negative Direction Velocity Limit Value	Sets the address decreasing direction velocity limit value for each axis.
VelocityLimit_OverOperation	Operation Setting at Velocity Limit Value Exceeded	Sets the operation when the velocity limit value during control is exceeded. • 0: Ignore (Ignore) • 3: Immediate Stop (ImmediateStop)

Axes group parameter (AxesGroupName.Pr.)

Variable/Structure name	Name	Details
VelocityLimit	Velocity Limit Value	Sets the velocity limit value of the axes group.

How to set the velocity limit function

Velocity limit value for each axis

Set Positive Direction Velocity Limit Value (AxisName.Pr.VelocityLimit_Positive) and Negative Direction Velocity Limit Value (AxisName.Pr.VelocityLimit_Negative), for each axis.

Setting item	Setting range	Setting details
Positive Direction Velocity Limit Value (AxisName.Pr.VelocityLimit_Positive)	0.0001 ≤ Setting value ≤ 2500000000.0	Sets the velocity limit value (the maximum velocity during control) for the positive direction operation.
Negative Direction Velocity Limit Value (AxisName.Pr.VelocityLimit_Negative)		Sets the velocity limit value (the maximum velocity during control) for the negative direction operation.

Velocity limit value for each axes group

Set Velocity Limit Value (AxesGroupName.Pr.VelocityLimit), for each axes group.

Setting item	Setting range	Setting details
Velocity Limit Value (AxesGroupName.Pr.VelocityLimit)	0.0001 ≤ Setting value ≤ 2500000000.0	Sets the velocity limit value (the maximum velocity during control) of the axes group.

Operation when the velocity limit value is exceeded

The operation when specifying the velocity exceeding the velocity limit value differs depending on single axis control, single axis synchronous control and interpolation control.

For single axis operation

Executes the velocity 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 velocity limit value (Positive Direction Velocity Limit Value (AxisName.Pr.VelocityLimit_Positive) for the positive direction and Negative Direction Velocity Limit Value (AxisName.Pr.VelocityLimit_Negative) for the negative direction) of each axis, "Velocity Limit Value Over (error code: 1A07H)" will occur, and the axis will not start.

■ During control

The velocity limit is not executed.

■ During control (At velocity change)

If exceeding the velocity limit value of each axis is specified by the override function and input label change in execution, "Velocity Limit Value Over Warning (warning code: 0D03H)" will occur, and the operation continues at the velocity before change.

For details on the override function and input label change in execution, refer to the following.

☞ Page 349 Override Function
☞ Page 375 Input Variable Change in Execution

For single axis synchronous control

Executes the velocity limit to the set velocity of the slave axis.

■ At start

The velocity limit is not executed.

■ During control

If the set velocity exceeds the velocity limit value (Positive Direction Velocity Limit Value (AxisName.Pr.VelocityLimit_Positive) for the positive direction and Negative Direction Velocity Limit Value (AxisName.Pr.VelocityLimit_Negative) for the negative direction) of each axis, follow Operation Setting at Velocity Limit Value Exceeded (AxisName.Pr.VelocityLimit_OverOperation).

If the absolute value of the set velocity exceeds the maximum velocity, "Velocity Range Over during Controlling (error code: 1AE8H)" will occur and stop the operation.

For interpolation control

For the velocity limit at the interpolation control, Positive Direction Velocity Limit Value (AxisName.Pr.VelocityLimit_Positive), Negative Direction Velocity Limit Value (AxisName.Pr.VelocityLimit_Negative), and Velocity Limit Value (AxesGroupName.Pr.VelocityLimit) are used.

The operation when the target velocity exceeds the velocity limit value is as follows.

■ At start

Variable to be used	Linear interpolation control			Circular interpolation control
	Vector velocity specification	Long axis velocity specification	Reference axis velocity specification	
Positive Direction Velocity Limit Value (AxisName.Pr.VelocityLimit_Positive) or Negative Direction Velocity Limit Value (AxisName.Pr.VelocityLimit_Negative)*1	Velocity is not restricted.	Long axis velocity is restricted by velocity limit value of long axis.*2	Reference axis velocity is restricted by velocity limit value of reference axis.*2	Velocity is not restricted.
Velocity Limit Value (AxesGroupName.Pr.VelocityLimit)	Velocity is restricted for vector velocity.*2	Velocity is not restricted.	Velocity is not restricted.	Velocity is restricted for vector velocity.*2

*1 About either Positive Direction Velocity Limit Value (AxisName.Pr.VelocityLimit_Positive) or Negative Direction Velocity Limit Value (AxisName.Pr.VelocityLimit_Negative) is restricted according to the movement direction at the limit value check.

*2 When the velocity limit value is exceeded, it will not start. "Velocity Limit Value Over (error code: 1A07H)" is output.

■ During control

Variable to be used	Linear interpolation control			Circular interpolation control
	Vector velocity specification	Long axis velocity specification	Reference axis velocity specification	
Positive Direction Velocity Limit Value (AxisName.Pr.VelocityLimit_Positive) or Negative Direction Velocity Limit Value (AxisName.Pr.VelocityLimit_Negative)*1	Velocity is restricted for set velocity of each configuration axis. When the velocity limit value is exceeded, it follows Operation Setting at Velocity Limit Value Exceeded (AxisName.Pr.VelocityLimit_OverOperation).			
Velocity Limit Value (AxesGroupName.Pr.VelocityLimit)	Velocity is not restricted.			

*1 About either Positive Direction Velocity Limit Value (AxisName.Pr.VelocityLimit_Positive) or Negative Direction Velocity Limit Value (AxisName.Pr.VelocityLimit_Negative) is restricted according to the movement direction at the limit value check.

■ During control (At velocity change)

Variable to be used	Linear interpolation control			Circular interpolation control
	Vector velocity specification	Long axis velocity specification	Reference axis velocity specification	
Positive Direction Velocity Limit Value (AxisName.Pr.VelocityLimit_Positive) or Negative Direction Velocity Limit Value (AxisName.Pr.VelocityLimit_Negative)*1	Velocity is not restricted.	Long axis velocity is restricted by velocity limit value of long axis.*2	Reference axis velocity is restricted by velocity limit value of reference axis. *2	Velocity is not restricted.
Velocity Limit Value (AxesGroupName.Pr.VelocityLimit)	*2	Velocity is not restricted.	Velocity is not restricted.	*2

*1 About either Positive Direction Velocity Limit Value (AxisName.Pr.VelocityLimit_Positive) or Negative Direction Velocity Limit Value (AxisName.Pr.VelocityLimit_Negative) is restricted according to the movement direction at the limit value check.

*2 When the velocity limit value is exceeded, the velocity change will not accepted. "Velocity Limit Value Over Warning (warning code: 0D03H)" is output.

Operation setting at velocity limit value exceeded

The operation when the velocity of any axis exceeds the velocity limit value during control can be specified in Operation Setting at Velocity Limit Value Exceeded (AxisName.Pr.VelocityLimit_OverOperation).

The following two types can be selected for the operation when the velocity exceeds the velocity limit value.

- 0: Ignore (Ignore)
- 3: Immediate Stop (ImmediateStop)

When the multiple axes exceed the limit value, the priority is "3: Immediate Stop (ImmediateStop)" > "0: Ignore (Ignore)".

Control continuation

When "0: Ignore (Ignore)" is specified in Operation Setting at Velocity Limit Value Exceeded (AxisName.Pr.VelocityLimit_OverOperation), "Velocity Limit Value Over Warning (warning code: 0D03)" occurs at the time of detecting the set velocity exceeding the velocity limit value, and the command exceeding the velocity limit value is output as it is.

Control stop

When "3: Immediate Stop (ImmediateStop)" is specified in Operation Setting at Velocity Limit Value Exceeded (AxisName.Pr.VelocityLimit_OverOperation), "Velocity Limit Value Over during Controlling (error code: 1A36H)" occurs at the time of detecting the set velocity exceeding the velocity 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.
- There are two methods of changing the override factor, the method using the dedicated FB and the method using the control data.

Change method	Description
Change using a dedicated FB	MC_SetOverride (Override Value Setting), MC_GroupSetOverride (Axes Group Override Value Setting)
Change using control data	Velocity Override Factor (AxisName.Cd.VelocityOverride), Velocity Override Factor (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 factor does not reflect immediately during deceleration stop after stop causes occur. It reflects at the timing which causes other stop causes.

Targets to be changed and corresponding FBs, control data

Targets to be changed			Corresponding FB	Corresponding control data
Velocity	Current value		MC_MoveVelocity (Speed Control) MC_SetOverride (Override Value Setting) MC_GroupSetOverride (Axes Group Override Value Setting)	Velocity Override Factor (AxisName.Cd.VelocityOverride), Velocity Override Factor (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 (Parameter Write). However, it cannot be changed during control.
*2 The maximum value of the limit value cannot be changed.

Relevant variables

Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
VelocityOverride	Velocity Override Factor	Sets the velocity override factor.
AccelerationOverride	Acceleration Override Factor	Sets the acceleration override factor.

Axis control data (AxisName.Cd.)

Variable/Structure name	Name	Details
VelocityOverride	Velocity Override Factor	Sets the velocity override factor.
AccelerationOverride	Acceleration Override Factor	Sets the acceleration override factor.

Axes group monitor data (AxesGroupName.Md.)

Variable/Structure name	Name	Details
VelocityOverride	Velocity Override Factor	Sets the velocity override factor.
AccelerationOverride	Acceleration Override Factor	Sets the acceleration override factor.

Axes group control data (AxesGroupName.Cd.)

Variable/Structure name	Name	Details
VelocityOverride	Velocity Override Factor	Sets the velocity override factor.
AccelerationOverride	Acceleration Override Factor	Sets the acceleration override factor.

Relevant FBs

For details on Motion control FBs, refer to the following.

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

Motion control FB	Name	Description
MC_SetOverride	Override value setting	Changes the target velocity, the target acceleration and the target deceleration of the specified axis.
MC_GroupSetOverride	Axes group override value setting	Changes the target velocity, the target acceleration and the target deceleration of the specified axes group.

How to change control data

Override factor setting range

The following shows each override factor setting range.

■ Axis control data

Control data	Setting range
VelocityOverride (AxisName.Cd.VelocityOverride)*1	0.0 (0%) ≤ Setting value ≤ 10.0 (1000%)
AccelerationOverride (AxisName.Cd.AccelerationOverride)*2	0.01 (1%) ≤ Setting value ≤ 10.0 (1000%)

*1 Registers "velocity override "0.00" in the event history when "0" is set to velocity override.

*2 Even if the acceleration/deceleration method specifying the acceleration/deceleration time, the override factor will be used to the acceleration/deceleration.

■ Axes group control data

Control data	Setting range
VelocityOverride (AxesGroupName.Cd.VelocityOverride)*1	0.0 (0%) ≤ Setting value ≤ 10.0 (1000%)
AccelerationOverride (AxesGroupName.Cd.AccelerationOverride)*2	0.01 (1%) ≤ Setting value ≤ 10.0 (1000%)

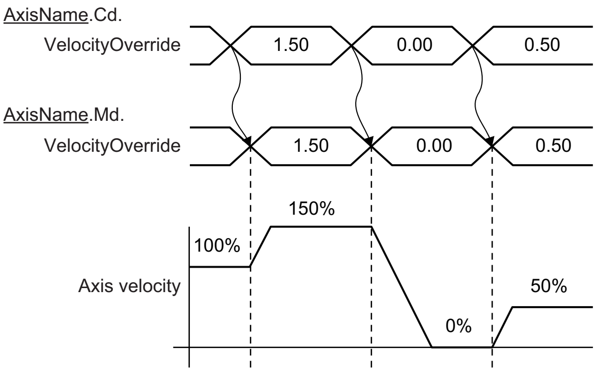
*1 Registers "velocity override "0.00" in the event history when "0" is set to velocity override.

*2 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 Velocity Override Factor (AxisName(AxesGroupName).Cd.VelocityOverride).

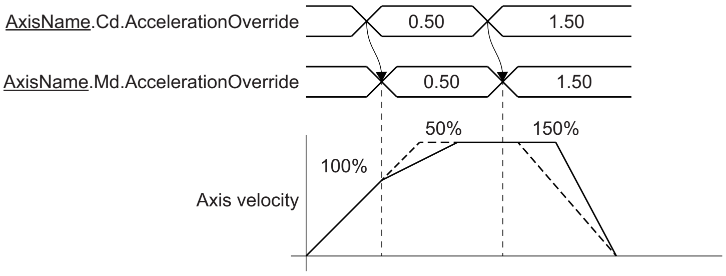
When "0.00" is set in Velocity Override Factor (AxisName(AxesGroupName).Cd.VelocityOverride), the axis is stopped without moving to the "4: Standby (Standstill)" status.



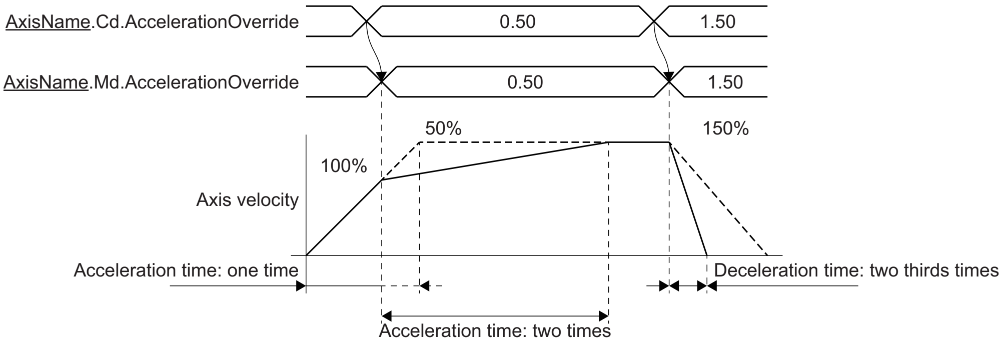
Operation when the acceleration override factor is changed

Change the acceleration/deceleration by changing Acceleration Override Factor (AxisName(AxesGroupName).Cd.AccelerationOverride).

■ Acceleration/deceleration method specifying acceleration/deceleration



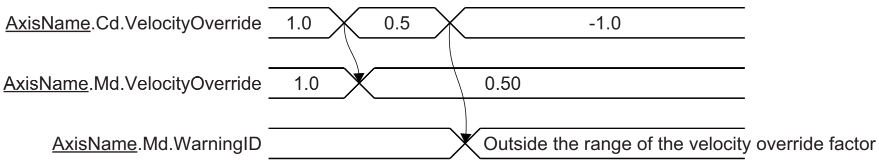
■ Acceleration/deceleration method specifying acceleration/deceleration time



Operation when a value outside the override factor setting range is set

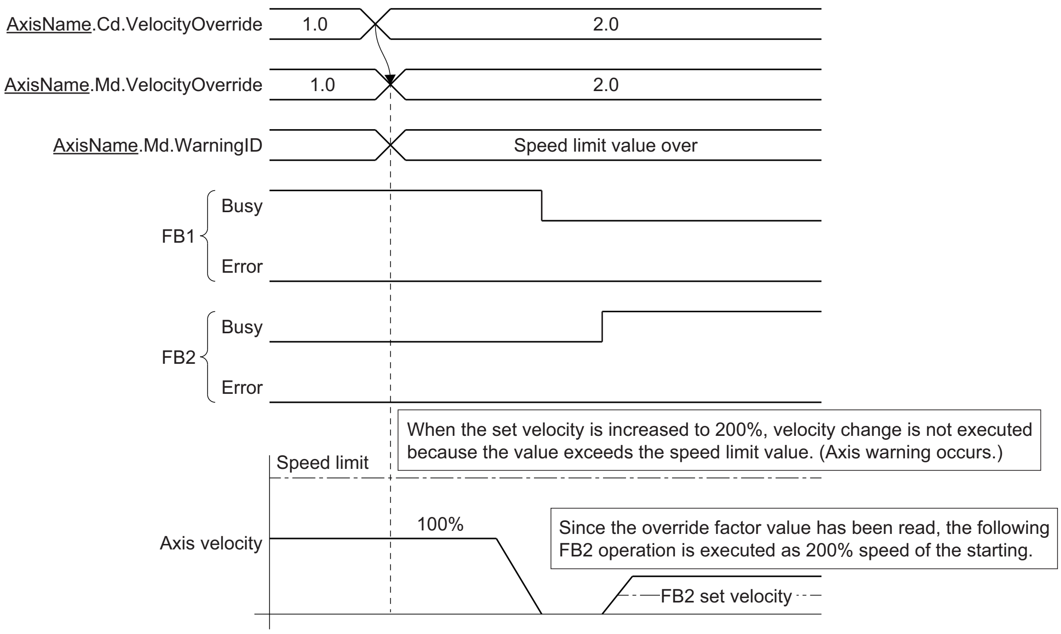
If a value outside the setting range is set in each override factor, the following warnings occur, and the override factor will remain unchanged from the previous value.

- "Out of Velocity Override Factor Range Warning (warning code: 0D0BH)"
- "Out of Acceleration Override Factor Range Warning (warning code: 0D0CH)"

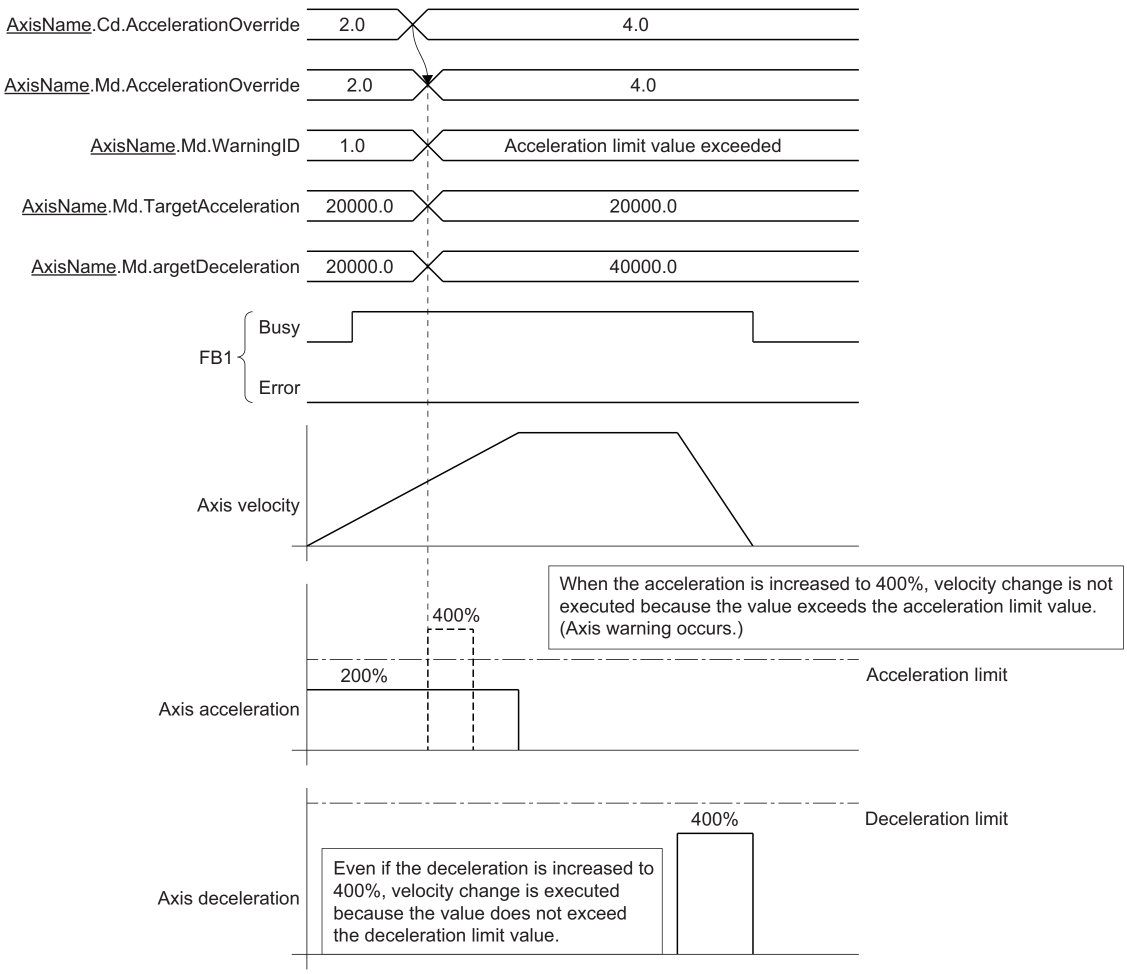


Operation at velocity/acceleration/deceleration limit value over by override

- When velocity limit value over occurs by velocity override, velocity over will occur during control execution and the velocity will not change.

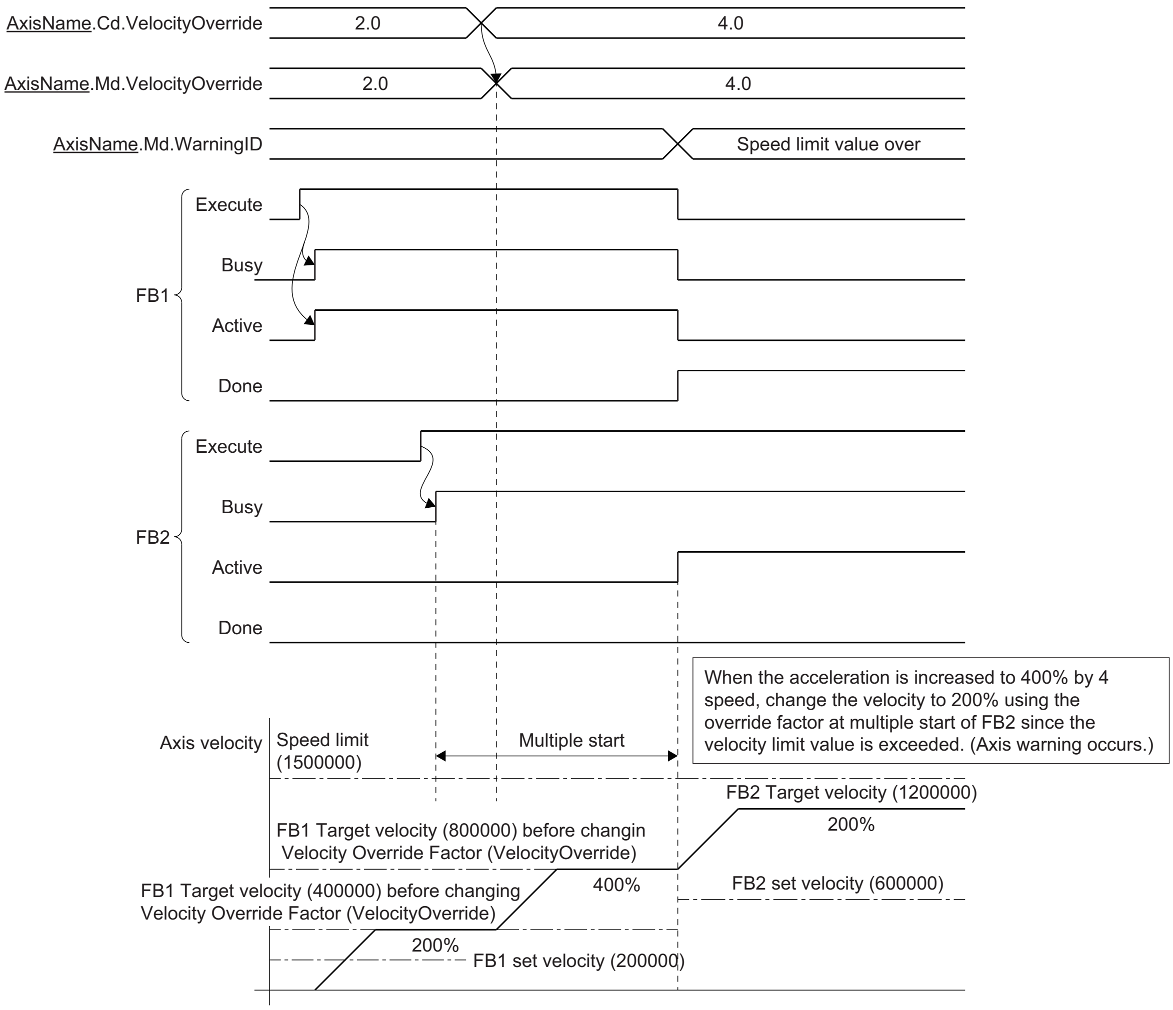


- When acceleration override factor is changed, the acceleration and deceleration will change. In this case, when acceleration limit value over occurs, the acceleration over will occur and the acceleration will change. When deceleration limit value over occurs, the deceleration over will occur and the deceleration will not change as well. (An error does not occur by a control data change, and the override factor is changed.)



■ When velocity/acceleration/deceleration limit value over occurs at switching of FB by buffer mode

- The control change of buffering FB is executed at switching 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 switching 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.



Override change operation by Motion control FB

The override factor can be changed by using the MC_SetOverride (Override Value Setting)/MC_GroupSetOverride (Axes Group Override Value Setting) Motion control FBs.

For details of Motion control FB operations, refer to the following.

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

Precautions

- Do not arrange two or more MC_SetOverride (Override Value Setting) and MC_GroupSetOverride (Axes Group Override Value Setting) 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 Velocity Override Factor (AxisName.Cd.VelocityOverride) or Acceleration Override Factor (AxisName.Cd.AccelerationOverride) directly during performing MC_SetOverride (Override Value Setting).
- Do not operate Velocity Override Factor (AxesGroupName.Cd.VelocityOverride) or Acceleration Override Factor (AxesGroupName.Cd.AccelerationOverride) directly during performing MC_GroupSetOverride (Axes Group Override Value Setting).
- When the velocity after the velocity override is out of the range, refer to the following.
  ☞ Page 83 Velocity Range
- When the acceleration, deceleration, acceleration time, and deceleration time after the acceleration override are out of the range, refer to the following.
  ☞ Page 317 Acceleration/deceleration Processing Function

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 (Torque Control), etc.".
- The torque limit value can be changed by changing the values of Positive Direction Torque Limit Value (AxisName.Cd.TorqueLimit_Positive) and Negative Direction Torque Limit Value (AxisName.Cd.TorqueLimit_Negative). When changing the torque limit value, specifies a value less than or equal to Torque Limit Maximum Value (AxisName.PrConst.TorqueLimit_Max) setting value. For details, refer to the following.
  ☞ Page 358 Torque Limit Value Change Function
- The torque limit value that is currently valid can be checked with Positive Direction Torque Limit Value (AxisName.Md.TorqueLimit_Positive) and Negative Direction Torque Limit Value (AxisName.Md.TorqueLimit_Negative).
- The torque limit value when connecting a driver, Positive Direction Torque Limit Value (AxisName.Cd.TorqueLimit_Positive) and Negative Direction Torque Limit Value (AxisName.Cd.TorqueLimit_Negative) are set.

Relevant variables

Axis parameter constant (AxisName.PrConst.)

Variable/Structure name	Name	Details
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 Positive Direction Torque Limit Value (AxisName.Cd.TorqueLimit_Positive).
TorqueLimit_NegativeInitial	Negative Direction Torque Limit Initial Value	Sets the initial value of Negative Direction Torque Limit Value (AxisName.Cd.TorqueLimit_Negative).

Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
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.

Axis control data (AxisName.Cd.)

Variable/Structure name	Name	Details
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.

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 65 Axes Group

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 (Torque Control), 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.
  For MR-J5(W)-G: 📖MR-J5 User's Manual (Function)

Torque limit value setting range

Set the settings of Positive Direction Torque Limit Value (AxisName.Cd.TorqueLimit_Positive) and Negative Direction Torque Limit Value (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 (Torque Control), etc.".
The torque limit value can be changed by changing the setting values. For details, refer to the following.

☞ Page 358 Torque Limit Value Change Function

When the power supply turns ON, the setting values of Positive Direction Torque Limit Initial Value (AxisName.PrConst.TorqueLimit_PositiveInitial) and Negative Direction Torque Limit Initial Value (AxisName.PrConst.TorqueLimit_NegativeInitial) are stored. In this time, if Positive Direction Torque Limit Initial Value (AxisName.PrConst.TorqueLimit_PositiveInitial) and Negative Direction Torque Limit Initial Value (AxisName.PrConst.TorqueLimit_NegativeInitial) are set with the larger value than Torque Limit Maximum Value (AxisName.PrConst.TorqueLimit_Max), "Out of Forward Direction Torque Limit Value Specification Range (error code: 1A79H)" or "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, "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.

Control data	Setting range
Positive Direction Torque Limit Value (AxisName.Cd.TorqueLimit_Positive)	0.0 (0%) ≤ Setting value ≤ Torque Limit Maximum Value (AxisName.PrConst.TorqueLimit_Max)
Negative Direction Torque Limit Value (AxisName.Cd.TorqueLimit_Negative)	

Torque limit maximum value setting range

Set the settings of Torque Limit Maximum Value (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, "Out of Parameter Range (Axis) (error code: 1D80H)" will occur.

Parameter	Setting range
Torque Limit Maximum Value (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 Torque Limit Maximum Value (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 (Torque Limit Value)
Change using control data	Positive Direction Torque Limit Value (AxisName.Cd.TorqueLimit_Positive), Negative Direction Torque Limit Value (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 Torque Limit Maximum Value (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 (Torque Control)	—
	Limit value		MCv_SetTorqueLimit (Torque Limit Value)	Positive Direction Torque Limit Value (AxisName.Cd.TorqueLimit_Positive) Negative Direction Torque Limit Value (AxisName.Cd.TorqueLimit_Negative)
	Limit value	Maximum value of the limit value	—*1	—

*1 Axis parameters can be changed by MC_WriteParameter (Parameter Write). However, it cannot be changed during control.

Relevant variables

For variables used in a method of changing control data, refer to the following.

☞ Page 355 Relevant variables

Relevant FB

For details on Motion control FBs, refer to the following.

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

Motion control FB	Name	Description
MCv_SetTorqueLimit	Torque limit value	Executes a torque limit value change.

How to change control data

Positive/Negative direction torque limit value setting range

The setting range of Positive Direction Torque Limit Value (AxisName.Cd.TorqueLimit_Positive) and Negative Direction Torque Limit Value (AxisName.Cd.TorqueLimit_Negative) is as the following table.

Control data	Setting range
Positive Direction Torque Limit Value (AxisName.Cd.TorqueLimit_Positive)	0.0 (0%) ≤ Setting value ≤ Torque Limit Maximum Value (AxisName.PrConst.TorqueLimit_Max)
Negative Direction Torque Limit Value (AxisName.Cd.TorqueLimit_Negative)	

Torque limit value change timing

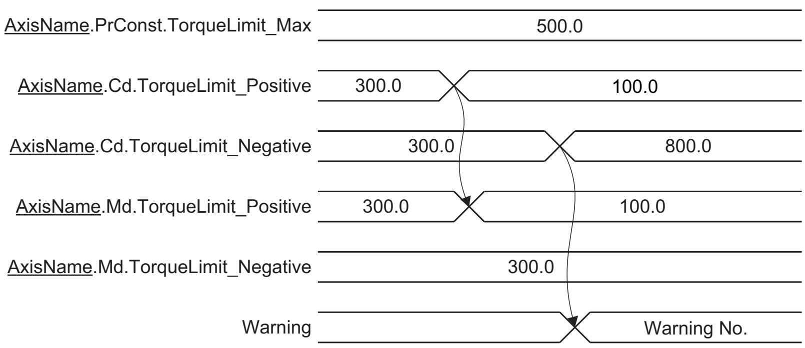
A torque limit value can be changed by changing Positive Direction Torque Limit Value (AxisName.Cd.TorqueLimit_Positive) and Negative Direction Torque Limit Value (AxisName.Cd.TorqueLimit_Negative).

If a value outside the setting range is set, "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, "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 Positive Direction Torque Limit Value (AxisName.Cd.TorqueLimit_Positive) and Negative Direction Torque Limit Value (AxisName.Cd.TorqueLimit_Negative) are changed.

Ex.

When a warning occurs after Negative Direction Torque Limit Value (AxisName.Cd.TorqueLimit_Negative) changes from "300.0" to "800.0" in the status where Torque Limit Maximum Value (AxisName.PrConst.TorqueLimit_Max) = "500.0" is set



Torque limit value change by Motion control FB

When changing the limit values of positive direction torque/negative direction torque at the same time, use the "MCv_SetTorqueLimit (Torque Limit Value)" Motion control FB.

For details of Motion control FB operations, refer to the following.

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

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

Axis parameter (AxisName.Pr.)

Variable/Structure name	Name	Details
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. • 1 to 2147483647
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. • 1 to 2147483647

Driver unit conversion function

- Real drive axis

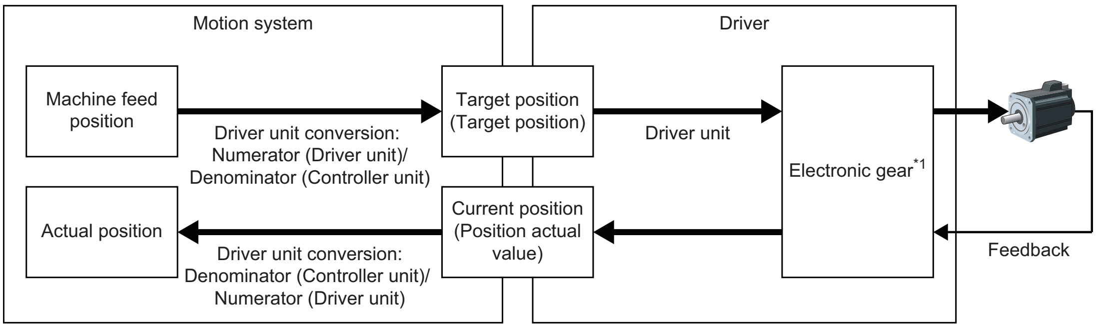
  "Driver unit conversion function" converts the machine feed value to the driver command value and pass it to the target position (Target position). Also, it converts the current position (Position actual value) of the driver and calculates the actual position.T
  he conversion formulas are shown below.

Target position = Machine feed value (Motion system command value) × Driver unit conversion numerator / Denominator

Actual position = Current position × Driver unit conversion denominator / Numerator

Point

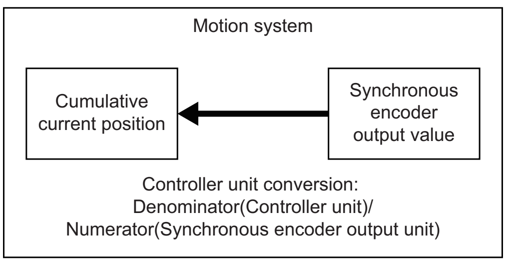
When the driver has the electronic gear, it is recommended to set the ratio to 1:1.



*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 Driver Unit Conversion Numerator (AxisName.Pr.Drive_UnitConvRatioNum) and Driver Unit Conversion Denominator (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.

For the following settings

Setting item	Setting value
Ball screw pitch	20 [mm]
Controller command unit	[mm]
Driver command unit	[pulse]
Motor encoder resolution	67108864 [pulse/rev]

Set the number of pulses required to rotate a motor once (the encoder resolution) in Driver Unit Conversion Numerator (AxisName.Pr.Drive_UnitConvRatioNum), and set the transfer distance of the machine per motor rotation in Driver Unit Conversion Denominator (AxisName.Pr.Drive_UnitConvRatioDen).

- Driver Unit Conversion Numerator (AxisName.Pr.Drive_UnitConvRatioNum) = 67108864 [pulses]
- Driver Unit Conversion Denominator (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]

$$\frac{\text{Number of pulses [pulse]}}{\text{Movement amount [μm]}} = \frac{1}{0.05} = \frac{20}{1.0}$$

- Driver Unit Conversion Numerator (AxisName.Pr.Drive_UnitConvRatioNum) = 20 [pulses]
- Driver Unit Conversion Denominator (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 80 Positioning Range
  If setting a value outside the range, "Driver Unit Conversion Magnification Upper Over (error code: 1AA4H)" will occur.
- A change of the driver unit conversion parameter is also imported when the PLC READY is turned ON. Turning ON it in the following status, otherwise, "Driver Unit Conversion Numerator/Denominator Change Timing Incorrect (error code: 1AA5H)" will occur, and the system will be operated with the set value before turning ON the PLC READY.
  · Axis Status (AxisName.Md.AxisStatus) is "0: Axis disabled (Disabled)"
  · Follow-up 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 becomes 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, 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 Driver Unit Conversion Denominator (AxisName.Pr.Drive_UnitConvRatioDen), and set [μm] in Position Command Unit (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.
The types of command filters are shown below.

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.
Moving direction restriction filter	• Filter function that is used to limit the travel direction of the axis. • The input movement amount 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 Enable (Enable) type of synchronous control FB and has Filter disable (FilterDisable) and Filter cumulated value (FilterPool).

Operation of this function for each system status

○: Possible, ×: Not possible

System status	Function operation
STOP	×
RUN	○
Moderate error	×
Major error	×

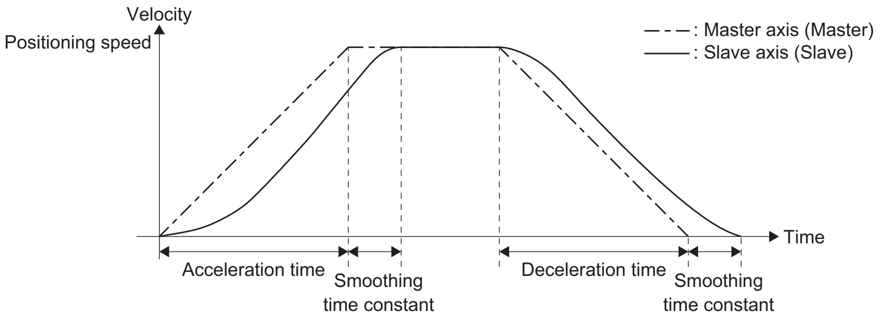
Relevant FBs

For details on Motion control FBs, refer to the following.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

Motion control FB	Name	Description
MCv_SmoothingFilter	Smoothing filter	Executes filter processing based on the specified frequency.
MCv_DirectionFilter	Moving direction restriction filter	Executes filter processing to the travel direction set up to restrict traveling.
MCv_SpeedLimitFilter	Speed limit filter	Executes filter processing to restrict to the speed of set up limit value.
MCv_BacklashCompensationFilter	Backlash compensation filter	Following the movement direction, executes filter processing to compensate the backlash amount in the machine system.

Smoothing filter

- Smoothing filter removes frequencies higher than the set Frequency (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 axis (Master value) set in Master axis data source selection (MasterValueSource) to Slave axis (Slave).
- Axis Status (AxisName.Md.AxisStatus) of Slave axis (Slave) is "7: During synchronous operation (SynchronizedMotion)" during smoothing filter execution.
- The time constant of smoothing is 1/Frequency (Frequency) [s], and the acceleration time and deceleration time become longer by the smoothing time constant. When Frequency (Frequency) is "10 [Hz]", the smoothing time constant is "0.1 [s] = 100 [ms]".

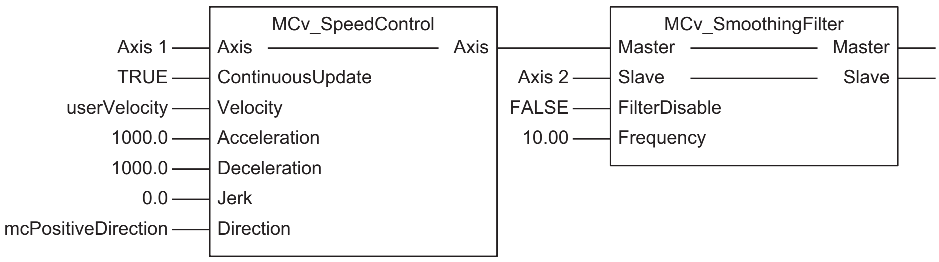


- If a smoothing filter is started while Slave axis (Slave) is in another operation instruction, the start request will be ignored, and "Start Not Possible (error code: 1AADH)" will occur. Start it when Axis Status (AxisName.Md.AxisStatus) of Slave axis (Slave) is "4: Standby (Standstill)".
- MC_Stop (Forced Stop) is the only function block that can be multiple started during smoothing filter execution.

Program examples

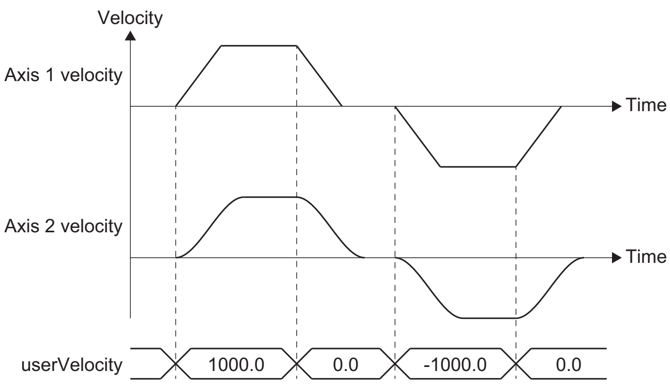
An example program to turn the trapezoidal acceleration/deceleration into smooth acceleration/deceleration waveform is shown below.

■ Sequence program



■ Operation

As Frequency (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).

$$\sum_{i=0}^{\text{Filter level - 1}} (i \text{ the position of master axis before operation cycle - the position of master axis at starting control})$$

- 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.

Necessary slave object

For details, refer to the following.

☞ Page 423 Required Slave Object

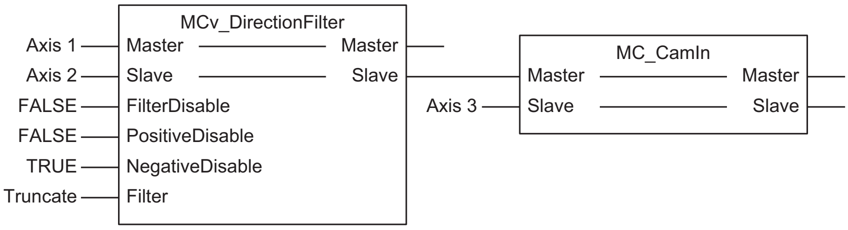
Moving direction restriction filter

- The moving direction restriction filter, to the master axis movement, restricts the movement of the slave axis to one direction.
- The moving direction restriction filter, outputs the result of filter processing that executed to the Master value set in Master axis data source selection (MasterValueSource) to Slave.
- Axis Status (AxisName.Md.AxisStatus) of Slave axis (Slave) will be "7: During synchronous operation (SynchronizedMotion)" during moving direction restriction filter execution.
- If a moving direction restriction filter is started while the slave axis is in another operation instruction, the start request will be ignored, and "Start Not Possible (error code: 1AADH)" will occur. Start when Axis Status (AxisName.Md.AxisStatus) of Slave axis (Slave) is "4: Standby (Standstill)".
- MC_Stop (Forced Stop) is the only function block that can be multiple started during moving direction restriction filter execution.
- When turning Enable (Enable) to FALSE, execute while the slave axis is in stop status. If Enable (Enable) is turned to FALSE during the slave axis in operation, the slave axis will stop immediately.

Program examples

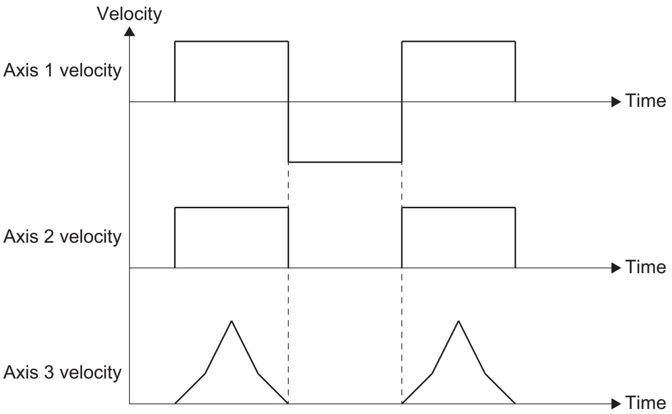
An example program for cam operation direction restriction is shown below.

■ Sequence program



■ Operation

When MCv_DirectionFilter (Moving Direction Restriction Filter) Master axis (Master) moves to the negative direction, Slave axis (Slave) does not move to the negative direction.



Necessary slave object

For details, refer to the following.

☞ Page 423 Required Slave Object

Speed limit filter

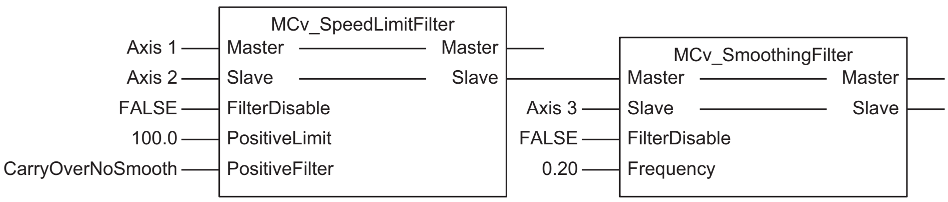
- The speed limit filter, to the input velocity of the master axis, sets a specified control value, and outputs the speed of the set control value to the slave axis.
- The speed limit filter, outputs the result of filter processing that executed to the Master axis (Master) value set in Master axis data source selection (MasterValueSource) to Slave axis (Slave).
- Axis Status (AxisName.Md.AxisStatus) of Slave axis (Slave) will be "7: During synchronous operation (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 "Start Not Possible (error code: 1AADH)" will occur. Start when Axis Status (AxisName.Md.AxisStatus) of Slave axis (Slave) is "4: Standby (Standstill)".
- MC_Stop (Forced Stop) is the only function block that can be multiple started during speed limit filter execution.
- When turning Enable (Enable) to FALSE, execute while the slave axis is in stop status. If Enable (Enable) is turned to FALSE during the slave axis in operation, the slave axis will stop immediately.

Program examples

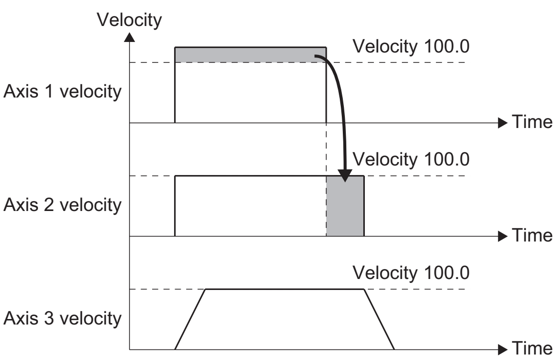
An example program for synchronous encoder speed limit is shown below.

■ Sequence program

The axis 1 sets the synchronous encoder.



■ Operation



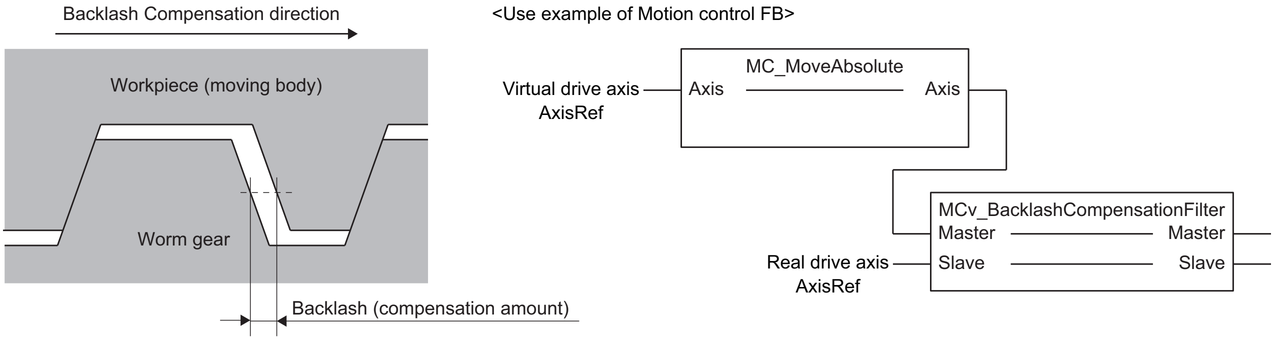
Necessary slave object

For details, refer to the following.

☞ Page 423 Required Slave Object

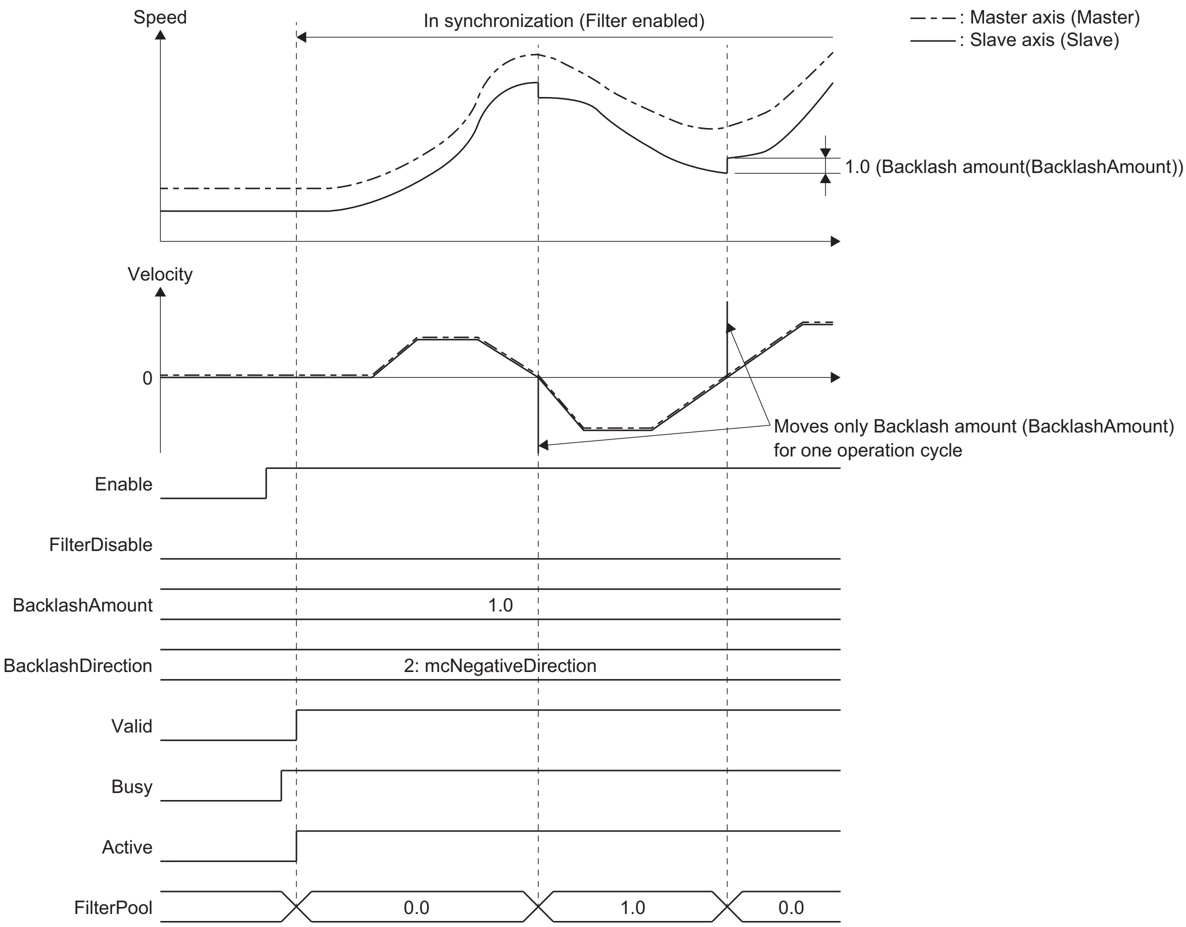
Backlash compensation filter

Backlash compensation filter, is used to compensate the backlash amount in the machine system.



When the movement direction of Master axis (Master) is the same as Backlash compensation direction (BacklashDirection), Slave axis (Slave) synchronizes to the position equivalent to the Master axis (Master) current position shifted in the Backlash direction (BacklashDirection) by the value of Backlash amount (BacklashAmount).

When the movement direction of Master axis (Master) is the reverse of Backlash compensation direction (BacklashDirection), Slave axis (Slave) synchronizes to the Master axis (Master) current position. Whenever the Master axis (Master) movement direction changes, backlash compensation is performed with the Slave axis (Slave) extra movement amount limited to only the value of Backlash amount (BacklashAmount).



- Stop operation

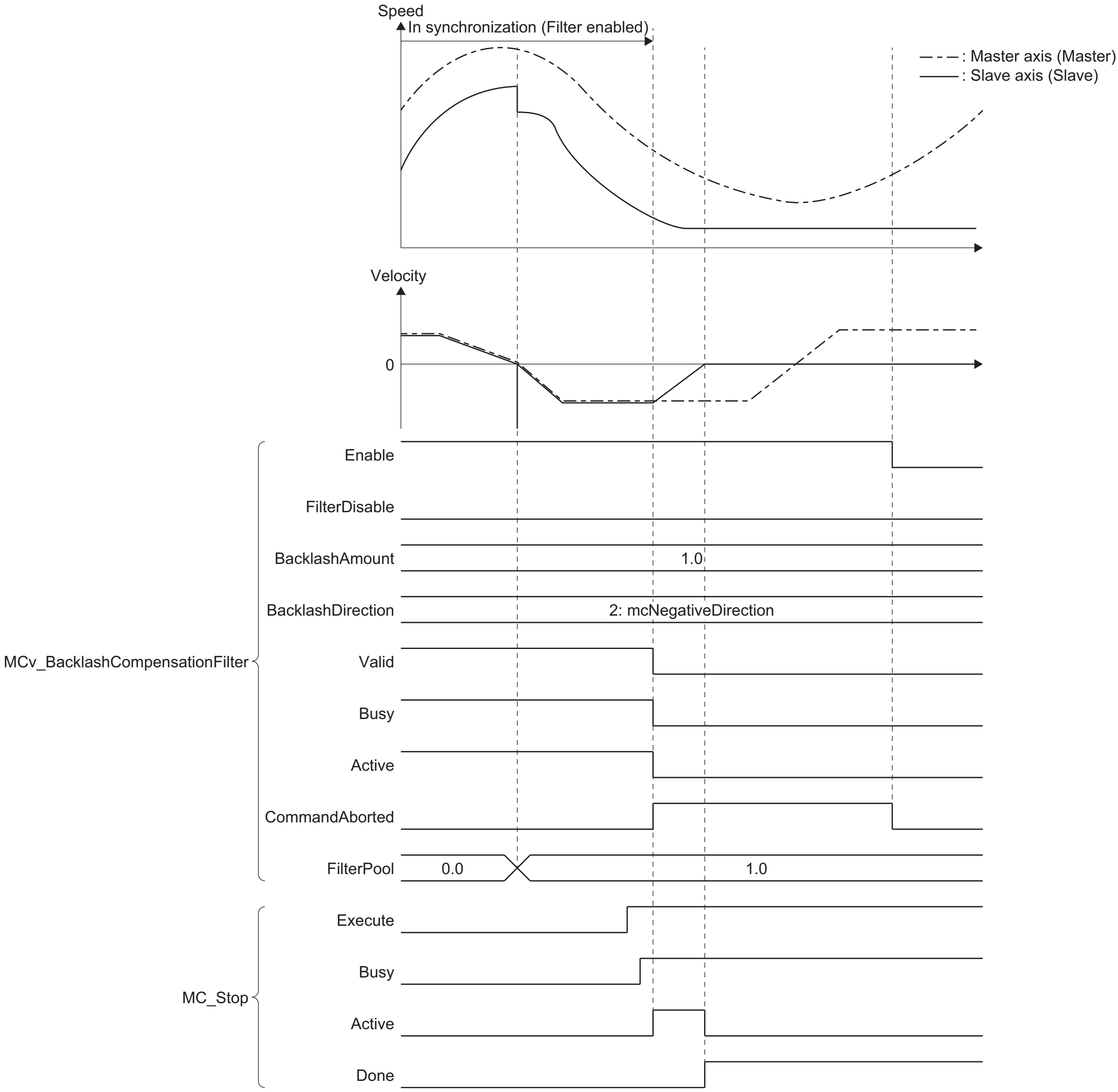
  For details of the stop operation, refer to the following.

  ☞ Page 423 Start and Stop Operation

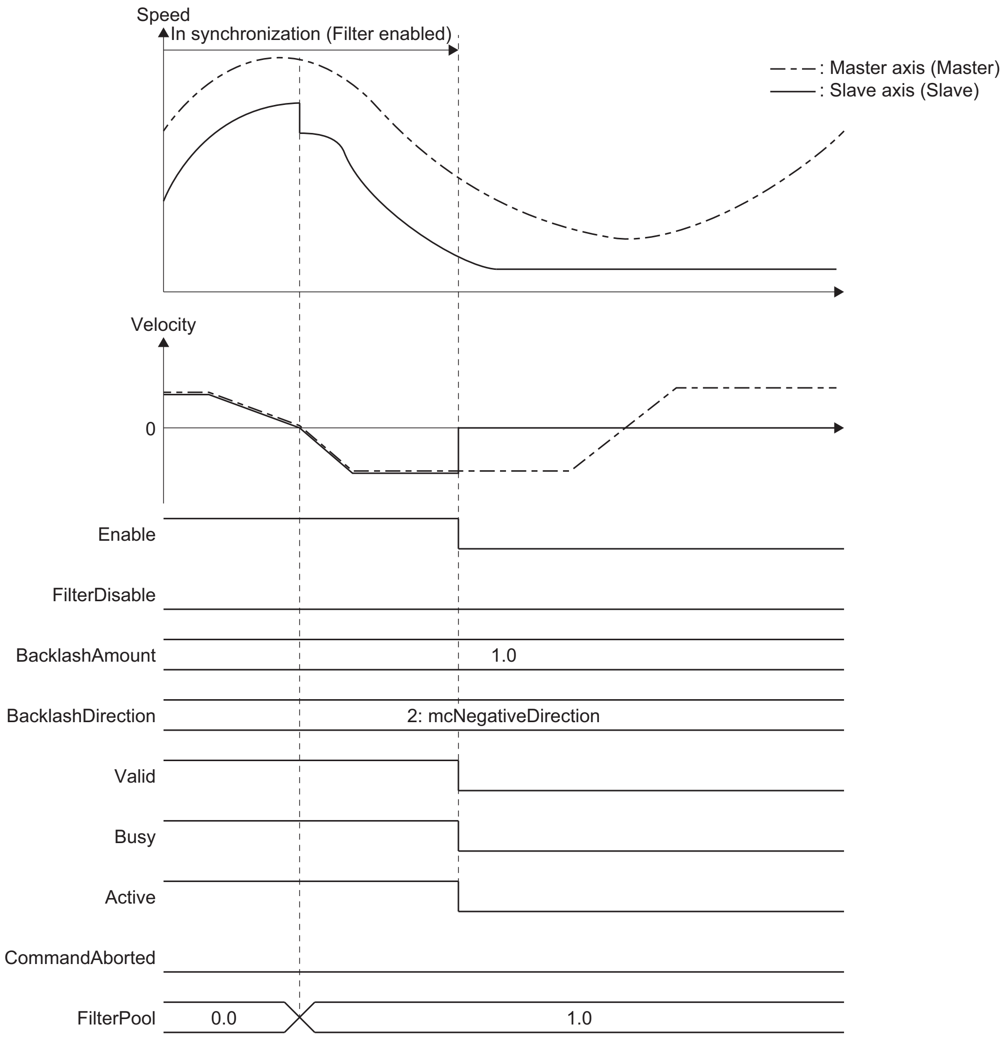
  This section describes the specific operation of the backlash compensation filter.

  When stop factor occurs to Slave axis (Slave), Slave axis (Slave) starts the stop operation from the synchronous position.

  The movement amount (Filter cumulated value (FilterPool) value) by the compensation will not be restored.



• Operation at Filter disabled (At Enable (Enable) falling edge)
Slave axis (Slave) stops immediately at the synchronous position when Enable (Enable) falling edge. The travel amount (Filter cumulated value (FilterPool) value) by the compensation will not be restored.

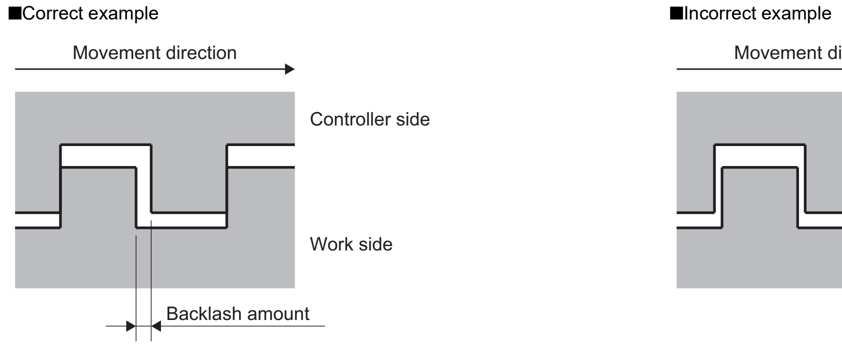


Procedure to enable the backlash compensation filter

Execute after the machine position is established by homing, or restart the backlash compensation filter after cancelling midway.

■ Example of the use procedure

1. Move the real drive axis by homing or JOG operation etc., and maintain the status so that the backlash of the machine system is generated in only one direction as shown in the correct example below.

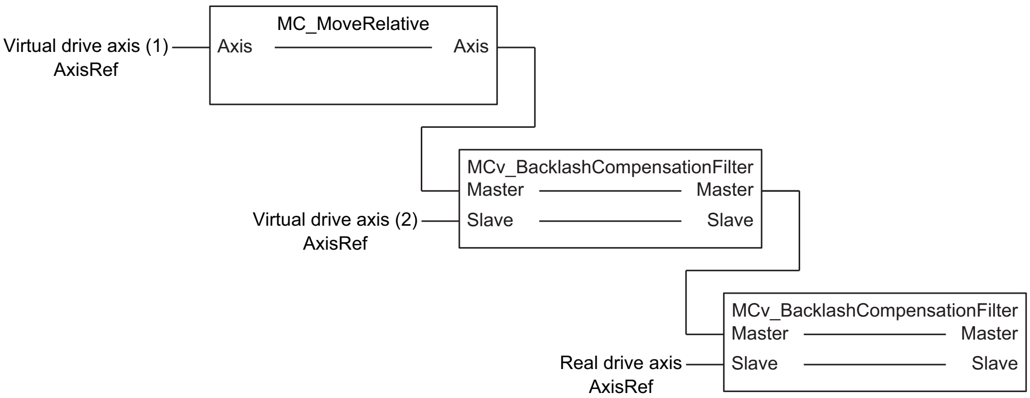


2. Match the Master axis (Master) set position with the real drive axis set position by using the current position change, etc.

3. Set Backlash compensation direction (BacklashDirection) so that it is the reverse direction of the direction that the real drive axis was moved to in Step 1., then start the backlash compensation filter. In addition, start any back part function blocks that are present.

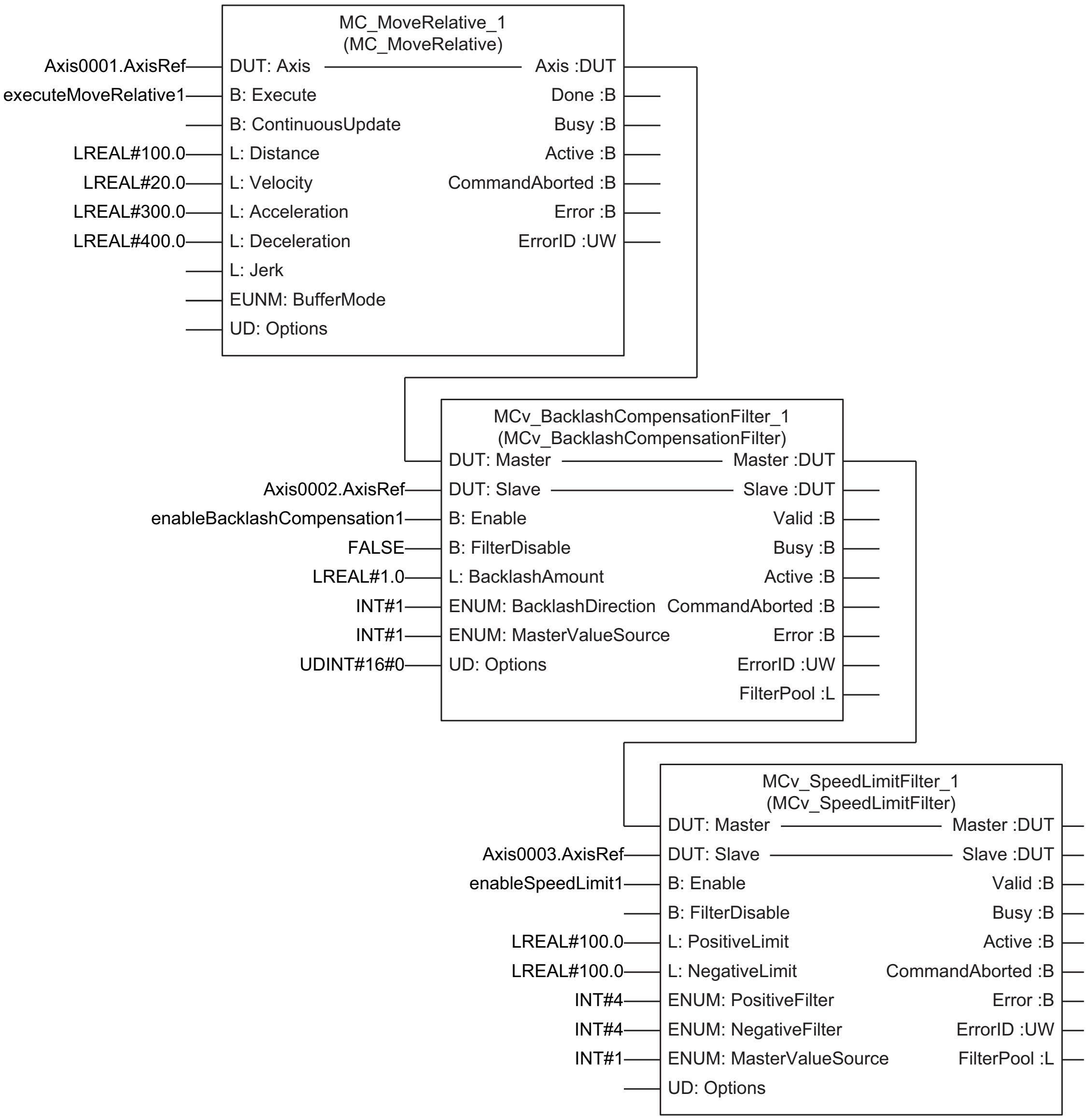
Program example

An example program for the backlash compensation filter in the program is shown below.
The example program is configured so the backlash compensation filter back part is connected with the speed limit filter.



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

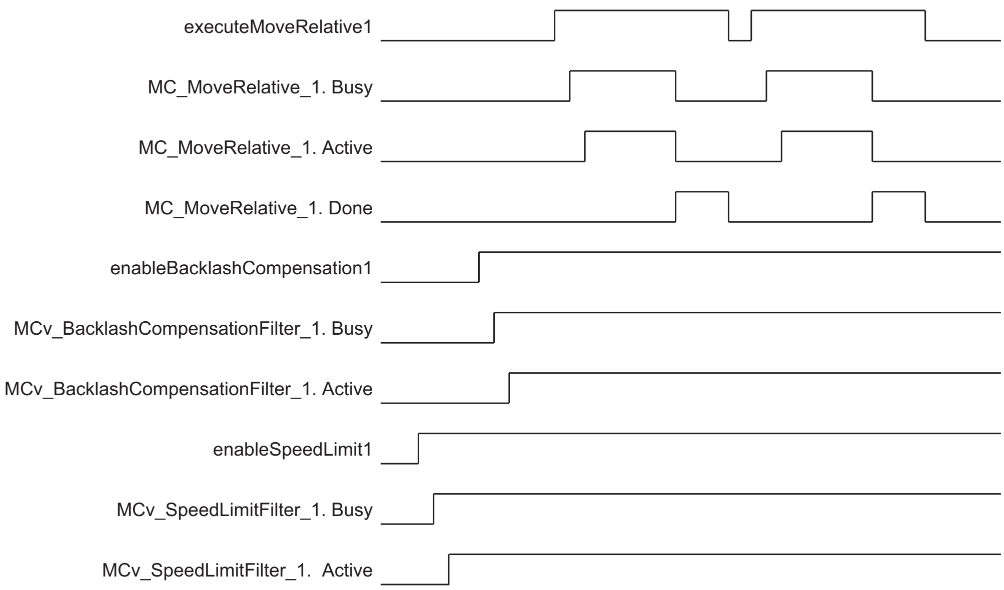
■ Structured text (ST language)

```
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 as follows.
  · A machine which always weighed by external force unidirectionally such as upper and lower axis.
  · A mechanism (such as rack and pinion mechanism) which the backlash amount changes depending on the machine position.
- Only Slave axis (Slave) Backlash amount (BacklashAmount) moves by one operation cycle at the time of Master axis (Master) movement direction changes. Depending on the value, it may exceed the command frequency that the velocity 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 axis (Slave) current position (such as set position, machine feed value etc.) will be the position that Backlash amount (BacklashAmount) compensation is added. Backlash amount (BacklashAmount) compensation will not be added to the Master axis (Master) current position.

Necessary slave object

For details, refer to the following.
☞ Page 423 Required Slave Object

Precautions

- Note that if Filter disable (FilterDisable) is switched during operation, Slave axis (Slave) velocity may change rapidly.
- After enabling the command filter, when Master axis (Master) command is not transmitted to Slave axis (Slave) or the back part synchronous control FB by stop factor occurs to Slave axis (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 command (Execute), data such as Target position (Position) and Velocity (Velocity) can be re-imported without interrupting the operation by Execute command (Execute) retrigger.
When ContinuousUpdate is TRUE at start, data such as Target position (Position) and Velocity (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.

11

Relevant variables

Axis parameter (AxisName.Pr.)

Variable/Structure name	Name	Details
OverrunOperation	Operation Setting at Overrun	Sets operation when an overrun occurs during operation. • 1: Immediate Stop (ImmediateStop) • 2: Keep Current Acceleration/Deceleration (KeepCurrentAcc)

Axes group parameter (AxesGroupName.Pr.)

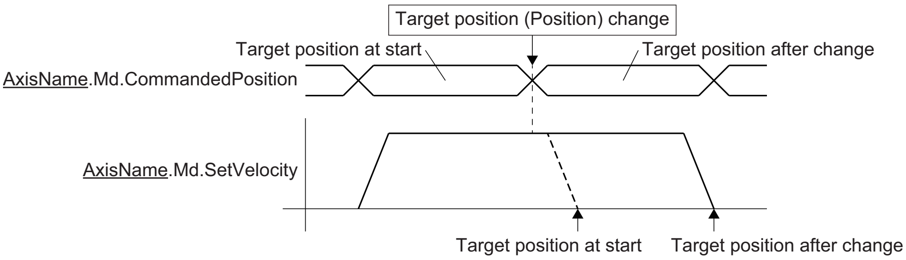
Variable/Structure name	Name	Details
OverrunOperation	Operation Setting at Overrun	Sets operation when an overrun occurs during operation. • 1: Immediate Stop (ImmediateStop)

Target position/movement distance change

This function changes the target position and the movement distance.

Target position change

The target position is changed.

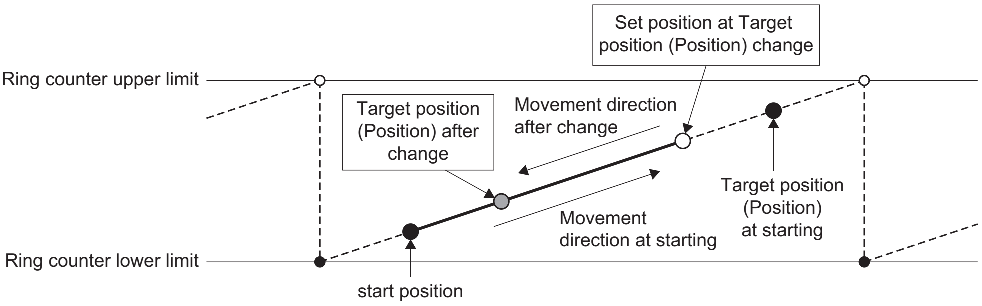


When Direction selection (Direction) can be set, the operation when changing the target position differs depending on the setting value.

■ 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 specified by Direction selection (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 (Options): Bit 5).



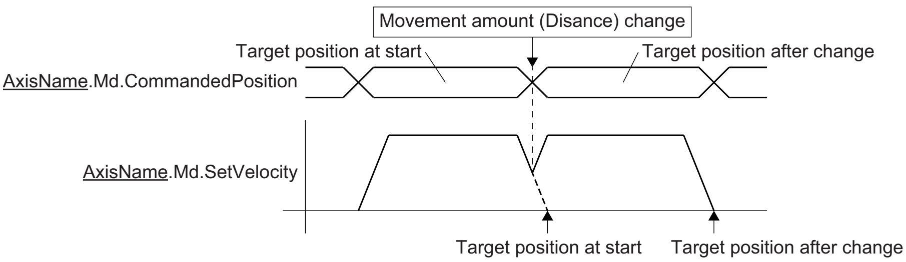
■ For the shortest path

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.



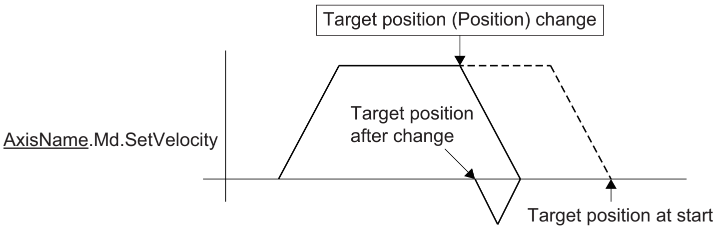
If the movement distance after change is shorter than the moved distance at movement distance change execution, the operation will be reversed, and the control will be performed according to the selection when having reverse rotation permission selection (Options (Options): Bit 5).

The deceleration distance is not kept by the target position/movement distance change

The control is performed according to the selection when having Operation Setting at Overrun (AxisName(AxesGroupName).Pr.OverrunOperation) and reverse rotation permission selection (Options (Options): Bit 5).

■ When the reverse rotation is allowed

When "0: Allow" is set to the reverse rotation permission selection (Options (Options): Bit 5), a deceleration stop is performed once regardless of the Operation Setting at Overrun (AxisName.Pr.OverrunOperation) setting. After the deceleration stop is completed, operation is started to the changed target position.

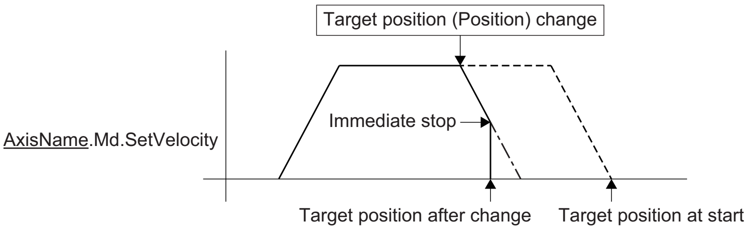


■ When the reverse rotation is not allowed

When "1: Do not allow" is set to the reverse rotation permission selection (Options (Options): Bit 5), the axis operates according to the Operation Setting at Overrun (AxisName.Pr.OverrunOperation) setting.

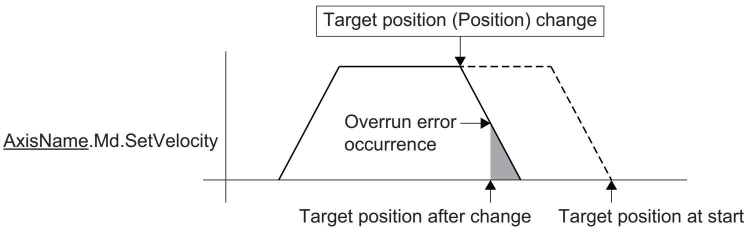
- When "1: Immediate Stop (ImmediateStop)" is set

  Deceleration is started, and the axis immediately stops after "Overrun Warning (warning code: 0D10H)" occurs when it reaches the stop address.



- When "2: Keep Current Acceleration/Deceleration (KeepCurrentAcc)" is set

  The deceleration is started, and the axis exceeds the stop address and stops. When the axis exceeds the stop address, "Overrun Error (error code: 1A7EH)" occurs. However, the operation stops according to the setting of Stop Selection at Stop Cause Occurrence (AxisName.Pr.StopMode_General) when the error occurs. To stop the operation with the address which exceeds the stop address, set "2: Keep Current Acceleration/Deceleration (KeepCurrentAcc)" in Stop Selection at Stop Cause Occurrence (AxisName.Pr.StopMode_General).

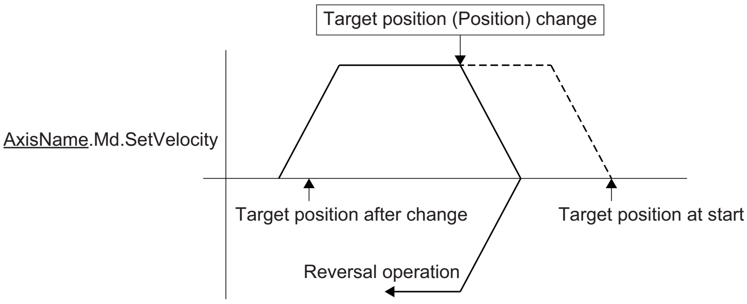


When an operation is reversed by changing the target position/movement distance

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 (Options): Bit 5).

■ When the reverse rotation is allowed

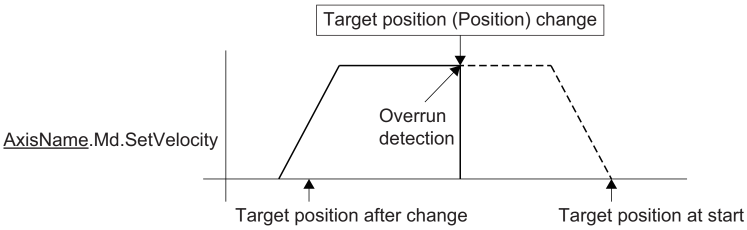
When "0: Allow" is set to the reverse rotation permission selection (Options (Options): Bit 5), a deceleration stop is performed once. After the deceleration stop is completed, operation is started to the changed target position.



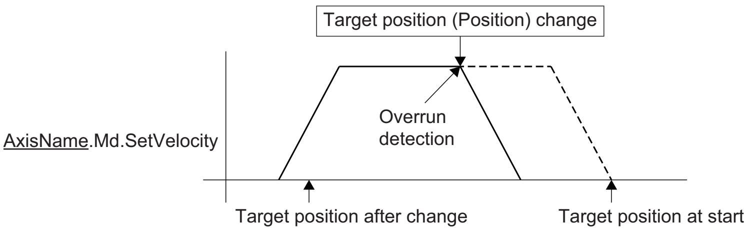
■ When the reverse rotation is not allowed

When "1: Do not allow" is set to the reverse rotation permission selection (Options (Options): Bit 5), the axis operates following the Operation Setting at Overrun (AxisName.Pr.OverrunOperation) setting. When the target position/movement distance is changed, "Overrun Error (error code: 1A7EH)" is detected.

- When setting "1: Immediate Stop (ImmediateStop)"
  When changing the target position/movement distance, "Overrun Error (error code: 1A7EH)" occurs, and immediately stops.

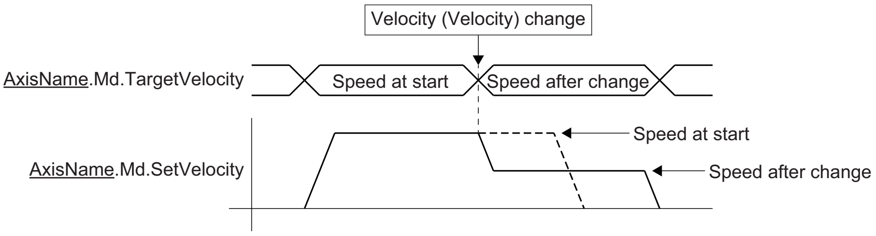


- When setting "2: Keep Current Acceleration/Deceleration (KeepCurrentAcc)"
  When changing the target position/movement distance, "Overrun Error (error code: 1A7EH)" occurs, and stops the operation following the Stop Selection at Stop Cause Occurrence (AxisName.Pr.StopMode_General) setting. To perform deceleration stop by current deceleration, set "2: Keep Current Acceleration/Deceleration (KeepCurrentAcc)" to Stop Selection at Stop Cause Occurrence (AxisName.Pr.StopMode_General).



Set velocity change

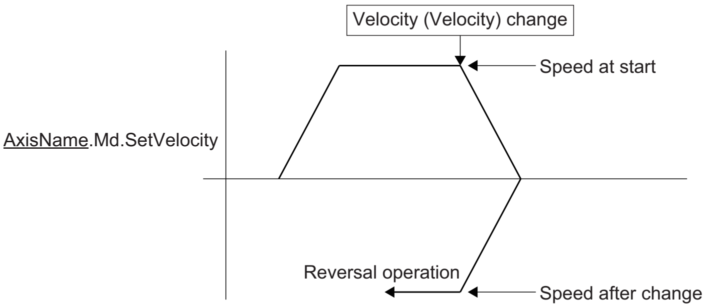
The set velocity is changed according to the acceleration/deceleration.



Set velocity sign inversion during speed control

The negative speed can be set in the speed control.

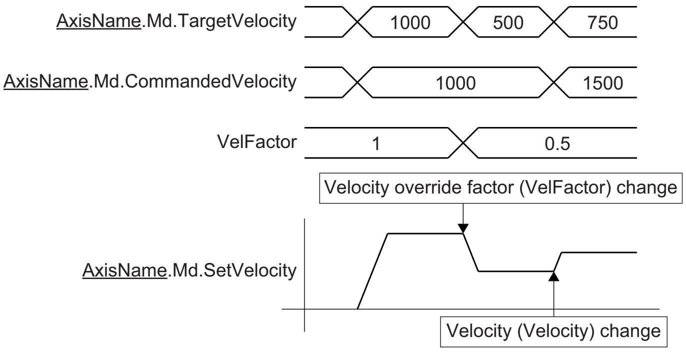
When the sign of set velocity is reversed, the reverse operation will start after a deceleration stop.



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.

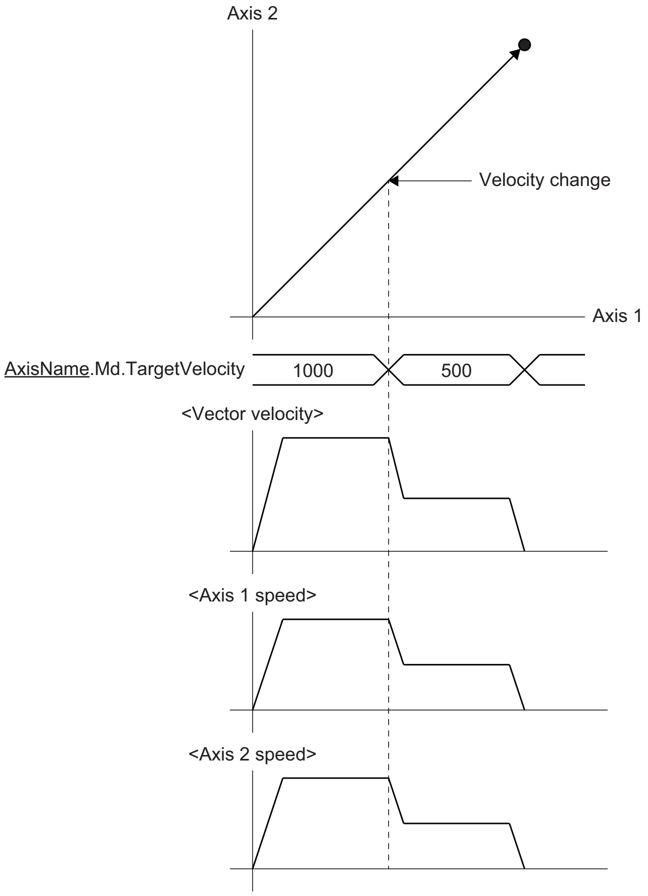


Operation during interpolation control

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 velocity 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 Operation Setting at Overrun (AxisName(AxesGroupName).Pr.OverrunOperation).
For operation when changing the acceleration/deceleration and the acceleration/deceleration time, refer to the following.
☞ Page 317 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 187 Forced Stop
☞ Page 311 Hardware Stroke Limit

Operation of this function for each system status

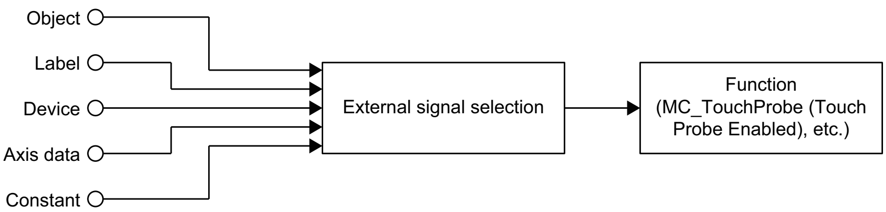
○: Possible

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

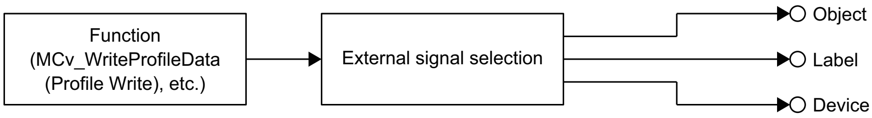
Overall block diagram

The relationship between an external signal selection and function is shown below.

Signal input



Signal output



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.

MC_TRIGGER_REF

Variable/Structure name	Name	Details
Signal	Trigger Signal	Sets the trigger signal. Trigger signal (Signal) is a SIGNAL_SELECT structure.

MC_INPUT_REF

Variable/Structure name	Name	Details
Signal	Input Signal	Sets the input signal. Input signal (Signal) is a SIGNAL_SELECT structure.

MC_OUTPUT_REF

Variable/Structure name	Name	Details
Signal	Output Signal	Sets the output signal. Output signal (Signal) is a SIGNAL_SELECT structure.

For the configuration of each structure, refer to the following.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

The SIGNAL_SELECT structure and the TARGET_REF structure which are structures common for each function, are described below.

SIGNAL_SELECT structure

This section describes SIGNAL_SELECT structure.

- SIGNAL_SELECT (Signal Select)

Variable/Structure name	Name	Details
Source	Signal	Sets BOOL type signals used in control.
Detection	Signal detection method	Sets the signal logic.
CompensationTime	Compensation time	Sets the compensation time of the I/O signal in units of seconds.
FilterTime	Filter time	Sets the filter time for removing chattering of the input signal in units of seconds.

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 described below.

Signal (Source)

- Specify BOOL type signals which are used in control with the TARGET_REF structure. The following data types can be used.

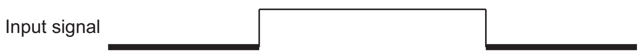
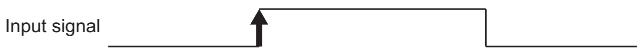
Data types
• [VAR] • [DEV] • [CONST] • [OBJ] (mapped to cyclic communication)

- For details of the TARGET_REF structure, refer to the following.
  ☞ Page 385 TARGET_REF structure
- For operation when a signal which cannot be used is set, refer to each function.

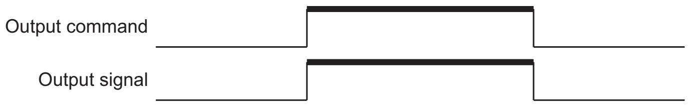
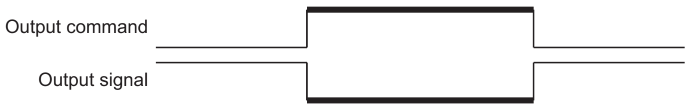
Signal detection method (Detection)

- Specify the signal logic with the MC_SIGNAL_LOGIC enumeration type. For details, refer to the following.

■ Bit input signals

Setting	Operation
0: Detection at TRUE (HighLevel)	When the corresponding signal is in the TRUE status, it operates as a signal input status. 
1: Detection at FALSE (LowLevel)	When the corresponding signal is in the FALSE status, it operates as a signal input status. 
2: Detection at FALSE → TRUE (Rising Edge) (RisingEdge)	When the corresponding signal is the FALSE to TRUE rising edge, it operates as a signal input timing. 
3: Detection at TRUE → FALSE (Falling Edge) (FallingEdge)	When the corresponding signal is the TRUE to FALSE falling edge, it operates as a signal input timing. 
4: Detection at Rising Edge/Falling Edge (BothEdges)	When the corresponding signal is the FALSE to TRUE rising edge and the TRUE to FALSE falling edge, it operates as a signal input timing. 

■ Bit output signals

Setting	Operation
0: Detection at TRUE (HighLevel)	When the output command is TRUE, the corresponding signal is turned to TRUE. 
1: Detection at FALSE (LowLevel)	When the output command is TRUE, the corresponding signal is turned to FALSE. 
2: Detection at FALSE → TRUE (Rising Edge) (RisingEdge)	Must not be specified.
3: Detection at TRUE → FALSE (Falling Edge) (FallingEdge)	
4: Detection at Rising Edge/Falling Edge (BothEdges)	

- Selectable logics vary depending on the function. For details, refer to each function.

Compensation time (CompensationTime)

- Specify the compensation time in seconds.
- Unavailability of the compensation time and the operation when a value is set depend on the function. For details, refer to each function.

Filter time (FilterTime)

- 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.

- TARGET_REF (Input Signal)

Variable/Structure name	Name	Details
StartIO	IO Number	Sets the I/O Number
Target	Target	This variable sets the signal used in control with the following character string formats. *: The setting is not case-sensitive. In addition, spaces are ignored. *: Up to 63 characters can be stored in Target (Target). ■[Classification](Type)Data name.Bit position@Target modification • [Classification] (☞ Page 385 [Classification] / data name) • (Type) (☞ Page 386 (Type)) • Data name (☞ Page 385 [Classification] / data name) • .Bit position (☞ Page 387 .Bit position) • @Target modification (☞ Page 387 @Target modification) ■[Classification](Data type)WSTRING type label^ • [Classification] (☞ Page 385 [Classification] / data name) • (Data type) (☞ Page 386 (Type)) • WSTRING type label^*1

*1 When specifying more than 63 characters string format, use the WSTRING type label and 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] / 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		Setting example
		Format	Description	
[OBJ]	CANopen object of the device station (mapped to cyclic communication)	[OBJ]0x□□□□**##	Sets the index, subindex, and size of the object. • □□□□: Index • **: Subindex • ##: Size (in bits) *: Specifying the target device may be required in @Target modification. Check the specification of the function. *: Match the specified type with the object size. *: Some objects are available as the high-accuracy input. For details, refer to the following. ☞ Page 388 External signal high-accuracy input *: An object whose size is 1 bytes is not available.	[OBJ](DINT)0x607A0020 *: MR-J5(W)-G Target position (Obj. 607Ah: 00h) For 4-byte object (20H)
[VAR]	Label of the motion system	[VAR]Label name	Specify a label in the motion system includes in Label name. The following labels are available as read only by specifying the member names (S: Contact, C: Coil, and N: Current value). • Timer type • Retentive timer type • Long timer type • Long retentive timer type • Counter type • Long counter type *: 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. *: Digit specification of bit-type array labels is not available.	[VAR]ADunit10.OutputEnable [VAR]Label_w1.S

Classification	Target	Specifying method of data name		Setting example
		Format	Description	
[AXIS]	Axis data	[AXIS]MC_SOURCE	Specify a MC_SOURCE enumeration type enumerator in MC_SOURCE. Only the data type "LREAL" can be specified. • 1: Set value (mcSetValue) • 2: Actual value (mcActualValue) • 101: Latest set value (mcLatestSetValue) • 102: Latest actual value (mcLatestActualValue) *: Specifying the target data may be required in @Target modification. *: This data cannot be written. *: Check the specification of the function.	[AXIS]mcSetValue
[DEV]	Device in the motion system (Including buffer memory and link device of the module itself)*2	[DEV]Device name	Specify a device in the module itself in Device name. Following devices can be set. • Buffer memory (G) • Remote input (RX) • Remote output (RY) • Remote register (RWw, RWr) • Link relay (LB) • Link register (LW) • Link special relay (SB) • Link special register (SW) *: @Target modification cannot be specified. (It is ignored.)	[DEV]RWr10.5 [DEV]G11500000.1
[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

*2 For restrictions when using devices, refer to "Direct access from motion built-in program to link devices" under "Cyclic Transmission" in the following manual.
📖MELSEC iQ-R Motion Module User's Manual (Network)

(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.

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.

Type	Description
(BOOL)	Bit
(INT)	Word [signed]
(DINT)	Double word [signed]
(WORD)	Word [unsigned]/bit string [16-bit]
(DWORD)	Double word [unsigned]/bit string [32-bit]
(REAL)	Single-precision real number
(LREAL)	Double-precision real number

Ex.

For [DEV](BOOL)G11500000

Handled as [DEV](BOOL)G11500000.0.

.Bit position

When .Bit position is added to the data whose [Classification] is as follows, 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, the (Type) specification is ignored and the data becomes BOOL type data.

Type	Bit position setting range
[OBJ]	0 to F
[VAR](Only WORD type, DWARD type, INT type, DINT type)	
[DEV]	

@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		Setting example
	Format	Description	
[OBJ]	@Station address	Specify the station address (IP address information to distinguish device stations.) Like multi-axis drive unit, when one station includes multiple logic axes, specify the No. in order to distinguish logic axes. @192.168.3.10#2 Multi-drop No. IP address *: The multi-drop No. is set as "# + No. (decimal)". ·#0: A-axis ·#1: B-axis ·#2: C-axis *: 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 @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 and @CumulativePos are set with the MC_SOURCE type enumerator set in "Data name".*2	@Position @CumulativePos
[DEV]	—	The target modification cannot be set. It is ignored if set.	
[CONST]	@+adding value @-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.	[CONST](INT)0@-10*3 [CONST](BOOL)0x00.3@+1*4

*1 POU: Program Organization Unit
*2 The enumerators that can be specified are shown below.
○: Can be specified, ×: Cannot be specified

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

*3 The referred value will be as "0, -10, -20, -30, ….".
Specifying [CONST] bit can be used from 0 to 7.
*4 The referred value set as [CONST] (BOOL) 0 x 00.3 @ + 1 switches between TRUE and 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.

External signal high-accuracy input

For some functions that have external signal input such as touch probe and advanced synchronous control, when signals which are compatible with the external signal high-accuracy input are used for functions which have Trigger input signal (TriggerInput) as the input, high-accuracy control using signal detection time of input devices can be performed.
For usable signals and setting methods, refer to the following.
When the signal type is [OBJ]: ☞ Page 863 Program example (When using MR-J5-G-RJ series)

List of function which is compatible with the external signal high-accuracy input

- Touch probe
- Advanced synchronous control

Precautions

When using [OBJ] as an output destination of a function, do not label the object and link device used as the output destination.

Relevant add-ons

The following add-on is required to use this function.

- SignalIO
- Axis*1
- ServoDriver_CANopen*2

*1 When using [AXIS] as the signal type
*2 When using [OBJ] as the signal type

12.2 Touch Probe

This function records (latches) arbitrary data at trigger input signal detection.

Operation of this function for each system status

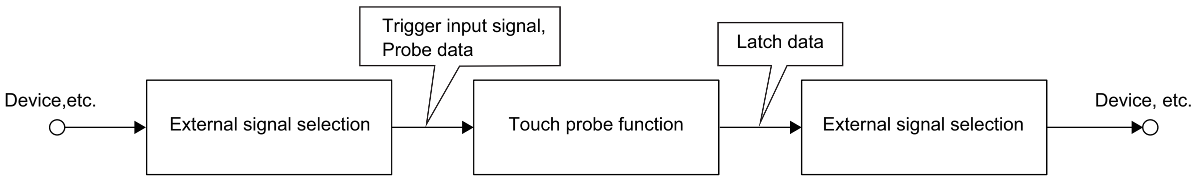
○: Possible, ×: Not possible

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

12

Overall block diagram

The relationship between the touch probe function and other functions is shown below.



Relevant FBs

For details on Motion control FBs, refer to the following.

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

Motion control FB	Name	Description
MC_TouchProbe	Touch probe enabled	Optional data will be recorded due to the trigger event occurrence.
MC_AbortTrigger	Touch probe disabled	Disables the latch in execution.

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 (Touch Probe Enabled). To stop this function, execute MC_AbortTrigger (Touch Probe Disabled).

Control details

The operation for when a trigger event occurs is shown below.

- Calculations for the latched data are estimated at rising edge/falling edge/both direction of the signal set in Trigger input signal (TriggerInput) (indicate trigger input signal as follows).
- When Enabled window (WindowOnly) is 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 Latch position (RecordedPosition) and Storage location of latch data (OutputBuffer) according to Latch mode (RecordMode), and Number of latch times (RecordedCounter) is updated.

■ Estimated calculation

The probe data during operation cycle interval is calculated by estimation. The value calculated by estimation at the timing inputted Trigger input signal (TriggerInput) is the latch data. The value is calculated as the figure below.

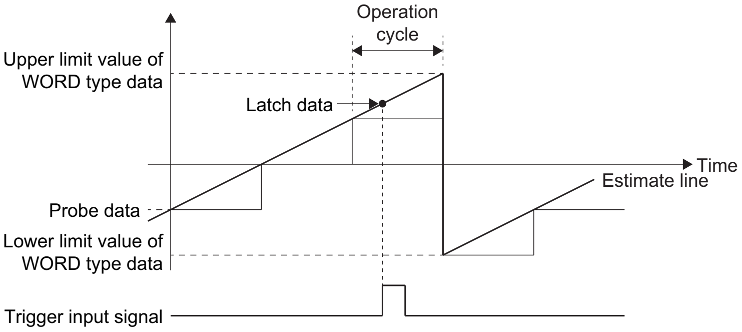
- When the prove data is the WORD type

■When using the external signal high-accuracy input

Trigger input timing is when the signals are detected at the device station.

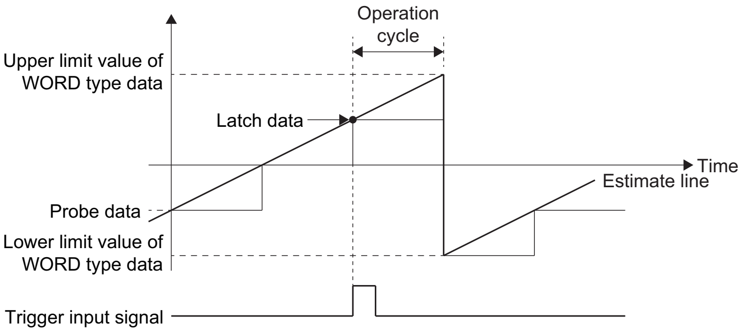
For setting method of high-accuracy input of the external signal selection, refer to the following.

☞ Page 388 External signal high-accuracy input



■When the external signal high-accuracy input is not used

The trigger input timing is operation cycle.



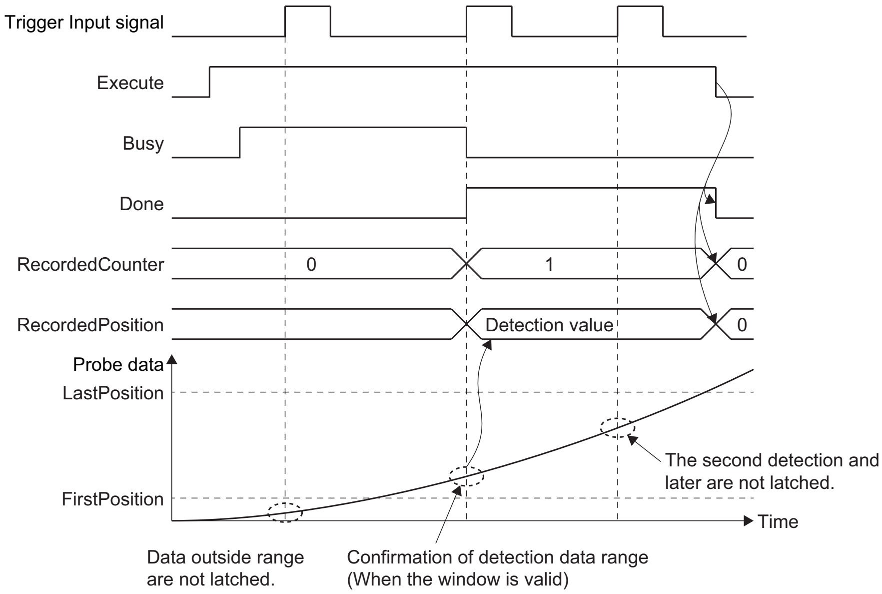
■ Timing chart

- When the FB is normally completed

  The following shows examples of latch operation by Latch mode (RecordMode).

Ex.

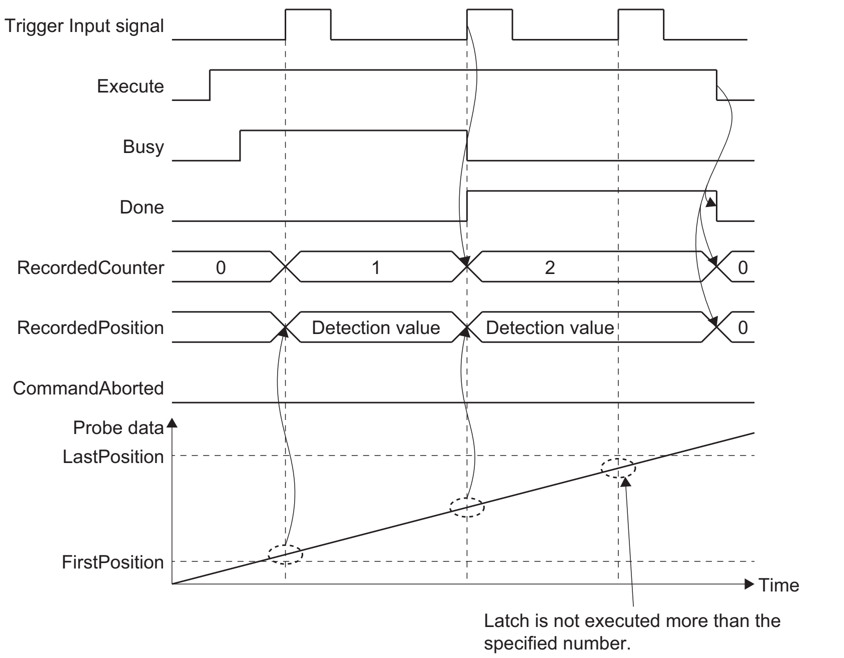
When Latch mode (RecordMode) is "0: One Shot Mode (OneShot)"



Ex.

When Latch mode (RecordMode) is "1: Frequency Specified Mode (RecordCount)"

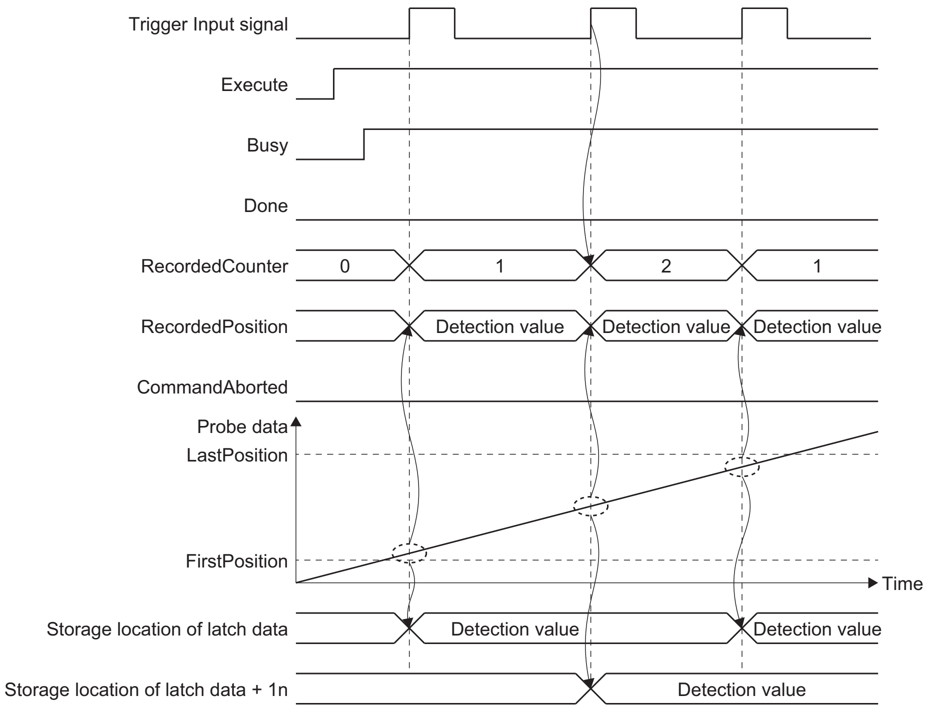
Number of latch times (RecordedCounter): 2



Ex.

When Latch mode (RecordMode) is "2: Ring Buffer Mode (RingBuffer)"

Number of latches: 2



- When an error occurs

  For details when an error occurs, refer to "Basic operation of Execute command (Execute) type Motion control FBs" in the following.

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

Precaution

- Refresh of the Motion control FB 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 (normal/fixed cycle).
- This FB is executed even if the target axis is not connected or an error occurs.
- If the operation cycle over occurs after Executing (Busy) of MC_TouchProbe (Touch Probe Enabled) is TRUE, accuracy of estimated calculation may deteriorate.

Touch probe invalid

Touch probe is invalidated.

Control details

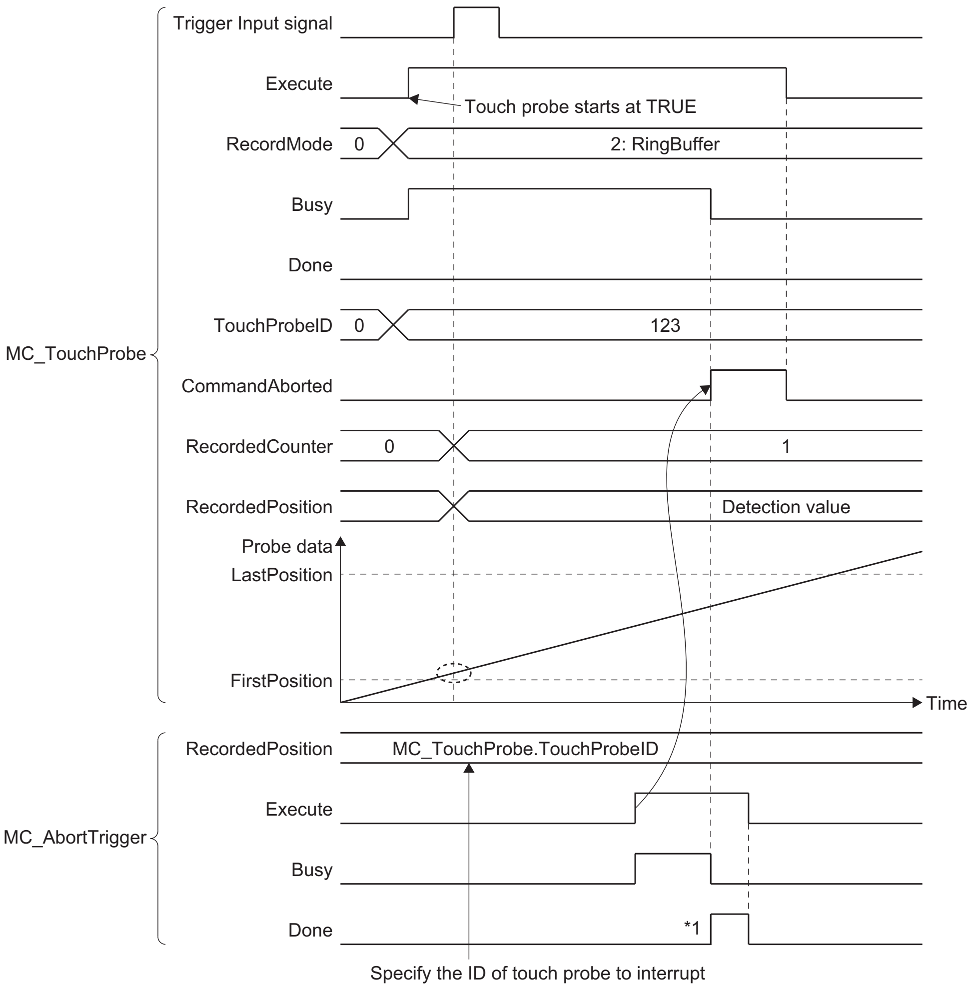
MC_TouchProbe (Touch Probe Enabled) which is specified in Touch probe ID (TouchProbeID) is invalidated.
When MC_AbortTrigger (Touch Probe Disabled) is executed, if the specified MC_TouchProbe (Touch Probe Enabled) specified Touch probe ID (TouchProbeID) is not running, the FB will immediately become Execution completion (Done) is TRUE.
Axis information (Axis) and Trigger input signal (TriggerInput) can be omitted, because those are ignored.

■ Timing chart

• When the FB is normally completed

Ex.

When Latch mode (RecordMode) is "2: Ring Buffer Mode (RingBuffer)"



*1 Execution completion (Done) may turn on immediately depending on the execution timing of a program.

• When an error occurs

For details when an error occurs, refer to "Basic operation of Execute command (Execute) type Motion control FBs" in the following.

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

Precautions

Relevant add-ons

The following add-on is required to use this function.

- MotionControl_General
- SignalIO
- Axis*1
- NetworkDriver_CCIETSN*2
- ServoDriver_CANopen*2

*1 For using the signal type [AXIS] of the external signals

*2 For using the signal type [OBJ] of the external signals

System memory capacity

■ RAM Usage

Memory capacity of MC_TouchProbe (Touch Probe Enabled) differs depending on Compensation Time (CompensationTime).

For details, refer to the following.

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

■ Backup RAM Usage

Backup RAM is not used.

12.3 Axis Emulation

Axis emulate function controls axes on the real axis without connecting to the device station.
When using emulate function, even if the device 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 device station. The operations unique to the device station can be simulated by installing emulate models corresponding to each device station as add-ons.

Relevant variables

12

Axis parameter constant (AxisName.PrConst.)

Variable/Structure name	Name	Details
SlaveEmulate_Enable	Axis Emulation Enabled	Sets whether the axis is used as an emulate axis. • FALSE: Invalid • TRUE: Valid

Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
SlaveEmulate_Enable	Axis 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 machine feed value at connection depending on the Absolute Position Control Setting (AxisName.PrConst.PosRestoration_AbsPosEnable) setting are as shown below.

Absolute Position Control Setting (AxisName.PrConst.PosRestoration_AbsPosEnable)	Values of the set position and the machine feed value at connection
-1: Automatic Setting (Acquire from Connected Device) (Auto)	Both the cumulative current position and the machine feed value are "0.0". (The set position is according to the ring counter setting)
0: Disable Absolute Position System (ABSDisabled)	
1: Enable Absolute Position System (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 machine feed value are "0.0". (The set position is according to the ring counter setting)

Specification of the emulated device 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 device station) is not carried out.
PDO-related	Emulation of feedback	○	Simulates feedback of MR-J5(W)-G. For details, refer to the following. ☞ Page 396 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 device 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 device 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 device 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 device station during power-OFF.
- In the emulation being disabled, when the device station is used with the setting which Absolute Position Control Setting (AxisName.PrConst.PosRestoration_AbsPosEnable) is "-1: Automatic Setting (Acquire from Connected Device) (Auto)" and the connected device is used in the setting of Use Absolute Position System, change to "1: Enable Absolute Position System (Enabled)" before switching emulation to enabled. When the emulation is enabled, "-1: Automatic Setting (Acquire from Connected Device) (Auto)" becomes "0: Disable Absolute Position System (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.

12.4 Inter-Module Synchronization Function

The inter-module synchronization function can synchronize the control timings among multiple modules on the same base.

Operation of this function for each system status

■ System operation status

○: Possible, △: Possible (restricted), ×: Not possible

Status	Operation availability
STOP	△*1
RUN	○
Moderate error	△*1
Major error	×

*1 Operation is possible when the CPU module is RUN.

■ Interrupt operation status

○: Possible, △: Possible (limited)

Status	Operation availability
STOP	△*1
RUN	○

*1 The inter-module synchronous interrupt program is not possible.

Control details

This function can synchronize the inter-module synchronous interrupt program (I44) execution cycle of the CPU module and the operation cycle of the Motion module.

For the parameter setting of the inter-module synchronization, refer to "CC-Link IE TSN Network Synchronous Communication Function" in the following,

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

or refer to the following.

MELSEC iQ-R Inter-Module Synchronization Function Reference Manual

Point

When using the inter-module synchronous function, the EI instruction must be executed on the CPU module to enable the inter-module synchronous interrupt program (I44). When the status is changed to RUN while the inter-module synchronous interrupt program (I44) is disabled, the buffer memory refresh task is not executed. "Cycle Over (error code: 320CH)" may occur.

Operation using the multiple Motion module

When the multiple Motion module is the synchronous target, execution timing of a fixed cycle program can be synchronized among modules.*1

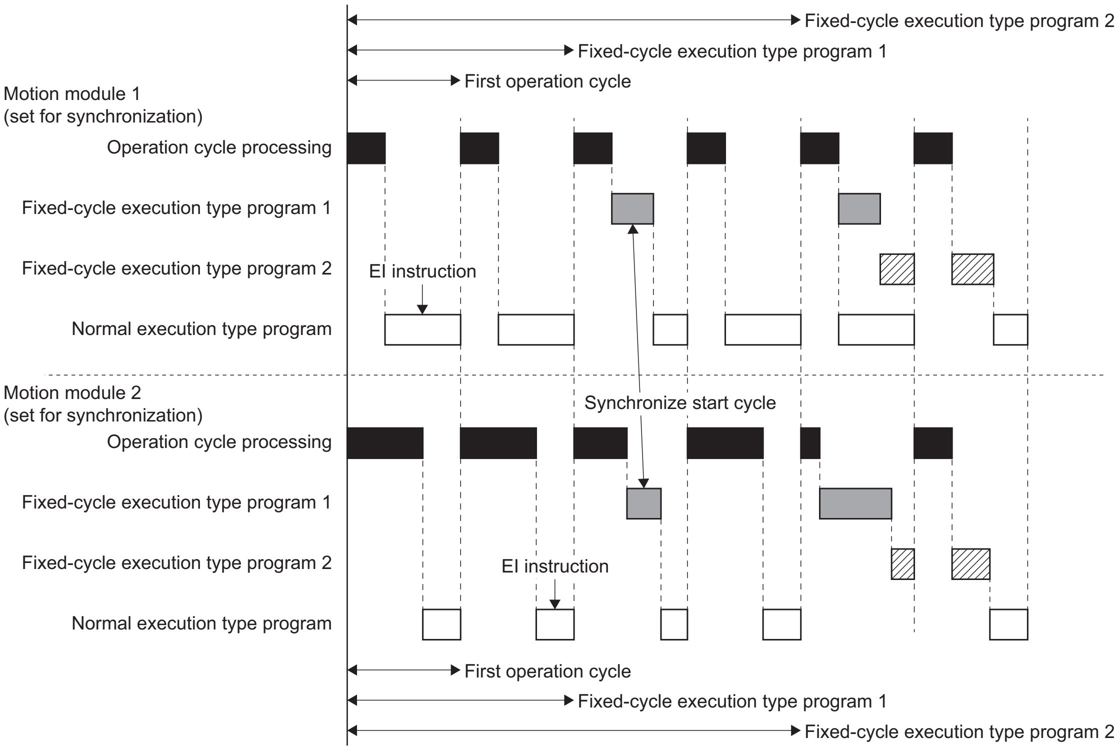
Note that the synchronization is applied only when the fixed scan interval setting of fixed cycle program is as follows.

Communication cycle $\times$ 2^n (Up to 128 ms interval)*2

Other fixed cycle programs of the fixed scan interval setting may not be synchronized the execution timing among modules.

*1 Even if the Motion control FBs is started in the fixed cycle program, start timing differs from each module because analysis complete timing of the instruction is different. For the method to synchronize the start timing among multiple modules, refer to the following.
☞ Page 400 Synchronization of positioning control start timing using the function

*2 After enabling the interrupt program (after executing the EI instruction), the first execution of the fixed cycle program should needs to match among the modules so that the execution is performed after the cycle set in the fixed scan interval setting. (Up to 128 [ms])



Synchronization of positioning control start timing using the function

The timing of positioning start can be synchronized among different modules. After the positioning is started, each Motion module operates independently. The setting and procedure are shown below.

■ Setting

1. Set the Motion module which is synchronized the timing of positioning start as the synchronous target module.

For the parameter setting of the inter-module synchronization, refer to "CC-Link IE TSN Network Synchronous Communication Function" in the following manual.

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

2. Set the same cycle between the inter-module synchronous cycle of the CPU module and the operation cycle of the Motion module. If not, the synchronization of the timing cannot be assured.

■ Procedure example

1. Set the velocity override factor "0.00" in the arbitrary buffer memory (user setting area).

2. Perform positioning start after setting the setting value of the buffer memory written in Step 1 to Velocity Override Factor (AxisName.Cd.VelocityOverride, AxesGroupName.Cd.VelocityOverride) by the fixed cycle program of the Motion module.

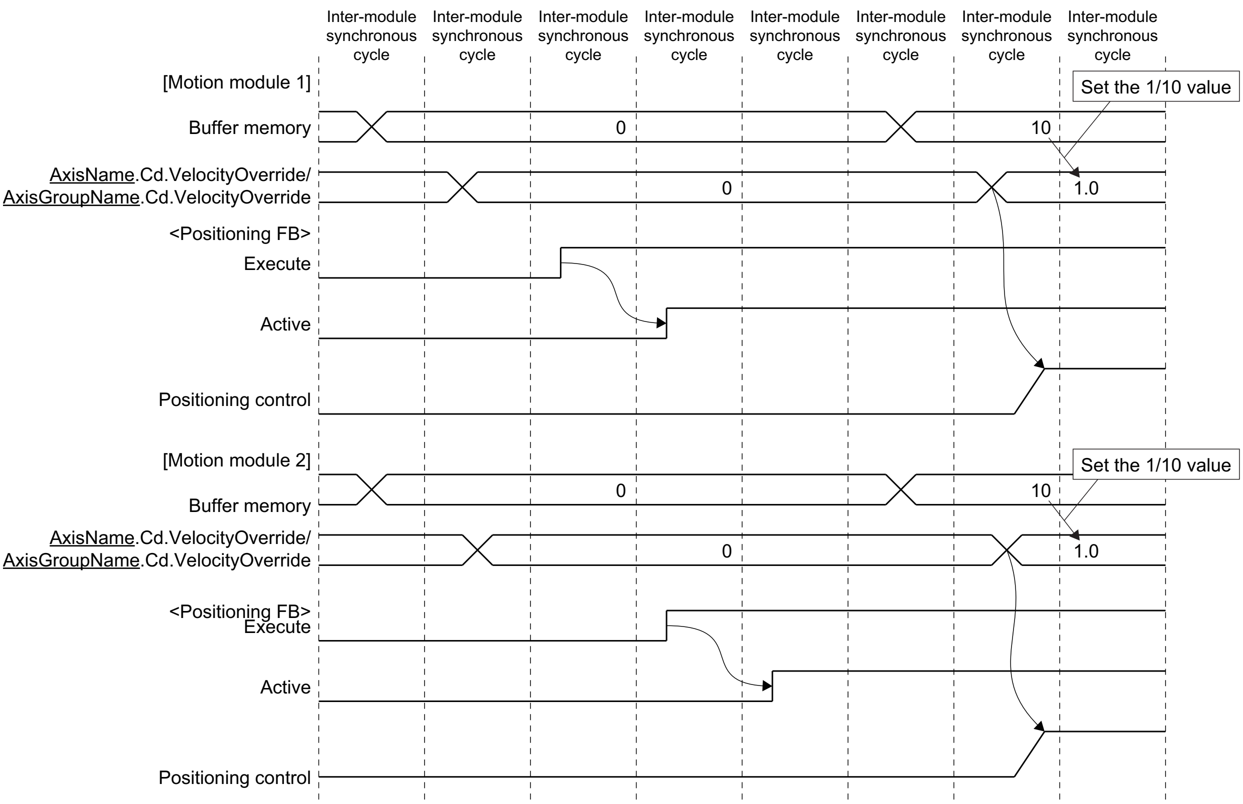
3. Confirm Controlling (Active) of all Motion control FBs becomes ON, and set the velocity override factor "10.00" in the buffer memory used in Step 1 within the same inter-module synchronous cycle while the inter-module synchronous interrupt program (I44) is performed.

4. Set the one-tenth value of the buffer memory setting value written in Step 3 to Velocity Override Factor (AxisName.Cd.VelocityOverride, AxesGroupName.Cd.VelocityOverride) by the fixed cycle program of the Motion module.

*1 When using the time-fixed method, set "1: Maximum Acceleration/Deceleration (MaximumAcceleration)" to Operation Selection at Start Acceleration/Deceleration 0 (AxisName.Pr.AccelerationZeroBehavior, AxesGroupName.Pr.AccelerationZeroBehavior). When setting "-1: Error (Not Started) (ACCError)", if Velocity Override Factor (AxisName.Cd.VelocityOverride) is set to "0.00", "Acceleration/Deceleration 0 Specified Operation Error at Start (error code: 1A0CH)" occurs and the axis will not start.

*2 The processing time of operation cycle temporarily increases because the velocity of all axes is changed at the same time. Take actions such as reducing the number of the target axis, when the operation cycle is exceeded.

■ Operation example



Synchronous control using inter-module synchronization

The synchronized operation synchronizing with the same input axis among different modules can be performed. The setting and procedure are shown below.

■ Setting

1. Set the Motion module used for the synchronized control as the inter-module synchronous target.

For the parameter setting of the inter-module synchronization, refer to "CC-Link IE TSN Network Synchronous Communication Function" in the following manual.

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

2. Set the same cycle between the inter-module synchronous cycle of the CPU module and the operation cycle of the Motion module. If not, the simultaneous start cannot be assured.

■ Procedure example

1. Prepare the synchronous control FB set the virtual encoder axis as the master axis for each module.

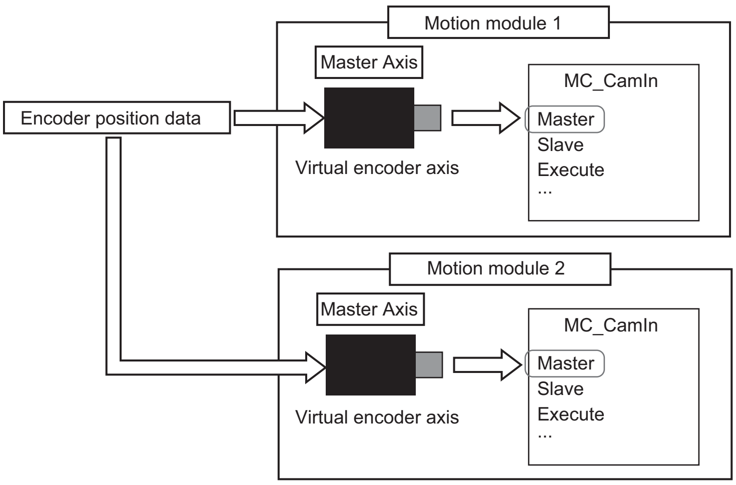
2. Set Connection Command (AxisName.Cd.Encoder_Connect) and Connection Status (AxisName.Md.Encoder_Connected) as public labels.

3. Set Connection Command (AxisName.Cd.Encoder_Connect) of each module to TRUE. Confirm whether the virtual encoder axis of all modules is connected completely by checking Connection Status (AxisName.Md.Encoder_Connected).

4. Start the synchronous control FB of all modules.

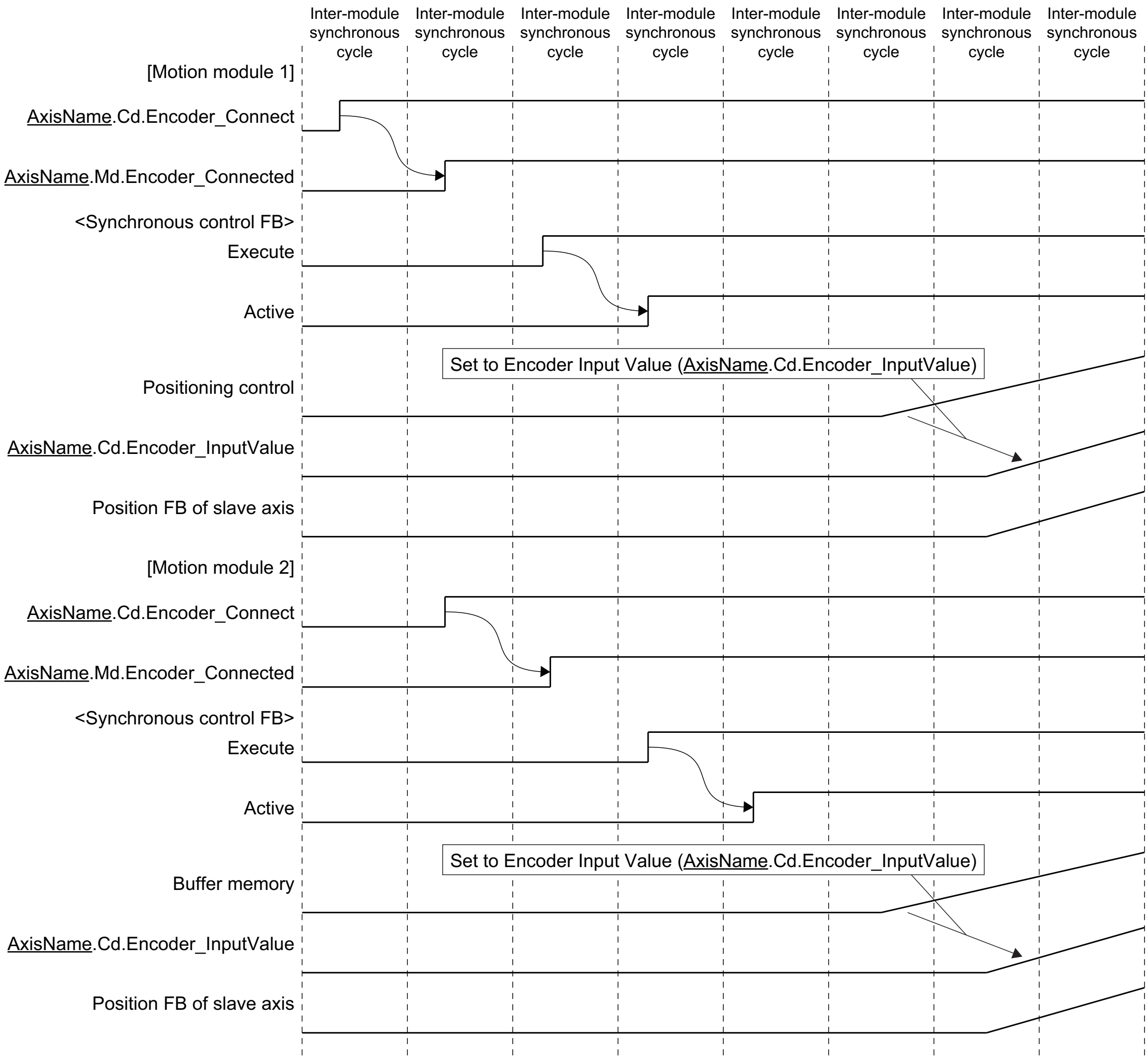
5. Write arbitrary values (position data of the virtual encoder) into arbitrary buffer memory (user setting area) in the inter-module synchronous interrupt program (I44).

6. Write the setting values of the buffer memory written in Step 5 into Encoder Input Value (AxisName.Cd.Encoder_InputValue) by the fixed cycle program of the Motion module.



*1 The public label is refreshed at END except executing the inter-module synchronous interrupt program (I44) so that the things which are communicated by synchronizing with the inter-module synchronous cycle cannot be used as public labels, but the things which are not communicated in inter-module synchronous cycle can be used as public labels.

■ Operation example



Program example

Simultaneous start with multiple modules

■ Scan program [CPU module]

ST program

```
// PLC READY ON
   Y10 := TRUE;
   Y40 := TRUE;

// Interruption enabled
   EI(TRUE);

// Confirm that Ready and Synchronization flag becomes ON
IF (X10 = TRUE) AND (X11 = TRUE) AND (X40 = TRUE) AND (X41 = TRUE) THEN
    // Servo ON
    MCv_AllPower_1.Enable := TRUE;
    MCv_AllPower_1.ServoON := TRUE;
    MCv_AllPower_2.Enable := TRUE;
    MCv_AllPower_2.ServoON := TRUE;
END_IF;

// Axes group enabled
IF (RD78_0040.Axis0001.Md.AxisStatus = 4) AND(RD78_0040.Axis0002.Md.AxisStatus = 4) THEN
    MC_GroupEnable_1.Execute := TRUE;
END_IF;

// Set the velocity override factor 0 to arbitrary buffer memory (user setting area)
IF (RD78_0010.Axis0001.Md.AxisStatus = 4) AND (RD78_0040.AxesGroup001.Md.GroupStatus = 4) AND (G_bAxisStart = FALSE) THEN
    U1¥G11500000 := 0;
    U4¥G11500000 := 0;
END_IF;

// Confirm that the velocity override factor is 0 and then execute the positioning start
IF (RD78_0010.Axis0001.Md.VelocityOverride = 0.0) AND (RD78_0040.AxesGroup001.Md.VelocityOverride = 0.0) THEN
    MC_MoveAbsolute_1.Execute := TRUE;
    MCv_MoveLinearInterpolateAbsolute_1.Execute := TRUE;
END_IF;

// Confirm that Active of all Motion control FBs becomes ON
IF (MC_MoveAbsolute_1.Active = TRUE) AND (MCv_MoveLinearInterpolateAbsolute_1.Active = TRUE) THEN
    G_bAxisStart := TRUE; // Set global labels of BOOL type to TRUE
END_IF;

// FB setting of the programmable controller
   MCv_AllPower_1(
        Axis:= RD78_0010.Axis0001.AxisRef );
   MCv_AllPower_2(
        Axis:= RD78_0040.Axis0001.AxisRef );
   MC_GroupEnable_1(
        AxesGroup:= RD78_0040.AxesGroup001.AxesGroupRef );
   MC_MoveAbsolute_1(
        Axis:= RD78_0010.Axis0001.AxisRef ,
        Position:= 10000.0 ,
        Velocity:= 1000.0 ,
        Acceleration:= 1000.0 ,
        Deceleration:= 1000.0 ,
        Direction:= 1 );
   MCv_MoveLinearInterpolateAbsolute_1.LinearAxes[0] := 1;
   MCv_MoveLinearInterpolateAbsolute_1.LinearAxes[1] := 2;
   MCv_MoveLinearInterpolateAbsolute_1.Position[0] := 10000.0;
   MCv_MoveLinearInterpolateAbsolute_1.Position[1] := 10000.0;
   MCv_MoveLinearInterpolateAbsolute_1.Direction[0] := 1;
   MCv_MoveLinearInterpolateAbsolute_1.Direction[1] := 1;
   MCv_MoveLinearInterpolateAbsolute_1(
        AxesGroup:= RD78_0040.AxesGroup001.AxesGroupRef ,
        Velocity:= 1000.0 ,
        Acceleration:= 1000.0 ,
        Deceleration:= 1000.0 );
```

■ Event (Inter-module synchronous interrupt program (I44)) [CPU module]

ST program

```
// Set the velocity override factor 10 to arbitrary buffer memory (user setting area)
IF G_bAxisStart = TRUE THEN
   // Actual setting value becomes one-tenth
      U1¥G11500000 := 10;
      U4¥G11500000 := 10;
END_IF;
```

■ Normal execution type program [Motion module 1] [Motion module 2]

ST program

```
// Interruption enabled
bEIout := EI(TRUE);
```

■ Fixed scan execution type program [Motion module 1]

ST program

```
// Set the one-tenth value of the arbitrary buffer memory (user setting area) setting value to the velocity override factor
wVelocityOverride_10 := G11500000;
leVelocityOverride_10 := INT_TO_LREAL( wVelocityOverride_10 );
leVelocityOverride := leVelocityOverride_10 / 10.0;
Axis0001.Cd.VelocityOverride := leVelocityOverride;
```

■ Fixed scan execution type program [Motion module 2]

ST program

```
// Set the one-tenth value of the arbitrary buffer memory (user setting area) setting value to the velocity override factor
wVelocityOverride_10 := G11500000;
leVelocityOverride_10 := INT_TO_LREAL( wVelocityOverride_10 );
leVelocityOverride := leVelocityOverride_10 / 10.0;
AxesGroup001.Cd.VelocityOverride := leVelocityOverride;
```

Simultaneous control with multiple modules

■ Scan program [CPU module]

ST program

```
// PLC READY ON
   Y10 := TRUE;
   Y40 := TRUE;

// Interruption enabled
   EI(TRUE);

// Confirm that Ready and Synchronization flag becomes ON
IF (X10 = TRUE) AND (X11 = TRUE) AND (X40 = TRUE) AND (X41 = TRUE) THEN
     // Servo ON
     MCv_AllPower_1.Enable := TRUE;
     MCv_AllPower_1.ServoON := TRUE;
     MCv_AllPower_2.Enable := TRUE;
     MCv_AllPower_2.ServoON := TRUE;
END_IF;

// Set AxisName.Cd.Encoder_Connect to TRUE
   RD78_0010.Axis0001.Cd.Encoder_Connect := TRUE;
   RD78_0040.Axis0001.Cd.Encoder_Connect := TRUE;

// Start the synchronous control FB of all modules after AxisName.Md.Encoder_Connected becomes ON
IF (RD78_0010.Axis0001.Md.Encoder_Connected = TRUE) AND
     (RD78_0040.Axis0001.Md.Encoder_Connected = TRUE) AND
     (RD78_0010.Axis0001.Md.AxisStatus = 4) AND
     (RD78_0040.Axis0001.Md.AxisStatus = 4) AND
     (RD78_0010.Axis0002.Md.AxisStatus = 4) AND
     (RD78_0040.Axis0002.Md.AxisStatus = 4) THEN
     MC_CamIn_1.Execute := TRUE;
     MC_CamIn_2.Execute := TRUE;
END_IF;

// Confirm that Active and InSync of synchronous control FB of all modules become ON
IF (MC_CamIn_1.Active = TRUE) AND (MC_CamIn_1.InSync = TRUE) AND
     (MC_CamIn_2.Active = TRUE) AND (MC_CamIn_2.InSync = TRUE) THEN
     G_bAxisStart := TRUE; // Set global labels of BOOL type to TRUE
END_IF;

// FB of the programmable controller
MCv_AllPower_1( Axis:= RD78_0010.Axis0001.AxisRef );
MCv_AllPower_2( Axis:= RD78_0040.Axis0001.AxisRef );
MC_CamIn_1.CamTableID.ProfileID.Number := 1;
MC_CamIn_1(
     Master:= RD78_0010.Axis0001.AxisRef,
     Slave:= RD78_0010.Axis0002.AxisRef );
MC_CamIn_2.CamTableID.ProfileID.Number := 1;
MC_CamIn_2(
     Master:= RD78_0040.Axis0001.AxisRef,
     Slave:= RD78_0040.Axis0002.AxisRef );
```

■ Event (Inter-module synchronous interrupt program (I44)) [CPU module]

ST program

```
// Write arbitrary values (position data of the virtual virtual encoder) into arbitrary buffer memory (user setting area)
IF G_bAxisStart = TRUE THEN
      IF dEncoderInputValue < 10000 THEN
            dEncoderInputValue := dEncoderInputValue + 1;
            U1\G11500000:D := dEncoderInputValue;
            U4\G11500000:D := dEncoderInputValue;
      END_IF;
END_IF;
```

■ Normal execution type program [Motion module 1] [Motion module 2]

ST program

```
// Interruption enabled
bEIout := EI(TRUE);
```

12

■ Fixed scan execution type program [Motion module 1] [Motion module 2]

ST program
// Write the setting value of arbitrary buffer memory (user setting area) into Encoder Input Value Axis0001.Cd.Encoder_InputValue := G11500000:D;

Precautions

- When performing the simultaneous start using the inter-module synchronization, note that the inter-module synchronous cycle needs to be the same as the buffer memory refresh cycle.
  When the inter-module synchronous cycle is different from the buffer memory refresh cycle, the data might be inconsistent. Perform mutual exclusion in a sequence program and a motion program.
- The maximum cycle is 128 ms assured by the inter-module synchronization. When using the instruction or the fixed cycle program which are operated in the cycle exceeded 128 ms, the synchronization among the modules are not assured.
- When using the inter-module synchronization in the multiple Motion module, even if PLC READY is changed OFF to ON at the same timing, the number of axis to used and required time to analyze the program so that READY does not become ON simultaneously. When performing the simultaneous start using the inter-module synchronization, execute the positioning start after READY becomes ON in the all modules performing the simultaneous start.
- When the CPU module status is changed from RUN to STOP, the buffer memory may not be updated for few cycles.
- When the inter-module synchronization is enabled, the public labels are refreshed at END except executing the inter-module synchronous interrupt program (I44). The refresh setting of the module parameter (Motion module) is disabled.

Relevant add-ons

The following add-on is required to use this function.

- baseSystem (version "1.16" or later)

The following boot software is required to use this function.

- Network boot software: version "7" or later
- Boot software: version "7" or later

When the Motion module installed the add-ons and the boot whose versions is earlier than the above is the target of the inter-module synchronization, an error occurs on the CPU modules.

Combination of the version

The versions of the engineering tool which is compatible with this function is as follows.

Engineering tool	Version
GX Works3	"1.075D" or later

Combination of the network modules

The network modules which are compatible with this function are as follows.

○: Supported, ×: Not supported

Inter-module synchronous master	Inter-module synchronous slave	Support information	Remark
RD78G(H)	RD78G(H)	○	—
	RJ71GN11-T2	×	The error "Inter-module Synchronization Master Setting Error" (error code: 3603H) occurs on the inter-module synchronous slave.
RJ71GN11-T2	RD78G(H)	○	—

12.5 Master-Slave Operation

Using the "master-slave operation function" of the servo amplifier, the "master-slave operation function" controls the master axis with the motion system, while the slave axes are controlled by data communication between servo amplifiers (driver communication) without using the motion system.
The master axis is specified from one of the real drive axes.
The slave axis is specified from a motion control station for which axes are not assigned.
One master axis can control up to eight slave axes.
To use the master-slave operation function, refer to the following combinations of model and version.

Item	Model	Version
Motion module	• RD78G_ • RD78GHV • RD78GHW	Add-on baseSystem version "1.28" or later
Servo amplifier	• MR-J5-_G_(-RJ) • MR-J5D1-_G_	"D0" or later*1

*1 When a servo amplifier model that does not support master-slave operation or a servo amplifier installed with a version that does not support master-slave operation is used mistakenly, the servo alarm [AL. 037.1_Parameter setting range error] will occur in the servo amplifier.

The following shows the number of axes that can be set when the maximum number of master axes are set for each communication period interval.

Communication period interval setting*1	Driver communication cycle*2	Motion control station		Number of standard stations*4	Total number of stations*5
		Number of master axes*3	Number of slave axes		
31.25 [μs]*6	Not supported	—	—	—	—
62.50 [μs]*6		—	—	—	—
125.00 [μs]	125.00 [μs]	1	1	0	2
250.00 [μs]		3	3	0	6
500.00 [μs]		5	5	0	10
1000.00 [μs]	250.00 [μs]	7	7	5	19
2000.00 [μs]		12	12	6	30
4000.00 [μs]		13	14	6	33
8000.00 [μs]*6	Not supported	—	—	—	—

*1 For operation cycles which can be set for the system configuration to be used, refer to the following.
☞ Page 414 Communication period setting
*2 This is the communication period of master-slave operation. For details, refer to the following.
☞ Page 414 Communication period setting
*3 This is based on "A rough standard for the number of axes and the operation cycle settings when using the positioning control".
(☞ Page 107 A rough standard for the number of axes and the operation cycle settings when using the positioning control)
*4 This is calculated based on the remote I/O module (model: NZ2GN12A42-16DTE, RX: 16 points, RY: 16 points, RWr: 4 points, RWw: 4 points).
*5 This is the number of stations when the communication mode is set to "unicast mode", communication speed is set to "1 Gbps", and MR-J5-G is connected without using a TSN hub. If this exceeds the standard number of stations, the connection becomes unstable. (For example, errors such as cyclic transfer failure occur.)
*6 If this value is set to "Communication Period Interval Setting" under "Basic Settings" of the module parameter, "Driver Communication Period Interval Setting Error (error code: 3236H)" occurs.

Operation of this function for each system status

○: Possible, ×: Not possible

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

Relevant variables

Axis parameter constant (AxisName.PrConst.)

Variable/Structure name	Name	Details
MasterSlave_Setting	Master-slave Operation Setting	Sets the station address of the slave axis to be used in master-slave operation. Master-slave operation setting (MasterSlave_Setting) is the structure of MASTER_SLAVE_SETTING type.

Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
MasterSlave_ReadyOnGroup	Master-slave Operation - Axes Group Ready	TRUE is stored for each group when the NMT state of the master-slave operation axes have all changed to Operational. Always becomes FALSE when "Master-slave Operation Setting" is set to "Not used" in module parameter (Motion) of GX Works3, or the axis emulate function is enabled. • FALSE: Not READY • TRUE: READY

System monitor data (System.Md.)

Variable/Structure name	Name	Details
MasterSlave_AxisSearching	Master-slave Operation - Searching for Axes	Indicates whether the master-slave operation axis search for the entire system is complete or not. Always becomes FALSE when "Master-slave Operation Setting" is set to "Not used" in module parameter (Motion) of GX Works3. • FALSE: Axis searching complete • TRUE: Searching for axes

Point

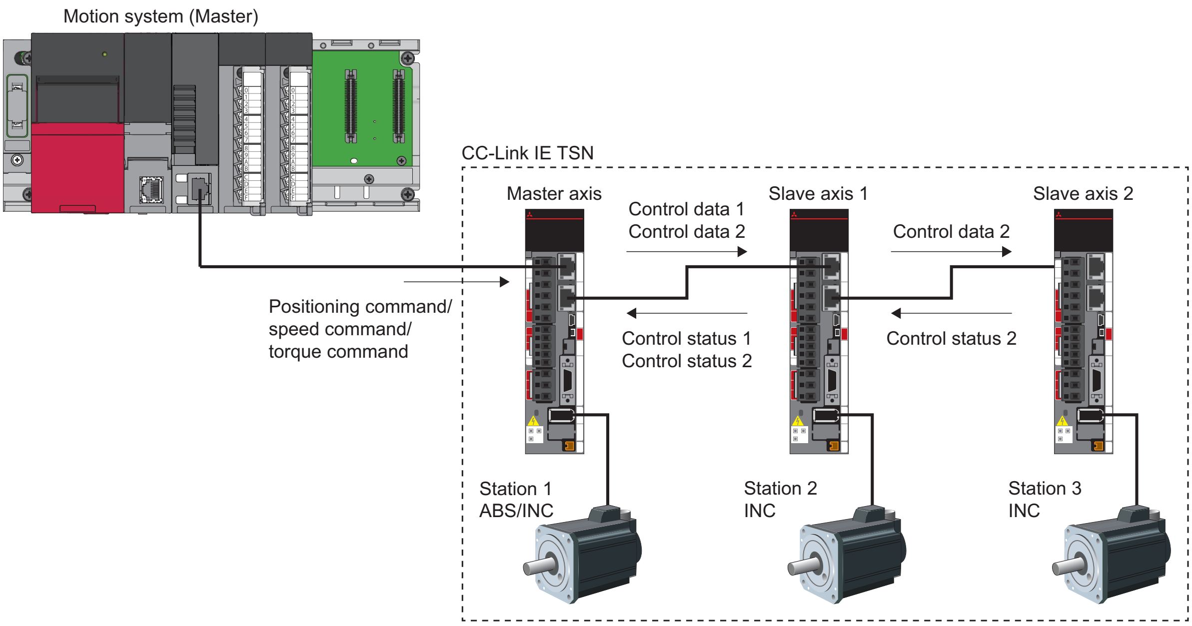
If all axes set to master-slave operation are not detected at the start of communication with the servo amplifier, all connected axes, including normal axes cannot be operated. (The LED display of the servo amplifier remains "b__" or "H__".)
When failing to connect to the servo amplifier and Master-slave operation - searching for axes (System.Md.MasterSlave_AxisSearching) is searching for axes, check that all axes that are set to master-slave operation are connected.

MASTER_SLAVE_SETTING

Variable/Structure name	Name	Details
AddressOfStation[1..8]	Slave Axis Station Address Setting	Sets the station address (IP address) of a slave axis of master-slave operation with a character string. <Example> 192.168.3.10

Control details

The servo amplifier set as the master axis receives commands (positioning command, speed command, torque command) from the motion system, and sends the control data to the servo amplifier set as slave axis by driver communication between servo amplifiers.
The servo amplifier set as the slave axis is controlled by the control data transmitted from the master axis by driver communication between servo amplifiers. Each control of the motion system is executed in the master axis.



■Master axis
- Operates according to received positioning command/speed command/torque command from the motion system.
- Sends control data to the slave axes.

■Slave axis
- Operates according to the received control data from the master axis by driver communication.
- Sends control status to the master axis. (Applicable only when the servo parameter "Master axis 1 control slave axis No. setting when the slave for driver communication is set (PD23.1)" is set)

Restriction

The inter-module synchronization function cannot be used when master-slave operation is enabled. When this function is used, "Driver Communication Setting Error 2 (error code: 3231H)" occurs.

Point

- There is no restriction on the connection order.
- Make settings to the master axis that allow it to control the slave axis so that the master axis can detect errors on the slave axis and stop control. (Set the servo parameter "Driver communication setting - slave - master axis 1 - control slave axis No. setting (PD23.1)" to "1 to 8 (Master axis control station)").
- Use this function when multiple motors are used to drive one mechanism. The master axis and slave axis should be coupled to prevent slippage.
- When slippage occurs and there is a risk of excessive movement of the slave axis, wire the upper limit signal (FLS) and the lower limit signal (RLS) to the master axis to make sure that the slave axis stops completely.

Master-slave operation control method

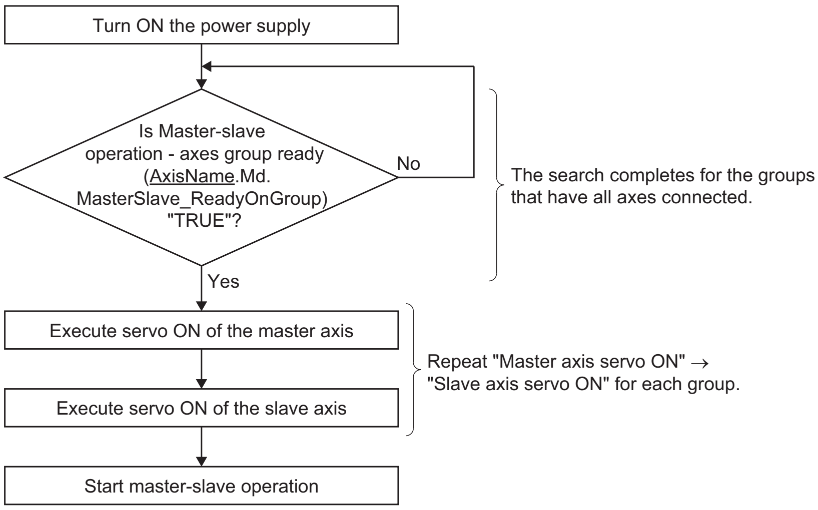
■ Checking that preparations for master-slave operation have been completed

Check if the master-slave operation control can be started.

After preparation, execute servo ON of the master axis and slave axis, and start the control of the master axis.

Check if the master-slave operation is ready with Master-slave operation - axes group ready (AxisName.Md.MasterSlave_ReadyOnGroup) of the master axis.

The procedure from connection to master-slave operation start is shown below.



Point

Only at system start, if data link is not enabled for all axes set to master-slave operation, all connected axes, including normal axes cannot be operated. (The LED display of the servo amplifier remains "b__".)

■ Servo ON

Since the master axis is a real drive axis, execute servo ON with MC_Power (Operation Available) or MCv_AllPower (All Axes Operation Available).

The slave axis executes servo ON in the user program by referring to the slave labels. For details on the user program, refer to the following.

☞ Page 865 Master-slave operation

■ Master axis control

The Motion control FB is used to control the master axis.

■ Slave axis monitoring

The slave axis performs monitoring in the motion system by referring to the slave labels.

■ Error reset

Since the master axis is a real drive axis, execute the error reset with MC_Reset (Axis Error Reset).

The slave axis executes the error reset in the Motion system by referring to the slave labels. For details on the user program, refer to the following.

☞ Page 865 Master-slave operation

Point

- Set the slave axis as a control slave axis (set the servo parameter "Driver communication setting - Slave - Master axis 1 - Control slave axis No. setting (PD23.1)" set to "1 to 8") to enable the master-slave operation simultaneous stop function. If the slave axis is not set as a control slave axis, the master-slave operation simultaneous stop function will not be enabled when an error occurs.
- If a forced stop occurs during positioning operation of the master axis, the master axis stops according to the forced stop method. The slave axis stops by the master-slave operation simultaneous stop function. For details, refer to "Master-slave operation function" of the following manual.
  📖MR-J5-G/MR-J5W-G User's Manual (Communication Function)

Slave axis operation mode

When the master-slave operation is enabled, the slave axis control mode operation is fixed as slave axis torque mode (slt).
Therefore, the "Modes of operation display (6061H)" slave object for the slave axis is fixed to "-110 (slave axis torque mode)".
For objects used in the slave axis torque mode, refer to "Master-slave operation function" of the following manual.
MR-J5-G/MR-J5W-G User's Manual (Communication Function)

Parameter setting

This section explains the parameters to be set in GX Works3 to perform master-slave operation.

Setting procedure

The setting procedure of the parameters is shown below.

Set the following in module parameter (Network).

For details of the parameters, refer to the following.

☞ Page 412 Module parameter (Motion)

☞ Page 413 Module parameter (Network)

☞ Page 417 Module extended parameter

In addition, refer to the following when using a TSN hub.

☞ Page 865 Master-slave operation

Operating procedure

1. Set [Module operation setting] ⇨ [Master-slave operation setting] under "Module Parameter (Motion)" to "Used".
2. Set the network configuration in the "CC-Link IE TSN Configuration" screen under [Basic Settings] ⇨ [Network Configuration Settings] ⇨ [Detailed Setting] of "Module Parameter (Network)".
3. In the "CC-Link IE TSN Configuration" screen, set the servo parameters to be automatically set to the servo amplifier. (Set the parameters related to the master-slave operation of the servo amplifier.)
4. Set parameters such as communication period settings for master-slave operation in [Basic Settings] and [Application Settings] of "Module Parameter (Network)".
5. With "Module Extended Parameter" ⇨ [Motion control setting function] ⇨ [Axis] ⇨ [Add New Data], generate a real drive axis for the servo amplifier specified as the master axis. No axis is generated for the slave axis.
   When generating an axis, do not specify the station address of the slave axis in Station address setting (AxisName.PrConst.AddressOfStation).
6. With "Module Extended Parameter" ⇨ [Motion control setting function] ⇨ [Axis] ⇨ [Real Drive Axis] ⇨ [Axis Parameter Setting], set the master-slave operation parameter settings.
7. Write the project to the CPU module and the motion system, and restart the system.
8. After the system restart, the servo parameters are automatically distributed to the servo amplifier. After the automatic parameter distribution to the servo amplifier, "Power Not Turned ON After Parameter Distribution (error code: 1C4CH)" occurs, however this is a normal operation.
9. Restart the system.

Module parameter (Motion)

Set the following parameters in module parameter (Motion) of GX Works3 to perform master-slave operation. (This section only describes the parameters to be set.)

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ [Target module] ⇨ [Module Parameter (Motion)] ⇨ "Parameter editor" screen

Item		Setting details
Module operation setting	Master-slave operation setting	Set whether to use the master-slave operation function or not.

Module parameter (Network)

Set the following parameters in module parameter (Network) of GX Works3 to perform master-slave operation. (This section only describes the parameters to be set.)

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ [Target module] ⇨ [Module Parameter (Network)] ⇨ "Parameter editor" screen

Item				Setting details
Basic settings	Network configuration settings		Motion control station	■When setting as the master axis Set to motion control station. ■When setting as the slave axis Set to motion control station.
			Parameter automatic setting	Set the servo parameters to be distributed from the motion system to the servo amplifier. For details, refer to the following. ☞ Page 415 Parameter automatic setting
			PDO mapping setting	Set the PDO mapping. For details on objects to be mapped to the slave axis, refer to the following. ☞ Page 416 Objects that require mapping for the slave axis
			Communication period setting	Set the communication period. *: When setting as the master axis or the slave axis, set to "Basic Period".
			CC-Link IE TSN Class	Set "CC-Link IE TSN ClassB" for the station set to the master axis or the slave axis.
	Communication period setting	Basic period setting	Setting in units of 1 μs	Set the communication period. For setting details of the cyclic transmission time and the communication period interval setting, refer to the following. ☞ Page 414 Communication period setting The master-slave operation supports the following communication period interval settings. • 125 μs • 250 μs • 500 μs • 1000 μs • 2000 μs • 4000 μs *: Set one of the above communication periods even when setting in units of 1 μs. When the setting value is other than the above, "Driver Communication Period Interval Setting Error (error code: 3236H)" occurs.
			Communication period interval setting (Do not set it in units of 1 μs)	
			Communication period interval setting (Set it in units of 1 μs)	
			System reservation time	
			Cyclic transmission time	
			Transient transmission time	
		Multiple period setting	Normal-speed	Set the following parameters. • Magnification at normal-speed • Magnification at low-speed *: When there is a station whose communication period setting in the network configuration settings is set to low speed, set the magnification for low speed so that it is "128 times" or less when multiplied by the driver communicationmagnification (basic). If set to larger than "128 times", "Driver Communication Setting Error 2 (error code: 3231H)" occurs. For details on the driver communication magnification (basic), refer to the following. ☞ Page 414 Communication period setting
			Low-speed	
	Connection device information	TSN HUB setting		When connecting a CC-Link IE TSN Class A device station, set "Use TSN HUB". In addition, set the total cyclic data size of CC-Link IE TSN Class A stations to "2k bytes" or less. *: When connecting CC-Link IE TSN Class A, if "Not to use TSN HUB" is set, "Driver Communication Setting Error 2 (error code: 3231H)" occurs.
Application settings	Communication speed	Communication speed		Set "1 Gbps" for the communication speed. *: If "100 Mbps" is set, "Driver Communication Setting Error 2 (error code: 3231H)" occurs in the motion system.

■ Communication period setting

Set the communication period interval setting and the cyclic transmission time as below when using master-slave operation.

- Communication period interval setting

  When using the master-slave operation, set a value equal to or more than the following formula for the communication period interval setting.

  When a value smaller than the formula value is set for the communication period interval setting, "Driver Communication Period Interval Setting Error (error code: 3236H)" occurs. When an error occurs, change the communication period interval setting according to the corrective action on the module diagnostic window.

Formula
Cyclic transmission time [μs]*1 × Driver communication magnification (basic)*2 + Cyclic processing time other than driver communication [μs]*3 + (Number of valid axes for driver communication*4 × 1.39 [μs])

*1 Indicates the cyclic transmission time set in the basic communication period setting.

*2 Indicates the value obtained by dividing the communication period interval time setting by the driver communication cycle. For details on the driver communication cycle, refer to "Driver communication cycle" in this section.

*3 For the calculation method of the cyclic processing time other than the driver communication, refer to the following.
📖MELSEC iQ-R Motion Module User's Manual (Network)

*4 Indicates the total number of stations set as the master axis and the slave axis.

- Cyclic transmission time

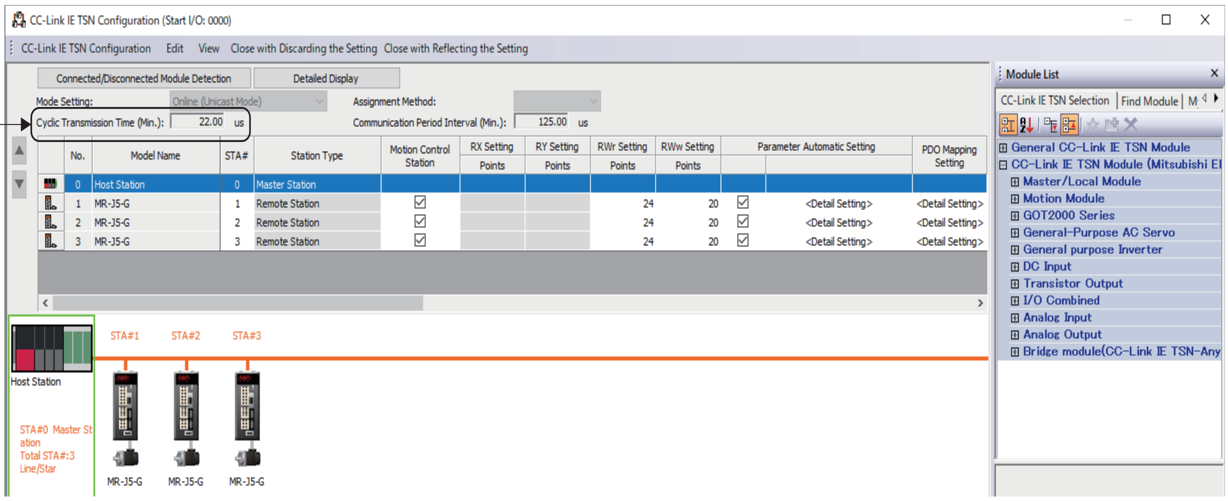
  When using master-slave operation, set the cyclic transmission time to a value equal to or more than the value obtained by the following formula, and equal to or less than 216.00 [μs].

  When a value outside the range is set for the cyclic transmission time, "Driver Communication Cyclic Transmission Time Setting Error (error code: 3235H)" occurs. When an error occurs, change the cyclic transmission time according to the corrective action on the module diagnostic window.

Formula
Cyclic processing time other than driver communication [μs]*1 + (Number of valid axes for driver communication × 2.715 [μs])

*1 The cyclic processing time other than driver communication is displayed under "Cyclic Transmission Time (Min.)" (1) in the "CC-Link IE TSN Configuration" window.
For the calculation method of the cyclic transmission time (minimum value), refer to the following.
📖MELSEC iQ-R Motion Module User's Manual (Network)

(1)



- Driver communication cycle
  The communication period (driver communication cycle) for master-slave operation is automatically set to 125 [μs] or 250 [μs] by "Communication period interval setting" in "Basic period setting".

Communication period setting	Driver communication period	Driver communication magnification (Basic)
125 [μs]	125 [μs]	1
250 [μs]	125 [μs]	2
500 [μs]	125 [μs]	4
1000 [μs]	250 [μs]	4
2000 [μs]	250 [μs]	8
4000 [μs]	250 [μs]	16

- Check the operating status of the current master-slave operation with the following link special registers (SW).

No.	Name
SW04C0	Driver communication cycle interval [μs]
SW04C1	Driver communication cycle system reservation time [μs]
SW04C2	Driver communication cycle cyclic transmission time [μs]
SW04C3	Driver communication cycle transient transmission time [μs]
SW04C4	Driver communication magnification (basic)
SW04C5	Driver communication magnification (medium-speed)
SW04C6	Driver communication magnification (low-speed)

12

■ Parameter automatic setting

Set the servo parameters for each master-slave operation station by the parameter automatic setting of GX Works3.
For the setting details, refer to the following.
MR-J5-G/MR-J5W-G User's Manual (Parameters)

- Driver communication operation setting

No.	Name	Setting details	Setting target	Setting example			
				Axis 1	Axis 2	Axis 3	Axis 4
PD15.0	Master axis operation selection	• Set the master axis. 0: Disabled (master-slave operation function is not used) 1: Enabled (set this servo amplifier for the master axis)	Master	1 (Master axis)	0	0	0
PD15.1	Slave axis operation selection	• Set the slave axis. 0: Disabled (master-slave operation function is not used) 1: Enabled (set this servo amplifier for the slave axis)	Slave	0	1 (Slave axis)	1 (Slave axis)	1 (Slave axis)

- Driver communication slave axis setting (network setting)

No.	Name	Setting details	Setting target	Setting example			
				Axis 1	Axis 2	Axis 3	Axis 4
PD22	Driver communication setting - Slave - Master axis 1 - Station No. setting	• Set the station No. of the master axis to which the slave axis belongs. 0: Disabled 1 to 255: Master axis station No.	Slave	0 (Disabled)	1 (Axis 1)	1 (Axis 1)	1 (Axis 1)
PD23.1	Driver communication setting - Slave - Master axis 1 - Control slave axis No. setting	• Set the identifier (control slave axis No.) of the own slave axis for the slave axis. • Set identifiers in order from 1 in a driver communication group (master axis to which the slave axis belongs). 0: Disabled 1 to 8: Control slave axis No.	Slave	0 (Disabled)	1 (Slave axis No.1)	2 (Slave axis No.2)	3 (Slave axis No.3)

- Driver communication slave axis setting (master-slave operation setting)

No.	Name	Setting details	Setting target	Setting example			
				Axis 1	Axis 2	Axis 3	Axis 4
PD30	Master-slave operation - Slave-side torque command coefficient	Set the command torque coefficient of the slave axis for the output torque of the master axis.	Slave	0 (Disabled)	100 [%]	100 [%]	100 [%]
PD31	Master-slave operation - Slave-side speed limit coefficient	Set the velocity limit value coefficient of the slave axis for the velocity limit value of the master axis.	Slave	0 (Disabled)	100 [%]	100 [%]	100 [%]
PD32	Master-slave operation - Slave-side speed limit adjusted value*1	Set a minimum value for velocity limit value of the slave axis.	Slave	0.00 (Disabled)	0.00 [r/min]	0.00 [r/min]	0.00 [r/min]
PV33	Master-slave operation - Speed limit adjusted value extension setting*2	Set a minimum value for velocity limit value of the slave axis.	Slave	0 (Disabled)	0	0	0

*1 This setting is valid only when the "Speed/acceleration/deceleration unit selection (PT01.1)" is set to "0".
*2 This setting is valid only when the "Speed/acceleration/deceleration unit selection (PT01.1)" is set to "1".
The unit of this parameter varies depending on the setting value of "Unit for position data (PT01.2)".

Point

- The servo parameters are transferred from the motion system to the servo amplifier after the system restart.
- The servo parameters for driver communication setting (PD15, PD22, PD23) are enabled by turning OFF to ON the power supply of the servo amplifier. After performing the above processes, turn OFF to ON the power supply of the servo amplifier, and then turn ON the power supply of the system again, or reset the CPU module.
- In the master-slave operation function, the torque generation direction for a slave axis can also be set in the servo parameter "Travel direction selection (PA14)".

■ Objects that require mapping for the slave axis

Set the PDO mapping for each master-slave operation station in the PDO mapping setting of GX Works3.
When using master-slave operation, map the following objects in the slave axis PDO. These objects are used for servo ON, alarm reset, etc.

- TPDO(RWr)

Index	Sub-index	Entry name
6041H	00H	Statusword

- RPDO(RWw)

Index	Sub-index	Entry name
6040H	00H	Controlword

The above objects are mapped to PDO as part of the default mapping pattern when [Batch Setting of PDO Mapping] is used from the [CC-Link IE TSN Configuration] menu of the "CC-Link IE TSN Configuration" screen.

Module extended parameter

The GX Works3 motion control setting function sets the following parameters. (This section only describes the parameters to be used.)

■ Axis parameters

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ [Target module] ⇨ [Module Extended Parameter] ⇨ [Motion Control Setting Function] ⇨ Navigation window ⇨ [Axis] ⇨ [Target axis] ⇨ "Axis Parameter Setting" screen

Item			Setting details
Axis parameter constant	Station address setting		Set the station address of the real drive axis set as the master axis. *: When a station address for other than the master axis is set in the network configuration setting, "Driver Communication Setting Error 1 (error code: 3230H)" occurs. *: When generating an axis, do not specify the station address of the slave axis in Station address setting (AxisName.PrConst.AddressOfStation). If the station address of the slave axis is set, "Slave Axis Specification Error (error code: 1C51H)" occurs.
	Master-slave operation setting	Slave axis station address setting [1]	Set the station address of the axis to be set as a slave axis. Up to eight slave axes can be set. In the following cases, "Out of Parameter Range (Axis) (error code: 1D80H)" occurs. • The station address for the slave axis is duplicated in the network configuration settings. • The slave axis station address setting is set when master-slave operation of the module parameter (Motion) is set to "Not used". In the following cases, "Driver Communication Setting Error 1 (error code: 3230H)" occurs. • The station address of the master axis is set in the network configuration settings. • The station address of the slave axis is not set in the network configuration settings. • When the master-slave operation setting is set to "Used" in the module parameter (Motion), the slave axis station address setting is not set.
		Slave axis station address setting [2]	
		Slave axis station address setting [3]	
		Slave axis station address setting [4]	
		Slave axis station address setting [5]	
		Slave axis station address setting [6]	
		Slave axis station address setting [7]	
		Slave axis station address setting [8]	
	Axis emulate enabled		Set to "Disabled" when using the master-slave operation function. Operates as a normal axis when set to "Enabled".

■ Network I/O

On the network I/O screen, generate the slave labels to be used from the program.

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ [Target module] ⇨ [Module Extended Parameter] ⇨ [Motion Control Setting Function] ⇨ Navigation window ⇨ [Network I/O] ⇨ "Network I/O" screen

The following slave axis objects are set as targets for slave label generation.

- Statusword
- Controlword

These objects are used for servo ON, alarm reset, etc. of the slave axis.
For program example of the procedure for servo ON or alarm reset of the slave axis, refer to the following.
☞ Page 865 Master-slave operation

In addition, when the following objects are mapped to PDO of the slave axis, set them as targets for slave label generation.

- Positive torque limit value
- Negative torque limit value

If no slave labels are generated for these objects of the slave axis, the torque limit value is fixed to 0%. Set an arbitrary torque limit value for the slave label generated on the program.
Also set other slave labels used for controlling and monitoring the slave axis as targets for slave label generation as required.

Communication period setting of TSN hub

When using a TSN hub, set the communication period interval and time for each timeslot by referring to the following values of the link special register (SW) when the driver communication parameters are set correctly.

No.	Name
SW04C0	Driver communication cycle interval [μs]
SW04C1	Driver communication cycle system reservation time [μs]
SW04C2	Driver communication cycle cyclic transmission time [μs]
SW04C3	Driver communication cycle transient transmission time [μs]

Precautions

Servo amplifier

- For the axes where master-slave operation is executed, use a servo amplifier that supports master-slave operation.
- When any axis used for master-slave operation is disconnected, disconnect all axes in the group of the master-slave operation to which the disconnected axis belongs in order to reconnect the axis, and then reconnect the axis.

■ Disconnection operation

Disconnection can be executed by either of the following methods.

- Disconnect the Ethernet cable from the Ethernet cable connector of a servo amplifier.
- Select the servo amplifier to be disconnected and execute a remote reset on the "CC-Link IE TSN/CC-Link IE Field Diagnostics" screen of GX Works3.

■ Reconnection operation

Reconnection can be executed by the following method:

- Reconnect the slave axis (execute servo ON again) in the user program.

*1 For details on the user program, refer to the following.
☞ Page 865 Master-slave operation

Point

- When communication fails due to a failure of a servo amplifier, communication with the servo amplifiers after the failed servo amplifier cannot be executed. For this reason, build the system using a star or ring topology using a TSN hub to minimize the effects.
- To stop operating when an error occurs, enable the master-slave operation simultaneous stop function, and set the slave axis as a control slave axis (set the servo parameter "Driver communication setting - Slave - Master axis 1 - Control slave axis No. setting (PD23.1)" to "1 to 8").
- If a forced stop occurs during positioning operation of the master axis, the master axis stops according to the forced stop method. The slave axis stops by the master-slave operation simultaneous stop function.

Software reboot

When executing reset after changing the slave axis station address determined at power ON, "Driver Communication Setting Error 1 (error code: 3230H)" occurs.

Motion control FB

When the master axis switches to continuous operation to torque control mode (ct) using MC_TorqueControl (Torque Control), "Control Mode Switching Disabled Warning (warning code: 0D31H)" occurs and the master axis operates in the current control mode.

Relevant add-ons

The following add-ons are required to use this function.

- NetworkDriver_CCIETSN
- ServoDriver_CANopen
- Axis
- baseSystem

Combination of the version

For details on the version, refer to the following.
☞ Page 875 Compatible versions for each Motion module function

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 axis (Slave) that is synchronized with Master axis (Master).

FB	Control details
MC_CamIn (Cam Operation Start)	Executes cam operation.
MC_GearIn (Gear Operation Start)	Sets the speed ratio between the master axis and the slave axis, and enters gear operation.
MC_CombineAxes (Addition/Subtraction Positioning)	Combines operations of two axes by a selectable combination method, and outputs it to the third axis.
MCv_ChangeCycle (Current Value Change per Cycle)	Changes the cam current value per cycle to the specified value during MC_CamIn (Cam Operation Start) control. It is used to interpolate the cam current value per cycle into an arbitrary value.
MCv_BacklashCompensationFilter (Backlash Compensation Filter)	Carries out the specific filter processing to input of Master axis (Master), and outputs the result to Slave axis (Slave). ☞ Page 363 Command Filter
MCv_SmoothingFilter (Smoothing Filter)	
MCv_DirectionFilter (Moving Direction Restriction Filter)	
MCv_SpeedLimitFilter (Speed Limit Filter)	
MCv_AdvancedSync (Advanced Synchronous Control)	Starts synchronous control based on the specified advanced synchronous control settings. ☞ Page 489 ADVANCED SYNCHRONOUS CONTROL

13.2 Axis Configuration

The following shows axes that can be specified to Master axis (Master) and Slave axis (Slave) in the single axis synchronization control FBs. For the specifications of each axis type, refer to the following.

☞ Page 30 Axis

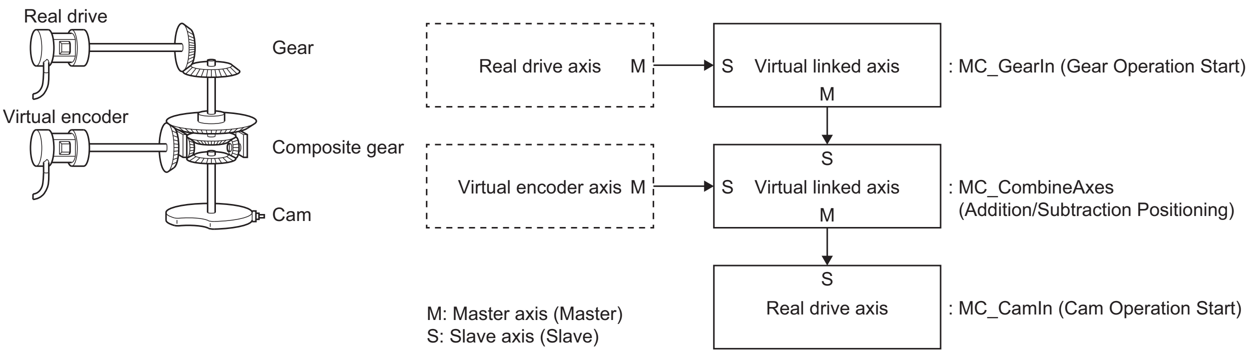
☞ Page 56 Axis Assignment

Axis type		Used for Master axis (Master)	Used for Slave axis (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 axis (Master). When it is used as Slave axis (Slave), "Necessary Slave Object Unset (error code: 1AA8H)" occurs and it does not start.
Virtual axis	Virtual drive axis	○	○	Commands can be mainly created by positioning control, etc.
	Virtual encoder axis	○	×	It is used as Master axis (Master). When it is used as Slave axis (Slave), "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.

To combine and execute controls such as gear and cam, use the virtual linked axis like the following example.

Ex.

To process gear following the instruction of the real drive axis and perform cam control adding the instruction from the virtual encoder



Position command unit of the master axis and the slave axis

Settings of position command unit of the master axis and the slave axis (Position Command Unit (AxisName.Pr.Unit_Position) and Position Command Unit String (AxisName.Pr.Unit_PositionString)) do not affect the control. The result of the calculation performed to the position information of the master axis which has no unit information is used as an instruction of the slave axis.

Ex.

To add each master axis by gear ratio 1 : 1 using MC_CombineAxes (Addition/Subtraction Positioning)

The slave axis operates 3.0 [degree] with the following settings of position command unit of each axis when the master axis 1 is moved 1.0 [pulse] and the master axis 2 is moved 2.0 [mm].

· Position command unit of the master axis 1: pulse

· Position command unit of the master axis 2: mm

· Position command unit of the slave axis: degree

Current position change of the master axis

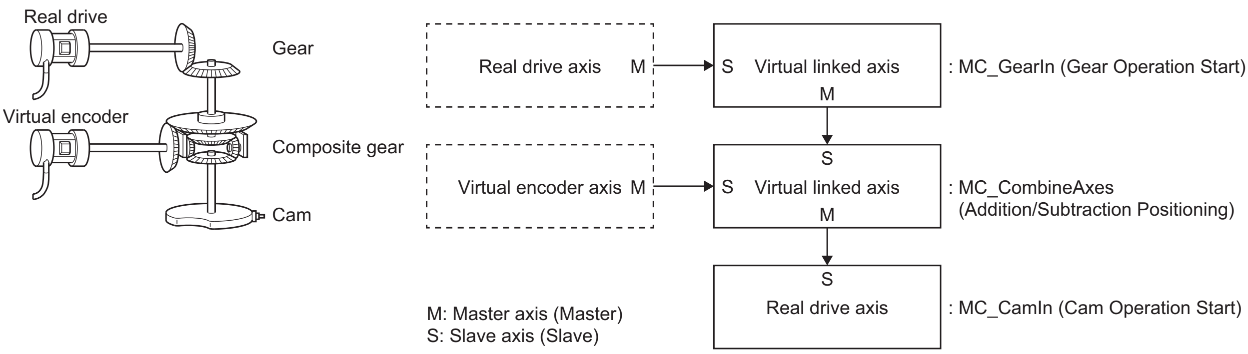
The set position of the slave axis does not change even when the current position change of the master axis is performed.

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 Master axis data source selection (MasterValueSource) of each FB.
When the actual position is specified for virtual axis (axis which does not have Actual Position (AxisName.Md.ActualPosition)), the FB operates by the value as the same with set value.
"101: Latest Set Value (mcLatestSetValue)" and "102: Latest Actual Value (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: Latest Set Value (mcLatestSetValue)" and "102: Latest Actual Value (mcLatestActualValue)" can transmit the multiple Motion control FB command in the same operation cycle. In this case, set the first call order and linking order of the linked Motion control FB 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 call order as follows.
- First call order

1. Motion control FB which moves the master axis (real drive axis) of MC_GearIn (Gear Operation Start)
2. MC_GearIn (Gear Operation Start)
3. MC_CombineAxes (Addition/Subtraction Positioning)
4. MC_CamIn (Cam Operation Start)

- Master axis data source selection (MasterValueSource) of each single synchronous control FB
  "101: Latest Set Value (mcLatestSetValue)"

Point

When Master axis data source selection (MasterValueSource) is set to "1: Set Value (mcSetValue)" or "101: Latest Set Value (mcLatestSetValue)" and the master axis becomes servo OFF by a servo alarm or forced stop, the amount of the value change may be too large. This can be prevented by setting Master axis data source selection (MasterValueSource) to "2: Actual Value (mcActualValue)" or "102: Latest Actual Value (mcLatestActualValue)".

Master axis data source selection (MasterValueSource)	Description
1: Set Value (mcSetValue)	Uses the commanded position of the master axis in the previous operation cycle.
2: Actual Value (mcActualValue)	Uses the actual position of the master axis in the previous operation cycle.
101: Latest Set Value (mcLatestSetValue)	Uses the commanded position of the master axis in the current operation cycle.
102: Latest Actual Value (mcLatestActualValue)	Uses the actual position of the master axis in the current operation cycle.

13.4 Start and Stop Operation

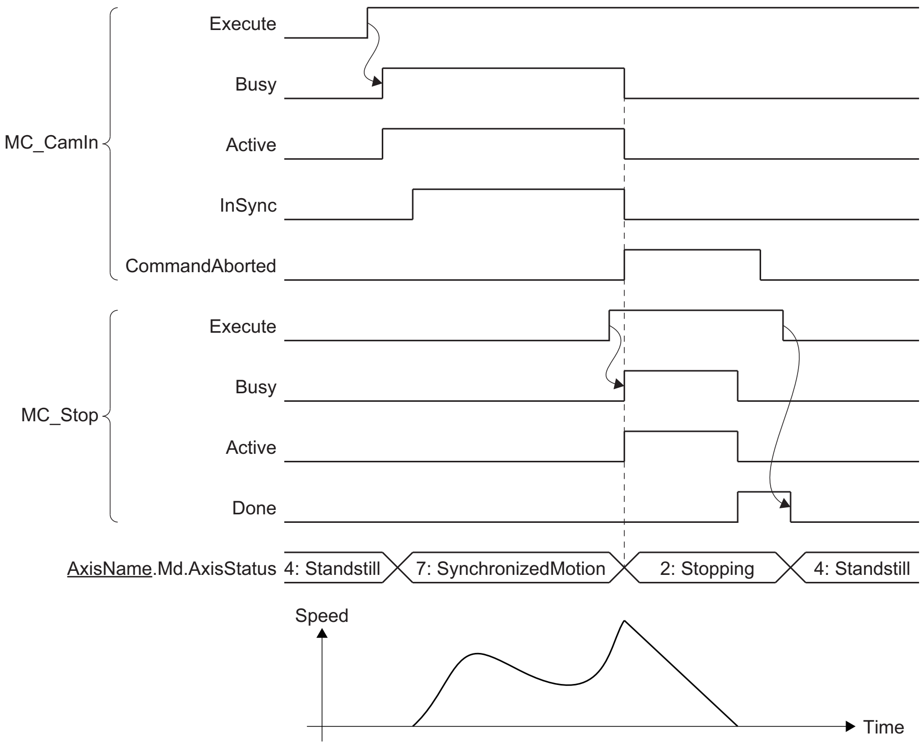
Slave axis (Slave) continues to control with the command transmitted by Master axis (Master) even if a stop cause occurs during execution of the synchronous control function blocks. The axis status of the slave axis does not change.
When a stop cause occurs to the slave axis, the axis status transits 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 166 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 (Forced 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 axis (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 "Necessary Slave Object Unset (error code: 1AA8H)" occurs and it will not start.

13.6 Restrictions

- When Master axis (Master) and Slave axis (Slave) are specified to the same axis, "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 axis (Slave) of the latter function block or more than 257 synchronous control FBs are linked, "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 open area and set to a state where it can be controlled.
For the open method to the open area, and for data types that can be controlled in cam operation, refer to the following.
☞ Page 469 Operating operation profile data
☞ Page 446 Operation profile data types

Relevant variable

Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
ProfileID	Execution Profile ID No.	Stores the executing Cam tableID (CamTableID). The update timing is as follows. ■Stores the specified Cam tableID (CamTableID) • When executing MC_CamIn (Cam Operation Start) (at the timing when the slave axis status becomes "7: During synchronous operation (SynchronizedMotion)") • When changing Cam tableID (CamTableID) The timing differs depending on the setting value of Start mode (StartMode). ■Cleared to "0" • When Cam cycle completion (EndOfProfile) becomes TRUE with the setting which Periodic (Periodic) is "FALSE: Non periodic" and Specifying operation after one cycle operation (Options (Options): Bit 16) is "0: Complete" • When stop cause occurs

Relevant FB

For details on Motion control FBs, refer to the following.
📖MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

Motion control FB	Name	Description
MC_CamIn	Cam operation start	Cam operation starts operation according to the specified cam table.

Control details

MC_CamIn (Cam Operation Start) sets Master axis offset (MasterOffset), Slave axis offset (SlaveOffset), Master axis scaling (MasterScaling), Slave axis scaling (SlaveScaling), Master axis follow-up distance (MasterStartDistance), Master axis synchronization start position (MasterSyncPosition), Start mode (StartMode), Master axis data source selection (MasterValueSource), Cam tableID (CamTableID), and Buffer mode (BufferMode) and then executes the cam operation.
To stop the operation, execute MC_Stop (Forced Stop).

Starting cam operation

Synchronous timing of the cam operation and reflection timing of the cam control data can be selected by setting of Start mode (StartMode).

■ Cam operation synchronous timing

Settings of Start mode (StartMode), Master axis synchronization start position (MasterSyncPosition), and Master axis follow-up distance (MasterStartDistance)) decide the synchronous start timing of Current value per cycle (InputPerCycle), Reference value (Reference), and Output value (OutputData) as follows.

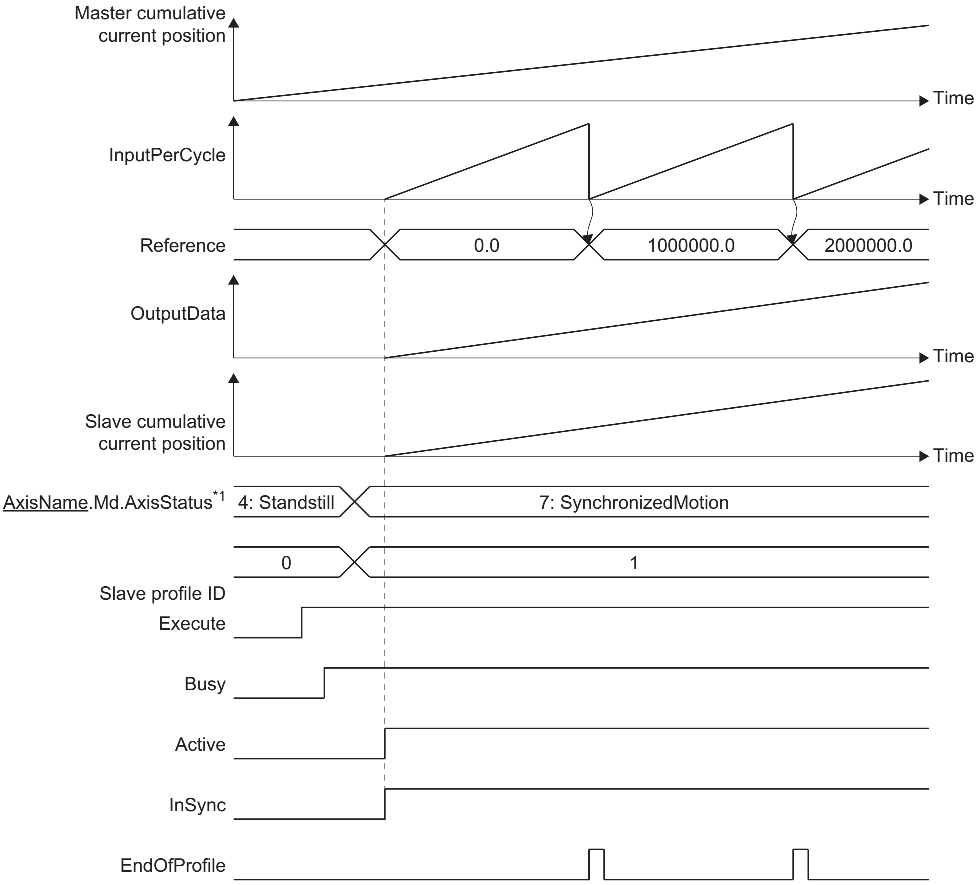
Start mode (StartMode)	Synchronous start timing	
	Current value per cycle (InputPerCycle)	Reference value (Reference), Output value (OutputData)
0: Immediate (mcImmediate)	When executing MC_CamIn (Cam Operation Start) (Ignores Master axis synchronization start position (MasterSyncPosition) and Master axis follow-up distance (MasterStartDistance))	
1: Absolute (mcAbsolute)	When the position*1 of master axis passes*2 Master axis synchronization start position (MasterSyncPosition) (Ignores Master axis follow-up distance (MasterStartDistance))	

*1 Set the position data to be used based on Specifying target of Master axis synchronization start position pass check (Option (Options): Bit 21) of MC_Camin (Cam Operation Start).
*2 The position of the Master axis has crossed over Master axis synchronization start position (MasterSyncPosition)(or, the Master axis synchronization start position (MasterSyncPosition) + Master axis follow-up distance (MasterStartDistance)). When these values are the same, the Master axis is considered to have not passed, and should the Master axis move from this value, it will be considered to have passed.

After MC_CamIn (Cam Operation Start) is executed, Controlling (Active) becomes TRUE after Current value per cycle (InputPerCycle) is synchronized, and then In synchronization (InSync) becomes TRUE after Output value (OutputData) is synchronized.

■ When Start mode (StartMode) is "0: Immediate (mcImmediate)"

The slave axis start moving after start of Execute command (Execute).

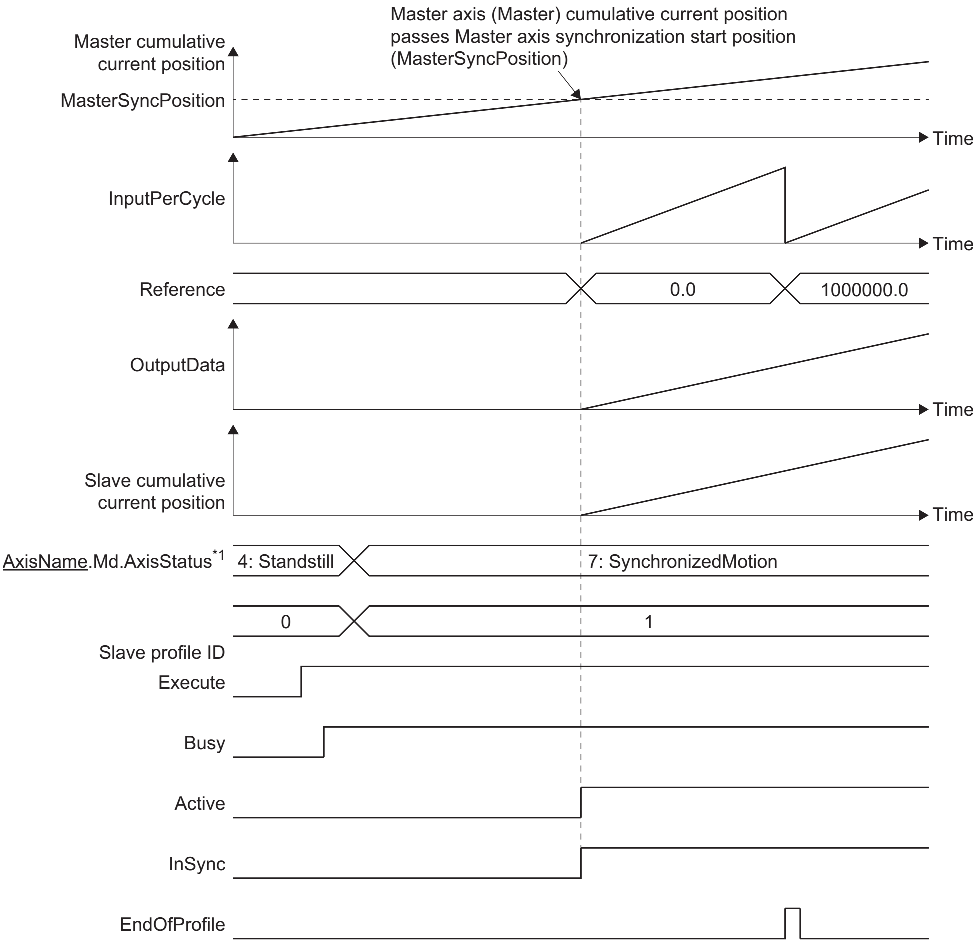


*1 This is the state of Slave axis (Slave).

■ When Start mode (StartMode) is "1: Absolute (mcAbsolute)"

After start of Execute command (Execute), the slave axis start moving after the cumulative current position of the master axis passes Master axis synchronization start position (MasterSyncPosition).

Checking of whether the Master axis cumulative current position has passed the Master axis synchronization start position (MasterSyncPosition) begins at the point when the Slave axis status becomes "7: During synchronous operation (SynchronizedMotion)".



*1 This is the state of Slave axis (Slave).

■ Timing of control change by FB restart/continuous update

Timing of reflecting changed parameters by FB restart/continuous update onto the control vary depending on Start Mode (StartMode) like the following.

Start Mode (StartMode)	Timing for reflecting parameters
0: Immediate (mcImmediate)	Immediate
1: Absolute (mcAbsolute)	When the current value per cycle (InputPerCycle) passes the first point on the CamTable

When the changed values exceed the range, the warning is output and the control is maintained by keeping the value before.

Parameter	Warning
Master axis offset (MasterOffset)	Out of spindle offset range (Synchronous Control Variable Out-of-Range Warning (warning code: 0D49H (detail code: 0001H)))
Slave axis offset (SlaveOffset)	Out of the follow shaft offset range (Synchronous Control Variable Out-of-Range Warning (warning code: 0D49H (detail code: 0002H)))
Master axis scaling (MasterScaling)	Out of spindle coefficient range (Synchronous Control Variable Out-of-Range Warning (warning code: 0D49H (detail code: 0003H)))
Slave axis scaling (SlaveScaling)	Out of the dependent coefficient range (Synchronous Control Variable Out-of-Range Warning (warning code: 0D49H (detail code: 0004H)))
Cam tableID (CamTableID)	■When the CamTableID (Cam tableID) numerical values are outside the range Out of cam table ID range (Synchronous Control Variable Out-of-Range Warning (warning code: 0D49H (detail code: 0005H))) ■When there is no CamTableID (Cam tableID) open area No Cam Table Warning (warning code: 0D44H)

Cam table

This section describes the cam data and the cam operation when MC_CamIn (Cam Operation Start) is executed.

○: Used in control, ×: Not used in control

Name	Variable name	Interpolation Method Specification (Interpolate)			Remark
		0: Linear Interpolation	1: Section Interpolation	2: Spline Interpolation	
Periodic	Periodic	○	○		The item set in the expand setting of the operation profile data or set in input variables of MC_CamTableSelect (Cam Table Selection) on the engineering tool
Input absolute coordinate	MasterAbsolute	○	○		
Output absolute coordinate	SlaveAbsolute	○	○		
Interpolation method specification	Interpolate	○	○		The item related to control among the data expanded to open area from the operation profile data of cam data format/rotary cutter format
Length per cycle	CycleLength	○	○		
Stroke amount	Stroke	○	○		
Start point	StartPoint	×	×		
Initial stroke amount	StartStroke	×	○		

■ Periodic (Periodic)

Periodic (Periodic) can set "FALSE: Non periodic" and "TRUE: Periodic (Periodic)". The following explains the operation when executing MC_CamIn (Cam Operation Start).

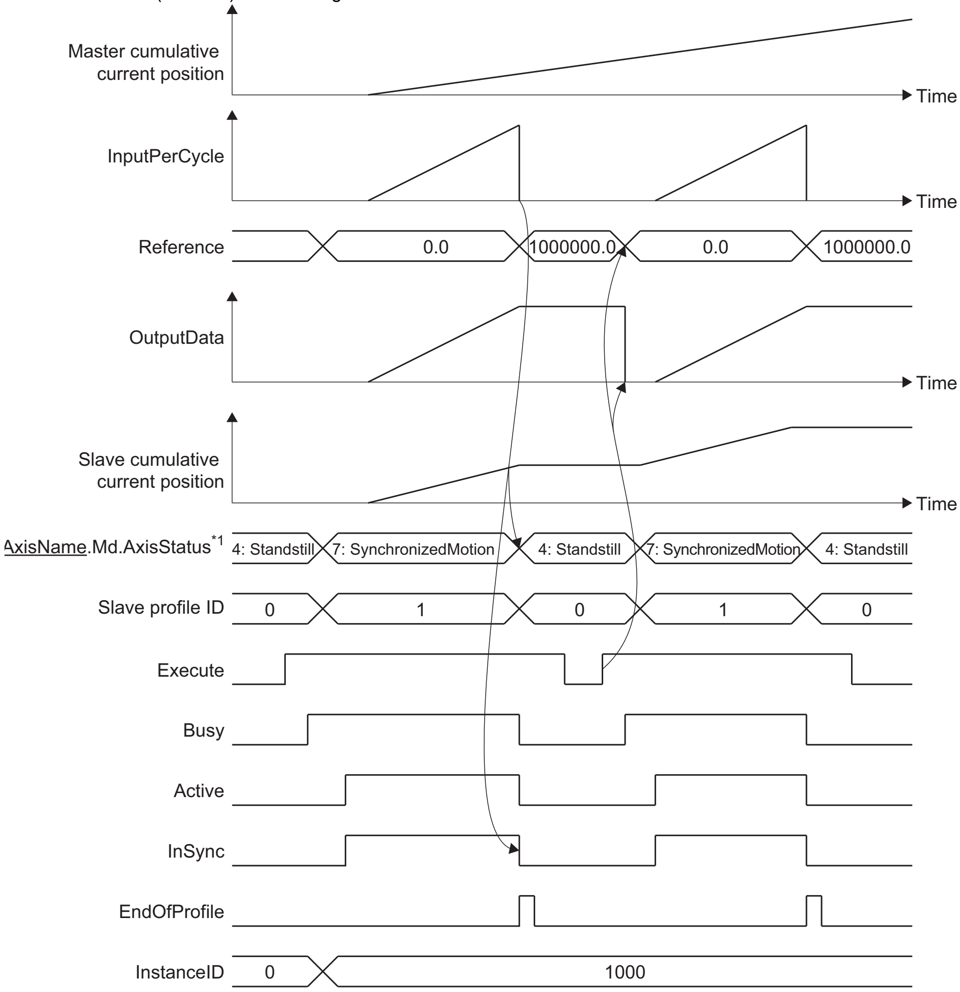
- Periodic (Periodic) "FALSE: Non periodic" is specified

  Only one cycle operation is carried out after Controlling (Active) becomes TRUE.The next operation in Specifying operation after one cycle operation (Options (Options): Bit 16).

Specifying operation after one cycle operation (Options (Options): Bit 16)	Operation after cycle operation				Reference value (Reference), Output value (OutputData) at restart
	Executing (Busy)	Controlling (Active)	In synchronization (InSync)	Axis Status (AxisName.Md.AxisStatus)	
0: Complete	FALSE			"4: Standby (Standstill)"	Initialized
1: Wait for restart	TRUE			Maintains "7: During synchronous operation (SynchronizedMotion)"	Maintained

■When Specifying operation after one cycle operation (Options (Options): Bit 16) is "0: Complete"

Synchronization status is canceled after one cycle, and Reference value (Reference) and Output value (OutputData) change to 0.0 (Default value) when Execute command (Execute) is started again.



*1 This is the state of Slave axis (Slave).

13

■When Specifying operation after one cycle operation (Options (Options): Bit 16) is "1: Wait for restart"

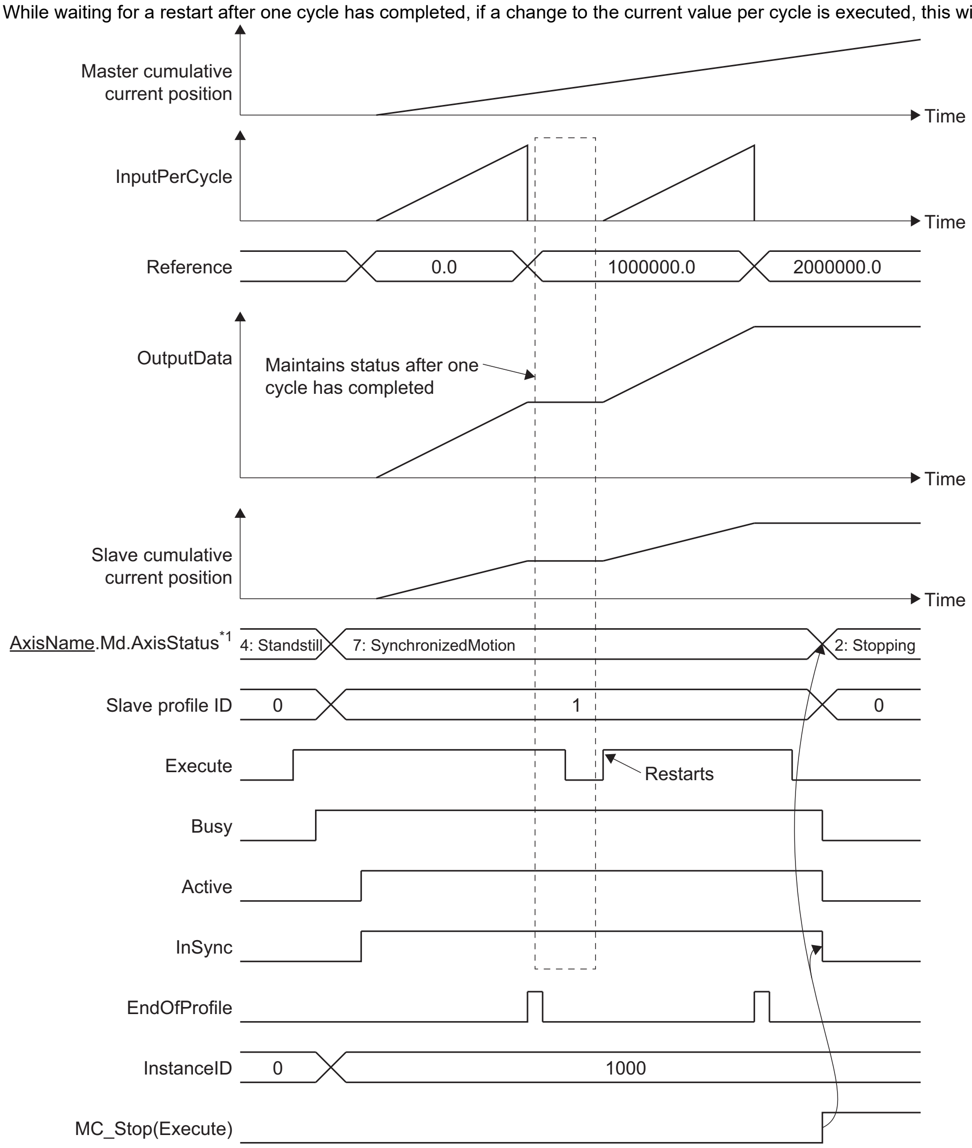
Synchronization status is maintained after one cycle, and Reference value (Reference) and Output value (OutputData) keep their value when Execute command (Execute) is started again.

Input stop cause to the slave axis to cancel synchronization status.

During a restart after one cycle has completed, even when values of input labels where continuous update is possible are changed, any input information will not be imported until Execute command (Execute) is restarted.

If Execute command (Execute) is restarted, regardless of the Start mode (StartMode) settings, the slave axis will immediately start moving.

While waiting for a restart after one cycle has completed, if a change to the current value per cycle is executed, this will be valid from the next cycle.

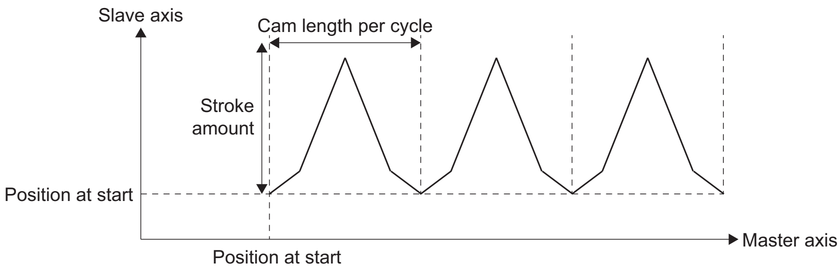


*1 Axis status of Slave axis (Slave).

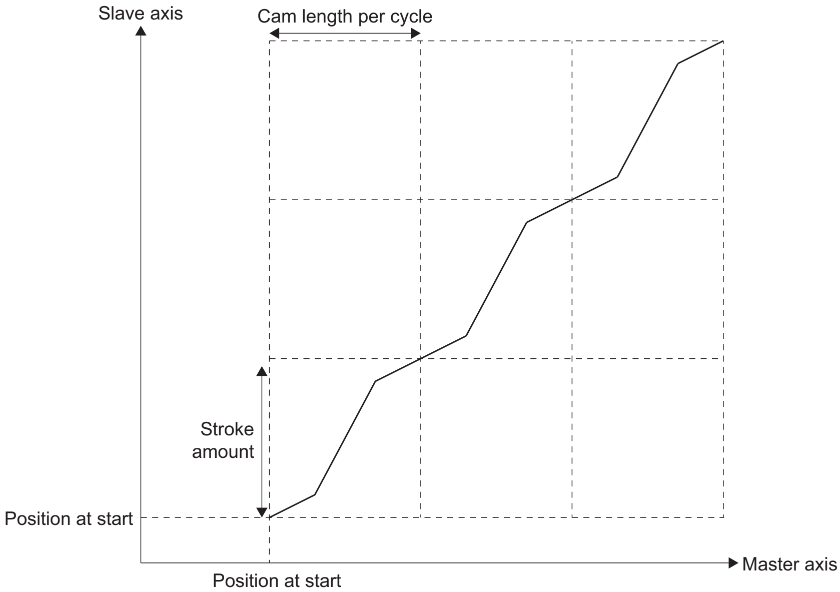
• Periodic (Periodic) "TRUE: Periodic" is specified

Cam table execution is repeated continuously. When the positions of Slave axis (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 positions 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 Master axis absolute coordinate (MasterAbsolute) set by MC_CamTableSelect.

- "0: Relative coordinate" is specified to the master axis absolute coordinate
  Master axis synchronization start position (MasterSyncPosition) becomes the start point of the cam table. When In synchronization (InSync) becomes TRUE, cam operation is executed corresponding to the relative movement amount of Master axis (Master). Cam operation is executed continuously, even when ring counters of Master axis (Master) and cam table are not matched.

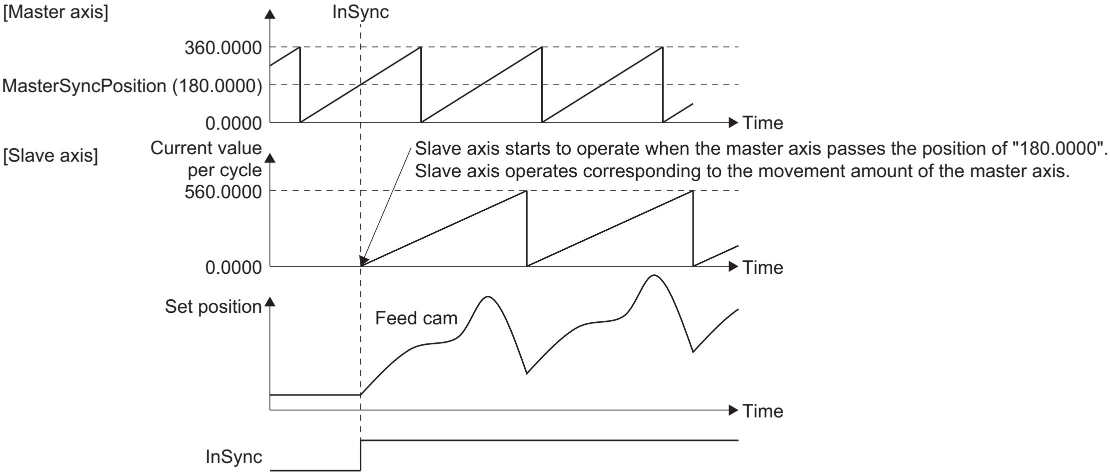
Ex.

For the following settings.

· Master axis: Ring counter of the current value: 0.0000 to 360.0000 [degree]

· Master axis synchronization start position (MasterSyncPosition): 180.0000 [degree]

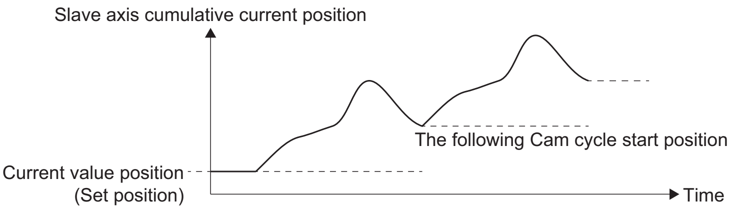
· Cam: Cam length per cycle: 540.0000 [degree]



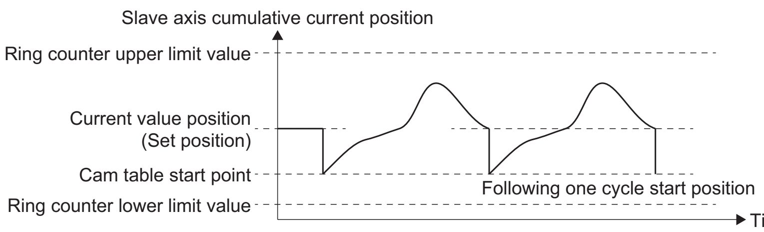
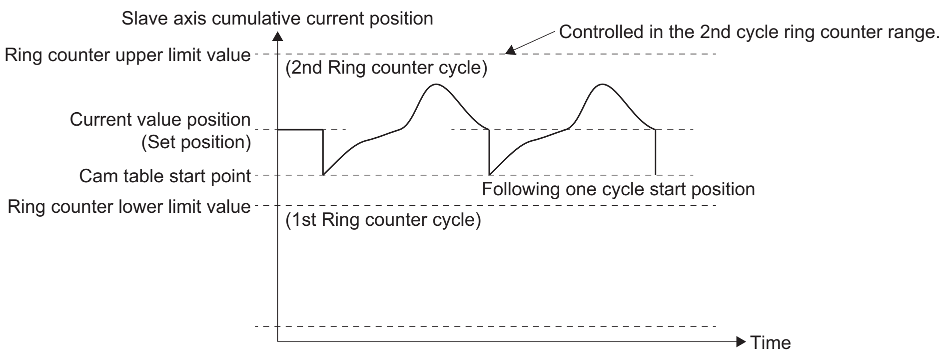
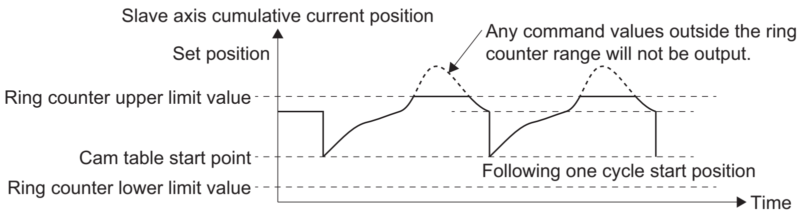
■ Slave axis absolute coordinate

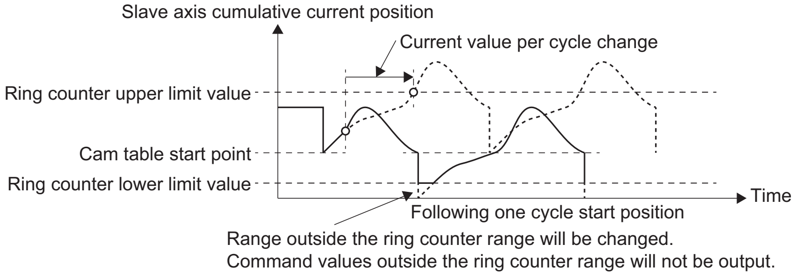
The following explains the operation at Slave axis absolute coordinate (SlaveAbsolute) set by MC_CamTableSelect (Cam Table Selection).

- "0: relative coordinate" is specified to Slave axis absolute coordinate (SlaveAbsolute)
  When In synchronization (InSync) becomes TRUE, the slave axis starts the operation of the cam table from the current position (set position).
  When Periodic (Periodic) of the cam table is set to "TRUE: Periodic", the next cycle starts from the stroke position (set position) where the last cycle completed.



- "1: absolute coordinate" is specified to Slave axis absolute coordinate (SlaveAbsolute)
  The cam is operated within the ring counter range including current values of Slave axis (Slave) at the point when In synchronization (InSync) becomes TRUE.
  Command is output per operation cycle to return the set position of Slave axis (Slave) to the start point of the cam table when In synchronization (InSync) and Cam cycle completion (EndOfProfile) become TRUE. If the command at this time is too large, the position command and velocity command to the drive unit also become too large, causing the servo error [AL. 035_Command frequency error] for the MR-J5(W)-G. The set position of Slave axis (Slave) will not output any values outside the ring counter range.

Status	Operation
When the stroke range is within the ring counter range of slave axis	
When the stroke range is within the ring counter range of slave axis (Cam started on the 2nd cycle)	
When the stroke range is outside the ring counter range of the slave axis*1	

Status	Operation
When current value per cycle is changed	When executing the current value per cycle change, to compensate the Reference value (Reference) so that the Slave axis (Slave) does not operate, any range outside of the ring counter range will be changed. When Slave axis (Slave) has moved and the current value per cycle has changed, change Master axis offset (MasterOffset). 

*1 If the version of ProfileControl is earlier than "1.15", command values outside of the ring counter range will also be output. Values outside of the ring counter range will be rounded so that they are within the range.

■ Length per cycle

Set the input amount that is required for one cycle. For details, refer to the following.

☞ Page 436 Current value per cycle (InputPerCycle)

■ Stroke amount

Set the stroke amount corresponding to its ratio 100%. For details, refer to the following.

☞ Page 438 Output value (MC_CamIn.OutputData)

■ Start point and Initial stroke amount

Start Point used for cam operation and the Initial stroke amount will differ depending on the interpolation method specification

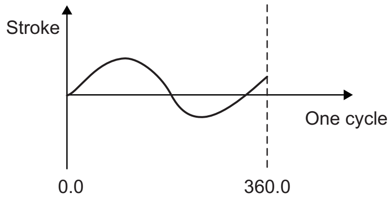
Interpolation Method Specification (Interpolate)	Cam operation Start Point	Initial stroke amount for cam operation
0: Linear Interpolation	Input value of the cam table start point	Output value of the cam table start point
1: Section Interpolation	Start Point (StartPoint) of the cam table start point	Initial stroke amount (StartStroke) of the cam table
2: Spline Interpolation		

- Cam operation start point

  Operates so that Current value per cycle (InputPerCycle) is "0.0".

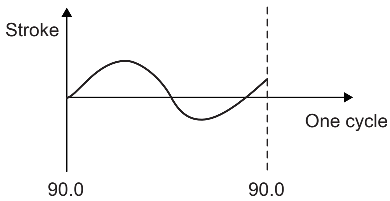
  - Cam table

    ■Cam (1) (Cam operation Start Point "0.0")



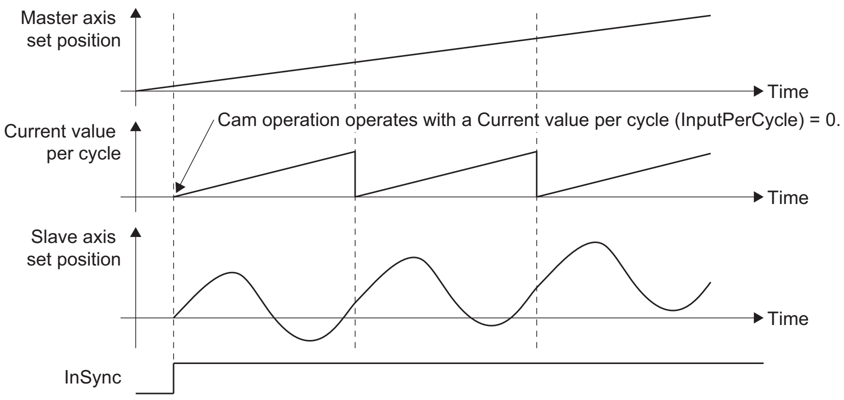
- Interpolation method specification: Section interpolation
- Length per cycle: 360.0
- Start Point: 0.0
- Initial stroke amount: 0.0

■Cam (2) (Cam operation Start Point "90.0")



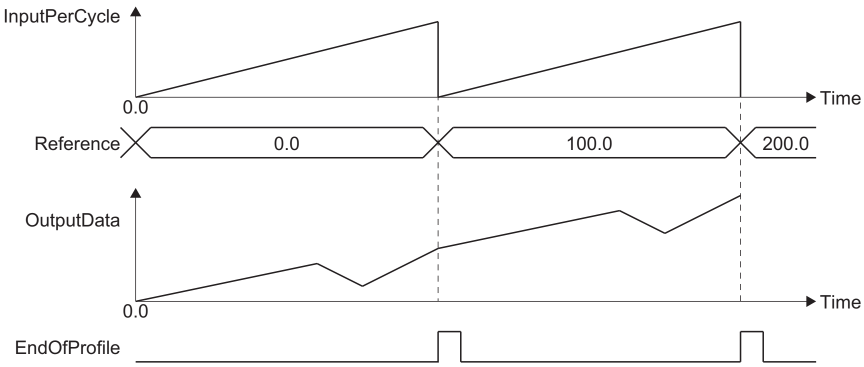
- Interpolation method specification: Section interpolation
- Length per cycle: 360.0
- Start Point: 90.0
- Initial stroke amount: 0.0

- Cam operation
  The operation in the following is performed regardless of whether cam (1) or cam (2) is used.

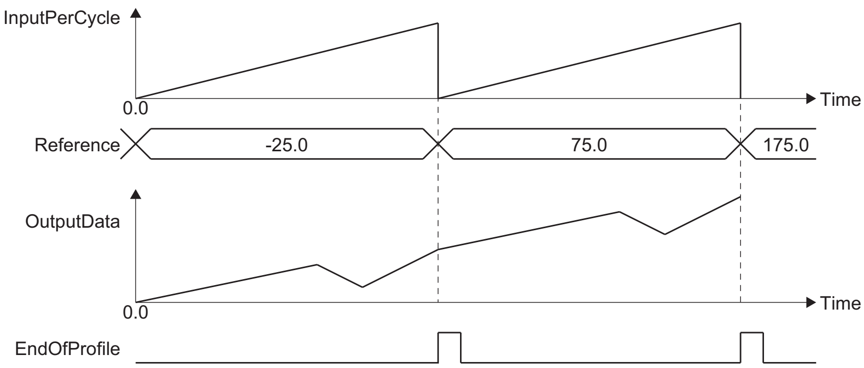


- Cam operation initial stroke amount.
  Operates so that it becomes the Output value (OutputData) at the start of synchronization. Because the initial stroke amount of the cam operation stops the slave axis from operating, the initial stroke amount from Reference value (Reference) is decreased.

■When Initial stroke amount of cam operation is "0.0"



■Initial stroke amount of cam operation is "25.0"



Monitor data during cam operation

During control in MC_CamIn (Cam Operation Start), the current value per cycle, the reference value, and the output value can monitor in Current value per cycle (InputPerCycle), Reference value (Reference), and Output value (OutputData).

Name	Variable name	Data type	Refresh cycle
Current value per cycle	InputPerCycle	LREAL	Execution cycle of POU which calls FB (program component)
Reference value	Reference	LREAL	
Output value	OutputData	LREAL	
Instance ID	InstanceID	WORD(UINT)	

■ Current value per cycle (InputPerCycle)

The initial value is "0.0".

The movement amount of Master axis (Master) is reflected after Controlling (Active) turns to TRUE as follows.

Current value per cycle = (Cumulative movement amount of the master axis - Master axis offset + Current value per cycle change value) MOD Length per cycle

The monitor value range is "0.0 ≤ Current value per cycle (MC_CamIn.InputPerCycle) < (One cycle length × Master axis scaling)"

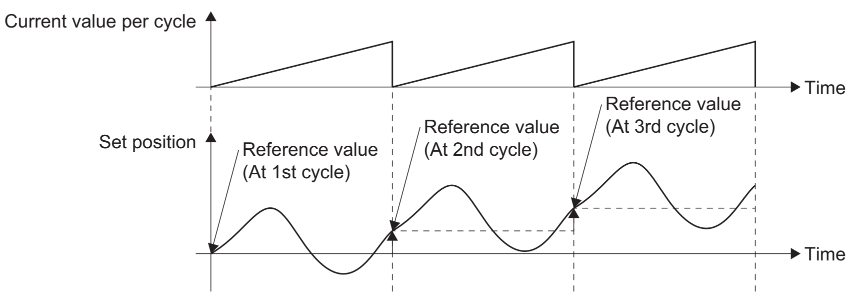
■ Reference value (Reference)

When MC_CamIn (Cam Operation Start) is started to operate, cam reference value is calculated by the cam position for start (current value per cycle) based on set position where In synchronization (InSync) becomes TRUE. The formula for initial value is as follows because the Initial stroke amount during cam table is subtracted.

Slave axis absolute coordinate (SlaveAbsolute)	Calculation formula
FALSE: Relative coordinate	■ProfileControl version is "1.15" or later Reference value = Slave axis position when synchronization starts - (Stroke value based on the cam position (current value per cycle) when In Synchronization (InSync) changes to TRUE + the initial stroke amount) × Slave axis scaling
	■ProfileControl version is earlier than "1.15" Reference value = 0 - (Stroke value based on the cam position (current value per cycle) when InSynchronization (InSync) changes to TRUE + the initial stroke amount) × Slave axis scaling
TRUE: Absolute coordinate	Reference value = The set position corresponding to the cumulative current position being 0 when In synchronization (In Sync) changes to TRUE.

When Two-way cam operation, or Slave axis absolute coordinate (SlaveAbsolute) is "TRUE: Absolute coordinate", the reference value is not updated. When Feed cam operation, as well as Slave axis absolute coordinate (SlaveAbsolute) are "FALSE: Relative coordinate", the reference value is calculated as shown below.

Calculation formula
Reference value = Original reference value + (Stroke value at the last point - Stroke value at the 1st point) × Slave axis scaling)



Reference value is updated at the following timing

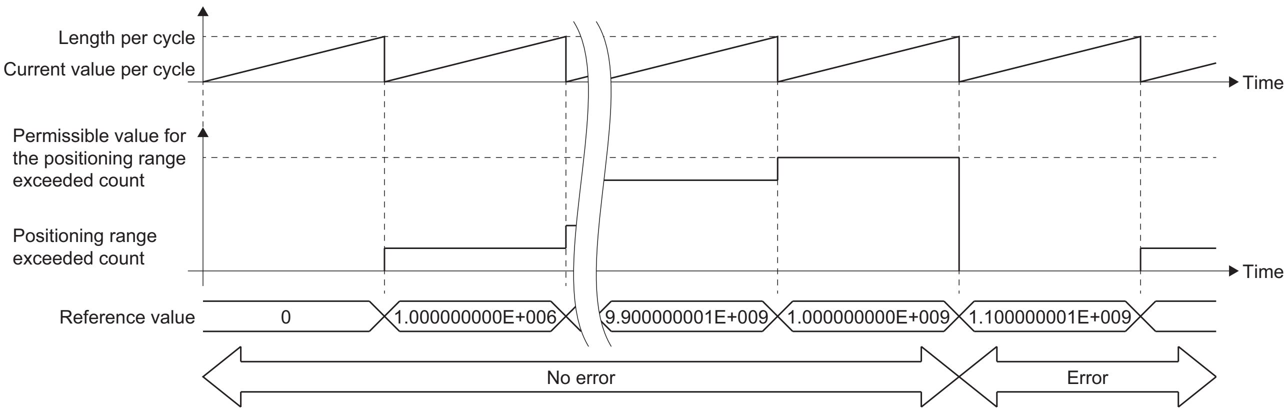
Update timing	The calculation formula of reference value
When Current value per cycle (InputsPerCycle) passes the last point of cam table	Original reference value + (Stroke value at the last point - Stroke value at the 1st point × Slave axis scaling)
When Current value per cycle (InputPerCycle) going in the address decreasing direction passes the first point of the cam table	Original reference value - ((Stroke value at the last point - Stroke value at the 1st point) × Slave axis scaling)
When changing current value per cycle (MCv_ChangeCycle (Current value per cycle) completion (Done) is TRUE)	Original reference value - (Stroke value corresponding to current value per cycle after change × Slave axis scaling)

Restriction

For the reference value error

Any of the following operation may cause the error of reference value.

- Condition 1: When the number of output TRUE of Cam cycle completion (EndOfProfile) after the feed cam and In synchronization (InSync) becomes TRUE exceeds the permissible value for the positioning range exceeded count.
  Although the error does not occur unless positioning range exceeded count exceeds the permissible value for the positioning range exceeded count, 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 feed cam and 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 too 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 80 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.

(Example)

When the output value at the 1st point is "0", the output value at the last point is "1000000", and the output coefficient is "10", the permissible value of the number of exceeding from maximum to minimum becomes "100000".

The monitor value range differs depending on ProfileControl version.

ProfileControl version	Monitor value range
"1.15" or later	Ring counter lower limit value of the slave axis ≤ Reference value < Ring counter upper limit value of the slave axis
Earlier than "1.15"	Lower limit value of the positioning range ≤ Reference value < Upper limit value of the positioning range

■ Output value (MC_CamIn.OutputData)

Reference value is the initial stroke amount during cam table. For details, refer to the following.

☞ Page 434 Start point and Initial stroke amount

While In synchronization (InSync) is TRUE, calculated as shown below.

Output value = Reference value+ ((Stroke amount × Stroke ratio corresponding to the current value per cycle) × Slave axis scaling) + Slave axis offset

or

Output value = Reference value + (Output value corresponding to the current value per cycle) × Slave axis scaling) + Slave axis offset

The monitor value range differs depending on ProfileControl version.

ProfileControl version	Monitor value range
"1.15" or later	Ring counter lower limit value of the slave axi ≤ Output value < Ring counter upper limit value of the slave axis
Earlier than "1.15"	Lower limit value of the positioning range ≤ Output value < Upper limit value of the positioning range

Precautions

- Changing Master axis offset (MasterOffset), Slave axis offset (SlaveOffset), Master axis scaling (MasterScaling), and Slave axis scaling (SlaveScaling) may give impact to the machine because the slave axis moves suddenly at the start of control or at control change. It is important to carefully check the setting values and the change timing.
- When monitoring the axis being cam controlled with the engineering tool, the position, velocity, etc. is based on the unit setting of the axis. Units for the cycle length, stroke amount, etc. set in the operation profile data are not used.

13.8 Gear Operation

This function sets the speed ratio between the master axis and the slave axis, and starts gear operation.

Relevant FB

For details on Motion control FBs, refer to the following.

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

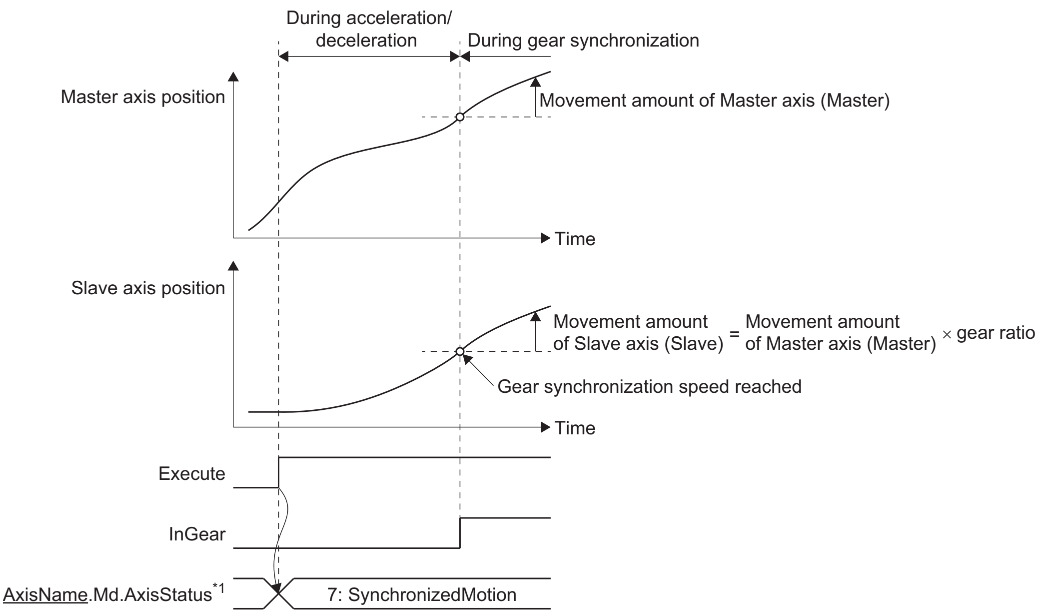
Motion control FB	Name	Description
MC_GearIn	Gear operation start	Gear operation starts operation according to the specified gear ratio.

Control details

In the MC_GearIn (Gear Operation Start), Gear ratio numerator (RatioNumerator), Gear ratio denominator (RatioDenominator), Master axis data source selection (MasterValueSource), Acceleration (Acceleration), Deceleration (Deceleration), Jerk (Jerk), and Buffer mode (BufferMode) are set and the gear operation starts.

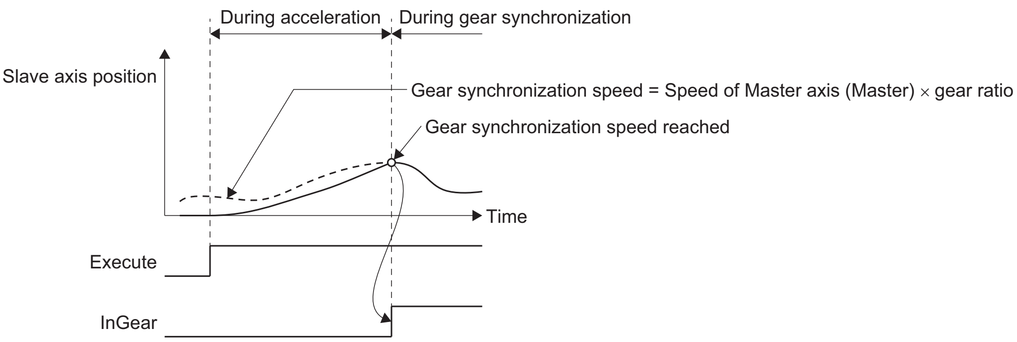
To stop the operation, execute the MC_Stop (Forced Stop).

After executing this FB, Slave axis (Slave) performs acceleration/deceleration until reaching the gear synchronization speed, the value of which is the velocity of Master axis (Master) converted by the gear ratio. After reaching the gear synchronization speed, Gear ratio reached (InGear) becomes TRUE, and Slave axis (Slave) is controlled using a velocity equal to the velocity of Master axis (Master) converted by the gear ratio.



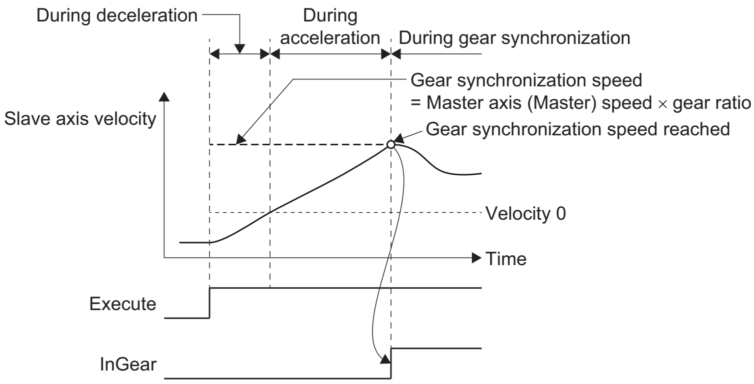
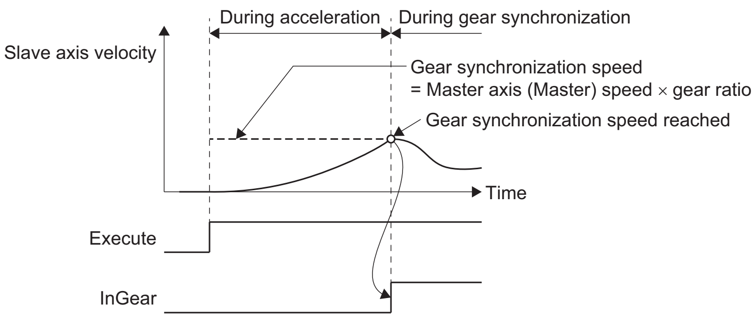
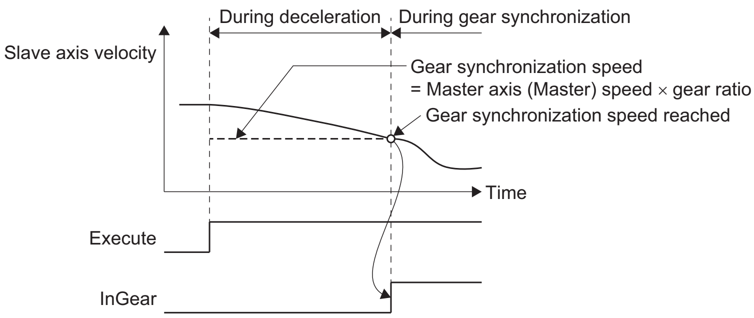
*1 This is axis state of Slave axis (Slave).

- When the speed of Master axis (Master) is changed during acceleration/deceleration, the gear synchronization speed is also updated.



Acceleration/deceleration until reaching the gear synchronization speed

- The acceleration/deceleration method for MC_GearIn (Gear Operation Start) is the acceleration/deceleration specification method.
  For details of operation of the acceleration/deceleration specification method, refer to the following.
  ☞ Page 317 Acceleration/deceleration Processing Function
- Depending on the velocity of Slave axis (Slave) and the gear synchronization speed at operation start, one of the following four types of acceleration/deceleration is performed.
  The type of acceleration/deceleration will not change even if the gear synchronization speed is updated during the acceleration/deceleration.

<table>
<tr><th>Velocity of Slave axis (Slave) and gear synchronization speed</th><th>Acceleration/deceleration operation</th></tr>
<tr><td>Velocity of Slave axis (Slave) < 0 < Gear synchronization speed
or
Velocity of Slave axis (Slave) > 0 > Gear synchronization speed</td><td>Deceleration is performed to velocity 0, then acceleration is performed from velocity 0 to the gear synchronization speed.

</td></tr>
<tr><td>Velocity of Slave axis (Slave) (Absolute value) < Gear synchronization speed (Absolute value)</td><td>Acceleration is performed until reaching the gear synchronization speed.

</td></tr>
<tr><td>Velocity of Slave axis (Slave) (Absolute value) > Gear synchronization speed (Absolute value)</td><td>Deceleration is performed until reaching the gear synchronization speed.

</td></tr>
<tr><td>Velocity of Slave axis (Slave) = Gear synchronization speed</td><td>Acceleration/deceleration is not performed.
After operation is started, Gear ratio reached (InGear) becomes TRUE immediately and Slave axis (Slave) is controlled using a velocity equal to the velocity of Master axis (Master) converted by the gear ratio.</td></tr>
</table>

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 (Smoothing Filter).
- When Master axis (Master) speed after conversion exceeds the acceleration/deceleration upper limit value during acceleration/deceleration operation, "Acceleration Time Limit Over Warning (warning code: 0D04H)" or "Deceleration Time Limit Over Warning (warning code: 0D05H)" will occur and 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 axis (Master) speed after conversion, Acceleration (Acceleration) or Deceleration (Deceleration) to not exceed the acceleration/deceleration upper limit value, and restart the acceleration/deceleration operation.

13.9 Addition/Subtraction Positioning

This function combines the movement amount of 2 axes and transmits.

Relevant FB

For details on Motion control FBs, refer to the following.

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

Motion control FB	Name	Description
MC_CombineAxes	Addition/subtraction positioning	Added or subtracted value which the movement amount of the specified master axis 2-axis is set as the commanded position, and the positioning is executed.

Control details

In MC_CombineAxes (Addition/Subtraction Positioning), Addition/subtraction method selection (CombineMode), Master axis 1 gear ratio numerator (GearRatioNumeratorM1), Master axis 1 gear ratio denominator (GearRatioDenominatorM1), Master axis 2 gear ratio numerator (GearRatioNumeratorM2), and Master axis 2 gear ratio denominator (GearRatioDenominatorM2), Master axis 1 data source (MasterValueSourceM1), Master axis 2 data source (MasterValueSourceM2), and Buffer mode (BufferMode) are set and the addition/subtraction positioning is executed.

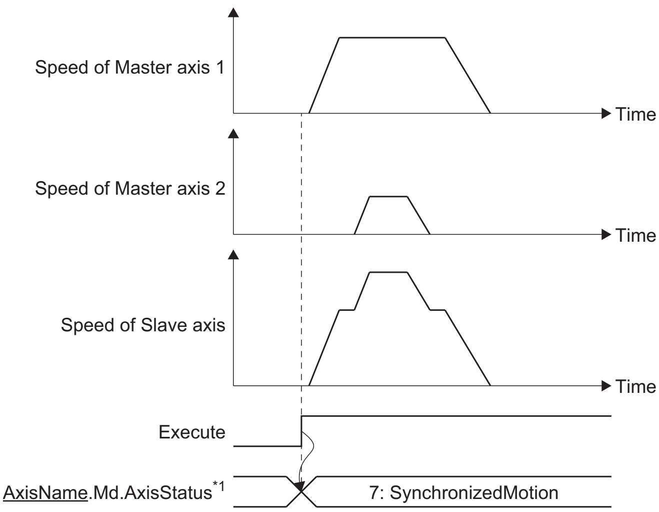
To stop the operation, execute MC_Stop (Forced Stop).

The movement amount of Master axis 1 (Master1) and Master axis 2 (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 Master axis 1 (Master1) and Master axis 2 (Master2).

When "0: mcAddAxes" is set to CombineMode

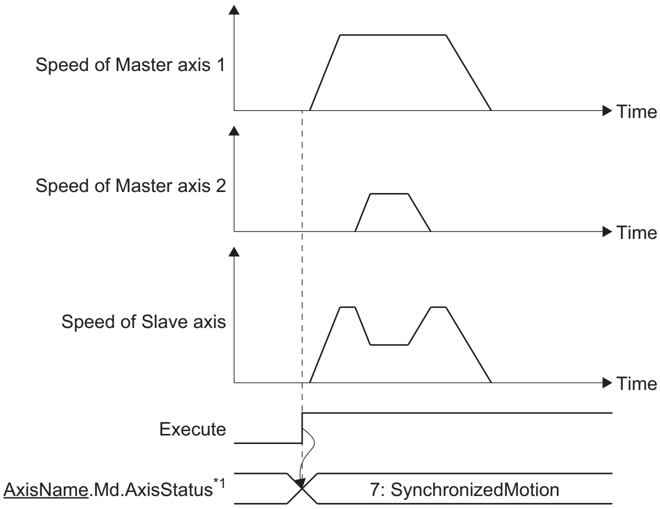
Movement amount of Slave axis (Slave) = Master axis 1 (Master1) + Movement amount of Master axis 2 (Master2)



*1 This is axis state of Slave axis (Slave).

When "1: mcSubAxes" is set to CombineMode

Movement amount of Slave axis (Slave) = Master axis 1 (Master1) - Movement amount of Master axis 2 (Master2)



*1 This is axis state of Slave axis (Slave).

13.10 Precautions

Relevant add-ons

The following add-ons are required to use the synchronous control function.

- ProfileControl
- MotionControl_Sync
- MotionEngine
- Axis
- MotionControl_General

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.

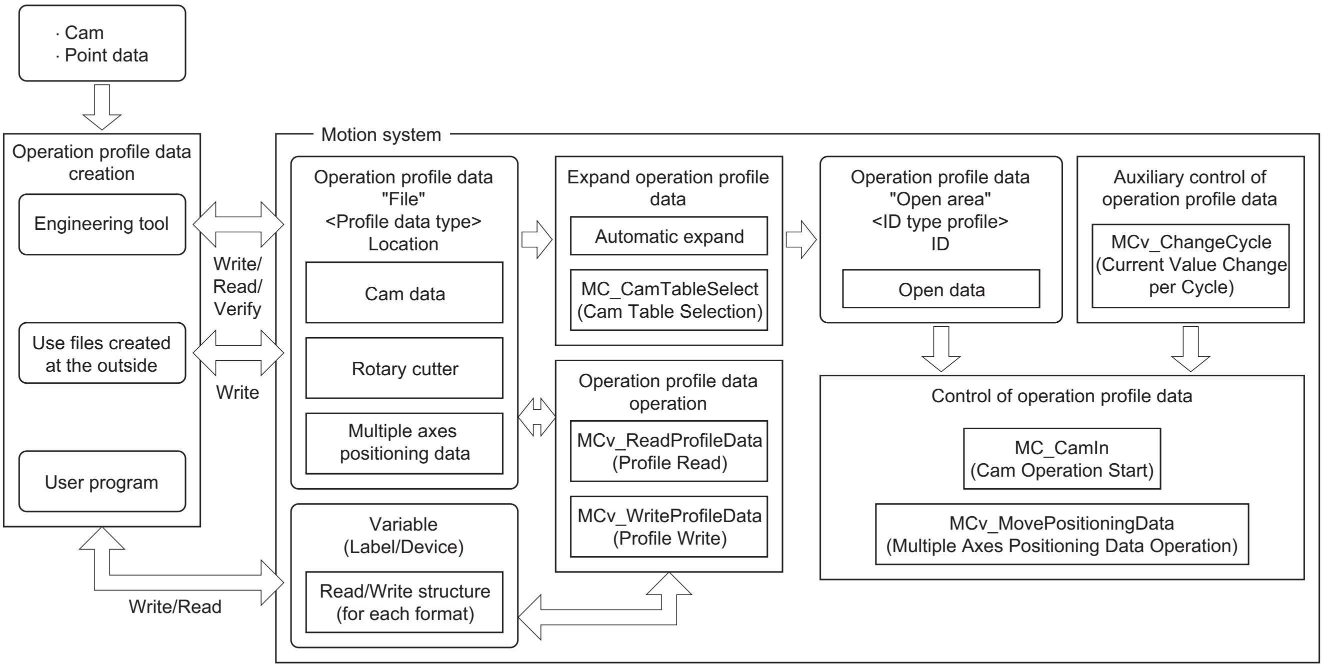
Operation of this function for each system status

○: Possible, ×: Not possible

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

Overall block diagram

The following diagram shows the relevancy between the data from creating to using of operation profile data and the Motion control FB about operation profile data used in various functions (e.g. the cam of single axis synchronous control).



Relevant variables

PROFILE_DATA

Variable/Structure name	Name	Details
Location	Operation Profile Data Storage Location	Sets the file name and storage location of the operation profile. FILE_LOCATION type
ID	Profile ID	Sets the number for the profile ID of the operation profile. PROFILE_ID type

FILE_LOCATION

Variable/Structure name	Name	Details
FileName	File Name	Specifies a file name. (Up to 63 characters including the extension can be set.)*1
Path	Folder Specification	Specifies the folder path in which the file is stored. (Up to 63 characters can be set.)*1

*1 Set a total of up to 127 characters (including Null string) for File name (FileName) and Folder specification (Path).

Point

- Make sure the folder path does not exceed the maximum number of characters for the folder path.
- Make sure the file name does not exceed the maximum number of characters for the file name.

PROFILE_ID

Variable/Structure name	Name	Details
Number	Profile ID No.	Sets the Profile ID No. • 0 to 60000

INSTANCE_ID

Variable/Structure name	Name	Details
Number	Instance ID No.	Sets the number for the instance ID

Relevant FBs

For details on Motion control FBs, refer to the following.

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

Motion control FB	Name	Description
MC_CamTableSelect	Cam table selection	Stores specified operation profile data in the open area.
MCv_ReadProfileData	Profile read	Reads the specified operation profile data from the open area or the file.
MCv_ReadProfileData	Profile write	Writes the specified operation profile data from the open area or the file.
MCv_ChangeCycle	Current value change per cycle	Changes the current value per cycle of the specified operation profile data control FB.

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 shows operation profile data format and I/O data that can be used to each function.

Function (Operation profile data control)	Operation profile data format	Input data	Output data	Operation profile data open
Cam operation (MC_CamIn (Cam Operation Start))	Cam data	Master axis position (selected with MC_SOURCE)	Slave axis position	Auto expand*1 MC_CamTableSelect (Cam Table Selection)
	Rotary cutter			
Advanced synchronous control (MCv_AdvancedSync)	Cam data	Master axis position, sub input axis, auxiliary shaft position (selected with MC_SOURCE)		
	Rotary cutter			
Advanced synchronous control current position per cycle calculation (MCv_AdvPositionPerCycleCalc)	Cam data	• Ring counter upper limit • Cam starting point • Length per cycle • Stroke amount • Current position per cycle • Cam reference position • Cam set position	Current position per cycle	
	Rotary cutter			
Advanced synchronous control cam set position calculation (MCv_AdvCamSetPositionCalc)	Cam data		Cam set position	
	Rotary cutter			
Multiple Axes Positioning Data Operation (MCv_MovePositioningData)	Multiple axes positioning data	Positioning data No.	Set position of axes group configuration axis	Auto expand*1

*1 When "Auto Expand" is set to "Yes".

Cam data

The relationship of output values (slave axis position) to input values (master axis position) is defined as the cam data.

The following interpolation method can be selected as the cam data.

Interpolation Method Specification (Interpolate)	Description
0: Linear Interpolation	Create cam data that specifies the input value (master axis position) of each point and the output position (slave axis position) and interpolates each point with straight lines.
1: Section Interpolation	Create cam data that specifies the input value (master axis position), stroke, and cam curve type of each point and interpolates with curve lines specified each point.
2: Spline Interpolation	Create cam data that specifies the input value (master axis position) and stroke and performs spline interpolation of each point.

Required data for the setting differs depending on the selected interpolation method. Refer to the following table.

◎: Necessary, ○: Optional, —: Unnecessary

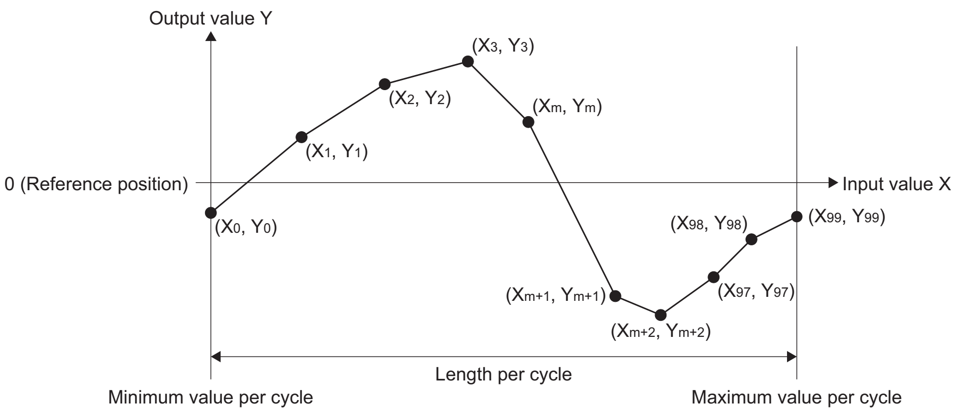
Setting items	Interpolation Method Specification (Interpolate)		
	0: Linear Interpolation	1: Section Interpolation	2: Spline Interpolation
Profile ID	○	○	○
Interpolation method specification	◎	◎	◎
Resolution/coordinate number	◎	◎	◎
Input unit character string	○	○	○
Output unit character string	○	○	○
Start point	—	◎	◎
Initial stroke amount	—	◎	◎
Initial velocity	—	○	—
Initial acceleration	—	○	—
I/O data	◎	—	—

Setting items		Interpolation Method Specification (Interpolate)		
		0: Linear Interpolation	1: Section Interpolation	2: Spline Interpolation
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		◎	◎	◎

■ Interpolation Method Specification (Interpolate)

- 0: Linear Interpolation

  The stroke data of each point data that is defined for a cam curve per cycle with two or more points (a pair data of current value per cycle and stroke) is opened in the open area.

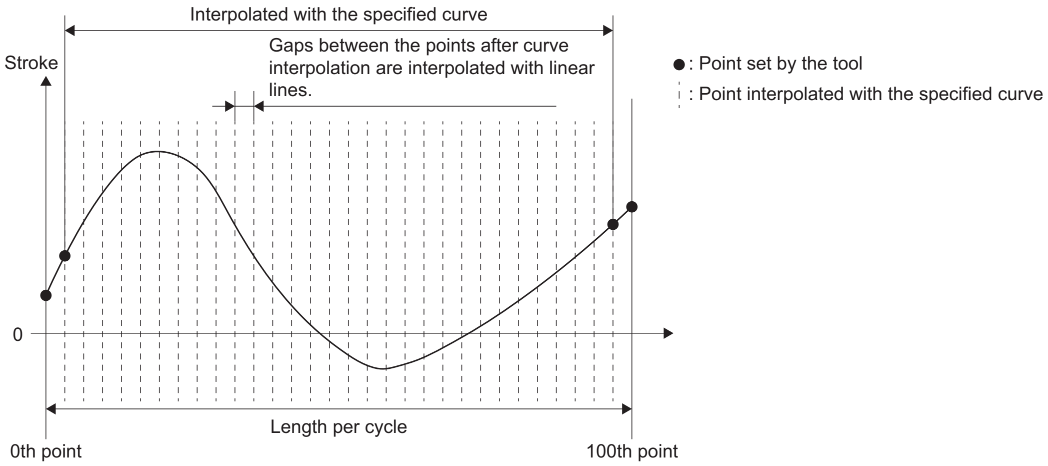


- 1: Section Interpolation

  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. (a pair data of current value per cycle and stroke)

Ex.

Resolution: 100



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.

- 2: Spline Interpolation
  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. (a pair data of current value per cycle and stroke)

■ Profile ID

It is a unique ID assigned by an open instruction.
The ID opened by the MC_CamTableSelect (Cam Table Selection) can be used in the control instruction (MC_CamIn (Cam Operation Start)) 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 "0.0".
If other than "0.0" is set, "Operation Profile Data Incorrect (error code: 3410H)" occurs.
Not applicable when the Interpolation Method Specification (Interpolate) is set to "0: Linear Interpolation".

■ 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.
Not applicable when the Interpolation Method Specification (Interpolate) is set to "0: Linear Interpolation".

■ Initial velocity/End point velocity

Set the velocity at the start point/end point.
Use when cam curve type is set to 5th Curve (Adj.).
The unit for setting value is [pulse/s] when output unit string is [pulse].
The unit for setting value is [output unit character string/min] when output unit string is other than [pulse].

Ex.

The unit for setting value is [degree/min] when output unit string is [degree].

■ Initial acceleration/End acceleration

Set the acceleration at the start point/end point.
Use when cam curve type is set to 5th Curve (Adj.).
The unit for setting value is [pulse/s] when output unit string is [pulse/s^2].
The unit for setting value is [output unit string/min^2] when output unit string is other than [pulse].

Ex.

The unit for setting value is [degree/min^2] when output unit string is [degree].

■ Length per cycle

Set the input amount that is required for one cycle.
Not applicable when the Interpolation Method Specification (Interpolate) is set to "0: Linear Interpolation".
The control that the difference between maximum value per cycle and minimum value per cycle is length per cycle is carried out.

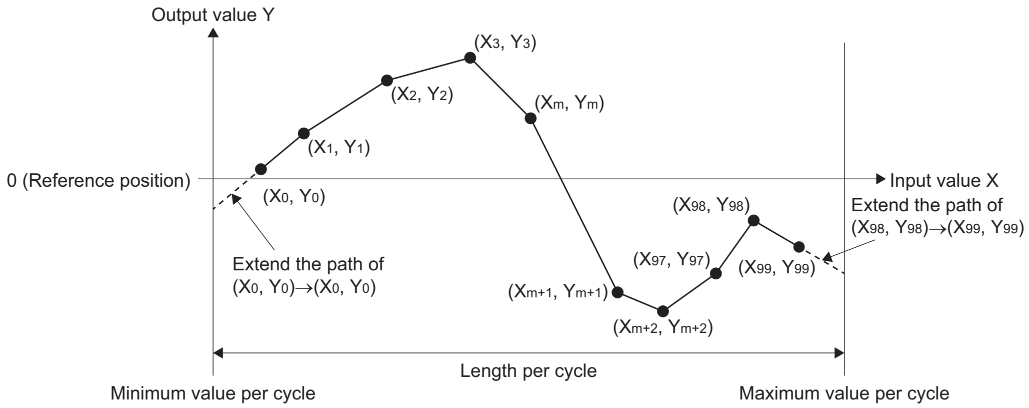
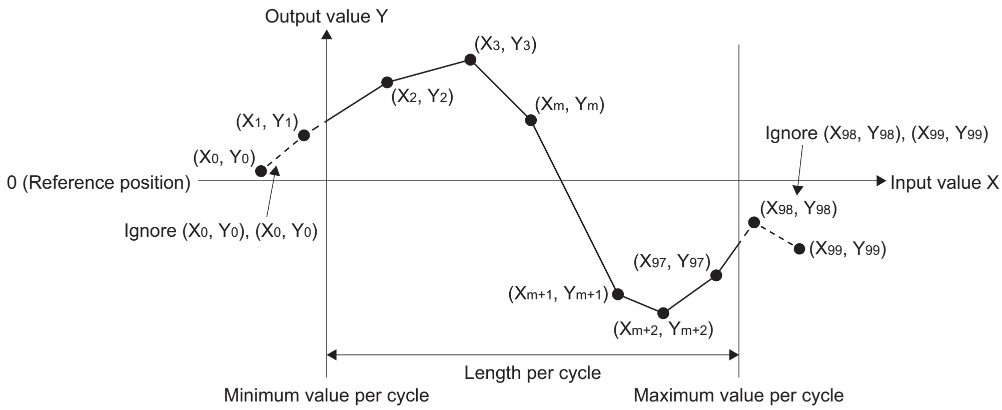
■ Maximum value per cycle/Minimum value per cycle

Set the maximum/minimum value per cycle.

It is valid only when Interpolation Method Specification (Interpolate) is "0: Linear Interpolation".

Set "0.0" to minimum value per cycle. If other than "0.0" is set to minimum value per cycle, "Operation Profile Data Incorrect (error code: 3410H)" occurs.

When Interpolation Method Specification (Interpolate) is "0: Linear Interpolation", the operation profile is as follows depending on the coordinate data.

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. <Example> When resolution is "100" 
"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 <Example> When the number of point data is "100" 

■ Time per cycle

Set time that is required for one cycle.

Use when cam curve type is set to 5th Curve (Adj.).

■ 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

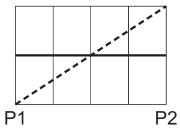
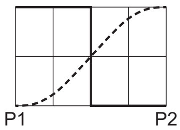
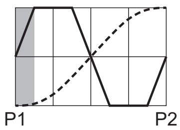
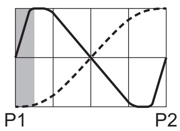
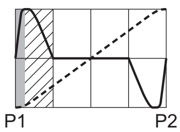
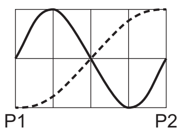
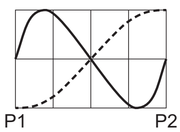
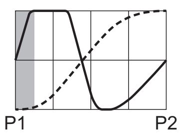
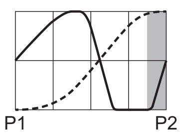
Set the end point of each section. End point should be set in ascending order.

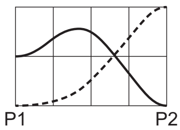
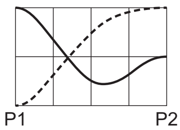
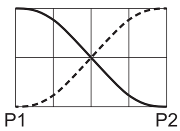
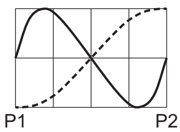
■ Stroke

Set the stroke of each section.

■ Cam curve types

A cam curve can be selected from the following curves.

Cam curve type (MC_CAM_CURVE_TYPE)			Acceleration curve shape*1	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 velocity	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)	—	—	—
DistortedConstantSpeedt(4)	Distorted constant velocity			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	—	—	—	—
Trapecloid (7)	Trapecloid	Two-dwelling asymmetrical		0.0 to 1.0	0.0001 < L1 < 0.2500 (0.125)	—	—	—
ReverseTrapecloid(8)	Reverse trapecloid			0.0 to 1.0	0.0001 < L1 < 0.2500 (0.125)	—	—	—

Cam curve type (MC_CAM_CURVE_TYPE)			Acceleration curve shape*1	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		
DoubleHypotenuse(9)	Double hypotenuse	One-dwelling		0.0 to 1.0	—	—	—	—
ReverseDoubleHypotenuse(10)	Reverse double hypotenuse			0.0 to 1.0	—	—	—	—
SingleHypotenuse(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	—	—	-2500000000.0 to 2500000000.0	-2147483647.0 to 2147483647.0

*1 ----: Stroke ratio ——: Acceleration ▭: Range L1 ▨: Range L2

■ Read/write data structure

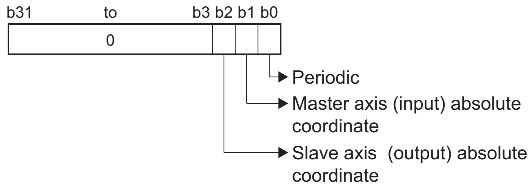
This section describes read data 1/write data 1 (Data1) and read data 2/write data 2 (Data2) specified by MCv_ReadProfileData (Profile Read)/MCv_WriteProfileData (Profile Write) when executing read/write operation profile data.

- Read data 1/Write data 1 (Data 1)

  Specify the PROFILE_CAM_DATA type structure.

PROFILE_CAM_DATA			Device offset	Description
Variable name	Name	Data type	(Total size: 108 points)	
Interpolate	Interpolation method specification	INT	+0	Set the interpolation method specification of the cam data. • 0: Linear Interpolation • 1: Section Interpolation • 2: Spline Interpolation
Resolution	Resolution/ number of coordinates	DWORD(UDINT)	+2	Set the resolution. Linear interpolation/section interpolation/spline interpolation • 8 to 65535
InputUnitString	Input unit character string	WSTRING(31)	+4	Sets the unit for input data with a character string.
OutputUnitString	Output unit character string	WSTRING(31)	+36	Sets the unit for output data with a character string.
StartPoint	Start point*1	LREAL	+68	Set the start point. *: Set "0.0".
StartStroke	Initial stroke amount*1	LREAL	+72	Set the stroke amount for the start point. (Stroke amount (Stroke) ≤ Initial stroke amount (StartStroke) ≤ Stroke amount (Stroke))
StartVelocity	Initial velocity*2	LREAL	+76	Set the velocity for the start point. (-2500000000.0 ≤ Initial velocity (StartVelocity) ≤ 2500000000.0)
StartAcceleration	Initial acceleration*2	LREAL	+80	Set the acceleration for the start point. (-2147483647.0 ≤ Initial acceleration (StartAcceleration) ≤ 2147483647.0)
CycleLength	Length per cycle*1	LREAL	+84	Set the input amount that is required for one cycle. (0.00000000000001 ≤ Length per cycle (CycleLength) ≤ 10000000000.0)
CycleMin	Minimum value per cycle	LREAL	+88	Set the minimum value per cycle. *: Set "0.0".

14

PROFILE_CAM_DATA			Device offset	Description
Variable name	Name	Data type	(Total size: 108 points)	
CycleMax	Maximum value per cycle	LREAL	+92	Set the maximum value per cycle. (Minimum value per cycle (CycleMin) < Maximum value per cycle (CycleMax) ≤ 10000000000.0)
CycleTime	Time per cycle[*2]	LREAL	+96	Set time per cycle. (0.001 < Time per cycle (CycleTime) ≤ 100000.0)
Stroke	Stroke amount	LREAL	+100	Set the stroke amount. (0.00000000000001 ≤ Stroke amount (Stroke) ≤ 10000000000.0)
NumberOfSections	Number of sections[*1]	DWORD(UDINT)	+104	Sets the number of sections ■When set to "1: Section Interpolation" • 1 to 360 ■When set to "2: Spline Interpolation" • 3 to 360
Options	Options	DWORD(HEX)	+106	Specify options.  ■Periodic (bit 0) • 0: Non periodic • 1: Periodic For details, refer to the following. ☞ Page 428 Periodic (Periodic) ■Master axis (input) absolute coordinate (bit 1) • 0: Relative coordinate • 1: Setting not possible[*3] For details, refer to the following. ☞ Page 432 Master axis absolute coordinate ■Slave axis (output) absolute coordinate (bit 2) • 0: Relative coordinate • 1: Absolute coordinate For details, refer to the following. ☞ Page 433 Slave axis absolute coordinate *: For bit 3 to 31, specify "0".[*4]

*1 It is not used in the cam data format (Interpolation Method Specification (Interpolate) is "0: Linear Interpolation"). (It is ignored.)
*2 Used only when cam curve is 5th Curve (Adj.). (Ignore for other curve)
*3 When "1" is set, "Out of MasterAbsolute Range (error code: 341DH)" occurs, and the FB will not start.
*4 When specifying other than "0", "Out of Options Range (error code: 1A4EH)" occurs, and the FB will not start.

Ex.

Setting example of Target (TARGET_REF.Target)

· [DEV](PROFILE_CAM_DATA)G11500000

· [VAR]CamData1 (CamData1 is PROFILE_CAM_DATA type label)

- Read data 2 (Data2)/Write data 2 (Data2)

  The structure to be specified differs depending on the setting of Read destination (Target)/Write destination (Target) and Interpolation Method Specification (Interpolate) of the target cam data.

  For details, refer to the following table.

○: Supported, —: Not supported

FB	Read destination/Write destination (Target)	Structure specified in Read data 2/Write data 2 (Data2)	Interpolation Method Specification (Interpolate)		
			0: Linear Interpolation	1: Section Interpolation	2: Spline Interpolation
MCv_ReadProfileData (Profile Read)	0: Open area	LREAL[]	○	○*1	○*1
	1: File	LREAL[]	○	—	—
		PROFILE_CAM_ELEMENT[]	—	○	○*2
	2: Open area (motion service processing)	LREAL[]	○	○*1	○*1
MCv_WriteProfileData (Profile Write)	0: Open area	LREAL[]	○	—	—
		PROFILE_CAM_ELEMENT[]	—	○	○*2
	1: File	LREAL[]	○	—	—
		PROFILE_CAM_ELEMENT[]	—	○	○*2
	2: Open area (motion service processing)	LREAL[]	○	—	—
		PROFILE_CAM_ELEMENT[]	—	○	○*2

*1 Read as the stroke data (a pair data of current value per cycle and stroke) expanded to the open area.
X of the 1st point data is the value set in start point and Y of the 1st point data is the value set in initial stroke amount.
The data of 2nd point or later is the stroke data divided by resolution.
Therefore, read/write point in open area is resolution + 1

*2 Used only by end point and Stroke amount (Stroke)

The following occurs depending on the version of ProfileControl used.

ProfileControl version	Description
Earlier than Version "1.19"	If the specified number of array elements does not match the resolution + 1/number of coordinates, "Number of Read/Write Data Mismatch (error code: 3464H)" occurs.
Version "1.19" or later	If the number of the elements of the specified array is smaller than the resolution+1/number of coordinates when reading/writing the entire operation profile data, "Number of Read/Write Data Mismatch (error code: 3464H)" occurs.

The structures specified by read data 2/write data 2 (Data2) are shown below.

- LREAL type

  Specify LREAL type two-dimensional array as follows.

  LREAL[m..n, o..p]

Item	Description
The number of elements (n - m + 1)	Set the array elements so that they are greater than the number of points that are to be read. When reading/writing entire operation profile data (Offset and Points are "0"), set it to be the resolution + 1/number of coordinates of the target cam data or more. However, an area larger than the resolution +1/number of coordinates read by Data1 is not updated when using MCv_ReadProfileData (Profile Read). An area larger than the resolution +1/number of coordinates specified by Data1 is not used when using MCv_WriteProfileData (Profile Write). If the ProfileControl version is "earlier than Ver.1.19", it is necessary to match the number of elements of the specified array to the target resolution +1/number of coordinates.
The number of dimensions (p - o + 1)	Read/Write current value per cycle and stroke of each point. Set the array elements to be "2". If the array element is set to "3" or more, a read/write is possible, but as the intended result of the read/write may not be obtained due to data being moved forward, make sure to set the array element to "2".

Ex.

When setting LREAL type with Interpolation Method Specification (Interpolate)

· The following shows examples of setting the two-dimensional device/label coordinate number "100" to Target (Target) of the TARGET_REF structure.

Interpolation Method Specification (Interpolate)	Specification method	Setting example
0: Linear Interpolation	Buffer memory	TARGET_REF.Target := "[DEV](LREAL[1..100,1..2])G1150000"
	Label	TARGET_REF.Target := "[VAR]CamData2" *: CamData2 declares the "LREAL[1..100,1..2]" label to data type and label name of the global label to "CamData2", and specifies the character string for Target (Target).Reading/Writing data is as follows. · CamData2[n,1]: Nth point current position per cycle · CamData2[n,2]: Nth point stroke
1: Section Interpolation 2: Spline Interpolation	Buffer memory	TARGET_REF.Target := "[DEV](LREAL[1..101,1..2])G1150000"
	Label	TARGET_REF.Target := "[VAR]CamData2" *: CamData2 declares the "LREAL[1..100,1..2]" label to data type and label name of the global label to "CamData2", and specifies the character string for Target (Target).Reading/Writing data is as follows. · CamData2[1,1]: Value set for the start point · CamData2[1,2]: Value set for the initial stroke amount · CamData2[n + 1,1]: Nth point current position per cycle · CamData2[n + 1,2]: Nth point stroke *: When specifying labels, local labels can also be set. However, when setting the local label, specify the POU name to "@target modification". For details, refer to the following. ☞ Page 385 TARGET_REF structure

Point

When executing MCv_WriteProfileData (Profile Write) whose Target (Target) is "1: File" in the LREAL type array, Interpolation Method Specification (Interpolate) of the cam data format is output to the file as "0: Linear Interpolation".

- PROFILE_CAM_ELEMENT type

  Specify PROFILE_CAM_ELEMENT type structure array as follows.

  PROFILE_CAM_ELEMENT[m..n]

Item	Description
The number of elements (n - m + 1)	Set the array elements so that they are greater than the number of points that are to be read. When reading/writing entire operation profile data (Offset and Points are set to 0), set this to be Number of Sections (PROFILE_CAM_DATA.NumberOfSections) of the target cam data or more. However, an area larger than Number of Sections (PROFILE_CAM_DATA.NumberOfSections) read by Data1 is not updated when using MCv_ReadProfileData (Profile Read). An area larger than Number of Sections (PROFILE_CAM_DATA.NumberOfSections) specified by Data1 is not used when using MCv_WriteProfileData (Profile Write). If the ProfileControl version is earlier than “1.19”, it is necessary to match the number of elements of the specified array to Number of Sections (PROFILE_CAM_DATA.NumberOfSections) of the target cam data.

Specify the following PROFILE_CAM_ELEMENT type structures.

PROFILE_CAM_ELEMENT			Device offset	Description
Variable name	Name	Data type	(Total size: 36 points)	
CurveType	Cam curve type	MC_CAM_CURVE_TYPE	+0	Set the cam curve. (When End point (EndPoint) is 0, it is invalid.) • 0: Constant Velocity (ConstantSpeed) • 1: Constant Acceleration (ConstantAcceleration) • 2: Distorted Trapezoid (DistortedTrapezoid) • 3: Distorted Sine (DistortedSine) • 4: Distorted Constant Velocity (DistortedConstantSpeed) • 5: Cycloid (Cycloid) • 6: 5th curve (FifthCurve) • 7: Trapecloid (Trapecloid) • 8: Reverse Trapecloid (ReverseTrapecloid) • 9: Double Hypotenuse (DoubleHypotenuse) • 10: Reverse Double Hypotenuse (ReverseDoubleHypotenuse) • 11: Single Hypotenuse (SingleHypotenuse) • 12: 5th Curve (Adj.) (FifthCurve_SpeedDesignation)
EndPoint	End point*1	LREAL	+4	Set the position to the length per cycle (the current value per cycle). (0.0 < End point (EndPoint) ≤ Length per cycle)
Stroke	Stroke	LREAL	+8	Set the stroke position. (The absolute value of Stroke amount (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*2	LREAL	+28	Set the end point velocity of the cam curve. (-2500000000.0 < End point velocity (EndVelocity) < 2500000000.0)
EndAcceleration	End point acceleration*2	LREAL	+32	Set the end point acceleration of the cam curve. (2147483647.0 < End point acceleration (EndAcceleration) < 2147483647.0)

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

*2 Setting items used for read/write differ depending on Interpolation Method Specification (Interpolate) of the operation profile data. For details, refer to the following.

☞ Page 446 Cam data

Ex.

For PROFILE_CAM_ELEMENT type

· The following shows examples of setting the device/label Number of sections "10" to Target (Target) of the TARGET_REF structure.

Specification method	Setting example
Buffer memory	TARGET_REF.Target := "[DEV](PROFILE_CAM_ELEMENT[1..10])G1150000"
Label	TARGET_REF.Target := "[VAR]CamData2" *: CamData2 declares the "PROFILE_CAM_ELEMENT[1..10]" label to data type and label name of the global label to "CamData2", and specifies the character string for Target (Target).Reading/Writing data is as follows. *: When specifying labels, local labels can also be set. However, when setting the local label, specify the POU name to "@target modification". For details, refer to the following. ☞ Page 385 TARGET_REF structure

Rotary cutter

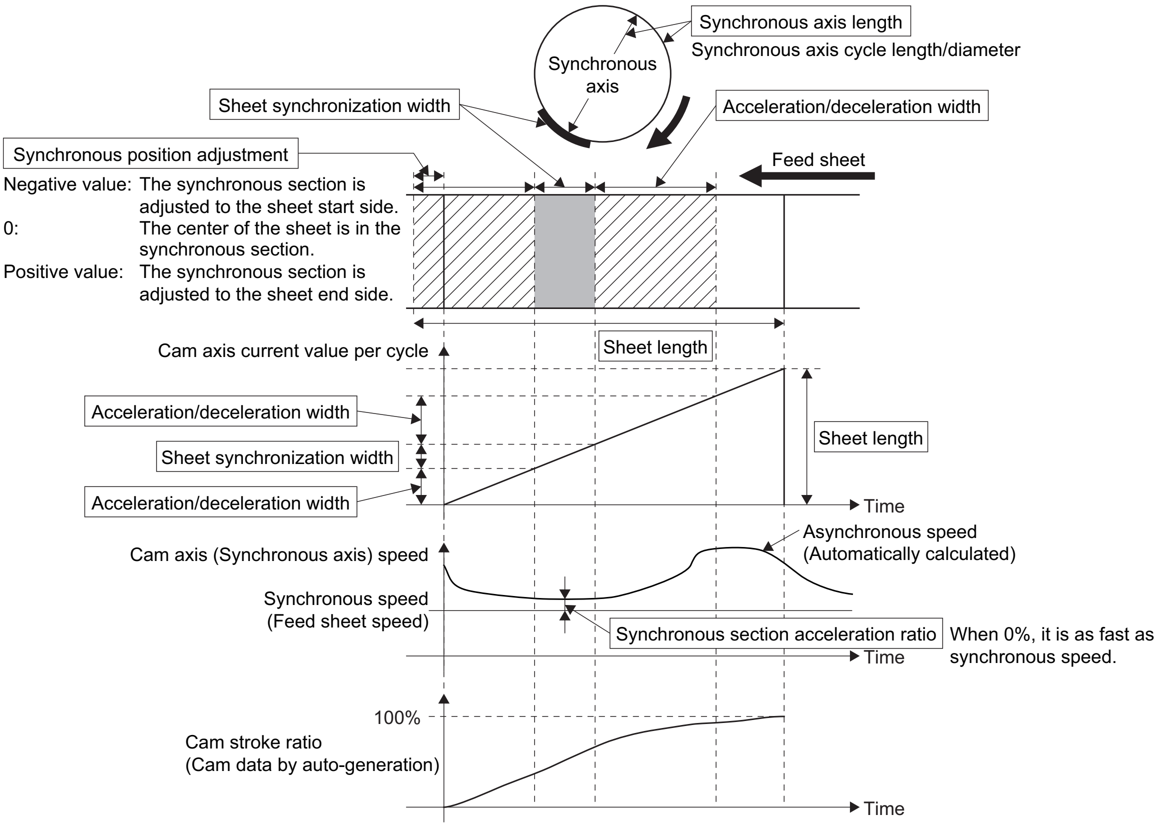
Generate cam data of the rotary cutter. The stroke data that is defined after equally dividing a cam pattern per cycle of rotary cutter by points of the resolution is opened in the open area.

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 options	◎
	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)



■ Read/write data structure

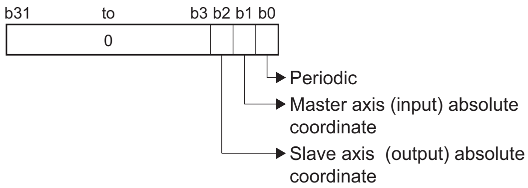
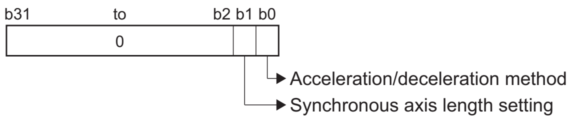
This section describes read data 1/write data 1 (Data1) specified by MCv_ReadProfileData (Profile Read)/ MCv_WriteProfileData (Profile Write) when executing read/write operation profile data. Read data 2/Write data 2 (Data2) is not used when executing read/write operation profile data of rotary cutter. However, when reading the expanded cam data by using the cam data for rotary cutter from the open area again, the cam data format (Interpolation Method Specification (Interpolate): "0: Linear Interpolation") should be specified. For details of the data structure for read/write specified with cam data format, refer to the following.

☞ Page 451 Read/write data structure

■ Rotary cutter (Specifying a double-precision real number)

- Read data 1/Write data 1 (Data1)

  Specify the PROFILE_ROTARY_CUTTER type structure.

PROFILE_ROTARY_CUTTER			Device offset	Description
Variable name	**Name**	**Data type**	**(Total size: 68 points)**	
Resolution	Resolution	DWORD(UDINT)	+0	Sets the resolution for generating the cam. • 8 to 32768
InputUnitString	Input unit character string	WSTRING(31)	+2	Sets the input unit character string.
Options	Options	DWORD(HEX)	+34	Specify Options.  ■Periodic (bit0) • 0: Non periodic • 1: Periodic For details, refer to the following. ☞ Page 428 Periodic (Periodic) ■Master axis (input) absolute coordinate (bit1) • 0: Relative coordinate • 1: Setting not possible[*1] For details, refer to the following. ☞ Page 432 Master axis absolute coordinate ■Slave axis (output) absolute coordinate (bit2) • 0: Relative coordinate • 1: Absolute coordinate For details, refer to the following. ☞ Page 433 Slave axis absolute coordinate *: For bit 3 to 31, specify "0".[*2]
AutoGeneration Options	Auto-generation options	WORD(HEX)	+36	Sets the auto-generation options with bit specification.  ■Acceleration/deceleration method (bit0) • 0: Trapezoidal acceleration/deceleration • 1: S-curve acceleration/deceleration ■Synchronous axis length setting (bit1) • 0: Diameter • 1: Cycle length *: Set bits 2 to 15 to "0".
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. (0.0 < Sheet length (SheetLength) ≤ 10000000000.0)
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. (0.0 < Sheet synchronization width (SheetSyncWidth) < Sheet length (SheetLength)) (0.0 < Sheet synchronization width (SheetSyncWidth) < Cycle length /Number of cutter (NumberOfCutter))

PROFILE_ROTARY_CUTTER			Device offset	Description
Variable name	Name	Data type	(Total size: 68 points)	
SyncAxisLength	Synchronous axis length	LREAL	+52	• Set the rotary cutter axis length. ■When the auto-generation options is set to "0: Diameter" it is calculated as "Cycle length = setting value × π ". ■When the auto-generation options is set to "1: Cycle length" it is calculated as "Cycle length = setting value". (0.0 < Cycle length < 10000000000.0)
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 Synchronous position tuning (SyncPositionAdjustment) < Sheet synchronization width (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 × Acceleration/deceleration width (AccDecWidth) ≤ Sheet length (SheetLength) - Sheet synchronization width (SheetSyncWidth))
AsyncSpd	Asynchronous velocity result	LREAL	+64	When the auto-generation is successfully completed, the asynchronous speed is stored as the ratio of the synchronous speed.

*1 When "1" is set, "Out of MasterAbsolute Range (error code: 341DH)" occurs and the FB will not start.
*2 When specifying other than "0", "Out of Options Range (error code: 1A4EH)" occurs and the FB will not start.

- Read data 2/Write data 2 (Data2)
  Specifying is not required. (It is ignored.)

Multiple axes positioning data

For a necessary definition as positioning data, refer to the following table.

◎: Necessary, ○: Optional, —: Unnecessary

Setting item		Multiple axes positioning data
Profile ID		○
Periodic		—
Input absolute coordinate		—
Output absolute coordinate		—
Operation profile data format		◎
Profile data type		◎
Interpolation method specification		—
Resolution/coordinate number		—
Start point		—
Initial stroke amount		—
Initial velocity		—
Initial acceleration		—
Options		—
Number of positioning data settings		○
Total number of positioning data		◎
Condition signal 1 to Condition signal 10 (ConditionSignal1 to ConditionSignal10)	IO Number (StartIO)	—
	Target (Target)	○
	Signal detection method (Detection)	○
	Compensation time (CompensationTime)	○
	Filter time (FilterTime)	○
Skip signal 1 to Skip signal 10 (SkipSignal1 to SkipSignal10)	IO Number (Source.StartIO)	—
	Target (Source.Target)	—
	Signal detection method (Detection)	—
	Compensation time (CompensationTime)	—
	Filter time (FilterTime)	—
Positioning data	Positioning data No. (DataNo)	◎
	Operation pattern (OperationPattern)	○[*1]
	Control method (ControlMethod)	◎
	Interpolation axes 1 to Interpolation axes 4 (InterpolationAxis1 to InterpolationAxis4)	○[*1]
	Circular interpolation mode (CircMode)	○[*1]
	Path selection (CircPathChoice)	○[*1]
	Target position/movement amount/end point 1 to Target position/movement amount/end point 4 (Position1 to Position4)	○[*1]
	Direction selection 1 to Direction selection 2 (Direction1 to Direction4)	○[*1]
	Sub point 1 to Sub point 4 (CircAuxPoint1 to CircAuxPoint4)	○[*1]
	Velocity mode (VelocityMode)	○[*1]
	Velocity (Velocity)	○[*1]
	Acceleration (Acceleration)	○[*1]
	Deceleration (Deceleration)	○[*1]
	Jerk (Jerk)	○[*1]
	Circular interpolation error tolerance (CircErrorTolerance)	○[*1]
	Positioning data options (DataOptions)	○[*1]
	Dwell time (DwellTime)	○[*1]
	M code (Mcode)	○[*1]
	M code output timing override (McodeOutput_Override)	○[*1]
	JUMP destination positioning data No. (JumpDestinationDataNo)	○[*1]
	Condition signal No. (ConditionSignalNo)	○[*1]
	Nuber of LOOP to LEND repetitions (LoopCount)	○[*1]

Setting item		Multiple axes positioning data
Positioning data	Skip signal No. (SkipSignalNo)	—
	Pitch (Pitch)	—
Length per cycle		—
Length per cycle unit setting		—
Length per cycle unit character string		—
Minimum value per cycle		—
Maximum value per cycle		—
Time per cycle		—
Stroke amount		—
Stroke amount unit setting		—
Stroke amount unit character string		—

*1 The items required to be set differ depending on the Control method (ControlMethod). For details, refer to the following.
☞ Page 241 Setting positioning data

■ Profile ID

It is a unique ID assigned by an open instruction.

ID can be specified with an arbitrary No.

14

■ Operation profile data format

When using the operation profile data format, set to "Multiple axes positioning data format".

■ Profile data type

When using the profile data type, set to "Multiple axes positioning data".

■ Number of positioning data settings

Specify the number of positioning data to be set.

Set the number of positioning data settings less than the total number of positioning data.

Only the settings for write can be omitted. (The setting value is ignored.)

When executing write, the motion system automatically calculates the number of positioning data settings.

When the value is out of setting range, or exceeding the total number of positioning data, "Operation Profile Data Incorrect (error code: 3410H)" occurs when opening.

- Setting range: 0 to 5000

■ Total number of positioning data

It is the total number of positioning data.

Specify the maximum value of Positioning data No. to be set by open and write.

When the value is out of setting range, "Operation Profile Data Incorrect (error code: 3410H)" occurs when opening.

- Setting range: 1 to 5000

Point

The larger the total number of positioning data is, the more positioning data can be set. However, the data size increases accordingly. For details on data size, refer to the following.
☞ Page 470 Capacity of operation profile data

■ Condition signal 1 to Condition signal 10 (ConditionSignal1 to ConditionSignal10)

When the Control method (ControlMethod) is set to "0082H: JUMP(JUMP)", set the execution conditions for JUMP. When the setting of each item is omitted, the value becomes the following initial value.

For details and setting range of each item, refer to the following.

☞ Page 266 Condition signal setting

Item	Initial value
I/O Number (StartIO)	0
Target (Target)	"" (a character string of length 0)
Signal detection method (Detection)	0: Detection at TRUE (HighLevel)
Compensation time (CompensationTime)	0.0
Filter time (FilterTime)	0.0

■ Positioning data

Set the interpolation operation to be executed with operation profile control. When the setting of each item is omitted, the value becomes the following initial value.

For details and the setting range of each item, refer to the following.

☞ Page 241 Setting positioning data

Item	Initial value
Positioning data No. (DataNo)	Omission not possible
Operation pattern (OperationPattern)	0: Positioning complete (PositioningComplete)
Control method (ControlMethod)	Omission not possible
Interpolation axes 1 to Interpolation axes 4 (InterpolationAxis1 to InterpolationAxis4)	0
Circular interpolation mode (CircMode)	0: Border point specification (mcBorder)
Path selection (CircPathChoice)	0: CW (mcCW)
Target position/movement amount/end point 1 to Target position/movement amount/end point 4 (Position1 to Position4)	0.0
Direction selection 1 to Direction selection 2 (Direction1 to Direction4)	0
Sub point 1 to Sub point 4 (CircAuxPoint1 to CircAuxPoint4)	0.0
Velocity mode (VelocityMode)	0: Vector velocity (VectorSpeed)
Velocity (Velocity)	0.0
Acceleration (Acceleration)	0.0
Deceleration (Deceleration)	0.0
Jerk (Jerk)	0.0
Circular interpolation error tolerance (CircErrorTolerance)	100.0
Positioning data options (DataOptions)	00000000H
Dwell time (DwellTime)	0.0
M code (Mcode)	0
M code output timing override (McodeOutput_Override)	0: Use FB options (UseFbOptions)
JUMP destination positioning data No. (JumpDestinationDataNo)	0
Condition signal No. (ConditionSignalNo)	0
Number of LOOP to LEND repetitions (LoopCount)	0
Skip signal No. (SkipSignalNo)	0
Pitch (Pitch)	0

■ Read/write data structure

This section describes read data 1/write data 1 (Data1) and read data 2/write data 2 (Data2) specified by MCv_ReadProfileData (Profile Read)/MCv_WriteProfileData (Profile Write) when executing read/write operation profile data.
Specify the section to be operated with Offset (Offset) or Number of read data/ Number of write data (Points).

- Read data 1/Write data 1 (Data 1)
  Specify the PROFILE_POSITIONING_DATA type structure.

PROFILE_POSITIONING_DATA			Device offset (Total size: 4 points)	Description
Variable name	Name	Data type		
NumberOfDataSettings	Number of positioning data settings	WORD(UINT)	+0	Sets the number of positioning data to be set.*1 • 0 to 5000
TotalNumberOfData	Total number of positioning data points	WORD(UINT)	+1	Sets the total number of positioning data points.*2 • 1 to 5000
Options	Options	DWORD(HEX)	+2	Sets the options. *: Set to "00000000H".

*1 It is not used in MCv_WriteProfileData (Profile Write). (The setting value is ignored.)
When executing MCv_WriteProfileData (Profile Write), the motion system automatically calculates the Number of positioning data settings (NumberOfDataSettings).

*2 When specifying a profile ID which has been opened/written, and overwriting it with MCv_WriteProfileData (Profile Write), the original open data will be discarded if the Total number of positioning data (TotalNumberOfData) does not match.
When setting the Number of write data (Points) to other than "0", and partially overwriting it with MCv_WriteProfileData (Profile Write), make the setting so that the Total number of positioning data (TotalNumberOfData) matches.

Ex.

Setting example of Target (TARGET_REF.Target)

· [DEV](PROFILE_POSITIONING_DATA)G11500000

· [VAR]PathData1 (PathData1 is PROFILE_POSITIONING_DATA type label)

- Read data 2/write data 2 (Data2)

  Specify the PROFILE_POSITIONING_DATA_ELEMENT type structure.

PROFILE_POSITIONING_DATA_ELEMENT[m..n]*1			Device offset (Total size: 88 points)	Description
Variable name	Name	Data type		
DataNo	Positioning data No.	WORD(UINT)	+0	Indicates the index No. of the positioning data. • 1 to total number of positioning data
OperationPattern	Operation Pattern	MC_OPERATION_PATTERN	+1	Sets the operation pattern.*2 • 0: Positioning complete (PositioningComplete) • 1: Continuous positioning (ContinuousPositioning) • 2: Continuous path (BlendingLow) (ContinuousBlendingLow) • 3: Continuous path (BlendingPrevious) (ContinuousBlendingPrevious) • 4: Continuous path (BlendingNext) (ContinuousBlendingNext) • 5: Continuous path (BlendingHigh) (ContinuousBlendingHigh)
ControlMethod	Control method	MC_CONTROL_METHOD	+2	Sets the control method.*2 • 0080H: NOP(NOP) • 0082H: JUMP(JUMP) • 0083H: LOOP(LOOP) • 0084H: LEND(LEND) • 0101H: Absolute value linear interpolation (LinearAbsolute) • 0102H: Relative value linear interpolation (LinearRelative) • 0103H: Absolute value circular interpolation (CircularAbsolute) • 0104H: Relative value circular interpolation (CircularRelative)
InterpolationAxis1	Interpolation axes 1	WORD(UINT)	+3	Sets the linear interpolation axes for linear interpolation, or the circular interpolation axes for circular interpolation depending on the setting of the control method (ControlMethod).*2 ☞ Page 253 Interpolation axes 1 to Interpolation axes 4 (InterpolationAxis1 to InterpolationAxis4)
InterpolationAxis2	Interpolation axes 2	WORD(UINT)	+4	
InterpolationAxis3	Interpolation axes 3	WORD(UINT)	+5	
InterpolationAxis4	Interpolation axes 4	WORD(UINT)	+6	
CircMode	Circular interpolation mode	MC_CIRC_MODE	+7	Set the specification method for circular interpolation.*2 • 0: Border point specification (mcBorder) • 1: Center point specification (mcCenter) • 2: Radius specification (mcRadius)
CircPathChoice	Path selection	MC_CIRC_PATHCHOICE	+8	Sets the rotation direction of the circular interpolation.*2 • 0: CW (mcCW) • 1: CCW (mcCCW • 2: Shortcut (mcShortWay) • 3: Detour (mcLongWay) • 4: CW Detour (mcCWLongWay) • 5: CCW Detour (mcCCWLongWay)
Reserve1[0..2]	Reserve 1	WORD(UINT)	+9	Not used. (For offset adjustment)
Position1	Target position/movement amount/end point 1	LREAL	+12	Set the target position, movement amount, and end point depending on the settings of Control method (ControlMethod).*2 ☞ Page 254 Target position/movement amount/end point 1 to target position/movement amount/end point 4 (Position1 to Position4)
Position2	Target position/movement amount/end point 2	LREAL	+16	
Position3	Target position/movement amount/end point 3	LREAL	+20	
Position4	Target position/movement amount/end point 4	LREAL	+24	
Direction1	Direction selection 1	MC_DIRECTION	+28	For linear interpolation control, this sets the direction used to move from the current position to the target position.*2 • 1: Positive direction (mcPositiveDirection) • 2: Negative direction (mcNegativeDirection) • 3: Shortest path (mcShortestWay)
Direction2	Direction selection 2	MC_DIRECTION	+29	
Direction3	Direction selection 3	MC_DIRECTION	+30	
Direction4	Direction selection 4	MC_DIRECTION	+31	

PROFILE_POSITIONING_DATA_ELEMENT[m..n]*1			Device offset (Total size: 88 points)	Description
Variable name	Name	Data type		
CircAuxPoint1	Sub point 1	LREAL	+32	For circular interpolation control, this sets the position from the current position at start to the border point and center point.*2 ■For "0: Border point specification (mcBorder)" and "1: Center point specification (mcCenter)" -10000000000.0 to 10000000000.0 ■For "2: Radius specification (mcRadius)" 0.000001 to 2147483647.0
CircAuxPoint2	Sub point 2	LREAL	+36	
CircAuxPoint3	Sub point 3	LREAL	+40	
CircAuxPoint4	Sub point 4	LREAL	+44	
VelocityMode	Velocity mode	MC_INTERPOLATE_SPEED_MODE	+48	Sets the velocity mode for linear interpolation control.*2 • 0: Vector velocity specification (VectorSpeed) • 1: Long axis velocity specification (LongAxisSpeed) • 2: Reference axis velocity specification (ReferenceAxisSpeed)
Reserve2[0..2]	Reserve 2	WORD(UINT)	+49	Not used. (For offset adjustment)
Velocity	Velocity	LREAL	+52	Sets the positioning speed of the multiple axes interpolation.*2 • 0.0, 0.0001 to 2500000000.0
Acceleration	Acceleration	LREAL	+56	Sets the acceleration for multiple axes interpolation.*2 • 0.0000, 0.0001 to 2147483647.0
Deceleration	Deceleration	LREAL	+60	Sets the deceleration for the multiple axes interpolation.*2 • 0.0000, 0.0001 to 2147483647.0
Jerk	Jerk	LREAL	+64	Sets the jerk for the multiple axes positioning control.*2 • 0.0000, 0.0001 to 2147483647.0
CircErrorTolerance	Circular interpolation error tolerance	LREAL	+68	Set the allowable range of a circular interpolation error.*2 • 0.000001 to 100000.0
DataOptions	Positioning data options	DWORD(HEX)	+72	Sets the linear interpolation or the circular interpolation options.*2 ☞ Page 256 Positioning data options (DataOptions)
Reserve3[0..1]	Reserve 3	WORD(UINT)	+74	Not used. (For offset adjustment)
DwellTime	Dwell time	LREAL	+76	Sets the dwell time.*2 • 0.0 [s]: Dwell function disabled • 0.000001 [s] to 8400.0 [s]: Dwell function enabled
Mcode	M code	WORD(UINT)	+80	Sets the M code.*2 • 1 to 65535
McodeOutput_Override	M code output timing override	MC_MCODE_OUTPUT_OVERRIDE	+81	M code output timing override.*2 • 0: Use FB options (UseFbOptions) • 1: WITH mode (WithMode) • 2: AFTER mode (AfterMode)
JumpDestinationDataNo	JUMP destination positioning data No.	WORD(UINT)	+82	Sets the positioning data No. of the jump destination.*2 • 1 to Total number of positioning data (5000)
ConditionSignalNo	Condition signal No.	WORD(UINT)	+83	Not used. *: Set to "0" at writing.
LoopCount	LOOP to LEND loop count	WORD(UINT)	+84	Sets the number of repetitions for loop control.*2 • 1 to 65535
SkipSignalNo	Skip signal No.	WORD(UINT)	+85	Not used. *: Set to "0" at writing.
Pitch	Pitch	WORD(UINT)	+86	Not used. *: Set to "0" at writing.
Reserve4	Reserve 4	WORD(UINT)	+87	Not used. (For offset adjustment)

*1 Set the number of elements (n – m + 1) of the specified array so that it is equal to or more than the number of points that are to be read/written.
When reading/writing the entire operation profile data (Offset and Points are "0"), set the number of array elements so that it equal to or more than Total Number of Positioning Data (PROFILE_POSITIONING_DATA.TotalNumberOfData).
If the number of the array elements is less, "Number of Read/Write Data Mismatch (error code: 3464H)" occurs.

*2 Errors caused by setting values do not occur at writing. When the written open area is used by the operation profile data control FB, an error will occur when the setting value is out of the setting range.

Ex.

When setting the device/label of the total number of positioning data "10" to Target (TARGET_REF.Target)

Specification method	Setting example
Buffer memory	TARGET_REF.Target := "[DEV](PROFILE_POSITIONING_DATA_ELEMENT[1..10])G1150000"
Label	TARGET_REF.Target := "[VAR]PositioningData2" *: Declare the "PROFILE_POSITIONING_DATA_ELEMENT[1..10]" label to data type and label name of the global label to "PositioningData2", and specify the character string of label name for Target (Target). Reading/Writing data is as follows. · PositioningData2[n]: The nth positioning data (PROFILE_POSITIONING_DATA_ELEMENT type) *: When specifying labels, local labels can also be set. However, when setting the local label, specify the POU name to "@target modification". For details, refer to the following. ☞ Page 385 TARGET_REF structure

Precautions

- When using condition signals in the operation profile data control FB, use only auto expand for operation profile data manipulation.
  Read/write of the condition signals cannot be performed by using MCv_ReadProfileData (Profile Read)/ MCv_WriteProfileData (Profile Write). When specifying an existing file and overwriting it by MCv_WriteProfileData (Profile Write), delete the condition signal and then overwrite the file. If a file created by using an engineering tool is overwritten, write the operation profile from the engineering tool to the motion system again.

Creating operation profile data

This section describes storage format and creating method of the operation profile data.

Storage format

The following storage format 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 Operation profile storage location (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 Operation profile storage location (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 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.

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.

When the profile data type includes the profile ID type of "function specific type" in the member, the set description is also reflected as the initial value in the profile ID type of "function specific type".

Ex.

When creating a operation profile data with "Name: ProfileData0001", "Format: Multiple axes positioning data", and "Profile ID: 1"

The initial value of ProfileData0001.PositioningData.ProfileID becomes "1". (Same as the initial value of ProfileData0001.ProfileData.ID.Number.)

The profile ID type of "function specific type" can be used as the I/O of operation profile data control FB or input parameter.

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 Operation profile storage location (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).

Operating operation profile data

Operation profile data uses files and open areas, and performs operation profile data manipulation 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 and the profile ID type can be used as the data type to perform operation profile data manipulation/ operation profile data controls.

- "File" is managed by the profile data type labels.
  Specify the file path and file name in Operation profile storage location (PROFILE_DATA.Location).
- "Open area" is managed by 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 arbitrary ID for Profile ID No. (PROFILE_ID.Number).
- Control can be carried out by specifying the label of profile ID type 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 FB execution (The write destination is "File")	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. • Operation profile data expand execution (Auto expand/operation profile data expand FB execution) • Operation profile data write FB execution (The write destination is "Open area")	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.

Point

To hold the change, write file by MCv_WriteProfileData (Profile Write) (file specification). Because the change is deleted when the power turns OFF, even if the data is changed in open area by MCv_WriteProfileData (Profile Write) (open area specification) during operation.

14

■ 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 • MC_POSITIONING_DATA_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 • MC_POSITIONING_DATA_ID	Data type to specify the operation profile data to use in the operation profile data control FB (MC_CamIn (Cam Operation Start), etc.). A unique identifiable No. is given by the operation profile data expand FB (MC_CamTableSelect (Cam Table Selection) , etc.). • Profile ID No. (Number)

The table shows data type and open FBs that can be used to each function.

Function	Profile data type	Profile ID type	Open FB
Cam Operation Start (MC_CamIn)	MC_CAM_REF	MC_CAM_ID	MC_CamTableSelect (Cam Table Selection)
Advanced synchronous control (MCv_AdvancedSync)			
Advanced synchronous control current position per cycle calculation (MCv_AdvPositionPerCycleCalc)			
Advanced synchronous control cam set position calculation (MCv_AdvCamSetPositionCalc)			
Multiple Axes Positioning Data Operation (MCv_MovePositioningData)	MC_POSITIONING_DATA_REF	MC_POSITIONING_DATA_ID	—

■ 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 Interpolation Method Specification (Interpolate) is "0: Linear Interpolation": Management information (1k bytes) + Coordinate number × 8 × 2 (byte) ■When Interpolation Method Specification (Interpolate) is "1: Section Interpolation" or "2: Spline Interpolation": Management information (1k bytes) + Resolution × 8 (byte)
	Rotary cutter	Management information (1k bytes) + Resolution × 8 (byte)
	Multiple axes positioning data	Management information (1k bytes) + 3200 (byte) + Total number of positioning data × 176 (byte)

In operation profile open processing, temporarily twice the data size memory of open area will be used for cam data (Interpolation Method Specification (Interpolate) is "1: Section Interpolation" and "2: Spline Interpolation") and rotary cutter, and the same data size memory of open area will be used for other than the above.

Operation profile data expansion

This section describes auto expand and expand.

Operation profile data created by "File" should be expanded to "Open area" because operation profile data of "Open area" is used for operation profile data control.

The method to expand to "Open area" has the following two types.

- Auto expand
- Operation profile data expand FB execution

■ Auto expand of operation profile data

Create on the engineering tool and the operation profile data set to auto expand is expanded automatically at power ON so that it can be used for operation profile data control without executing operation profile data expand FB.

The operation profile data satisfying the following all conditions can be used for its control without executing operation profile data expand FB because it is also expanded automatically at power ON.

- Other than "0" is specified to the initial value of Profile ID No. (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.
- No error such as out of the setting range in the data to be expanded.

The operation profile for automatic expansion will be searched for in both "%PROJECT_ROOT%/calc_profile/auto_open" and "%PROJECT_ROOT_DRIVE%/calc_profile/auto_open".

When both paths contain an operation profile with the same ID, the operation profile data stored in "%PROJECT_ROOT_DRIVE%/calc_profile/auto_open" is prioritized.

Point

- When setting auto expand, Profile ID (PROFILE_DATA.ID.Number), Periodic (Periodic), Master axis absolute coordinate (MasterAbsolute), and Slave axis absolute coordinate (SlaveAbsolute) open the same parameters in the csv file which can be specified in operation profile data expand FB.
- When there is an operation profile data which does not meet the requirement above during auto expand, Details of the error are registered in the event history without expanding automatically the profile data. In this case, the operation continues until automatically expanding the remaining operation profile data, without ending because of error.

■ Expand operation profile data

Expands "File" specified in Operation Profile Data Storage Location (PROFILE_DATA.Location) to "Open area" by using MC_CamTableSelect (Cam Table Selection).

- Expanded data to "Open area" is assigned profile IDs expressed in the profile ID type.
- An operation profile data expand FB that can be executed differs depending on the defined profile data type. For details, refer to the following.

  ☞ Page 446 Operation profile data types
- When an operation profile data expand FB that does not support the profile data type is executed, "Operation Profile Data Incorrect (error code: 3410H)" occurs.
- When an error has occurred while expanding the operation profile data, "Open area" in processing is discarded.

Point

- Expansion process of the operation profile data is carried out with motion service processing. It may take time to open depending on the processing details and the operation profile data.
- When PROFILE_DATA.ID.Number (ProfileID), Periodic (Periodic), Master axis absolute coordinate (MasterAbsolute), and Slave axis absolute coordinate (SlaveAbsolute) are present as input arguments for the operation profile data expand FB, the value set in "File" is ignored and the expansion is performed while referencing the input argument settings.
  (The setting value of "File" is not overwritten.)

Precautions

When rewriting "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.

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 (Profile Read)/ MCv_WriteProfileData (Profile Write) 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 446 Operation profile data types
When an error has occurred while writing operation profile data, the open area and the file 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 (Profile Read).

- The file specified in Profile data (ProfileData) and Read destination (Target), or the operation profile data in open area are read for the number of points specified in Number of read data (Points) from data specified with Offset (Offset).
- To read the whole operation profile data, specify "0" in both Offset (Offset) and Number of read data (Points).
- Read data is stored in variables specified in Read data 1 (Data1) and Write data 1 (Data 1).

■ Write operation profile data

Exexcutes write operation of operation profile data.
Operation profile data can be written by using the MCv_WriteProfileData (Profile Write).

- The file specified in Profile data (ProfileData) and Write destination (Target), or the operation profile data in open area are written for the number of points specified in Write destination (Points) from data specified with Offset (Offset).
- To write the whole operation profile data, specify "0" in both Offset (Offset) and Write destination (Points).
- Read data is stored in variables specified in Read data 1 (Data1) and Write data 1 (Data 1).

Precautions

When the operation profile data is written in the open area or file area using MCv_WriteProfileData (Profile Write) with the ProfileControl version being "1.9" or earlier, the setting values of Periodic (Periodic), Master axis absolute coordinate (MasterAbsolute) and Slave axis absolute coordinate (SlaveAbsolute) will become FALSE.
When changing setting values, follow the procedures below.

1. Write the operation profile data to the file area.
2. Specify the file written in step 1 in MC_CamTableSelect (Cam Table Selection), and set Periodic (Periodic), Master axis absolute coordinate (MasterAbsolute), and Slave axis absolute coordinate (SlaveAbsolute).

Control of operation profile data

Operation profile data is used to control of each function.

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.

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.

For the operation profile data expand FB corresponding to each operation profile data, refer to the following.

Operation profile data format	Operation profile data expand FB	Detail
Cam data	• MC_CamIn (Cam Operation Start) • MCv_AdvancedSync (Advanced Synchronous Control) • MCv_AdvPositionPerCycleCalc (Advanced Synchronous Control Current Position Per Cycle Calculation) • MCv_AdvCamSetPositionCalc (Advanced Synchronous Control Cam Set Position Calculation)	For details, refer to the following. MELSEC iQ-R Programming Manual (Motion Control Function Blocks)
Rotary cutter		
Multiple axes positioning data	Multiple Axes Positioning Data Operation (MCv_MovePositioningData)	

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.

Execution mode (ExecutionMode)

For the operation profile data expand FB and the operation profile data write FB, the operation for simultaneous execution can be specified with Execution mode (ExecutionMode).

Point

When no FB is in execution, the FB is executed immediately regardless of specification of Execution mode (ExecutionMode).

While the FB (open, write, read, control) that accesses open areas is in execution, Executing (Busy) of various FBs are TRUE. To give priority to the FB execution order based on the execution time of the FB (open, write, read), use this signal to the interlock.

○: Execution possible, ×: Error

On-going FB	FB to be executed			
	Open/Write			Read/Control
	0: Execute Immediately (mcImmediately)	1: Execute at Completion of Previous One (mcQueued)	3: Execute Speculatively (mcSpeculatively)	
Open/Write	○	○	×	×
Read/Control	○	○	×	○

■ 0: Execute Immediately (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, when executing open/write during FB execution (open, write, read, control), "Operation Profile Data Being Operated (error code: 3411H)" will occur in the following cases.

- The format of the operation profile data for executing open/write does not match the operation profile format of the open area.
- The capacity of the operation profile data for executing open/write does not match the capacity of the open area[*1].

*1 For the capacity of the open area, refer to the following.
☞ Page 470 Capacity of operation profile data

■ 1: Execute at Completion of Previous One (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.

The status of execution completion of each FB is shown below.

FB	Status
MC_CamTableSelect (Cam Table Selection)	Execution completion (Done) = TRUE
MCv_ReadProfileData (Profile Read)	Execution completion (Done) = TRUE
MCv_WriteProfileData (Profile Write)	Execution completion (Done) = TRUE
MC_CamIn (Cam Operation Start)	Periodic (MC_CamTableSelect.Periodic) = TRUE : Abortion of execution (MC_CamIn.CommandAborted) = TRUE Periodic (MC_CamTableSelect.Periodic) = FALSE : Abortion of execution (MC_CamIn.CommandAborted) = TRUE, Cam cycle completion (MC_CamIn.EndOfProfile) = TRUE
MCv_AdvancedSync (Advanced Synchronous Control)	Executing (Busy) = FALSE
MCv_AdvPositionPerCycleCalc (Advanced Synchronous Control Current Position Per Cycle Calculation)	Executing (Busy) = FALSE
MCv_AdvCamSetPositionCalc (Advanced Synchronous Control Cam Set Position Calculation)	Executing (Busy) = FALSE
MCv_MovePositioningData (Multiple Axes Positioning Data Operation)	Executing (Busy) = FALSE, and the Profile No. being executed (CurrentProfileNo) of MCv_MovePositioningData (Multiple Axes Positioning Data Operation) = 0

■ 3: Execute Speculatively (mcSpeculatively)

"Operation Profile Data Being Operated (error code: 3411H)" occurs, and the open area is not changed.

Sub function of operation profile data

Current value per cycle change function

Change the current value per cycle of an operation profile data control FB to an arbitrary value.

- Sets Instance ID (MC_CamIn.InstanceID) of corresponding to the operation profile data control FB (hereinafter called to as MC_CamIn) to Instance ID (InstanceID) of MCv_ChangeCycle (Current Value Change per Cycle).
- Public variables of MC_CamIn are shown below when current value per cycle change is performed.

Public variable	Updating value	Remark
Current value per cycle (InputsPerCycle)	Value specified in Current value per cycle (Cycle[])	—
Reference value (Reference)	Stroke value equivalent to Current value per cycle (Cycle[])	Reference value (MC_CamIn.Reference) is refreshed to fix Output value (MC_CamIn.OutputData).
Output value (OutputData)	Not updated	—

Point

Even if Slave axis absolute coordinate (MC_CamTableSelect.SlaveAbsolute) is TRUE (absolute coordinate), Reference value (Reference) is refreshed to fix Output value (OutputData) if current value per cycle change is performed.

14

- The method for current value per cycle change is as follows.

Method	Description
MCv_ChangeCycle (Current Value Change per Cycle)	Change it to the specified value at control execution.

- For specification of the current value per cycle to be changed, "absolute specification" or "relative specification" can be selected by Relative position selection (Relative).
- The current value per cycle of the following operation profile data control FB can be changed. When the current value per cycle of other operation profile data control FBs are changed "Current Value per Cycle Change Unsupported (error code: 3467H)" occurs.
  · MC_CamIn (Cam Operation Start)

■ Current value per cycle change

Uses Current value per cycle (MCv_ChangeCycle).

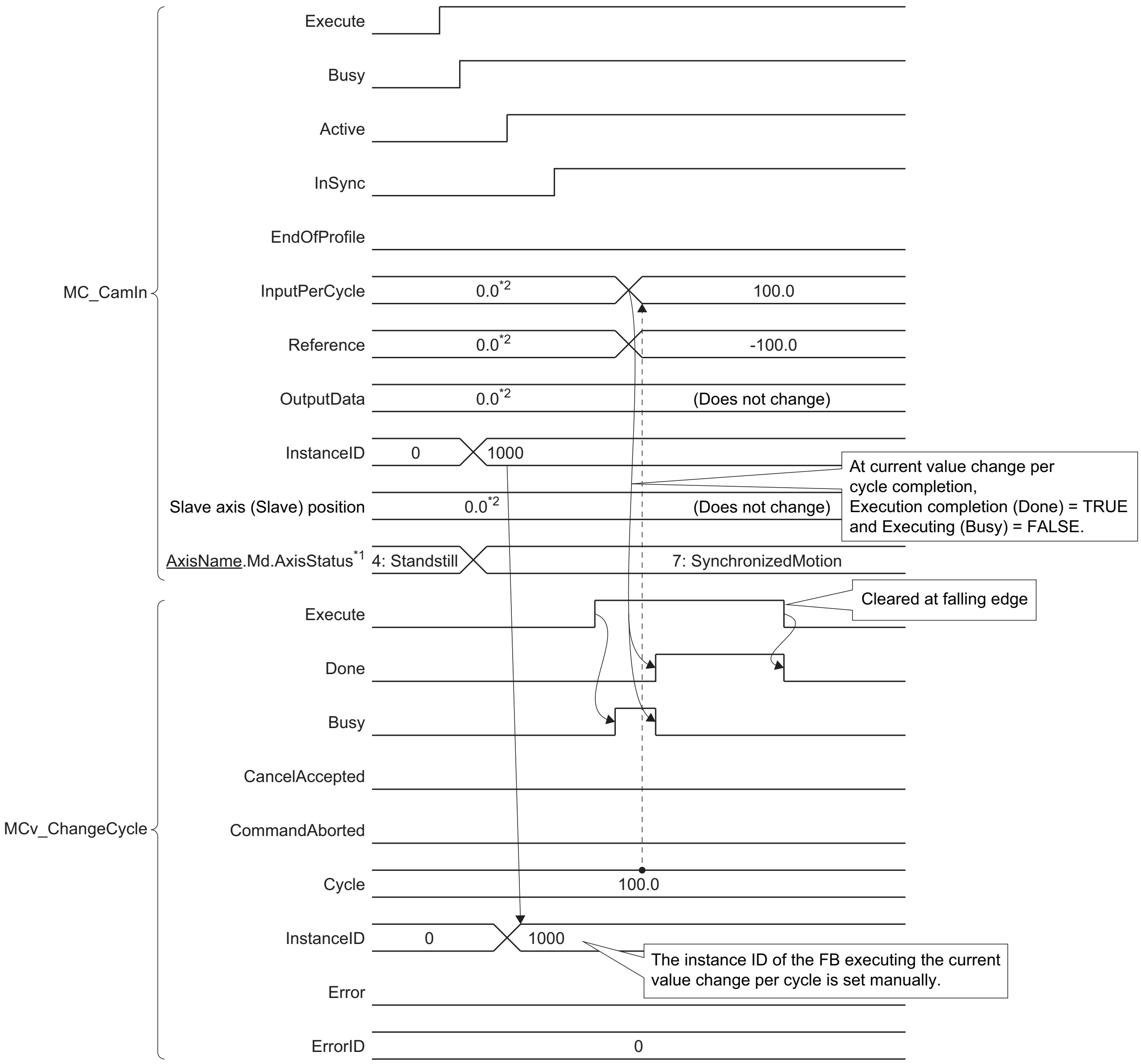
Changes the current value per cycle of the specified operation profile data control FB.

Executes current value per cycle change at rising edge detection of Execute command (Execute).

- For "0: Execute Immediately (mcImmediately)"
  Executed immediately at rising edge detection of Execute command (Execute) with Executing (MC_CamIn.Busy) = TRUE.

Ex.

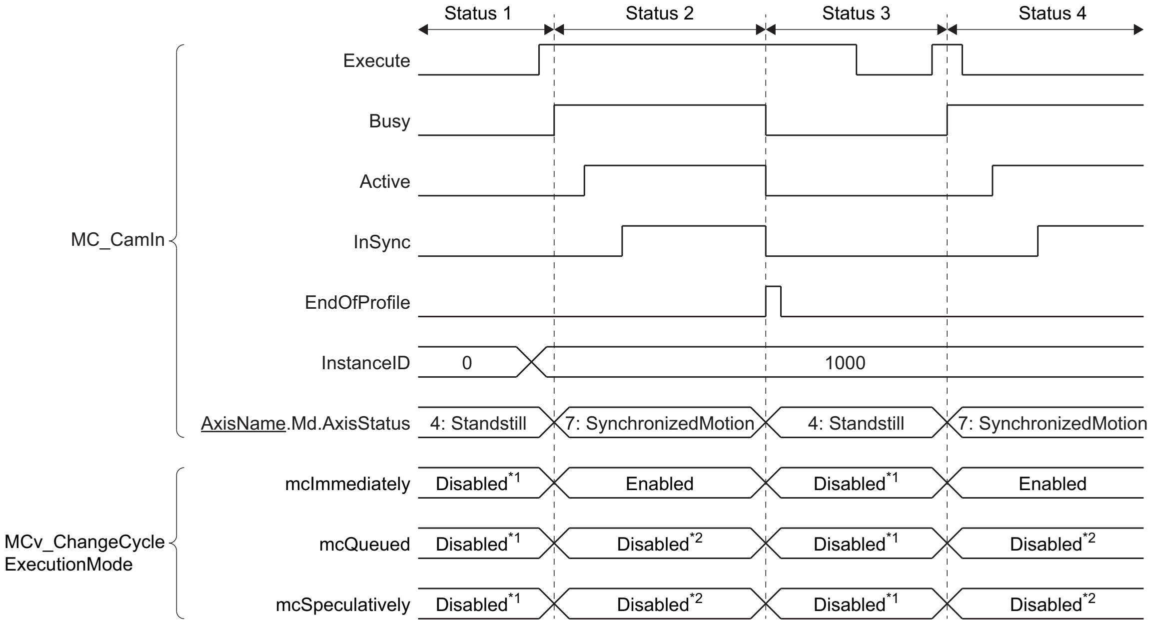
Sets Current value per cycle (Cycle) = 100.0 when Executing (MC_CamIn.Busy) = TRUE, and when executing current value per cycle change. (Current value of Master axis (Master) of MC_CamIn (Operation Available) shall not change.)



*1 This is axis state of Slave axis (Slave).

*2 Does not change because the master axis does not move.

• Current value per cycle change timing by each value



*1 "MCv_ChangeCycle Instruction Error (error code: 3497H (detail code: 000AH))" occurs.
*2 "MCv_ChangeCycle Instruction Error (error code: 3497H (detail code: 0002H))" occurs.

Status of FB of current value per cycle change target		Condition of state transition (Status change of the slave axis)		Operation of current value per cycle change
		Before transition	After transition	
Status 1	FB is not started	—(Initial status)	4: Standby (Standstill)	Current value per cycle change cannot be executed regardless of Execution mode (ExecutionMode). "MCv_ChangeCycle Instruction Error (error code: 3497H (detail code: 000AH))" occurs.
Status 2	FB is starting	4: Standby (Standstill)	7: During synchronous operation (SynchronizedMotion)	Depending on Execution mode (ExecutionMode), the timing is as follows ■0: Execute Immediately (mcImmediately) Current value per cycle change can be executed. The execution cycle of target FB is reflected. ■1: Execute at Completion of Previous One (mcQueued) "MCv_ChangeCycle Instruction Error (error code: 3497H (detail code: 0002H))" occurs. ■3: Execute Speculatively (mcSpeculatively) "MCv_ChangeCycle Instruction Error (error code: 3497H (detail code: 0002H))" occurs.
Status 3	FB after the control completed	7: During synchronous operation (SynchronizedMotion)	4: Standby (Standstill)	Current value per cycle change cannot be executed regardless of Execution mode (ExecutionMode). "MCv_ChangeCycle Instruction Error (error code: 3497H (detail code: 000AH))" occurs.
Status 4	FB after restarting	4: Standby (Standstill)	7: During synchronous operation (SynchronizedMotion)	The operation is the same as Status 2. Repeats after that as Status 4 (Status 2) → Status 3 → ... Current value per cycle change operation is not affected even Execute command (Execute) has TRUE/FALSE differences in Status 2 and Status 4.

Precautions

Relevant add-ons

The following add-ons are required to use this function.

- ProfileControl
- PackagingApp (when using the rotary cutter)

System memory capacity

■ RAM usage

Refer to the following.

☞ Page 470 Capacity of operation profile data

■ Backup RAM usage

Backup RAM is not used.

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 the operation profile data format and Interpolation method specification (Interpolate).

Operation profile data format	Interpolation method specification (Interpolate)	N
Cam data format	0: Linear interpolation	7 (start row of previous step) + (coordinate number × 2) + 2 (length per cycle, stroke amount)
	1: Section interpolation	7 (start row of previous step) + Number of sections + 2 (length per cycle, stroke amount)
	2: Spline interpolation	
Multiple axes positioning data format	—	7 (start row of previous step) + 1 (row: total number of positioning data) + 1 (row: condition signal) + 1 (row: skip signal) + Number of sections + 2 (length per cycle, stroke amount)

14

Common items

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 444 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 444 Operation Profile Data	Integer	Specifies periodic.	0x0103
	3	Input absolute coordinate	○	☞ Page 444 Operation Profile Data	Integer	Specifies input absolute coordinate.	0x0103
	4	Output absolute coordinate	○	☞ Page 444 Operation Profile Data	Integer	Specifies output absolute coordinate.	0x0103
3	1	Operation profile data format	×	99: Cam data format 123: Cam data format for rotary cutter (double precision specification) 1000: Multiple axes positioning data format	Integer	Specifies operation profile data format.	0x0103
	2	Profile data type	○	0: General profile data 1: Cam data 6: Digital output data 200: Multiple axes positioning data • Initial value: 0	Integer	Specifies the profile data type. ☞ Page 444 Operation Profile Data	0x0103
	3	Interpolation method specification	○	0: Linear Interpolation 1: Section Interpolation 2: Spline Interpolation • Initial value: 0	Integer	Specifies the interpolation method between data.	0x0103
4	1	Resolution/ coordinate number	×	☞ Page 444 Operation Profile Data • Initial value: 0	Unsigned integer	Sets the resolution or the number of coordinates.	0x0103
5	1	Start point	○	*1 99: Valid for the cam data format (section interpolation, spline interpolation) • Initial value: 0	Real number	Specifies the start point.	0x0103
	2	Initial stroke amount	○	*1 99: Valid for the cam data format (section interpolation, spline interpolation) • Initial value: 0	Real number	Specifies the initial stroke amount.	0x0103
	3	Initial velocity	○	*1 99: Valid for the cam data format (section interpolation) • Initial value: 0	Real number	Specifies the initial velocity	0x0103
	4	Initial acceleration	○	*1 99: Valid for the cam data format (section interpolation) • Initial value: 0	Real number	Specifies the initial acceleration	0x0103
	5	Options	○	*2 • Initial value: 0	Unsigned integer	Defines for each operation profile data format.	0x0103
6	1	Number of sections	×	*2 • 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

*1 Differs depending on the data format.

*2 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 74 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 444 Operation Profile Data	0x0103
	3	Unit character string	○	Up to 31 characters • Initial value: "" ☞ Page 74 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 (Stroke)

○: 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 74 Technical Units	Unsigned integer	Specifies the unit of stroke amount. Units that can be specified differ depending on the profile data type. ☞ Page 444 Operation Profile Data	0x0103
	3	Unit character string	○	Up to 31 characters • Initial value: "" ☞ Page 74 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 (Interpolate)	0: Linear Interpolation

■ 6th line: Number of sections

Specify "0".

■ 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 80 Positioning Range	Real number	☞ Page 446 Cam 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 80 Positioning Range	Real number	☞ Page 446 Cam data	0x0103

Cam data format (Interpolation method specification: "1: Section 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 (Interpolate)	1: Section Interpolation

■ 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 446 Cam data	Integer	☞ Page 446 Cam data	0x0103
	2	End point	×	☞ Page 446 Cam 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	1	Operation profile data format	99: Cam data format
	3	Interpolation Method Specification (Interpolate)	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 446 Cam data	Real number	☞ Page 446 Cam data	0x0103
	2	Stroke	×		Real number		0x0103
8	1-2	Same as line No.7					
⋮ (For the number of sections)							

Cam data format for rotary cutter (double precision specification)

The following shows a case for specifying setting value below.

Line No.	Column No.	Item	Setting value
3	1	Operation profile data format	123: Cam data format for rotary cutter (double precision specification)

■ 6th line: Number of sections

Specify "0".

■ 7th line: Parameter

×: Omission not possible

Line No.	Column No.	Item	Omission	Setting range (Initial value)	Setting format	Description	Supported file version
7	1	Auto-generation options	×	☞ Page 456 Rotary cutter	Unsigned integer	☞ Page 456 Rotary cutter	0x0103
	2	Synchronous section acceleration ratio	×		Real number		0x0103
	3	Sheet length	×		Real number	☞ Page 456 Rotary cutter *: The current value per cycle is ignored.	0x0103
	4	Sheet synchronization width	×		Real number	☞ Page 456 Rotary cutter	0x0103
	5	Synchronous axis length	×		Real number		0x0103
	6	Synchronous position adjustment	×		Real number		0x0103
	7	Acceleration/ deceleration width	×		Real number		0x0103
	8	Number of cutter	×		Unsigned integer		0x0103

Multiple axes positioning data format

The following shows a case for specifying setting value below.

Line No.	Column No.	Item	Setting value
3	1	Operation profile data format	1000: Multiple axes positioning data format

■6th line: Number of sections

×: Omission not possible

Line No.	Column No.	Item	Omission	Setting range (Initial value)	Setting format	Description	Supported file version
6	1	Number of sections	×	0 to (total number of positioning data)	Unsigned integer	Specifies the number of positioning data settings. ☞ Page 460 Multiple axes positioning data	0x0103

■7th line: Parameter

×: Omission not possible

Line No.	Column No.	Item	Omission	Setting range (Initial value)	Setting format	Description	Supported file version
7	1	Total number of positioning data	×	1 to 5000	Unsigned integer	☞ Page 460 Multiple axes positioning data	0x0103

■ 8th line: Condition signal

○: Omission possible

Line No.	Column No.	Item		Omission	Setting range (Initial value)	Setting format	Description	Supported file version
8*1	1	Condition signal 1 (Condition Signal1)	I/O Number (StartIO)	○*2	—*3	Unsigned integer	☞ Page 460 Multiple axes positioning data	0x0103
	2		Target (Target)	○*2	—*3 Up to 63 characters	Character string		0x0103
	3		Signal Detection Method (Detection)	○*2	—*3	Integer		0x0103
	4		Compensation Time (CompensationTime)	○*2	—*3	Real number		0x0103
	5		Filter Time (FilterTime)	○*2	—*3	Real number		0x0103
	6 ⋮ 10	Condition signal 2 (Condition Signal2)	Same as Column No.1 to No.5					
	11 ⋮ 15	Condition signal 3 (Condition Signal3)	Same as Column No.1 to No.5					
	16 ⋮ 20	Condition signal 4 (Condition Signal4)	Same as Column No.1 to No.5					
	21 ⋮ 25	Condition signal 5 (Condition Signal5)	Same as Column No.1 to No.5					
	26 ⋮ 30	Condition signal 6 (Condition Signal6)	Same as Column No.1 to No.5					
	31 ⋮ 35	Condition signal 7 (Condition Signal7)	Same as Column No.1 to No.5					
	36 ⋮ 40	Condition signal 8 (Condition Signal8)	Same as Column No.1 to No.5					
	41 ⋮ 45	Condition signal 9 (Condition Signal9)	Same as Column No.1 to No.5					
	46 ⋮ 50	Condition signal 10 (Condition Signal10)	Same as Column No.1 to No.5					

*1 If the setting of none of the columns is required, specify " " (a character string of length 0) to the line.

*2 This item cannot be omitted when setting the column after.
When the setting of the corresponding column itself is not required, specify " " (a character string of length 0).

*3 An arbitrary value can be set. Errors caused by setting values do not occur at operation profile data opening.
However, there are setting ranges according to the operation profile data control FB to be used. For details, refer to the following.
☞ Page 266 Condition signal setting

■ 9th line: Skip signal (SkipSignal)

Specify " " (a character string of length 0) to the line.

■ 10th line or later: Positioning data

○: Omission possible, ×: Omission not possible

Line No.	Column No.	Item	Omission	Setting range (Initial value)	Setting format	Description	Supported file version
10	1	Positioning data No. (DataNo)	×	1 to (Total number of positioning data)	Unsigned integer	☞ Page 460 Multiple axes positioning data	0x0103
	2	Operation pattern (OperationPattern)	○[*1]	—[*2]	Integer		0x0103
	3	Control method (ControlMethod)	×	—[*2]	Integer		0x0103
	4	Interpolation Axes 1 (InterpolationAxis1)	○[*1]	—[*2]	Integer		0x0103
	5	Interpolation Axes 2 (InterpolationAxis2)	○[*1]	—[*2]	Integer		0x0103
	6	Interpolation Axes 3 (InterpolationAxis3)	○[*1]	—[*2]	Integer		0x0103
	7	Interpolation Axes 4 (InterpolationAxis4)	○[*1]	—[*2]	Integer		0x0103
	8	Circular interpolation mode (CircMode)	○[*1]	—[*2]	Integer		0x0103
	9	Path selection (CircPathChoice)	○[*1]	—[*2]	Integer		0x0103
	10	Target position/ movement amount/ end point 1 (Position1)	○[*1]	—[*2]	Real number		0x0103
	11	Target position/ movement amount/ end point 2 (Position2)	○[*1]	—[*2]	Real number		0x0103
	12	Target position/ movement amount/ end point 3 (Position3)	○[*1]	—[*2]	Real number		0x0103
	13	Target position/ movement amount/ end point 4 (Position4)	○[*1]	—[*2]	Real number		0x0103
	14	Direction selection 1 (Direction1)	○[*1]	—[*2]	Integer		0x0103
	15	Direction selection 2 (Direction2)	○[*1]	—[*2]	Integer		0x0103
	16	Direction selection 3 (Direction3)	○[*1]	—[*2]	Integer		0x0103
	17	Direction selection 4 (Direction4)	○[*1]	—[*2]	Integer		0x0103
	18	Sub point 1 (CircAuxPoint1)	○[*1]	—[*2]	Real number		0x0103
	19	Sub point 2 (CircAuxPoint2)	○[*1]	—[*2]	Real number		0x0103
	20	Sub point 3 (CircAuxPoint3)	○[*1]	—[*2]	Real number		0x0103
	21	Sub point 4 (CircAuxPoint4)	○[*1]	—[*2]	Real number		0x0103
	22	Velocity mode (VelocityMode)	○[*1]	—[*2]	Integer		0x0103
	23	Velocity (Velocity)	○[*1]	—[*2]	Real number		0x0103
	24	Acceleration (Acceleration)	○[*1]	—[*2]	Real number		0x0103

Line No.	Column No.	Item	Omission	Setting range (Initial value)	Setting format	Description	Supported file version
10	25	Deceleration (Deceleration)	○*1	—*2	Real number	☞ Page 460 Multiple axes positioning data	0x0103
	26	Jerk (Jerk)	○*1	—*2	Real number		0x0103
	27	Circular interpolation error tolerance (CircErrorTolerance)	○*1	—*2	Real number		0x0103
	28	Positioning data options (DataOptions)	○*1	—*2	Unsigned integer		0x0103
	29	Dwell time (DwellTime)	○*1	—*2	Real number		0x0103
	30	M codes (Mcode)	○*1	—*2	Unsigned integer		0x0103
	31	M code output timing override (McodeOutput_Ove rridev	○*1	—*2	Integer		0x0103
	32	JUMP destination positioning data No. (JumpDestinationD ataNo)	○*1	—*2	Unsigned integer		0x0103
	33	Condition signal No. (ConditionSignalNo)	○*1	0	Unsigned integer		0x0103
	34	Number of LOOP to LEND repetitions (LoopCount)	○*1	—*2	Unsigned integer		0x0103
	35	Skip signal No. (SkipSignalNo)	○*1	0	Unsigned integer	Not used. *: Omit or specify "0".	0x0103
	36	Pitch (Pitch)	○*1	0	Unsigned integer		0x0103
11	1-36	Same as Line No. 10.					
⋮ (For the number of sections)							

*1 This item cannot be omitted when setting the column after. When the setting of the corresponding column itself is not required, specify "" (a character string of length 0).

*2 An arbitrary value can be set. Errors caused by setting values do not occur at operation profile data opening.
However, there are setting ranges according to the operation profile data control FB to be used. For details, refer to the following.
☞ Page 241 Setting positioning data

14

The following is an example of writing a csv file for setting positioning data.

Ex.

When executing 2 axes absolute value circular interpolation control

• Positioning data

Line No.	Column No.	Item	Setting value
10	1	Positioning data No. (DataNo)	1
	2	Operation pattern (OperationPattern)	1: Continuous positioning (ContinuousPositioning)
	3	Control method (ControlMethod)	0103h: Absolute value circular interpolation (CircularAbsolute)
	4	Interpolation axes 1 (InterpolationAxis1)	Configuration axis 2
	5	Interpolation axes 2 (InterpolationAxis2)	Configuration axis 3
	8	Circular interpolation mode (CircMode)	1: Center point specification (mcCenter)
	9	Path selection (CircPathChoice)	0: CW (mcCW)
	10	Target position/movement amount/end point 1 (Position1)	Configuration axis 2: 100.0
	11	Target position/movement amount/end point 2 (Position2)	Configuration axis 3: 100.0
	18	Sub point 1 (CircAuxPoint1)	Configuration axis 2: 50.0
	19	Sub point 2 (CircAuxPoint2)	Configuration axis 3: 50.0
	23	Velocity (Velocity)	10.0
	24	Acceleration (Acceleration)	1.0
	25	Deceleration (Deceleration)	1.0
	26	Jerk (Jerk)	2.0
	27	Circular interpolation error tolerance (CircErrorTolerance)	100.0
	28	Positioning data options (DataOptions)	0
	29	Dwell time (DwellTime)	100.0

• csv file (delimiters: comma (,))

Writing
1,1,0x0103,2,3,,,1,0,100.0,100.0,,,,,,,50.0,50.0,,,,10.0,1.0,1.0,2.0,100.0,0,100.0

Ex.

When making data No. 10 NOP

• csv file (delimiters: comma (,))

Writing
10,,0x0080

Notation for different setting formats

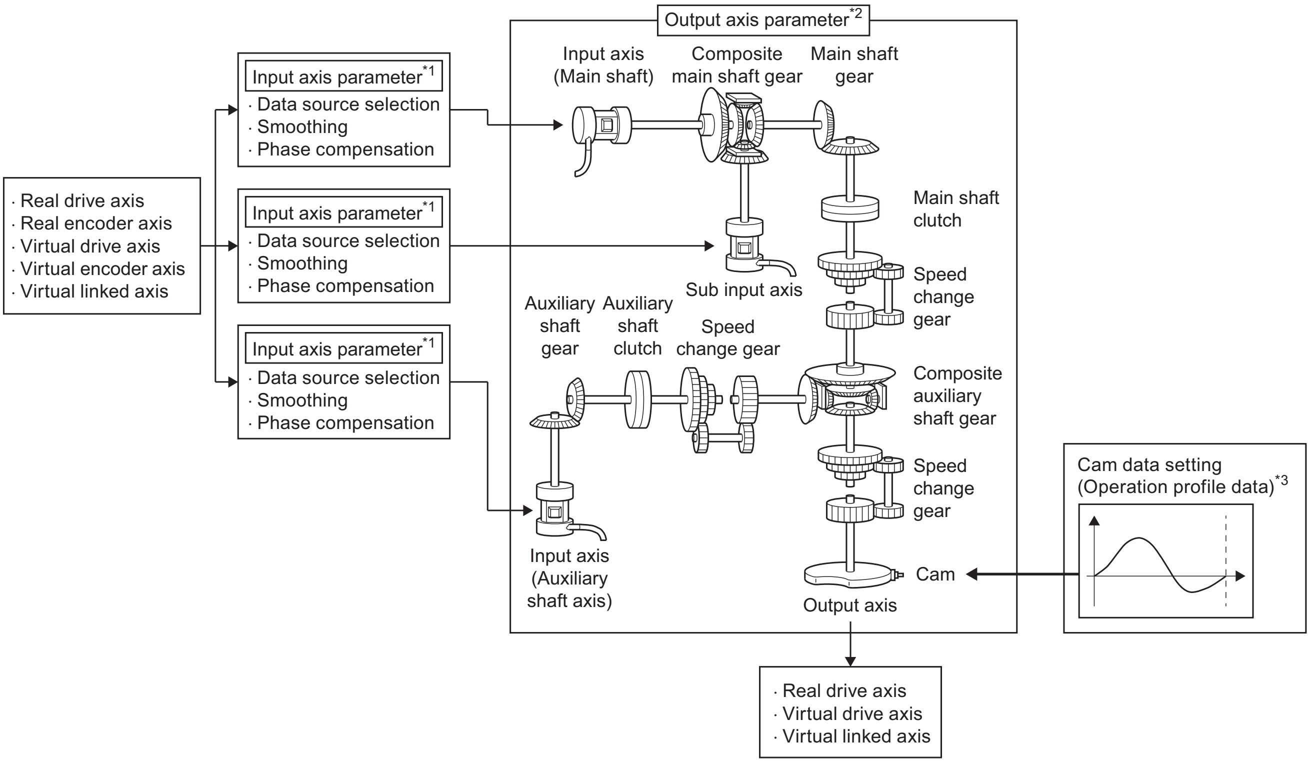
The following shows the notation for different setting formats.

Type		Notation	Example
Integer	Decimal	Directly enter a signed or unsigned decimal number.	123 +123 -123
	Hexadecimal	Add "0x" or "0X" before a hexadecimal number.	0xFF 0XEAD
Real number	Decimal notation	Directly enter a signed or unsigned real number with a decimal.	2.34 +2.34 -2.34
	Exponential notation	Add "e" or "E", and a signed or unsigned exponent (decimal) at the end of the real number.	1.0e6 1.0E-6
Character string		Directly enter a character string.	ABC

15 ADVANCED SYNCHRONOUS CONTROL

15.1 Overview of Advanced Synchronous Control

Advanced synchronous control can be achieved using software instead of controlling mechanically with gears, clutches, speed change gears or cams etc.
Advanced synchronous control synchronizes control with the input axis by setting parameters for advanced synchronous control and starting MCv_AdvancedSync (Advanced Synchronous Control) on each output axis.
Set the input axes for advanced synchronous control and the filter settings using the input axis parameters.
Set the output axis for advanced synchronous control using the output axis parameters. The setting of the auxiliary axes and sub input axes, as well as the settings for gears, clutches, speed change gears, and cams are included under the output axis parameters.



*1 For details of input axes, refer to the following.
☞ Page 509 Input Axis Function
*2 For details of output axes, refer to the following.
☞ Page 518 Output Axis Function
*3 For details of cams, refer to the following.
☞ Page 562 Output axis module

Relevant variables

Input axis (AdvInputName.)

Variable name	Name	Reference
LabelID	Axis label ID	Page 511 Input axis (AdvInputName.)
Axis	Axis information	
PrConst.	Input axis parameter constant	Page 490 Input axis parameters constant (AdvInputName.PrConst.)
Pr.	Input axis parameter	Page 490 Input axis parameters (AdvInputName.Pr.)
Md.	Input axis monitor data	Page 490 Input axis monitor data (AdvInputName.Md.)

■ Input axis parameters constant (AdvInputName.PrConst.)

Variable name	Name	Reference
SourceValue	Data source selection	Page 512 Input axis parameter constant (AdvInputName.PrConst.)
SmoothingTimeConstant	Smoothing time constant	
PhaseCompensationTimeConstant	Phase compensation time constant	
DirectionRestriction	Moving direction restriction	

■ Input axis parameters (AdvInputName.Pr.)

Variable name	Name	Reference
PhaseCompAdvanceTime	Phase compensation advance time	Page 514 Input axis parameter (AdvInputName.Pr.)

■ Input axis monitor data (AdvInputName.Md.)

Variable name	Name	Reference
ActiveStatus	Active status	Page 515 Input axis monitor data (AdvInputName.Md.)
CumulativePosition	Cumulative current position	
SetVelocity	Set velocity	
PhaseCompensationAmount	Phase Compensation Amount	
DirectionRestrictionAmount	Moving direction restriction amount	

Output axis (AdvOutputName.)

Variable name	Name	Reference
LabelID	Axis label ID	Page 519 Output axis (AdvOutputName.)
Axis	Axis information	
PrConst.	Output axis parameter constant	Page 491 Output axis parameters constant (AdvOutPutName.PrConst.)
Pr.	Output axis parameter	Page 491 Output axis parameters (AdvOutPutName.Pr.)
Md.	Output axis monitor data	Page 493 Output axis monitor data (AdvOutPutName.Md.)
Cd.	Output axis control data	Page 494 Output axis control data (AdvOutPutName.Cd.)

■ Output axis parameters constant (AdvOutPutName.PrConst.)

Variable name	Name	Reference
SmoothingTimeConstant	Output axis smoothing time constant	☞ Page 520 Output axis parameters constant (AdvOutPutName.PrConst.)
MasterClutchSmoothingTimeConstant	Main shaft clutch smoothing time constant	☞ Page 524 Output axis parameter constant (AdvOutPutName.PrConst.)
AuxClutchSmoothingTimeConstant	Auxiliary shaft clutch smoothing time constant	☞ Page 535 Output axis parameters constant (AdvOutPutName.PrConst.)
MasterSpeedChangeGearSmoothingTimeConstant	Main shaft speed change gear smoothing time constant	☞ Page 560 Output axis parameters constant (AdvOutPutName.PrConst.)
AuxSpeedChangeGearSmoothingTimeConstant	Auxiliary shaft speed change gear smoothing time constant	
OutSpeedChangeGearSmoothingTimeConstant	Output axis speed change gear smoothing time constant	

■ Output axis parameters (AdvOutPutName.Pr.)

Variable name	Name	Reference
SubAxis	Sub input axis object	☞ Page 520 Output axis parameter (AdvOutPutName.Pr.)
AuxAxis	Auxiliary shaft object	
MasterOnClutchSignal	Main shaft clutch ON signal setting	☞ Page 524 Output axis parameters (AdvOutPutName.Pr.)
MasterOffClutchSignal	Main shaft clutch OFF signal setting	
AuxOnClutchSignal	Auxiliary shaft clutch ON signal setting	☞ Page 535 Output axis parameters (AdvOutPutName.Pr.)
AuxOffClutchSignal	Auxiliary shaft clutch OFF signal setting	
Gear.	Gear parameter	*1
Clutch.	Clutch parameter	*2
SpeedChangeGear.	Speed change gear parameter	*3
Cam.	Cam parameter	*4
Restore.	Synchronous control initial position parameter	*5

*1 Gear parameter (AdvOutPutName.Pr.Gear.)

Variable name	Name	Reference
MasterCompositeGear	Composite main shaft gear	☞ Page 523 Gear parameter of output axis parameters (AdvOutPutName.Pr.Gear.)
MasterNumerator	Main shaft gear: Numerator	
MasterDenominator	Main shaft gear: Denominator	
AuxCompositeGear	Composite auxiliary shaft gear	☞ Page 534 Gear parameter of output axis parameters (AdvOutPutName.Pr.Gear.)
AuxNumerator	Auxiliary shaft gear: Numerator	
AuxDenominator	Auxiliary shaft gear: Denominator	

15

*2 Clutch parameter (AdvOutPutName.Pr.Clutch.)

Variable name	Name	Reference
MasterOnControl	Main shaft clutch ON control setting	Page 526 Clutch parameter of output axis parameters (AdvOutputName.Pr.Clutch.)
MasterOffControl	Main shaft clutch OFF control setting	
MasterReference	Main shaft clutch reference address setting	
MasterOnAddress	Main shaft clutch ON address	
MasterMovementAmountBeforeOn	Movement amount before main shaft clutch ON	
MasterOffAddress	Main shaft clutch OFF address	
MasterMovementAmountBeforeOff	Movement amount before main shaft clutch OFF	
MasterSmoothingMethod	Main shaft clutch smoothing method	
MasterOnSlippageAmount	Slippage amount at main shaft clutch ON	
MasterOffSlippageAmount	Slippage amount at main shaft clutch OFF	
AuxOnControl	Auxiliary shaft clutch ON control setting	Page 537 Clutch parameter of output axis parameters (AdvOutputName.Pr.Clutch.)
AuxOffControl	Auxiliary shaft clutch OFF control setting	
AuxReference	Auxiliary shaft clutch reference address setting	
AuxOnAddress	Auxiliary shaft clutch ON address	
AuxMovementAmountBeforeOn	Movement amount before auxiliary shaft clutch ON	
AuxOffAddress	Auxiliary shaft clutch OFF address	
AuxMovementAmountBeforeOff	Movement amount before auxiliary shaft clutch OFF	
AuxSmoothingMethod	Auxiliary shaft clutch smoothing method	
AuxOnSlippageAmount	Slippage amount at auxiliary shaft clutch ON	
AuxOffSlippageAmount	Slippage amount at auxiliary shaft clutch OFF	

*3 Speed change gear parameter (AdvOutPutName.Pr.SpeedChangeGear.)

Variable name	Name	Reference
MasterGearIn	Main shaft speed change gear valid setting	Page 559 Speed change gear parameter of output axis parameters (AdvOutPutName.Pr.SpeedChangeGear.)
MasterRatioNumerator	Main shaft speed change ratio: Numerator	
MasterRatioDenominator	Main shaft speed change ratio: Denominator	
AuxGearIn	Auxiliary shaft speed change gear valid setting	
AuxRatioNumerator	Auxiliary shaft speed change ratio: Numerator	
AuxRatioDenominator	Auxiliary shaft speed change ratio: Denominator	
OutGearIn	Output axis speed change gear valid setting	
OutRatioNumerator	Output axis speed change ratio: Numerator	
OutRatioDenominator	Output axis speed change ratio: Denominator	

*4 Cam parameter (AdvOutPutName.Pr.Cam.)

Variable name	Name	Reference
LengthPerCycle	Length per cycle	Page 563 Cam parameter of output axis parameters (AdvOutputName.Pr.Cam.)
StrokeAmount	Cam stroke amount	
CamNo.	Cam No.	
StartingPoint	Cam starting point	
LengthPerCycleChange	Length per cycle change setting	
PhaseCompensationAdvancedTime	Phase compensation advance time	
PhaseCompensationTimeConstant	Phase compensation time constant	

*5 Synchronous control initial position parameter (AdvOutPutName.Pr.Restore.)

Variable name	Name	Reference
MasterGearPositionPerCycleMethod	Setting method of current position per cycle after main shaft gear	Page 602 Synchronous control initial position parameter (AdvOutputName.Pr.Restore.)
MasterGearInitialPositionPerCycle	Current position per cycle after main shaft gear (initial setting)	
AuxGearPositionPerCycleMethod	Setting method of current position per cycle after auxiliary shaft gear	
AuxGearInitialPositionPerCycle	Current position per cycle after auxiliary shaft gear (initial setting)	
PositionRestorationObject	Cam axis position restoration object	
ReferenceSetPositionMethod	Setting method of cam reference position	
PositionPerCycleMethod	Setting method of current position per cycle	
InitialReferenceSetPosition	Cam reference position (initial setting)	
InitialPositionPerCycle	Current position per cycle (initial setting)	
RestorationAllowablePosition	Cam set position restoration: Allowable movement amount setting	

■ Output axis monitor data (AdvOutPutName.Md.)

Variable name	Name	Reference
SyncStatus	Synchronization status	Page 521 Output axis monitor (AdvOutputName.Md.)
MasterAxisNo	Main shaft	
SubAxisNo	Sub input axis	
AuxAxisNo	Auxiliary shaft	
Clutch.	Clutch monitor	*1
Cam.	Cam monitor	*2

*1 Clutch monitor (AdvOutputName.Md.Clutch.)

Variable name	Name	Reference
MasterOnOffStatus	Main shaft clutch ON/OFF status	Page 530 Clutch monitor of output axis monitor data (AdvOutputName.Md.Clutch.)
MasterSmoothingStatus	Main shaft clutch smoothing status	
MasterCumulativeSlippage	Main shaft clutch slippage (accumulative)	
AuxOnOffStatus	Auxiliary shaft clutch ON/OFF status	Page 543 Clutch monitor of output axis monitor data (AdvOutputName.Md.Clutch.)
AuxSmoothingStatus	Auxiliary shaft clutch smoothing status	
AuxCumulativeSlippage	Auxiliary shaft clutch slippage (accumulative)	

*2 Cam monitor (AdvOutputName.Md.Cam.)

Variable name	Name	Reference
MasterCompositeGearSetPosition	Current position after composite main shaft gear	Page 567 Cam monitor of output axis monitor data (AdvOutputName.Md.Cam.)
MasterGearPositionPerCycle	Current position per cycle after main shaft gear	
AuxGearPositionPerCycle	Current position per cycle after auxiliary shaft gear	
PhaseCompensationAmount	Phase compensation amount	
PositionPerCycle	Current position per cycle	
ReferenceSetPosition	Cam reference position	
SetPosition	Cam set position	
CamNo	Execution profile No.	
StrokeAmount	Execution cam stroke amount	
LengthPerCycle	Execution length per cycle	
StartingPoint	Execution starting point	
SyncControlChangeStatus	Synchronous control change status	Page 582 Cam monitor of output axis parameters (AdvOutputName.Md.Cam.)

■ Output axis control data (AdvOutPutName.Cd.)

Variable name	Name	Reference
Clutch.	Clutch control data	*1
SpeedChangeGear.	Speed change gear control data	*2
Cam.	Cam control data	*3

*1 Clutch control data (AdvOutputName.Cd.Clutch.)

Variable name	Name	Reference
MasterCommand	Main shaft clutch command	Page 531 Clutch control data of output axis control data (AdvOutputName.Cd.Clutch.)
MasterInvalidCommand	Main shaft clutch control invalid command	
MasterForcedOff	Main shaft clutch forced OFF command	
MasterClutchSmoothingTimeConstant	Main shaft clutch smoothing time constant change value	
AuxCommand	Auxiliary shaft clutch command	Page 544 Clutch control data of output axis monitor data (AdvOutputName.Cd.Clutch.)
AuxInvalidCommand	Auxiliary shaft clutch control invalid command	
AuxForcedOff	Auxiliary shaft clutch forced OFF command	
AuxClutchSmoothingTimeConstant	Auxiliary shaft clutch smoothing time constant change value	

*2 Speed change gear control data (AdvOutputName.Cd.SpeedChangeGear.)

Variable name	Name	Reference
MasterSmoothingTimeConstant	Main shaft speed change gear smoothing time constant change value	Page 560 Speed change gear control data of output axis parameters (AdvOutPutName.Cd.SpeedChangeGear.)
AuxSmoothingTimeConstant	Auxiliary shaft speed change gear smoothing time constant change value	
OutSmoothingTimeConstant	Output axis speed change gear smoothing time constant change value	

*3 Cam control data (AdvOutputName.Cd.Cam.)

Variable name	Name	Reference
OutSmoothingTimeConstant	Output axis smoothing time constant change value	Page 569 Cam control data of output axis parameters (AdvOutputName.Cd.Cam.)
SyncControlChangeRequest	Synchronous control change request	Page 583 Cam control data of output axis parameters (AdvOutputName.Cd.Cam.)
SyncControlChangeCommand	Synchronous control change command	
SyncControlChange	Synchronous control change value	
SyncControlReflectionTime	Synchronous control reflection time	

Relevant FBs

For details on Motion control FBs, refer to the following.

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

Motion control FB	Name	Description
MCv_AdvancedSync	Advanced synchronous control	Starts synchronization control according to the set advanced synchronous control setting.
MCv_AdvPositionPerCycleCalc	Advanced synchronous control current position per cycle calculation	Calculates Current Position per Cycle based on the set cam data.
MCv_AdvCamSetPositionCalc	Advanced synchronous control cam set position calculation	Calculates Cam Set Position based on the set cam data.

15.2 Control Details for Advanced Synchronous Control

This section describes the procedure for conducting advanced synchronous control.

Phase	Step	Procedure
Preparation	STEP1	Generate the axes to be set as the input axis and output axis for advanced synchronous control.
	STEP2	Set the input axis parameters.
	STEP3	If using user-defined cam data, set the operation profile data.
	STEP4	Set the output axis parameters.
Synchronous control start	STEP5	Start MCv_AdvancedSync (Advanced Synchronous Control).
		Confirm synchronization is being conducted.
		Start the input axis.
Synchronous control monitor, control change	STEP6	Change the speed change ratio and cam No. that monitor the synchronous control operation status.
Synchronous control end	STEP7	End synchronous control.

End of control

Procedure for executing advanced synchronous control

This section describes the details of the procedure for conducting advanced synchronous control.

1. Generate the axes to be set as the input axis and output axis for advanced synchronous control.

In order to use advanced synchronous control, the axes to be set as the input axis and output axis must be generated.

Axes that can be set as an input axis and output axis are shown in the following table.For the specifications of each axis type, refer to the following.

☞ Page 30 Axis
☞ Page 56 Axis Assignment

○: Settable, ×: Not settable

Axis type		Input axis	Output axis	Remark
Real axis	Real drive axis	○	○	When control for multiple FBs is required, execute each FB with a virtual linked axis, and configure the axes to transmit the results (command) to the real drive axis.
	Real encoder axis	○	×	Use as Master axis (Master). When it is used as Slave axis (Slave), "Necessary Slave Object Unset (error code: 1AA8H)" occurs and it does not start.
Virtual axis	Virtual drive axis	○	○	Commands can be generated mainly with positioning control.
	Virtual linked axis	○	○	Use as an intermediary axis to transmit a command to the real drive axis or general output axis. When using multiple FBs, assign them to virtual linked axes.
	Virtual encoder axis	○	×	Use as Master axis (Master). When it is used as Slave axis (Slave), "Necessary Slave Object Unset (error code: 1AA8H)" occurs and it does not start.

Generating the advanced synchronous control input axis and output axis will consume memory per each setting. Adjust the maximum RAM size (MotionControl_AdvancedSync_System.PrConst.RamSizeMax) supported to match the system structure. For memory usage, refer to the following.

☞ Page 508 System memory usage

Also, if the axes set as an input axis or output axis are using an absolute position system, backup RAM will be consumed. For backup RAM usage for each setting, refer to the following.

☞ Page 588 Backup data management for advanced synchronous control

2. Set the input axis parameters

The GX Works3 motion control setting function generates an input axis from "Input Setting" (1) under "Advanced Synchronous Control". When an input axis is generated, ADV_INPUT type labels are generated in global label "Adv+Global" (2).*1

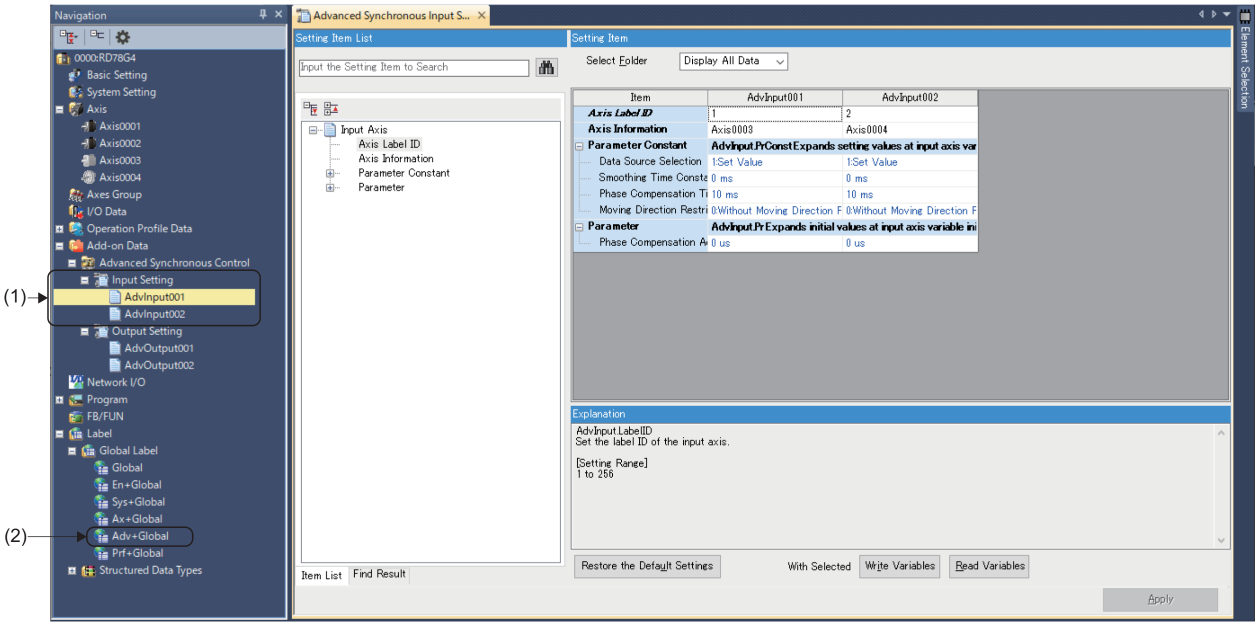
Set the axis to be used as an input axis in the Axis information (AdvInputName.Axis) of the input axis under input settings. It is not possible to set the same axis to the axis information of different axes. For details of settable axis types, refer to step 1.

For details on input settings, refer to the following.

☞ Page 509 Input Axis Function

*1 ADV_INPUT type labels defined in local label cannot be used in advanced synchronous control.

Navigation window ⇨ "Add-on Data" ⇨ "Advanced Synchronous Control" ⇨ "Input Setting"



3. Set the cam data

Set the cam data from the operation profile.

Set this when using a cam whose slave axis synchronizes it's position with the master axis according to the cam data and input settings.

Before using the cam function, the cam table must be transmitted to the open area and made controllable. For details on transmitting a cam table to the open area and data types that can be controlled by the cam operation, refer to the following.

☞ Page 562 Output axis module

☞ Page 570 Cam function

☞ Page 444 OPERATION PROFILE FUNCTION

Set the set operation profile to the output setting Cam No. (AdvOutputName.Pr.Cam.CamNo).

When using a Cam No. other than "0", it is required to install the add-on "ProfileControl" in advance.

If MCv_AdvancedSync (Advanced Synchronous Control), MCv_AdvPositionPerCycleCalc (Advanced Synchronous Control Current Position Per Cycle Calculation), MCv_AdvCamSetPositionCalc (Advanced Synchronous Control Cam Cet Position Calculation) is started without "ProfileControl" installed, the following errors occur.

Motion control FB	Error code
MCv_AdvancedSync (Advanced Synchronous Control)	"Add-on Acquisition Failure (error code: 1AF6H)"
MCv_AdvPositionPerCycleCalc (Advanced Synchronous Control Current Position Per Cycle Calculation)	"Add-on Acquisition Failure (error code: 34ABH)"
MCv_AdvCamSetPositionCalc (Advanced Synchronous ControlCam Set Position Calculation)	

4. Set the output axis

The GX Works3 motion control setting function generates an output axis from "Output Setting" (3) under "Advanced Synchronous Control". When an output axis is generated, ADV_OUTPUT type labels are generated in global label "Adv+Global" (2).

Set the axis to be used as an output axis in the Axis information (AdvOutputName.Axis) of the output axis under output settings. It is not possible to set the same axis to the axis information of different axes.

By setting the input axis set in step 1 to the Auxiliary shaft object (AdvOutputName.Pr.AuxAxis) and Sub input axis object (AdvOutputName.Pr.SubAxis), the auxiliary shaft and sub input axis inputs can be combined to the main shaft.

For details on output settings, refer to the following.

☞ Page 518 Output Axis Function

If parameters are not supported, the function name will be displayed in the setting image, but the parameters will not be displayed in the setting item.

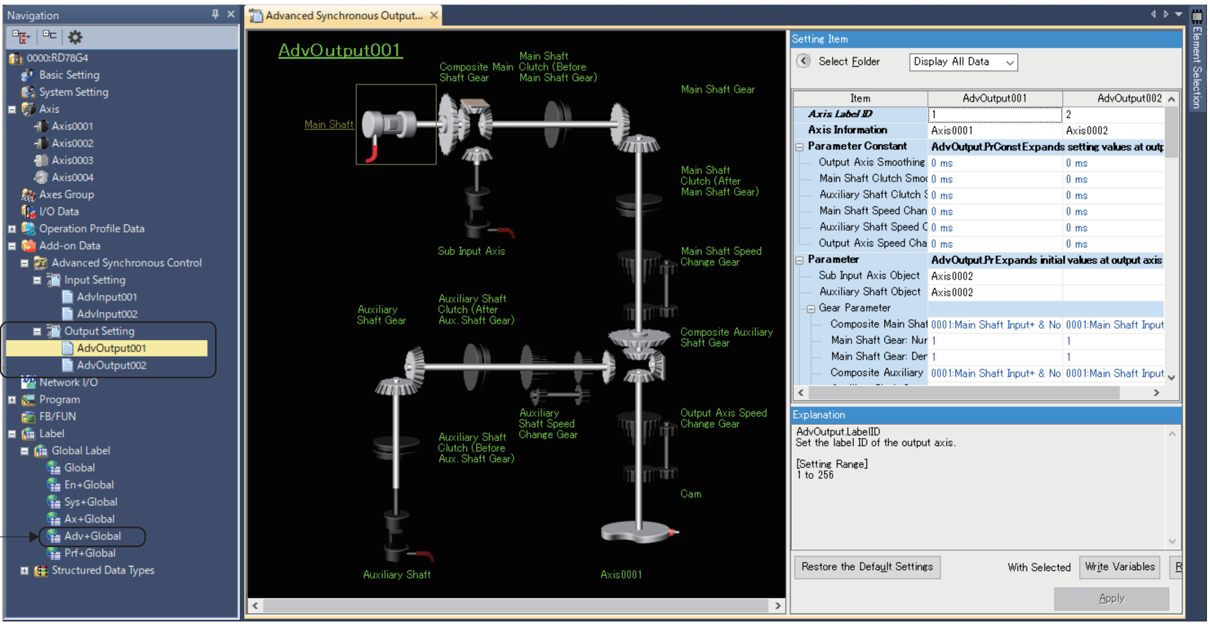
For details on the supported version for each parameter, refer to the following.

☞ Page 508 Combining versions

Navigation window ⇨ "Add-on Data" ⇨ "Advanced Synchronous Control" ⇨ "Output Setting"

(3)

(2)



5. Start synchronous control

Set the input axis and output axis set in steps 1. to 3. to the MCv_AdvancedSync (Advanced Synchronous Control) input/output variables.

- Input axis: Master axis (Master)
- Output axis: Slave axis (Slave)

If using analysis mode, set the following input variables.

- Synchronous control analysis mode setting (Options (Options): Bit 16)

Start synchronous control by starting MCv_AdvancedSync (Advanced Synchronous Control). For details on starting and confirmation during synchronization, refer to the following.

☞ Page 500 Starting/stopping for advanced synchronous control

6. Change the synchronous control monitor and control

Monitors such as the advanced synchronous control position per cycle can be confirmed in the Output axis monitor data (AdvOutputName.Md.).

The advanced synchronous control gear, clutch, speed change gear, and synchronous control change function can be used in the output axis control data (AdvOutputName.Cd) and the output axis parameters (AdvOutputName.Pr) of the output axis while MCv_AdvancedSync (Advanced Synchronous Control) is running.

For details on output axis functions, refer to the following.

☞ Page 518 Output Axis Function

7. Stop synchronous control

To stop the operation, use the following method.

- Change MCv_AdvancedSync (Advanced Synchronous Control) Enable (Enable) from TRUE to FALSE.
- Start MC_Stop (Forced Stop).

Advanced synchronous control can be ended even while the input axis is in operation. However, as the output axis loses synchronization while the input axis is stopped, it is recommended to stop the operation of the input axis before ending.

For details on stopping advanced synchronous control, refer to the following.

☞ Page 500 Starting/stopping for advanced synchronous control

Starting/stopping for advanced synchronous control

This section describes how to start and stop the operation for advanced synchronous control.

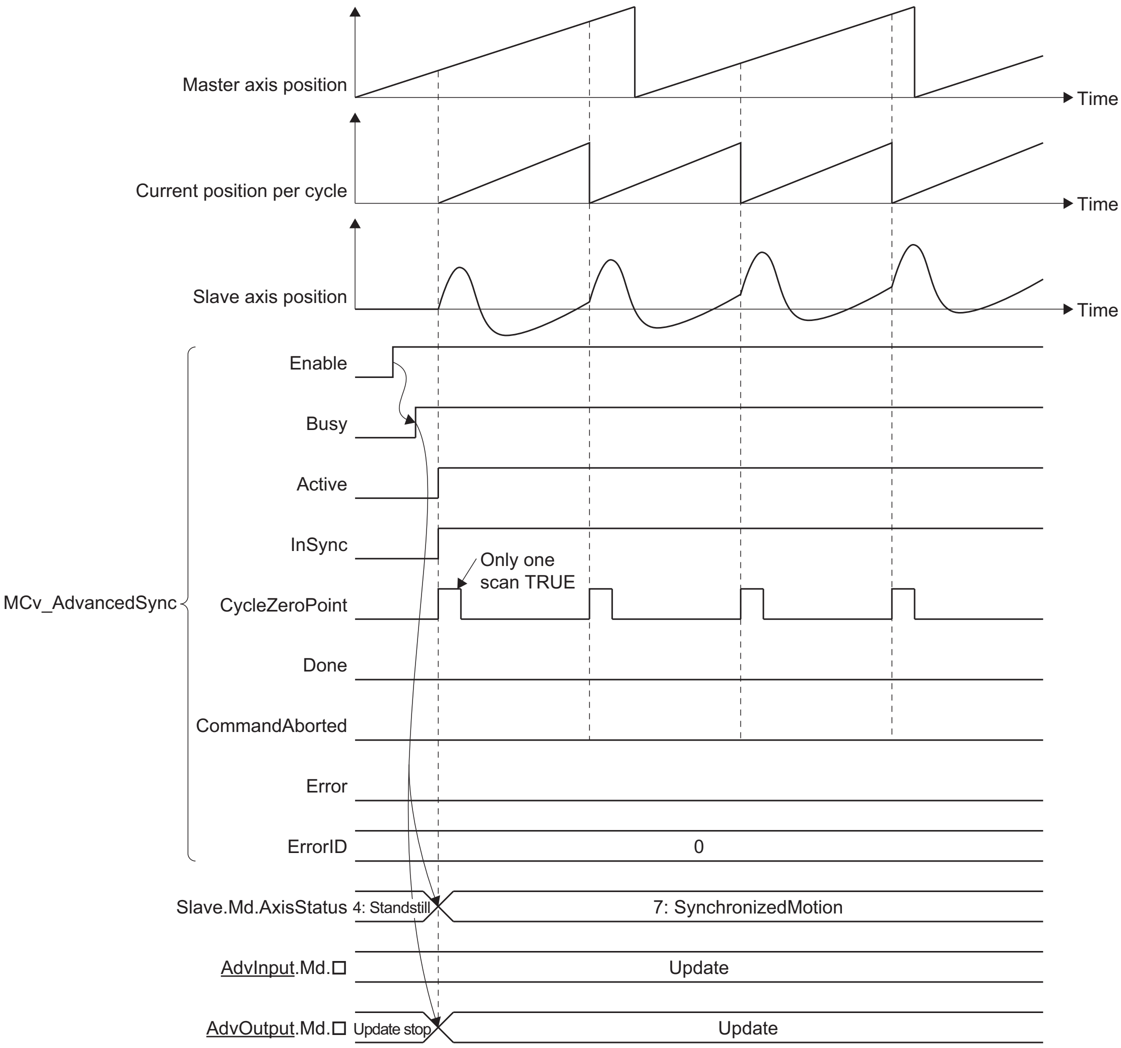
Operation when starting

To start advanced synchronous control, set MCv_AdvancedSync (Advanced Synchronous Control) Enable (Enable) to TRUE. Synchronous control can only be started when the slave axis Axis status (AxisName.Md.AxisStatus) is "4: Standby (Standstill)".

Upon starting advanced synchronous control , the slave axis Axis status (AxisName.Md.AxisStatus) changes to "7: During synchronous operation (SynchronizedMotion)". After starting synchronous control, if the status is not "4: Standby (Standstill)", the slave axis Axis status (AxisName.Md.AxisStatus) will change to "1: Stopping on error (ErrorStop)" and "Start Not Possible (error code: 1AADH)" occurs.

The Input axis monitor data (AdvInputName.Md.□) will continuously update regardless of whether synchronous control has started.

The Output axis monitor data (AdvOutputName.Md.□) will update after the slave axis Axis status (AxisName.Md.AxisStatus) changes to "7: During synchronous operation (SynchronizedMotion)".



Stop operation

When MCv_AdvancedSync (Advanced Synchronous Control) Enable (Enable) is set to FALSE, or if a stop cause occurred in the axis set in axis information of the output axis, the Slave axis (Slave) immediately stops or decelerates to a stop according to the stop cause and stopping process.

Stop causes that occur during axis operation and the stopping process when each cause occurs are shown below.

Stop cause*1	Stopped axis	Axis status (AxisName.Md.AxisStatus)		Stopping process*2
		During deceleration stop	After stopping	
Enable (Enable) is set to FALSE during synchronous control by MCv_AdvancedSync (Advanced Synchronous Control)	Each axis	7: During synchronous operation (SynchronizedMotion)	4: Standby (Standstill)	Deceleration stop/Immediate stop (according to Stop Selection at Stop Cause Occurrence (AxisName.Pr.StopMode_General)

*1 For other stop causes, refer to the following.

☞ Page 166 Stop causes during axis operation

*2 For each operation of stopping processes, refer to the following.

☞ Page 168 Stop processes for single axis

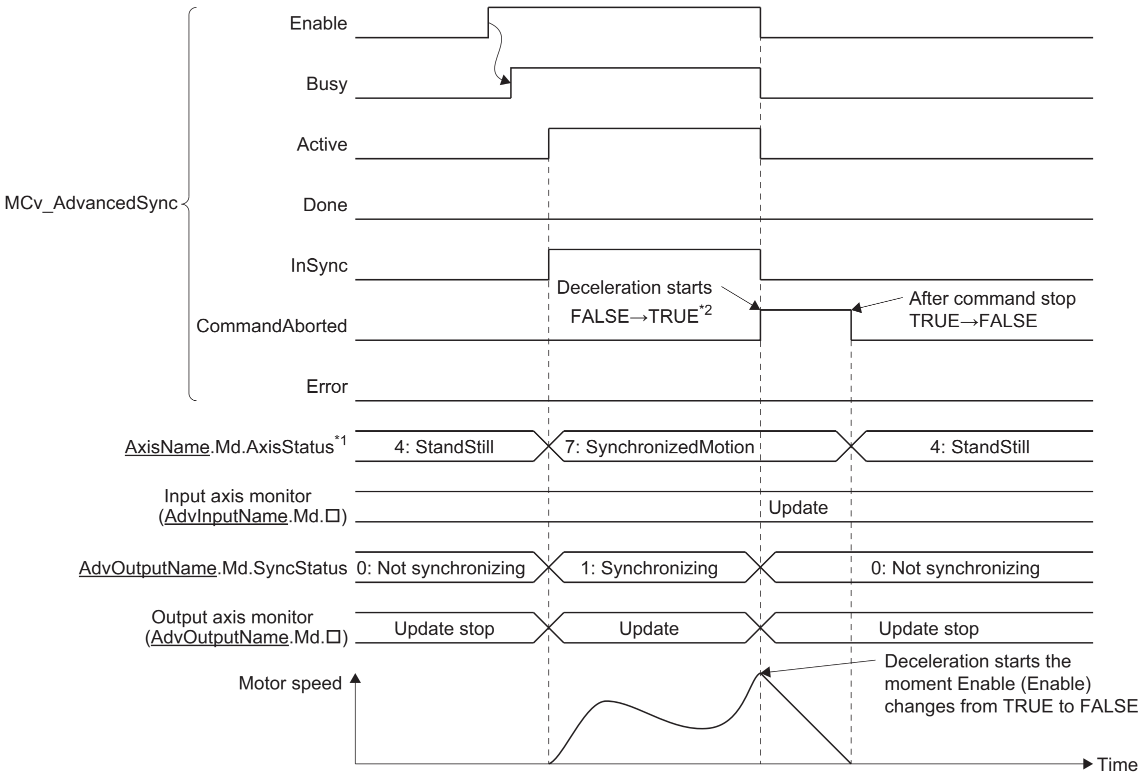
After the stop cause occurs, the Input axis monitor data (AdvInputName.Md.□) update will not stop, but the Output axis monitor data (AdvOutputName.Md.□) will stop updating and will retain the values the moment when the stop cause occurred. The Output axis monitor data (AdvOutputName.Md.□) will update the next time MCv_AdvancedSync (Advanced Synchronous Control) is executed.

Time charts showing what happens when a stop cause occurs are shown below.

Point

To stop using the deceleration set in Deceleration (Deceleration) under MCv_AdvancedSync (Advanced Synchronous Control), set the Stop Selection at Stop Cause Occurrence (AxisName.Pr.StopMode_General) to "2: Keep Current Acceleration/Deceleration (KeepCurrentAcc)".

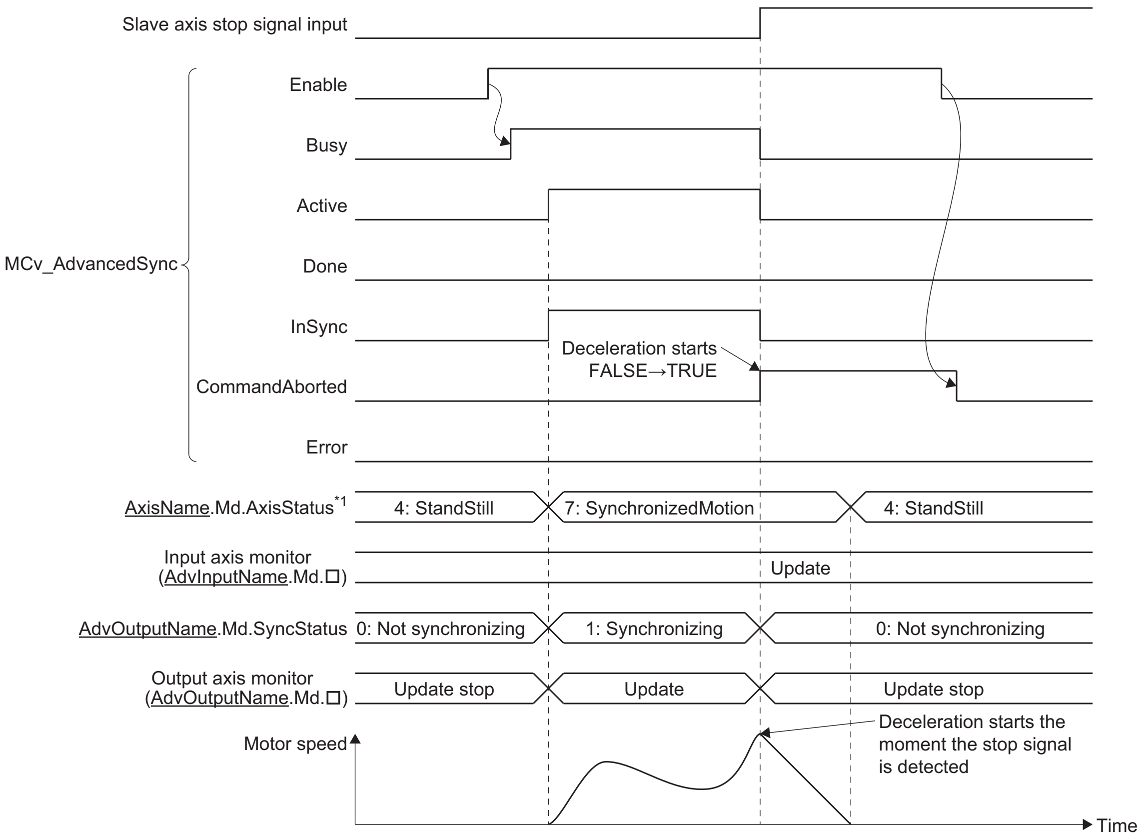
■ When MCv_AdvancedSync (Advanced Synchronous Control) Enable (Enable) is set to FALSE during advanced synchronous control



*1 Slave axis (Slave) axis status.

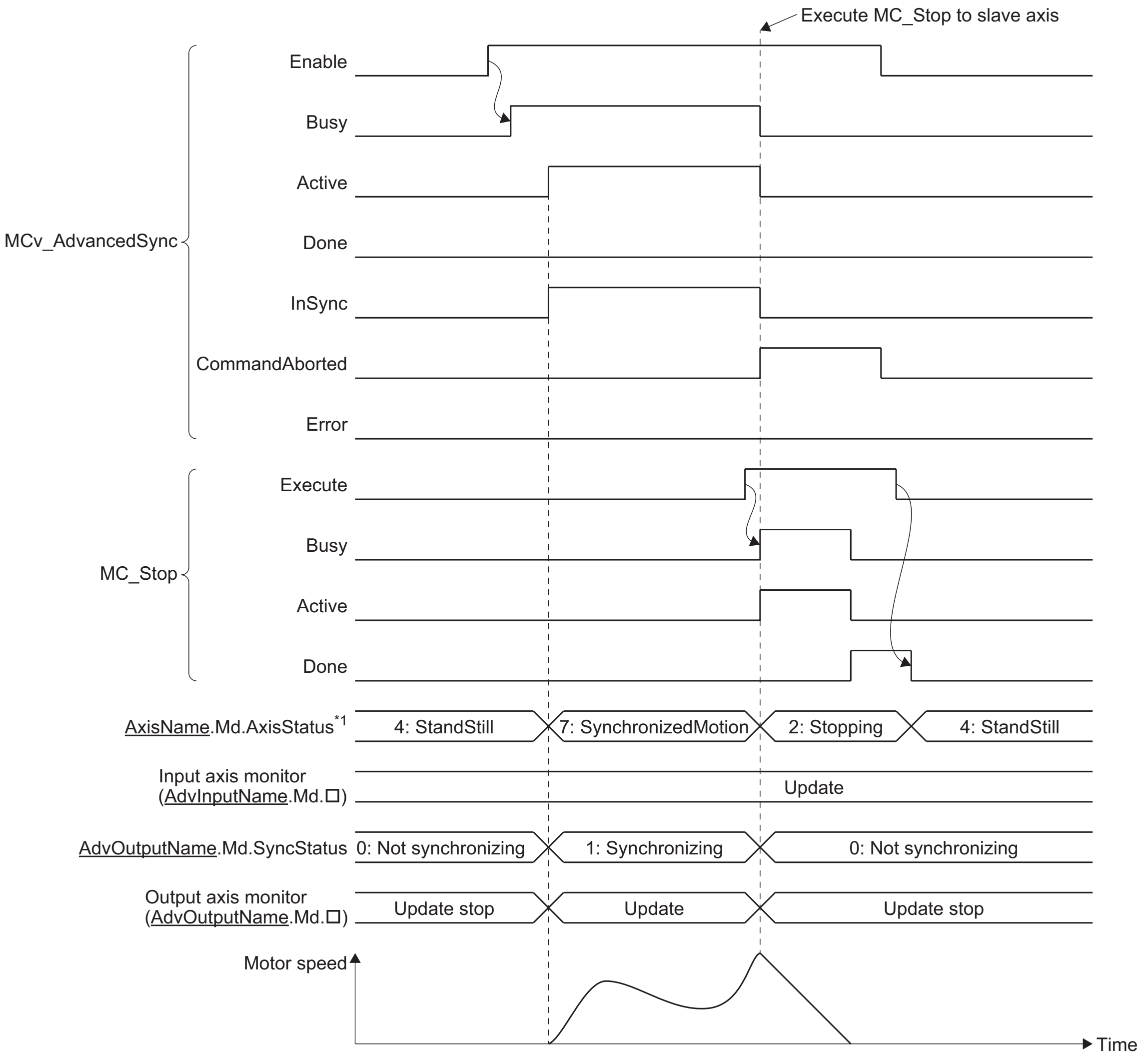
*2 If Immediate stop is selected, only one scan will be TRUE.

■ When a slave axis stop signal is detected during advanced synchronous control



*1 Slave axis (Slave) axis status.

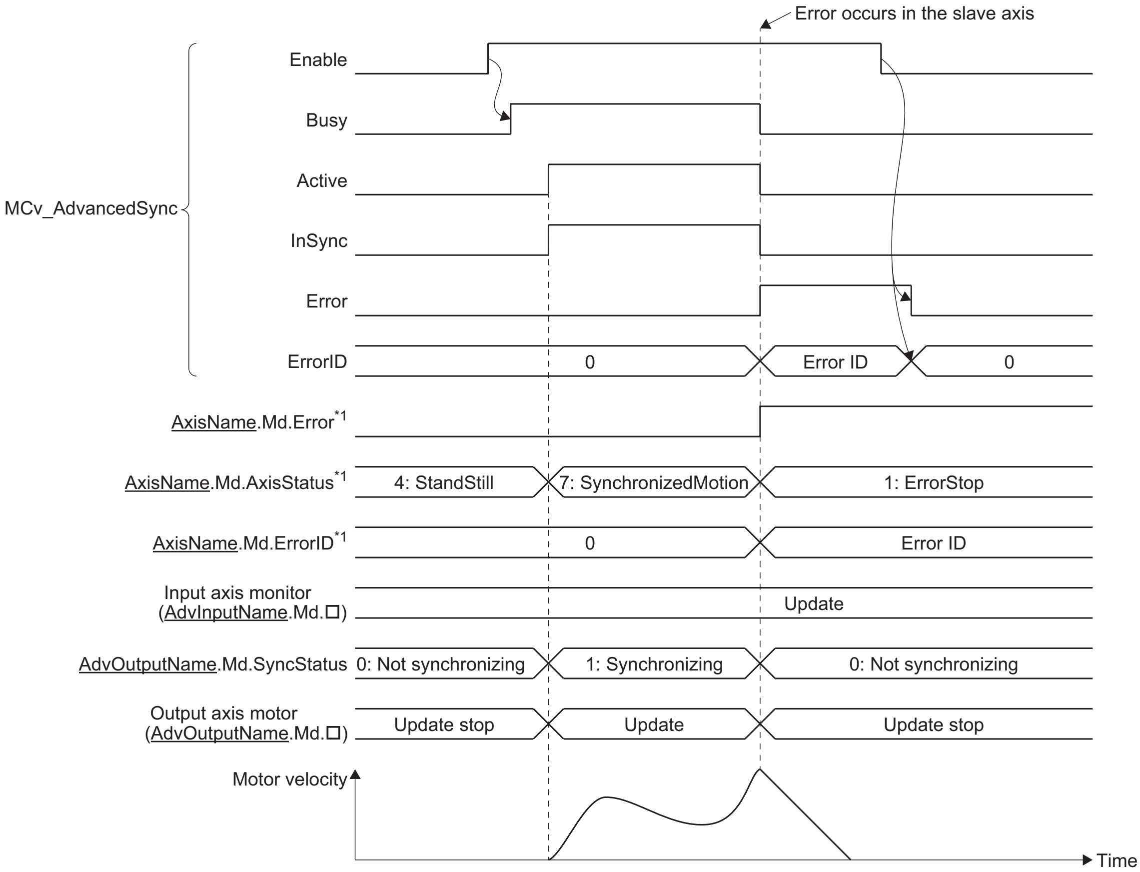
■ When stop cause (MC_Stop (Forced Stop) execution/stop signal detection) occurs during advanced synchronous control



*1 Slave axis (Slave) axis status.

15

■ When a stop cause (Slave axis error, Forced stop) occurs during advanced synchronous control



*1 Slave axis (Slave) axis status.

Precaution

- During execution of MCv_AdvancedSync (Advanced Synchronous Control), if a stop case occurs in the axes set in the input axis information of the Master axis (Master), Auxiliary shaft object (AdvOutputName.Pr.AuxAxis), and Sub input axis object (AdvOutputName.Pr.SubAxis), the Slave axis (Slave) will continue control using commands transmitted from the input axis. The Slave axis (Slave) axis status will not change. Also, if a stop cause occurs in the axis is set in the output Axis axis information (AdvOutputName.Axis) it will not affect the axis is set in the input axis Axis information (AdvInputName.Axis).
- When linking multiple axes with MCv_AdvancedSync (Advanced Synchronous Control) using a virtual linked axis, do not set the Master axis (Master) to a Slave axis (Slave) of a function block afterwards. The operation when set to a function block afterwards cannot be guaranteed.
- When the Master axis (Master), Auxiliary shaft object (AdvOutputName.Pr.AuxAxis), and Sub input axis object (AdvOutputName.Pr.SubAxis) changes to servo OFF due to a servo alarm or a forced stop, stopping the operation of MCv_AdvancedSync (Advanced Synchronous Control) using MC_Stop (Forced stop) is recommended because the Slave axis (Slave) may cause an unexpected operation.
- If an input axis is operated during the analyzing of MCv_AdvancedSync (Advanced Synchronous Control) startup (until the Slave axis (Slave) axis status (AxisName.Md.AxisStatus) becomes "7: During synchronous operation (SynchronizedMotion)"), the change amount based on current value change and the movement amount of the input axis during analyzing is reflected immediately after synchronous control starts. As the output axis may suddenly accelerate depending on the change amount and movement amount of the input axis, start the operation of the input axis after confirming synchronous control is in operation.
- If the member variable setting value of ADV_INPUT or ADV_OUTPUT that imported setting values during synchronous control are changed to a value out of range while MCv_AdvancedSync (Advanced Synchronous Control) is being executed, "Out of Advanced Synchronous Control Variable Range Warning (error code: 0D3FH)" occurs, and control will continue using the setting values before the change. If the member variable setting value of ADV_INPUT or ADV_OUTPUT is out of range when MCv_AdvancedSync is started, "Out of Parameter Range (Advanced Synchronous Control Input Axis) (error code: 1D85H)" or "Out of Parameter Range (Advanced Synchronous Control Output Axis) (error code: 1D86H)" occurs, and advanced synchronous control will not start.

Advanced synchronous control precautions

This section describes precautions when using advanced synchronous control.

When using public labels

When setting the parameters of input axes and output axes to public labels, set to the public label only the parameters that need to be changed using the CPU module side program.

Parameters set to the public labels will initially be "0" regardless of the values set using the engineering tool. Set these values using the CPU module side program before starting the advanced synchronous control FB.

Decimal points

- The decimal point range guaranteed for the input axis and output axis together is up to 5 decimal places. When setting the units, set the appropriate unit for the system to the Position command unit (AxisName.Pr.Unit_Position).

Ex.

Control using degree units

To accurately control values less than the decimal "0.00001" using control, set the position command unit to "$\times 10^{-3}$ degree" or "$\times 10^{-6}$ degree" instead of "degree". As a result, the value of "0.00001" can be handled as "10.0" in terms of control, meaning it can be controlled accurately.

- If the position command unit of the axis specified as an output axis has been set to "$\times 10^{-6}$ degree", in addition to operating the slave axis by synchronous control restoration, it is possible to go over the velocity limit value with the axis set as the output axis. If going over the velocity limit value occurs, move the set current position of the axis set as the output axis in advance to a position close to the restoration position, or change the position command unit to a large unit such as "$\times 10^{-3}$ degree".
- As the guaranteed range for decimals is up to 5 decimal places, the numbers of the current position of the axis set as the input axis are truncated after the 5th decimal place. In addition, if a value with more than 5 decimal places is set for a parameter or control data variable with a double-precision real number data type, the values after the 5th decimal place are ignored.
- If double-precision real numbers are monitored with the engineering tool, any values after the 5th decimal place may differ from the actual values.

Ex.

If the current position per cycle is in "180.00000" position

The monitor may display this as "179.999999...".

Smoothing

■ Import timing

The smoothing time constant becomes enabled when the system is started. To reflect any changed values, a system restart is required.

■ Smoothing time constant setting value and operation cycle setting

When the smoothing time constant is set to maximum value (5000 [ms]), in addition to the operation cycle being less than or equal to 125 [μs], there will be restriction on the movement amount that can be processed per operation cycle. The relationships between the operation cycle and movement amount is shown in the following table.

Operation cycle [μs]	Movement amount per operation cycle when the smoothing time constant is 5000 [ms]
62.5	-536870912.0 to 536870911.0
125	-1073741824.0 to 1073741823.0
More than 250	-2147483648.0 to 2147483647.0

If a value larger than the movement amount corresponding to the operation cycles in the above table is input into one operation cycle, "Advanced Synchronization Control Operation Overflow (error code: 1AF1H)" occurs.

- The filter level can be calculated by the following formula. Any decimal values rounded to the nearest integer. However, if the result is rounded to "0", the filter level will operate using level 1.
  Filter level = (smoothing time [ms]) ÷ operation cycle [ms]

■ Smoothing function memory usage

- When using the smoothing function, the longer the total of the smoothing time constant used by advanced synchronized control, the more system memory the add-onMotionControl_AdvancedSync consumes. The overall smoothing time is calculated using the following formula.
  Total smoothing time = Total Output axis smoothing time constant (AdvOutputName.PrConst.SmoothingTimeConstant) + Total Main shaft clutch smoothing time constant (AdvOutputName.PrConst.MasterClutchSmoothingTimeConstant) + Total Auxiliary shaft clutch smoothing time constant (AdvOutputName.PrConst.AuxClutchSmoothingTimeConstant) + Total Main shaft speed change gear smoothing time constant (AdvOutputName.PrConst.MasterSpeedChangeGearSmoothingTimeConstant) + Total auxiliary shaft speed change gear smoothing time constant (AdvOutputName.PrConst.AuxSpeedChangeGearSmoothingTimeConstant) + Total Output axis speed change gear smoothing time constant (AdvOutputName.PrConst.OutSpeedChangeGearSmoothingTimeConstant)
- The smoothing function secures memory used by the smoothing process at power ON. Memory usage can be calculated using the following formula.
  Memory usage = Filter level × 8 bytes
- Operation when smoothing memory is insufficient.
  Depending on the smoothing time constant setting, "Advanced Synchronous Control Smoothing Memory Shortage (error code: 1C9CH)", or "Advance Synchronous Control Add-on Memory Shortage (error code: 1C9BH)" will occur at startup. If an error occurs, reduce the smoothing time constant used by advanced synchronous control, or set the add-on MotionControl_AdvancedSync parameter (MotionControl_AdvancedSync_System.PrConst.AddonMemory) to satisfy the required memory amount.

Combination with other functions

■ Touch probe

When latching the Input axis variables (AdvInputName.□) and Output axis variables (AdvOutputName.□) using MC_TouchProbe (Touch Probe), latch is possible by specifying the data type to [VAR]. However, the latch data stores a value delayed by one operation cycle.
In addition, latch data estimations do not take into account any values exceeding the permissible value.

■ Synchronous control function

The following Motion control FBs cannot be combined with advanced synchronous control.
To change the cam reference position, the current position per cycle, and current position per cycle after main shaft/auxiliary shaft gear during synchronous control, use the synchronous control change function.
- MCv_ChangeCycle (Current Position Change Per Cycle)

Related add-ons

Using this function requires the following add-ons.

Item	Add-on
Required	• baseSystem • Axis • MotionEngine • MotionControl_General • MotionControl_AdvancedSync
When using synchronous control initial position function	AbsSystem
When using operation profile	ProfileControl
When using a clutch with an external input signal	SignalIO

Point

To use this function, it is recommended to disable the following add-ons.
- SimpleMotion

System memory usage

■ RAM usage

- When there are no input settings, output settings, and Motion control FBs, 270k bytes of memory will be used. The memory usage for input settings and output settings and each Motion control FB is shown below. Change the add-on MotionControl_AdvancedSync parameter (MotionControl_AdvancedSync_System.PrConst.AddonMemory) where necessary.

Data type	RAM usage
Input axis	2.7k bytes
Output axis	17.7k bytes
MCv_AdvancedSync (Advanced Synchronous Control)	0.9k bytes

- For memory usage when using smoothing, refer to the following.
  ☞ Page 507 Smoothing function memory usage
- Memory usage when using an external input signal
  When using an external input signal with a variable such as Main shaft clutch ON signal setting (AdvOutputName.Pr.MasterOnClutchSignal), 1.8k bytes of add-on "SignalIO" memory are used for each setting. Change the Maximum RAM size (System.PrConst.Addon_SignalIO.RamSizeMax) in accordance with the number settings used for the external input signal.

■ When the add-on memory capacity is insufficient

When the memory used for add-ons is insufficient, "Insufficient Add-on System Memory (RAM) (error code: 3209H)" occurs. When an error occurs, set the add-on MotionControl_AdvancedSync parameter (MotionControl_AdvancedSync_System.PrConst.AddonMemory) to satisfy the required amount of memory. The usage for add-on memory and filter functions can be checked with the following variable.

- System Memory Usage (MotionControl_AdvancedSync_System.Md.AddonMemory)

■ Backup RAM usage

For backup RAM usage, refer to the following.

☞ Page 588 Backup data management for advanced synchronous control

Combining versions

The engineering tool version that supports this function is shown below.

Engineering tool	Version
Motion control setting function	"1.040S" or later

Buffer mode

For the buffer modes that can be specified with MCv_AdvancedSync (Advanced Synchronous Control), refer to the following.

☞ Page 146 Buffer mode that can be set

15.3 Input Axis Function

The motion control setting function of GX Works3 is used to configure the input settings for advanced synchronous control and generate and initialize the input axes necessary for advanced synchronous control. Any set axes will be assigned to the global label data as an input axis variable.

Input axes are defined as an axis variable consisting of monitor information such as parameter information and current positions.

As the input axis settings are enabled after power ON, the status of input axes can be monitored even when MCv_AdvancedSync (Advanced Synchronous Control) has not been started.

When a warning or error occurs in the input axis, the warning or error is stored in the axis set in the input axis Axis information (AdvInputName.Axis).

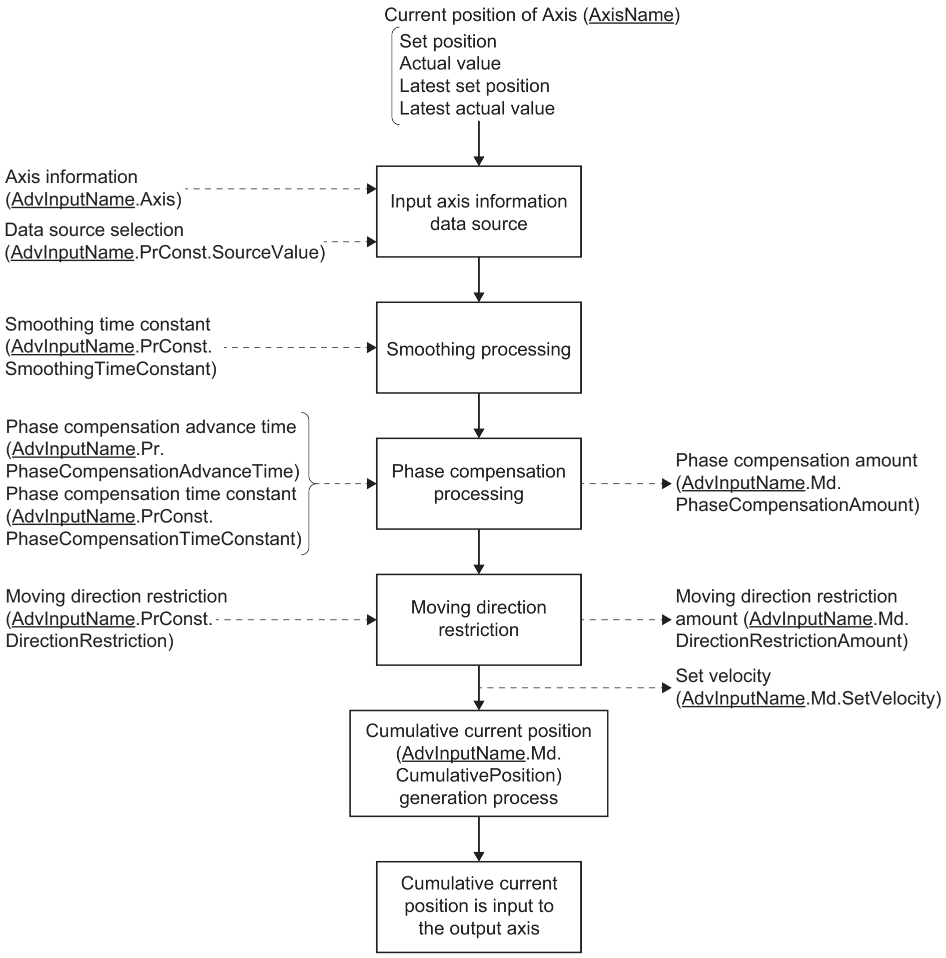
The input axis functions are shown below.

Function	Description	Application
Type setting	Sets the axis and data source used for the input axis.	Set the axis to transmit to the advanced synchronous control output axis and its input values.
Smoothing	Smoothing processing will be conducted for the input value of the axis set in type setting.	It is possible to suppress the velocity change by smoothing processing when "Actual value" or "Latest actual value" is set as the input value in the data source. However, the input response is delayed by the set time due to the smoothing process.
Phase compensation	Phase compensation processing will be conducted for the input values of the axis set in type setting.	When an real drive axis or real encoder axis is set to the input axis, a phase delay occurs between the input axis and the motor axis end of the output axis. The phase compensation function is used to prevent phase deviation. For phase compensation of the input axis delay time, set the delay time peculiar to the system to the input axis Phase compensation advance time (AdvInputName.Pr.PhaseCompAdvanceTime)*1.
Moving direction restriction	Moving direction restriction filter will be conducted for the input values of the axis set in type setting.	When "Actual value" or "Latest actual value" is set as the input value in the data source, it is possible to prevent reverse operation due to mechanical vibration, etc.
Monitor	Monitors the input axis. Input axis monitor values are stored in the following monitor data. • Active status (AdvInputName.Md.ActiveStatus) • Cumulative current position (AdvInputName.Md.CumulativePosition) • Set velocity (AdvInputName.Md.SetVelocity) • Phase compensation amount (AdvInputName.Md.PhaseCompensationAmount) • Moving direction restriction amount (AdvInputName.Md.DirectionRestrictionAmount)	When the status of the input axis is monitored, this is described on the program.

*1 For the delay time peculiar to the system, refer to the following.
☞ Page 612 Phase compensation function

15

After the input axis is generated, specify the input axis Axis information (AdvInputName.Axis) to the Master axis (Master) of the MCv_AdvancedSync (Advanced Synchronous Control). If using an auxiliary shaft and a sub input axis, specify the input axis Axis information (AdvInputName.Axis) to the Auxiliary shaft object (AdvOutputName.Pr.AuxAxis) and the Sub input axis object (AdvOutputName.Pr.SubAxis).
The relationship between the axis current position set in the input axis Axis information (AdvInputName.Axis) and each input axis function is shown below.



Input axis initialization timing

The input axis variables are initialized at the following timing.

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

Input axis position command unit

The input axis position command unit is the Position command unit (AxisName.Pr.Unit_Position) and the Position command unit string (AxisName.Pr.Unit_PositionString) of the axis set in the input axis Axis information (AdvInputName.Axis).
In addition, the position command unit of the input axis and output axis during synchronized control will not affect control.
The result of the calculation performed for the position information of the input axis which has no unit information is used as the command of the output axis.
Refer to the explanation of each monitor data for details on input axis monitor display units.

Input axis positioning range

Regardless of the Ring counter upper limit value (AxisName.PrConst.RingCount_UpperValue) or Ring counter lower limit value (AxisName.PrConst.RingCount_LowerValue) of the axis positioning range set in the input axis Axis information (AdvInputName.Axis), the input axis Cumulative current position (AdvInputName.Md.CumulativePosition) will be within the following range.

Positioning Range
-10000000000.0 ≤ Positioning range < 10000000000.0

Input axis relevant variables

15

Input axis (AdvInputName.)

Variable name	Name	Import*1	Data type*1	Setting value	Initial value
LabelID	Axis label ID	At system start	WORD(UINT)	1 to 256	0
Axis	Axis information		AXIS_REF	—	—

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Axis label ID(AdvIntputName.LabelID)

Sets the label ID of the input axis.
The setting range is from 1 to 256.

■ Axis information (AdvIntputName.Axis)

- Sets the Axis information (AxisName.AxisRef) of the axis that will be the input axis for advanced synchronous control.
  Up to 256 input axes can be generated.
  It is not possible to set the same axis to the axis information of different input axes. If it is set, "Advanced Synchronous Control Input Axis Incorrect (error code: 1C9DH)" will occur.
- Do not set the Axis information (AxisName.AxisRef) of an axis defined in local labels. If it is set, input axes cannot be generated.

Input axis parameter constant (AdvInputName.PrConst.)

Variable name	Name	Import*1	Data type*1	Setting value	Initial value
SourceValue	Data source selection	At system start	INT (MC_SOURCE)	1: Set Value (mcSetValue) 2: Actual Value (mcActualValue) 101: Latest Set Value (mcLatestSetValue) 102: Latest Actual Value (mcLatestActualValue)	1: Set Value (mcSetValue)
SmoothingTimeConstant	Smoothing time constant		WORD(UINT)	0 to 5000 [ms]	0 [ms]
PhaseCompensationTimeConstant	Phase compensation time constant		WORD(UINT)	0 to 65535 [ms]	10 [ms]
DirectionRestriction	Moving direction restriction		INT (MC_INPUT_DIRECTION)	0: Without moving direction restriction (NoDirectionRestriction) 1: Enable only for current position increase direction (mcPositiveDirection) 2: Enable only for current position decrease direction (mcNegativeDirection)	0: Without movingdirection restriction (NoDirectionRestriction)

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Data source selection (AdvInputName.PrConst.SourceValue)

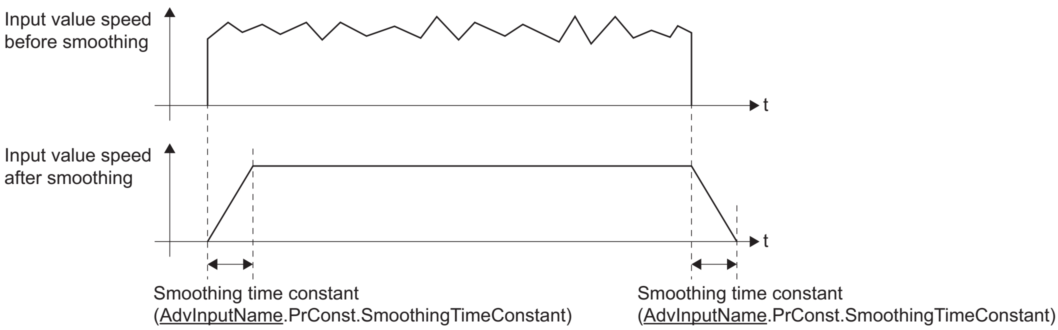
Set the input axis data source to be referenced in advanced synchronous control.

Setting value	Description
1: Set Value (mcSetValue)	Uses the set position of the axis specified to the input axis in the previous operation cycle.
2: Actual Value (mcActualValue)	Uses the actual position of the axis specified to the input axis in the previous operation cycle.
101: Latest Set Value (mcLatestSetValue)	Uses the set position of the axis specified to the input axis in the curent operation cycle.
102: Latest Actual Value (mcLatestActualValue)	Uses the actual position of the axis specified to the input axis in the current operation cycle.

- If using "101: Latest set value (mcLatestSetValue)" and "102: Latest actual value (mcLatestActualValue)", the commands of multiple Motion control FBs can be transmitted in the same operation cycle. In this case, make the first calling order and the linking order of linked Motion control FBs the same.
- When Data source selection (AdvInputName.PrConst.SourceValue) is set to "1: Set value (mcSetValue)" or "101: Latest set value (mcLatestSetValue)" and the master axis is disconnected or changes to servo OFF due to a servo alarm or forced stop, the change in values may become large. This can be prevented by setting Data source selection (AdvInputName.PrConst.SourceValue) to "101: Latest set value (mcLatestSetValue)" or "102: Latest actual value (mcLatestActualValue)".
- When actual value is specified for an axis (virtual axis, virtual linked axis, virtual encoder axis) that does not have an Actual position (AxisName.Md.ActualPosition), the axis operates by the value that is the same as Set Position (AxisName.Md.SetPosition).
- For real encoder axes, since the set position is calculated from the actual position, operation will be the same regardless of the set value.

■ Smoothing time constant (AdvInputName.PrConst.SmoothingTimeConstant)

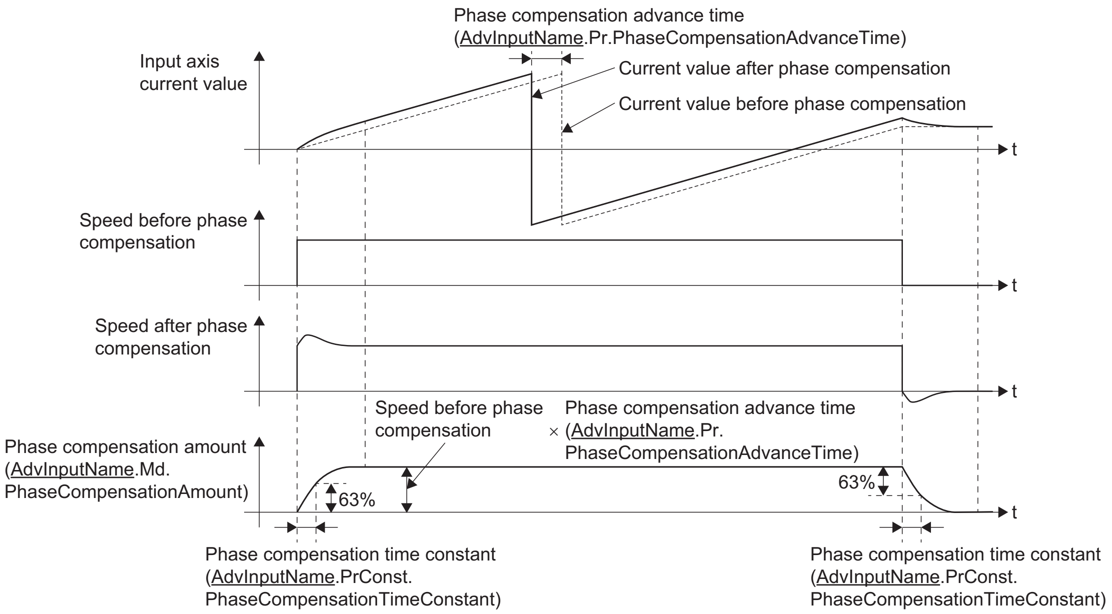
Set the averaged time for the smoothing process of the input movement amount from the input axis. The smoothing process can moderate velocity change when "Actual value" or "Latest actual value" is set to the input value. However, the input response is delayed by the set time due to the smoothing process.



■ Phase compensation time constant (AdvInputName.PrConst.PhaseCompensationTimeConstant)

Set the time constant for reflecting the phase compensation amount for the first order delay.

63% of the phase compensation amount is reflected in the specified time constant.



15

■ Moving direction restriction (AdvInputName.PrConst.DirectionRestriction)

Set this parameter to restrict the input movement amount from the input axis to one direction.

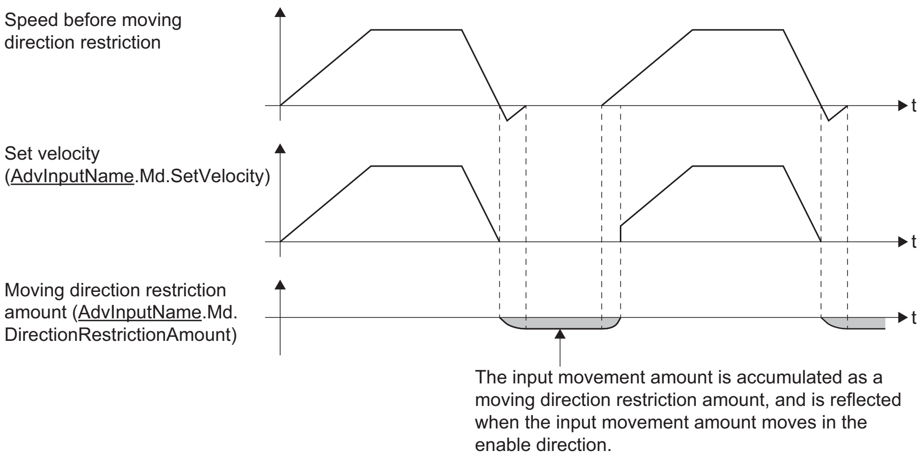
Setting value	Description
0: Without movement direction restriction (NoDirectionRestriction)	Moving direction restriction is not executed.
1: Enable only for current position increase direction (mcPositiveDirection)	Enable only the input travel value in a positive direction of the input axis cumulative current position.
2: Enable only for current position decrease direction (mcNegativeDirection)	Enable only the input travel value in a negative direction of the input axis cumulative current position.

The input movement amounts of the enable direction and opposite direction are accumulated as a moving direction restriction amount, and is reflected when the input movement amount moves in the enable direction. As such, even when reverse operations are repeated, there will be no deviation in the input axis cumulative current position.

When the following operations are conducted for the input axis, the movement amount will be cleared to 0.

- When a servo amplifier is connected
- When homing is executed
- When the current position is changed

When "1: Enable only for current position increase direction (mcPositiveDirection)" is set to Moving direction restriction amount (AdvInputName.Md.DirectionRestrictionAmount).



Input axis parameter (AdvInputName.Pr.)

Variable name	Name	Import*1	Data type*1	Setting value	Initial value
PhaseCompensationAdvanceTime	Phase compensation advance time	Operation cycle	DINT	-100000000 to 100000000 [μs]	0

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Phase compensation advance time (AdvInputName.Pr.PhaseCompAdvanceTime)

Sets the time to advance or delay the phase (input response) of the input axis.

Setting value	Description
1 to 100000000 [μs]	Advance the phase (input response) according to the setting time.
0 [μs]	Do not execute phase compensation.
-100000000 to -1 [μs]	Delay the phase (input response) according to the setting time.

If the setting time is too long, the system experiences overshoot or undershoot at acceleration/deceleration of the input speed. In this case, set longer time to reflect the phase compensation amount in Phase compensation in time constant (AdvInputName.PrConst.PhaseCompensationTimeConstant).

Input axis monitor data (AdvInputName.Md.)

Variable name	Name	Import*1	Data type*1	Stored value
ActiveStatus	Active status	Immediate	INT	• -1: Disabled • 0: Not connected • 1: Input axis enabled
CumulativePosition	Cumulative current position	Operation cycle	LREAL	-10000000000.0 ≤ Input axis cumulative current position < 10000000000.0 [Position command unit of the axis set as the input axis]
SetVelocity	Set velocity		LREAL	-2500000000.0 to 2500000000.0 [Position command unit of the axis set as the input axis]
PhaseCompensationAmount	Phase compensation amount		LREAL	-2147483648.0 to 2147483647.0 [Position command unit of the axis set as the input axis]
DirectionRestrictionAmount	Direction restriction amount		LREAL	-2147483648.0 to 2147483647.0 [Position command unit of the axis set as the input axis]

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Active status (AdvInputName.Md.ActiveStatus)

Stores the status of the input axis.

Setting value	Description
-1: Disabled	Input axis generation has failed due to being parameters being outside the possible range.
0: Not connected	The initial processing of the axis set as an input axis has not been completed, such as when a servo amplifier is not connected.
1: Input axis enabled	The input axis can be used for synchronous control.

■ Cumulative current position (AdvInputName.Md.CumulativePosition)

The cumulative current position of the input axis is stored in the position units of the axis set in the input axis Axis information (AdvInputName.Axis) as shown below.

The current position of the input axis is the value after smoothing, phase compensation, and movement direction restriction processes.

Data source selection (AdvInputName.PrConst.SourceValue) setting value	Description
1: Set value (mcSetValue)	• The set position of the axis set in the input axis Axis information (AdvInputName.Axis) in the previous operation cycle is stored. • When the set current position or the actual current position is changed by homing or current position change, it will be changed to the current position after change.
2: Actual value (mcActualValue)	• The value calculated using the position address returned by the device station of the axis set in the input axis Axis information (AdvInputName.Axis) in the previous operation cycle is stored.
101: Latest set value (mcLatestSetValue)	• The set position of the axis set in the input axis Axis information (AdvInputName.Axis) in this operation cycle is stored. • When the set current position or the actual current position is changed by homing or current position change, it will be changed to the current position after change.
102: Latest sctual value (mcLatestActualValue)	• The value calculated using the position address returned by the device station of the axis set in the input axis Axis information (AdvInputName.Axis) in this operation cycle is stored.

■ Set velocity (AdvInputName.Md.SetVelocity)

The speed of the input axis is stored in speed units of the axis set in the input axis Axis information (AdvInputName.Axis).

The speed of the input axis is the value after smoothing, phase compensation, and moving direction restriction processes.

Point

Depending on the movement amount per input axis operation cycle, the range of "-2500000000.0 to 2500000000.0" may be exceeded.

15

■ Phase compensation amount (AdvInputName.Md.PhaseCompensationAmount)

The phase compensation amount of the input axis is stored in position command units of the axis set in the input axis Axis information (AdvInputName.Axis).

The phase compensation amount of the input axis is the value after smoothing and phase compensation processes.

■ Moving direction restriction amount (AdvInputName.Md.DirectionRestrictionAmount)

During moving direction restriction of the input axis, the cumulative value of the enabled direction and opposite input movement amount is stored as shown below in position units of the axis specified as the input axis.

Moving direction restriction (AdvInputName.PrConst.DirectionRestriction) setting value	Description
1: Enable only for current position increase direction	A negative accumulation is stored during moving direction restriction. "0.0" is stored if there is no restriction.
2: Enable only for current position decrease direction	Stores positive cumulative values during moving direction restriction. "0.0" is stored if there is no restriction.

Since the moving direction restriction is processed after the phase compensation process, if an undershoot occurs due to phase compensation during a deceleration stop, the moving direction restriction amount may remain.

Precautions

The following are precautions for using an input axis.

Advanced synchronous control operation overflow warning

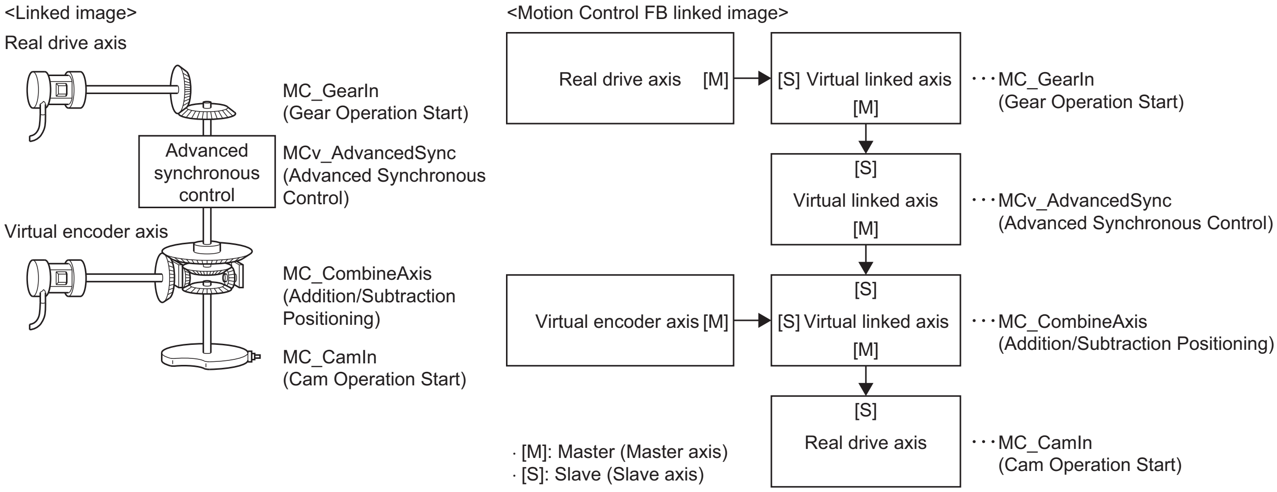
When the driver conversion numerator/driver conversion denominator setting value of the axis set in the input axis Axis information (AdvInputName.Axis) or the input axis speed is significantly large, an overflow in the internal operation of the input axis will occur, and "Advanced Synchronous Control Operation Overflow Warning (error code: 0D3EH)" error may occur.
When a warning occurs, the synchronized position of the input axis and output axis may shift out of position. In this case, fix the setting values of the driver conversion numerator/driver conversion denominator or reduce the speed of the input axis.

Advanced synchronous control linking order

The Latest Set Value (mcLatestSetValue) and Latest actual value (mcLatestActualValue) of the Data Source Selection (AdvInputName.PrConst.SourceValue) use 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 the Latest set value (mcLatestSetValue) and Latest Actual Value (mcLatestActualValue) can transmit the multiple Motion control FB command in the same operation cycle. In this case, set the first call order and linking order of the linked Motion control FB to be the same.

Ex.

When transmitting movement of the real drive axis, the virtual encoder axis and advanced synchronous control to MC_CamIn (Cam Operation Start) control in the same operation cycle



When using the above combination of MCv_AdvancedSync (Advanced Synchronous Control) and single synchronous control (linked order), set the first call order as follows.

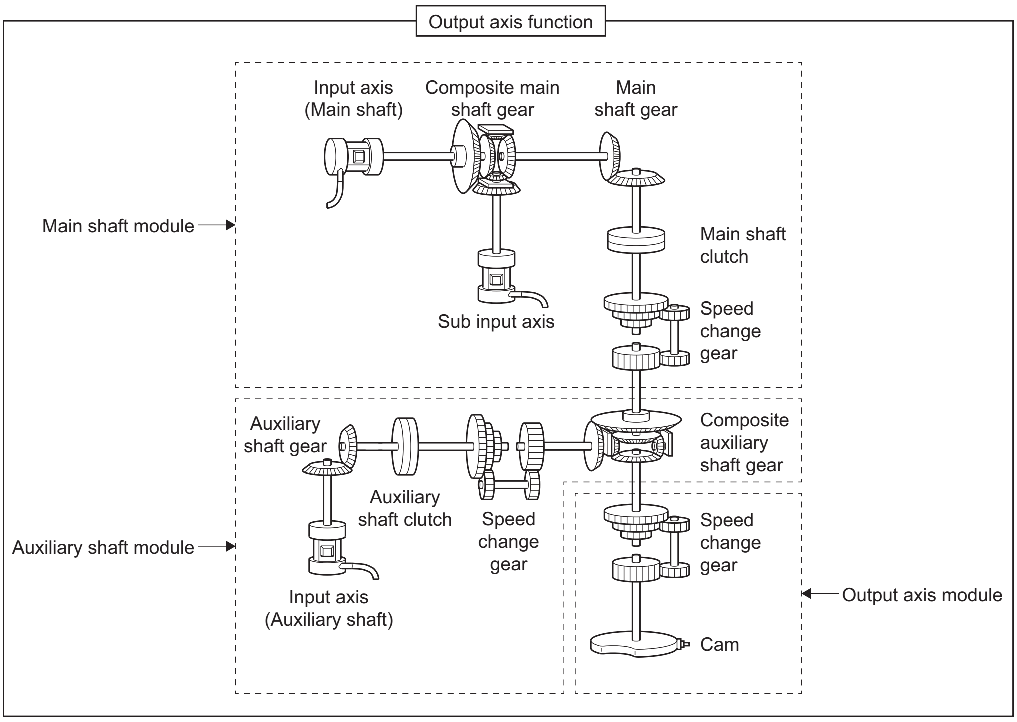
- First call order

1. Motion control FB which moves the real drive axis set as a Master axis (Master) of MC_GearIn (Gear Operation Start)
2. MC_GearIn (Gear Operation Start)
3. MCv_AdvancedSync (Advanced Synchronous Control)
4. MC_CombineAxes (Addition/Subtraction Positioning)
5. MC_CamIn (Cam Operation Start)

- Master axis data source selection (MasterValueSource) of each single synchronous control FB and advanced synchronous control data source selection (AdvInputName.PConst.PosSource) setting:
  101: Latest Set Value (mcLatestSetValue)

15.4 Output Axis Function

The GX Works3 motion control setting function is used to configure output settings and for generating and initializing the output axes necessary for advanced synchronous control.
Output axes are defined as the output axis variables consisting of parameter information, control data, and monitor information such as current positions.
Any generated output axes are specified in the output axis Axis information (AdvOutPutName.Axis) of the Slave Axis (Slave) of MCv_AdvancedSync (Advanced Synchronous Control).
When an error or warning occurs in the output axis, the warning or error is stored in the axis set in the output axis Axis information (AdvOutPutName.Axis).
This section describes the parameters, control data, and monitor data of the "Main shaft module", "Auxiliary shaft module", and "Output axis module"of advanced synchronous control.
The "Main shaft module", "Auxiliary shaft module" and "Output axis module" are defined as shown in the following figure.



Point

• The arrangement and number of modules in advanced synchronous control are fixed.

Output axis initialization timing

The output axis variables are initialized at the following timings.

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

Output axis positioning range

- Set the Ring counter lower limit value (AxisName.PrConst.RingCount_LowerValue) and Ring counter upper limit value (AxisName.PrConst.RingCount_UpperValue) to the values below. If any other values are set, an "Out of Advanced Synchronous Control Output Axis Ring Counter Range (error code: 1AF4H)" error will occur at system start.

Setting range	
Ring counter lower limit value (AxisName.PrConst.RingCount_LowerValue	**Ring counter upper limit value (AxisName.PrConst.RingCount_UpperValue)**
0.0	0.0 < Setting value ≤ 2147483647.0
-10000000000.0	10000000000.0

- Do not set the Ring counter lower limit value (AxisName.PrConst.RingCount_LowerValue) and Ring counter upper limit value (AxisName.PrConst.RingCount_UpperValue) of the axis positioning range of the axis set in the output axis Axis information (AdvOutputName.Axis) to a decimal number. If set, an error will occur with the value output to the slave axis and the Set position (AxisName.Md.SetPosition) of the axis set to the output axis in synchronous control when the maximum permissible value is exceeded.

Output axis relevant variables

Output axis (AdvOutputName.)

Variable name	Name	Import*1	Data type*1	Setting axis	Initial value
LabelID	Axis label ID	At system start	WORD(UINT)	1 to 256	0
Axis	Axis information		AXIS_REF	—	—

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Axis label ID (AdvOutputName.LabelID)

Sets the label ID of the output axis.

The setting range is from 1 to 256.

■ Axis information (AdvOutputName.Axis)

- Sets the Axis information (AxisName.AxisRef) of the axis that will be the output axis for advanced synchronous control. Up to 256 output axes can be generated.
  It is not possible to set the same axes to the axis information of different output axes.
- Do not set the Axis information of an axis (AxisName.AxisRef) defined in local labels. If it is set, output axes cannot be generated.
- Settable axis types to the output axis are shown below.

○: Settable, ×: Not settable

Axis type		Output axis setting possible
Real axis	Real drive axis	○
	Real encoder axis	×*1
Virtual axis	Virtual drive axis	○
	Virtual encoder axis	×*1
	Virtual linked axis	○

*1 If set, "Advanced Synchronous Control Output Axis Incorrect (error code: 1C9EH)" will occur.

Output axis parameters constant (AdvOutPutName.PrConst.)

Variable name	Name	Import*1	Data type*1	Setting value	Initial value
SmoothingTimeConstant	Output axis smoothin time constant	At system start	WORD(UINT)	0 to 5000 [ms]	0 [ms]

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Output axis smoothing time constant (AdvOutPutName.PrConst.SmoothingTimeConstant)

Set the averaging time to execute a smoothing process for the movement amount to the output axis after cam conversion.
The smoothing process can moderate rapid velocity changes on the cam, etc. of cam data (linear interpolation).
The output response is delayed depending on the time corresponding to the setting by smoothing process setting.

Output axis parameter (AdvOutPutName.Pr.)

Variable name	Name	Import*1	Data type*1	Setting value	Initial value
SubAxis	Sub input axis object	At start	AXIS_REF	—	—
AuxAxis	Auxiliary axis object		AXIS_REF	—	—

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Sub input axis object (AdvOutPutName.Pr.SubAxis)

Set the sub input axis object axis.
This variable will be disabled when set to "0".
When an axis that has not been assigned input settings is set, "Advanced Synchronization Control Axis Unset (error code: 1AF0H)" will occur.

■ Auxiliary shaft object (AdvOutPutName.Pr.AuxAxis)

Set the auxiliary axis object axis.
This variable will be disabled when set to "0".
When an axis that has not been assigned input settings is set, "Advanced Synchronization Control Axis Unset (error code: 1AF0H)" will occur.

Output axis monitor (AdvOutputName.Md.)

Structure/Variable name	Name	Import[*1]	Data type[*1]	Stored value
Synchronous status	SyncStatus	At start	INT	-1: Disabled 0: Not synchronizing 1: In synchronization
Master shaft	MasterAxisNo		WORD(UINT)	0 to 10000
Sub input axis	SubAxisNo		WORD(UINT)	0 to 10000
Auxiliary axis	AuxAxisNo		WORD(UINT)	0 to 10000

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Synchronization status (AdvOutputName.Md.SyncStatus)

The synchronization status of the output axis is stored.

Setting value	Description
-1: Disabled	The output axis cannot be used in this status due to being in initial processing or because of out of range parameters.
0: Not synchronizing	The output axis is not synchronized with the input axis by MCv_AdvancedSync (Advanced Synchronous Control).
1: In synchronization	The output axis is synchronized with the input axis by MCv_AdvancedSync (Advanced Synchronous Control).

■ Main shaft (AdvOutputName.Md.MasterAxisNo)

The axis number of the main shaft set in the input axis Axis information (AdvInputName.Axis) in synchronization with the output axis is stored.

"0" is stored when a main shaft is not connected.

■ Sub input axis (AdvOutputName.Md.SubAxisNo)

The axis number of the sub input axis set in the input axis Axis information (AdvInputName.Axis) in synchronization with the output axis is stored.

"0" is stored when a sub input axis is not connected.

■ Auxiliary shaft (AdvOutputName.Md.AuxAxisNo)

The axis number of the auxiliary shaft set in the input axis Axis information (AdvInputName.Axis) in synchronization with the output axis is stored.

"0" is stored when an auxiliary shaft is not connected.

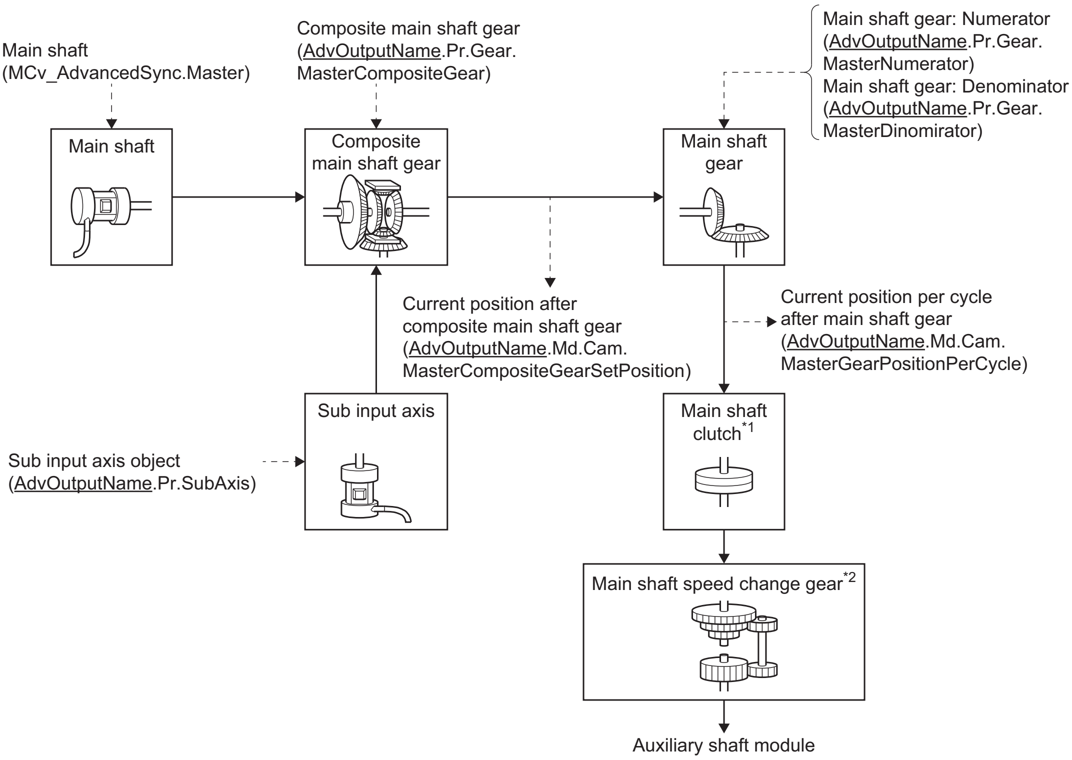
Main shaft module

The main shaft control modules transmit the movement amount inputted from an input axis to the output axis.
The functions for a main shaft module include gears, clutches, and speed change gears.
The main shaft module corresponds to the axis set as the Master axis (Master) of MCv_AdvancedSync (Advanced Synchronous Control).
Refer to the following for the setting items for gears, clutches, and speed change gears of the main shaft module.

Function	Member	Variable	Reference
Gear	Output axis parameter (AdvOutPutName.Pr.)	Gear parameter (Gear.)	☞ Page 523 Gear parameter of output axis parameters (AdvOutPutName.Pr.Gear.)
Clutch	Output axis parameter constant (AdvOutPutName.PrConst.)	—	☞ Page 524 Output axis parameter constant (AdvOutPutName.PrConst.)
	Output axis parameter (AdvOutPutName.Pr.)	—	☞ Page 524 Output axis parameters (AdvOutPutName.Pr.)
		Clutch parameter (Clutch.)	☞ Page 526 Clutch parameter of output axis parameters (AdvOutputName.Pr.Clutch.)
	Output axis monitor data (AdvOutPutName.Md.)	Clutch monitor (Clutch.)	☞ Page 530 Clutch monitor of output axis monitor data (AdvOutputName.Md.Clutch.)
	Output axis control data (AdvOutPutName.Cd.)	Clutch control data (Clutch.)	☞ Page 531 Clutch control data of output axis control data (AdvOutputName.Cd.Clutch.)
Speed change gear	Output axis parameter constant (AdvOutPutName.PrConst.)	—	☞ Page 560 Output axis parameters constant (AdvOutPutName.PrConst.)
	Speed change gear parameter of output axis parameters (AdvOutPutName.Pr.SpeedChangeGear.)	Speed change gear parameter (SpeedChangeGear.)	☞ Page 559 Speed change gear parameter of output axis parameters (AdvOutPutName.Pr.SpeedChangeGear.)
	Speed change gear control data of output axis parameters (AdvOutPutName.Cd.SpeedChangeGear.)	Speed change gear control data (SpeedChangeGear.)	☞ Page 560 Speed change gear control data of output axis parameters (AdvOutPutName.Cd.SpeedChangeGear.)

- Main shaft module

  The input value is generated as a composite value from two input axes (the main shaft and sub input axis) through the composite main shaft gear. The composite input value can be converted by the main shaft gear that provides the deceleration ratio, rotation direction, etc. for the machine system.



*1 Clutch (☞ Page 545 Clutch)
*2 Speed change gear (☞ Page 558 Speed change gear)

Gear parameter of output axis parameters (AdvOutPutName.Pr.Gear.)

Variable name	Name	Import*1	Data type*1	Setting value	Initial value
MasterCompositeGear	Composite main shaft gear	Operation cycle (FB is starting)	WORD(HEX)	H□□□□ → Main shaft input method 0: No input 1: Input + 2: Input - → Sub input method 0: No input 1: Input + 2: Input -	0001H
MasterNumerator	Main shaft gear: Numerator	At start	DINT	-2147483647 to 2147483647	1
MasterDenominator	Main shaft gear: Denominator		DWORD(UDINT)	1 to 2147483647	1

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Composite main shaft gear (AdvOutputName.Pr.Gear.MasterCompositeGear)

Set the composite method for the current positions from the main shaft and sub input axis.

Setting value	Description
0: No input	The input value from the input axis is calculated as 0.
1: Input +	The input value from the input axis is calculated as it is.
2: Input -	The input value from the input axis is calculated with its opposite sign.

Operation assumes "0: No input" if the value is set out of the range from 0 to 2.

Point

The composite method for the composite main shaft gear can be changed during synchronous control. It can also be used as a clutch to switch input values of the main shaft and auxiliary shafts.

■ Main shaft gear: Numerator (AdvOutputName.Pr.Gear.MasterNumerator)

Set the numerator for the main shaft gear input value conversion.

The input value is converted as follows.

$$\text{Input value after conversion} = \text{Input value before conversion} \times \frac{\text{Main shaft gear: Numerator}}{\text{Main shaft gear: Denominator}}$$

The input value direction can be reversed by setting a negative value in Main shaft gear: Numerator.

Point

Setting a reduced fraction is recommended for "Main shaft gear: Numerator/Main shaft gear: Denominator".

■ Main shaft gear: Denominator (AdvOutputName.Pr.Gear.MasterDenominator)

Set the denominator for the main shaft input value conversion.

Set together with Main shaft gear: Numerator.

Output axis parameter constant (AdvOutPutName.PrConst.)

Variable name	Name	Import*1	Data type*1	Setting value	Initial value
MasterClutchSmoothing TimeConstant	Main shaft clutch smoothing time constant	At system start	WORD(UINT)	0 to 5000 [ms]	0 [ms]

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Main shaft clutch smoothing time constant (AdvOutPutName.PrConst.MasterClutchSmoothingTimeConstant)

Set a time constant when the time constant method is set to "1: Time constant method (Exponent) (TimeConstantExponent)" or "2: Time constant method (Linear) (TimeConstantLinear)" in the Main shaft clutch smoothing method (AdvOutputName.Pr.Clutch.MasterSmoothingMethod).
The time constant setting applies for both clutch ON/OFF.

Output axis parameters (AdvOutPutName.Pr.)

Variable name	Name	Import*1	Data type*1	Setting value	Initial value
MasterOnClutchSignal	Main shaft clutch ON signal setting	At system start	SIGNAL_SELECT	—	—
MasterOffClutchSignal	Main shaft clutch OFF signal setting			—	—

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Main shaft clutch ON signal setting (AdvOutPutName.Pr.MasterOnClutchSignal)

Set the variable and device used as the clutch external signal in SIGNAL_SELECT structure when the Main shaft clutch ON control setting (AdvOutputName.Pr.Clutch.MasterOnControl) is set to "15: I/O data specification (ClutchSignal)".
For SIGNAL_SELECT structure, refer to the following.
Page 383 SIGNAL_SELECT structure
The following shows the settings peculiar to Main shaft clutch ON signal setting (AdvOutPutName.Pr.MasterOnClutchSignal).

Structure	Variable name	Type	Setting range
SIGNAL_SELECT (Signal Select)	Source (Signal)	TARGET_REF	■Type • BOOL ■Data type • [OBJ] • [VAR] • [DEV]
	Detection (Signal detection method)	INT (MC_SIGNAL_LOGIC)	• 2:Detection at FALSE → TRUE (risingedge) (RisingEdge) • 3:Detection at TRUE → FALSE (falling edge) (FallingEdge) • 4:Detection at rising edge/falling edge (BothEdges)
	CompensationTime (Compensation time)	LREAL	-5.0 to 5.0 [s]
	FilterTime (Filter time)	LREAL	0.0 [s]

Point

When setting [OBJ] for Target (Source.Target), start MCv_AdvancedSync (Advanced Synchronous Control) after the communication with device stations is established. If the communication is not established, "External Signal String Incorrect (error code: 1AB6H)" occurs.

■ Main shaft clutch OFF signal setting(AdvOutPutName.Pr.MasterOffClutchSignal)

Set the variable and device used as the clutch external signal in SIGNAL_SELECT structure when the Main shaft clutch OFF control setting (AdvOutputName.Pr.Clutch.MasterOffControl) is set to "15: I/O data specification (ClutchSignal)".

For SIGNAL_SELECT structure, refer to the following.

☞ Page 383 SIGNAL_SELECT structure

The following shows the settings peculiar to Main shaft clutch OFF signal setting (AdvOutPutName.Pr.MasterOnClutchSignal).

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

Point

When setting [OBJ] for Target (Source.Target), start MCv_AdvancedSync (Advanced Synchronous Control) after the communication with device stations is established. If the communication is not established, "External Signal String Incorrect (error code: 1AB6H)" occurs.

Clutch parameter of output axis parameters (AdvOutputName.Pr.Clutch.)

Variable name	Name	Import*1	Data type*1	Setting value	Initial value
MasterOnControl	Main shaft clutch ON control setting	Operation cycle (FB is starting)	INT (MC_CLUTCH_ METHOD)	0: Invalid (NoClutch)(Direct coupled operation) 1: Clutch command (ClutchCommand) 2: Clutch command leading edge (ClutchLeadingEdge) 3: Clutch command trailing edge (ClutchTrailingEdge) 4: Address mode (ClutchAddress) 15: I/O data specification (ClutchSignal)	0: Invalid (NoClutch)
MasterOffControl	Main shaft clutch OFF control setting		INT (MC_CLUTCH_ METHOD)	0: Invalid (NoClutch)(OFF control invalid) 1: Clutch command (ClutchCommand) (One-shot operation) 2: Clutch command leading edge (ClutchLeadingEdge) 3: Clutch command trailing edge (ClutchTrailingEdge) 4: Address mode (ClutchAddress) 15: I/O data specification (ClutchSignal)	0: Invalid (NoClutch)
MasterReference	Main shaft clutch reference address setting	At start	INT (MC_CLUTCH_ REFERENCE)	0: Current position before gear (GearFrontPosition) 1: Current position per cycle after gear (GearPositionPerCycle)	0: Current position before gear (GearFrontPosition)
MasterOnAddress	Main shaft clutch ON address	Operation cycle (FB is starting)	LREAL	-10000000000.0 to 10000000000.0	0.0
MasterMovementAmountBeforeOn	Movement amount before main shaft clutch ON	At completing clutch ON condition	LREAL	-2147483648.0 to 2147483647.0	0.0
MasterOffAddress	Main shaft clutch OFF address	Operation cycle (FB is starting)	LREAL	-10000000000.0 to 10000000000.0	0.0
MasterMovementAmountBeforeOff	Movement amount before main shaft clutch OFF	At completing clutch ON condition	LREAL	-2147483648.0 to 2147483647.0	0.0
MasterSmoothingMethod	Main shaft clutch smoothing method	At start	INT (MC_CLUTCH_ SMOOTHING_ METHOD)	0: Direct (ClutchSmoothingDisabled) 1: Time constant method (Exponent) (TimeConstantExponent) 2: Time constant method (Linear) (TimeConstantLinear) 3: Slippage method (Exponent) (SlippageExponent) 4: Slippage method (Linear) (SlippageLinear) 5: Slippage method (Linear: Input value follow up) (SlippageLinearFollow)	0: Direct (ClutchSmooth ingDisabled)
MasterOnSlippageAmount	Slippage amount at main shaft clutch ON	At turning clutch ON	LREAL	0.0 to 2147483647.0	0.0
MasterOffSlippageAmount	Slippage amount at main shaft clutch OFF	At turning clutch OFF	LREAL	0.0 to 2147483647.0	0.0

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Main shaft clutch ON control setting (AdvOutputName.Pr.Clutch.MasterOnControl)

Set the control method for main shaft clutch ON.

The Main shaft clutch ON control setting (AdvOutputName.Pr.Clutch.MasterOnControl) can be changed during synchronous control. However, changes from settings other than "0: Invalid (NoClutch)" to "0: Invalid (NoClutch)" are not possible.

Setting value	Description
0: Invalid (NoClutch) (Direct coupled operation)	Execute direct coupled operation without clutch control.
1: Clutch command (ClutchCommand)	The clutch is turned ON/OFF by the operations of the Main shaft clutch command (AdvOutputName.Cd.Clutch.MasterCommand) TRUE/FALSE. (Setting in the Main shaft clutch OFF control setting (AdvOutputName.Pr.Clutch.MasterOffControl) is not applicable in the clutch command ON/OFF mode.)
2: Clutch command leading edge (ClutchLeadingEdge)	The clutch is turned ON when the Main shaft clutch command (AdvOutputName.Cd.Clutch.MasterCommand) passes the leading edge (from FALSE to TRUE).
3: Clutch command trailing edge (ClutchTrailingEdge)	The clutch is turned ON when the Main shaft clutch command (AdvOutputName.Cd.Clutch.MasterCommand) passes the trailing edge (from TRUE to FALSE).
4: Address mode (ClutchAddress)	The clutch is turned ON when the reference address (the current position after composite main shaft gear or the current position per cycle after main shaft gear) reaches the Main shaft clutch ON address (AdvOutputName.Pr.Clutch.MasterOnAddress). The movement amount after passing through the ON address is calculated as the output movement amount of the clutch based on the reference address passing through, thereby controlling the clutch with an accurate movement amount.
15: I/O data specification (ClutchSignal)	The clutch is turned ON when the external input signal set in the Main shaft clutch ON signal setting (AdvOutputName.Pr.MasterOnClutchSignal) is detected.

Point

- Other clutch parameters are not applicable during direct coupled operation by setting the Main shaft clutch ON control setting (AdvOutputName.Pr.Clutch.MasterOnControl) to "0: Invalid (NoClutch)". The Main shaft clutch forced OFF command (AdvOutputName.Cd.Clutch.MasterForcedOff) and the change of the clutch control setting are ignored during direct coupled operation.
- Changes from settings other than "0: Invalid (NoClutch)" to "0: Invalid (NoClutch)" are not possible. If the setting is changed in that manner, "Out of Advanced Synchronous Control Variable Range Warning (error code: 0D3FH)" occurs.

■ Main shaft clutch OFF control setting (AdvOutputName.Pr.Clutch.MasterOffControl)

Set the control method for main shaft clutch OFF.

Setting value	Description
0: Invalid (NoClutch) (OFF control invalid)	Clutch OFF control is not used. This setting is applicable only for execution with clutch ON control.
1: Clutch command (ClutchCommand) (One-shot operation)	The clutch is turned OFF after moving the distance set in the Movement amount before main shaft clutch OFF (AdvOutputName.Pr.Clutch.MasterMovementAmountBeforeOff) (One-shot operation) after the clutch command turns ON. If the Movement amount before main shaft clutch OFF (AdvOutputName.Pr.Clutch.MasterMovementAmountBeforeOff) is "0.0", the Main shaft clutch ON/OFF status (AdvOutputName.Md.Clutch.MasterOnOffStatus) does not turn to "TRUE (Clutch ON status)" in order to turn back OFF immediately.
2: Clutch command leading edge (ClutchLeadingEdge)	The clutch is turned OFF when the Main shaft clutch command (AdvOutputName.Cd.Clutch.MasterCommand) passes the leading edge (from FALSE to TRUE).
3: Clutch command trailing edge (ClutchTrailingEdge)	The clutch is turned OFF when the Main shaft clutch command (AdvOutputName.Cd.Clutch.MasterCommand) passes the trailing edge (from TRUE to FALSE).
4: Address mode (ClutchAddress)	The clutch is turned OFF when the reference address (the current position after composite main shaft gear or the current position per cycle after main shaft gear) reaches the Main shaft clutch OFF address (AdvOutputName.Pr.Clutch.MasterOffAddress). The movement amount after passing through the OFF address is calculated as the output movement amount of the clutch based on the reference address passing through, thereby controlling the clutch with an accurate movement amount.
15: I/O data specification (ClutchSignal)	The clutch is turned OFF when the external input signal set in the Main shaft clutch OFF signal setting (AdvOutputName.Pr.MasterOffClutchSignal) is detected.

■ Main shaft clutch reference address setting (AdvOutputName.Pr.Clutch.MasterReference)

Select the address type to be used as the reference address for clutch control.

Note that the processing order of gears and clutches varies by the reference address.

Setting value	Description
0: Current position before gear (GearFrontPosition)	The clutch is controlled by using the current position after composite main shaft gear, which is before the main shaft gear conversion. The movement amount after the clutch control is converted then output through the main shaft gear. *: The clutch processing is performed for the current position after composite main shaft gear. The current position after composite main shaft gear is used as the reference address when the Main shaft clutch ON control setting (AdvOutputName.Pr.Clutch.MasterOnControl) is set to "4: Address mode (ClutchAddress)".
1: Current position per cycle after gear (GearPositionPerCycle)	The clutch is controlled by using the current position per cycle after main shaft gear. Output after the main shaft clutch control is movement amount without conversion. *: The clutch processing is performed for the current position per cycle after main shaft gear. The current position per cycle after main shaft gear is used as the reference address when the Main shaft clutch ON control setting (AdvOutputName.Pr.Clutch.MasterOnControl) is set to "4: Address mode (ClutchAddress)".

The unit settings of the following parameters are in position command units of the main shaft.

- Main shaft clutch ON address (AdvOutputName.Pr.Clutch.MasterOnAddress)
- Main shaft clutch OFF address (AdvOutputName.Pr.Clutch.MasterOffAddress)
- Movement amount before main shaft clutch ON (AdvOutputName.Pr.Clutch.MasterMovementAmountBeforeOn)
- Movement amount before main shaft clutch OFF (AdvOutputName.Pr.Clutch.MasterMovementAmountBeforeOff)
- Slippage amount at main shaft clutch ON (AdvOutputName.Pr.Clutch.MasterOnSlippageAmount)
- Slippage amount at main shaft clutch OFF (AdvOutputName.Pr.Clutch.MasterOffSlippageAmount)

■ Main shaft clutch ON address (AdvOutputName.Pr.Clutch.MasterOnAddress)

Set the clutch ON address when "4: Address mode (ClutchAddress)" is set in the Main shaft clutch ON control setting (AdvOutputName.Pr.Clutch.MasterOnControl).

When "1: Current position per cycle after gear (GearPositionPerCycle)" is set in the Main shaft clutch reference address setting (AdvOutputName.Pr.Clutch.MasterReference), the setting address is converted for control within the range from "0.0 to (length per cycle - 0.00001)".

Setting value
-10000000000.0 ≤ Main shaft clutch ON address (AdvOutputName.Pr.Clutch.MasterOnAddress) < 10000000000.0

Ex.

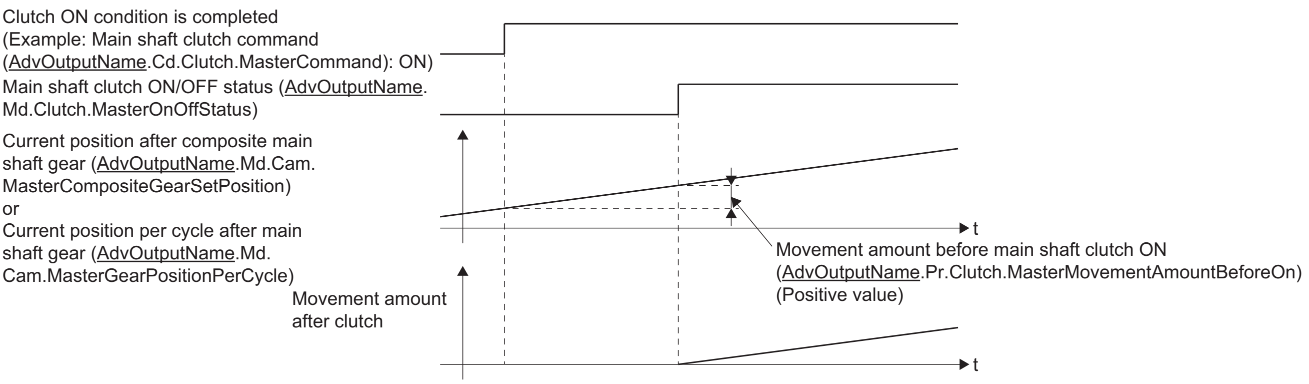
Length per cycle: 20000.0

The ON address is controlled as "19000.0" when the setting value is "-1000.0".

■ Movement amount before main shaft clutch ON (AdvOutputName.Pr.Clutch.MasterMovementAmountBeforeOn)

Set the movement amount for the reference address with a signed number for the distance between the main shaft clutch ON condition completing and the clutch closing.

Setting value	Description
0.00001 to 2147483647.0 (Positive value)	Used when the reference address is increasing in direction.
0.0	No movement (The clutch is immediately turned ON with the clutch ON condition completing.)
-2147483648.0 to -0.00001 (Negative value)	Used when the reference address is decreasing in direction.



■ Main shaft clutch OFF address (AdvOutputName.Pr.Clutch.MasterOffAddress)

Set the clutch OFF address when "4: Address mode (ClutchAddress)" is set in the Main shaft clutch OFF control setting (AdvOutputName.Pr.Clutch.MasterOffControl).

When "1: Current position per cycle after gear (GearPositionPerCycle)" is set in the Main shaft clutch reference address setting (AdvOutputName.Pr.Clutch.MasterReference), the setting address is converted for control within the range from "0.0 to (length per cycle - 0.00001)".

Setting value
-10000000000.0 ≤ Main shaft clutch OFF address (AdvOutputName.Pr.Clutch.MasterOffAddress) < 10000000000.0

Ex.

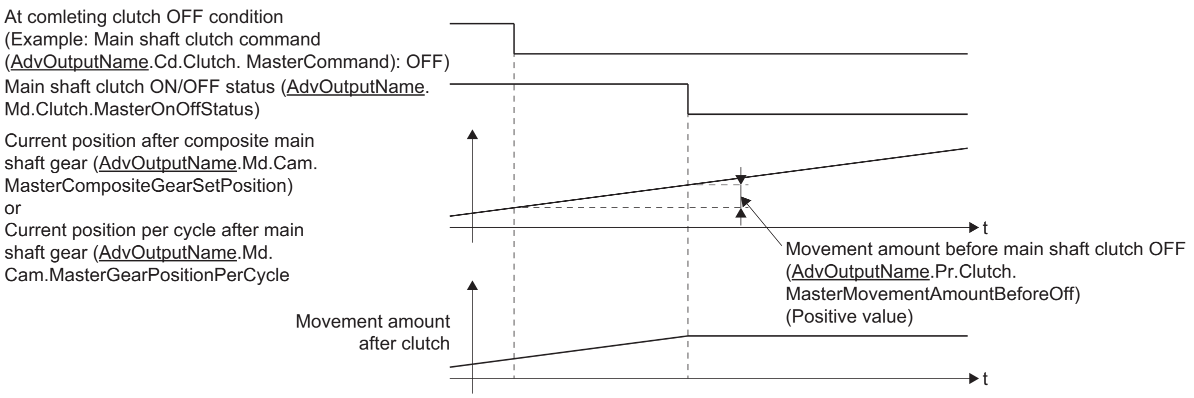
Length per cycle: 20000.0

The OFF address is controlled as "60.0" when the setting value is "40060.0".

■ Movement amount before main shaft clutch OFF (AdvOutputName.Pr.Clutch.MasterMovementAmountBeforeOff)

Set the movement amount for the reference address with a signed number for the distance between the clutch OFF condition completing and the clutch opening.

Setting value	Description
0.00001 to 2147483647.0 (Positive value)	Used when the reference address is increasing in direction.
0.0	No movement (The clutch is immediately turned OFF with the clutch OFF condition completing.)
-2147483648.0 to -0.00001 (Negative value)	Used when the reference address is decreasing in direction.



■ Main shaft clutch smoothing method (AdvOutputName.Pr.Clutch.MasterSmoothingMethod)

Set the smoothing method for clutch ON/OFF.

Setting value	Description
0: Direct (ClutchSmoothingDisabled)	No smoothing.
1: Time constant method (Exponent) (TimeConstantExponent)	Smoothing with an exponential curve based on the time constant setting.
2: Time constant method (Linear) (TimeConstantLinear)	Smoothing with linear acceleration/deceleration based on the time constant setting.
3: Slippage method (Exponent) (SlippageExponent)	Smoothing with an exponential curve based on the slippage amount setting.
4: Slippage method (Linear)(SlippageLinear)	Smoothing with linear acceleration/deceleration based on the slippage amount setting.
5: Slippage method (Linear: Input value follow up) (SlippageLinearFollow)	Smoothing with linear acceleration/deceleration (input value follow up) based on the slippage amount setting.

■ Slippage amount at main shaft clutch ON (AdvOutputName.Pr.Clutch.MasterOnSlippageAmount)

Set the slippage amount at clutch ON when any of the following is set in the Main shaft clutch smoothing method (AdvOutputName.Pr.Clutch.MasterSmoothingMethod).

- "3: Slippage method (Exponent) (SlippageExponent)"
- "4: Slippage method (Linear) (SlippageLinear) (SlippageLinear)"
- "5: Slippage method (Linear: Input value follow up) (SlippageLinearFollow)"

The slippage amount is set in units based on the setting in the Main shaft clutch reference address setting (AdvOutputName.Pr.Clutch.MasterReference).

If the set amount is negative, slippage amount at clutch ON is controlled as 0 (direct).

Point

Do not set digits after the decimal point when "3: Slippage method (Exponent) (SlippageExponent)" is set in the Main shaft clutch smoothing method (AdvOutputName.Pr.Clutch.MasterSmoothingMethod). If set, smoothing is performed with the digits after the decimal point ignored.

■ Slippage amount at main shaft clutch OFF (AdvOutputName.Pr.Clutch.MasterOffSlippageAmount)

Set the slippage amount at clutch OFF when any of the following is set in the Main shaft clutch smoothing method (AdvOutputName.Pr.Clutch.MasterSmoothingMethod).

- "3: Slippage method (Exponent) (SlippageExponent)"
- "4: Slippage method (Linear) (SlippageLinear)"
- "5: Slippage method (Linear: Input value follow up) (SlippageLinearFollow)"

The slippage amount is set in units based on the setting in the Main shaft clutch reference address setting (AdvOutputName.Pr.Clutch.MasterReference).

If the set amount is negative, slippage amount at clutch ON is controlled as 0 (direct).

Point

Do not set digits after the decimal point when "3: Slippage method (Exponent) (SlippageExponent)" is set in the Main shaft clutch smoothing method (AdvOutputName.Pr.Clutch.MasterSmoothingMethod). If set, smoothing is performed with the digits after the decimal point ignored.

Clutch monitor of output axis monitor data (AdvOutputName.Md.Clutch.)

Variable name	Name	Import*1	Data type*1	Monitor value
MasterOnOffStatus	Main shaft clutch ON/OFF status	Operation cycle (FB is starting)	BOOL	FALSE: Clutch OFF status TRUE: Clutch ON status
MasterSmoothingStatus	Main shaft clutch smoothing status		BOOL	FALSE: Not on clutch smoothing TRUE: On clutch smoothing
MasterCumulativeSlippage	Main shaft clutch slippage (accumulative)		LREAL	-2147483648.0 to 2147483647.0

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Main shaft clutch ON/OFF status (AdvOutputName.Md.Clutch.MasterOnOffStatus)

The main shaft clutch ON/OFF status is stored.

■ Main shaft clutch smoothing status (AdvOutputName.Md.Clutch.MasterSmoothingStatus)

The smoothing status of the main shaft clutch is stored.

The status is updated by the Main shaft clutch smoothing method (AdvOutputName.Pr.Clutch.MasterSmoothingMethod).

Method	Description
Time constant method	The status is always "TRUE: On clutch smoothing" during the clutch ON status. The status will be "FALSE: Not on clutch smoothing" when the clutch is turned OFF and smoothing is completed.
Slippage method	The status is "TRUE: On clutch smoothing" till the clutch accumulative slippage amount reaches the slippage at clutch ON when the clutch is turned ON. The status will change to "FALSE: Not on clutch smoothing" when the clutch accumulative slippage amount reaches the slippage at clutch ON. The status is "TRUE: On clutch smoothing" till the clutch accumulative slippage amount reaches 0 when the clutch is turned OFF. The status will change to "FALSE: Not on clutch smoothing" when the clutch accumulative slippage amount reaches 0.

■ Main shaft clutch slippage (accumulative) (AdvOutputName.Md.Clutch.MasterCumulativeSlippage)

The accumulative slippage amount with the slippage method is stored as a signed value.

The absolute value of the accumulative slippage increases to reach the slippage at clutch ON during clutch ON.

The absolute value of the accumulative slippage decreases to reach "0.0" during clutch OFF.

Monitoring of the accumulative slippage is used to check the smoothing progress with the slippage method.

Clutch control data of output axis control data (AdvOutputName.Cd.Clutch.)

15

Variable name	Name	Import*1	Data type*1	Setting value	Initial value
MasterCommand	Main shaft clutch command	Operation cycle (FB is starting)	BOOL	FALSE: Clutch command OFF TRUE: Clutch command ON	FALSE
MasterInvalidCommand	Main shaft clutch control invalid command		BOOL	FALSE: Clutch control valid TRUE: Clutch control invalid	FALSE
MasterForcedOff	Main shaft clutch forced OFF command		BOOL	FALSE: Clutch normal control TRUE: Clutch forced OFF	FALSE
MasterClutchSmoothing TimeCon stant	Main shaft clutch smoothing time constant change value	At start	INT	0 to 5000[ms]	0

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Main shaft clutch command (AdvOutputName.Cd.Clutch.MasterCommand)

Sets ON/OFF for the main shaft clutch command.

This command is used when any of the following is set in the Main shaft clutch ON control setting (AdvOutputName.Pr.Clutch.MasterOnControl).

- 1: Clutch command (ClutchCommand)
- 2: Clutch command leading edge (ClutchLeadingEdge)
- 3: Clutch command trailing edge (ClutchTrailingEdge)

Status is considered as clutch command OFF just before starting synchronous control. If synchronous control is started while the clutch command is ON, the condition is established just after starting synchronous control, by setting "2: Clutch command leading edge (ClutchLeadingEdge)". The condition is not established just after starting, by setting "3: Clutch command trailing edge (ClutchTrailingEdge)".

■ Main shaft clutch control invalid command (AdvOutputName.Cd.Clutch.MasterInvalidCommand)

The main shaft clutch control is invalid if "TRUE: Clutch control invalid" is set. The previous clutch ON/OFF status remains before clutch control becomes invalid.

Clutch control will not become invalid during movement before clutch ON and during movement before clutch OFF. Instead, clutch control will become invalid after movement is completed.

■ Main shaft clutch forced OFF command (AdvOutputName.Cd.Clutch.MasterForcedOff)

Set "TRUE: Clutch forced OFF" to force the clutch OFF. The output value from the clutch becomes "0" immediately, even during clutch smoothing. The slippage (accumulative) amount is set to 0 if smoothing with a slippage method.

Reset to "FALSE: Clutch normal control" to restart the clutch control from the clutch OFF status after using the clutch forced OFF command.

■ Main shaft clutch smoothing time constant change valuex (AdvOutputName.Cd.Clutch.MasterClutchSmoothingTimeConstant)

The smoothing time constant of the main shaft clutch is changed. The setting is imported when MCv_AdvancedSync (Advanced Synchronous Control) is started.

Setting value	Description
When a negative value is set	No smoothing.
When 0 is set	Smoothing based on the setting value of the Main shaft clutch smoothing time constant (AdvOutPutName.PrConst.MasterClutchSmoothingTimeConstant).
When a positive value is set	Smoothing based on the setting value of the Main shaft clutch smoothing time constant change value (AdvOutPutName.Cd.Clutch.MasterClutchSmoothingTimeConstant).

Set within the following setting range.

If the set value exceeds the range below, "Out of Advanced Synchronous Control Variable Range Warning (error code: 0D3FH)" occurs, and smoothing is executed based on the setting value of the Main shaft clutch smoothing time constant (AdvOutPutName.PrConst.MasterClutchSmoothingTimeConstant).

Setting range
Main shaft clutch smoothing time constant (AdvOutPutName.PrConst.MasterClutchSmoothingTimeConstant) ≤ Main shaft clutch smoothing time constant change value (AdvOutPutName.Cd.Clutch.MasterClutchSmoothingTimeConstant)

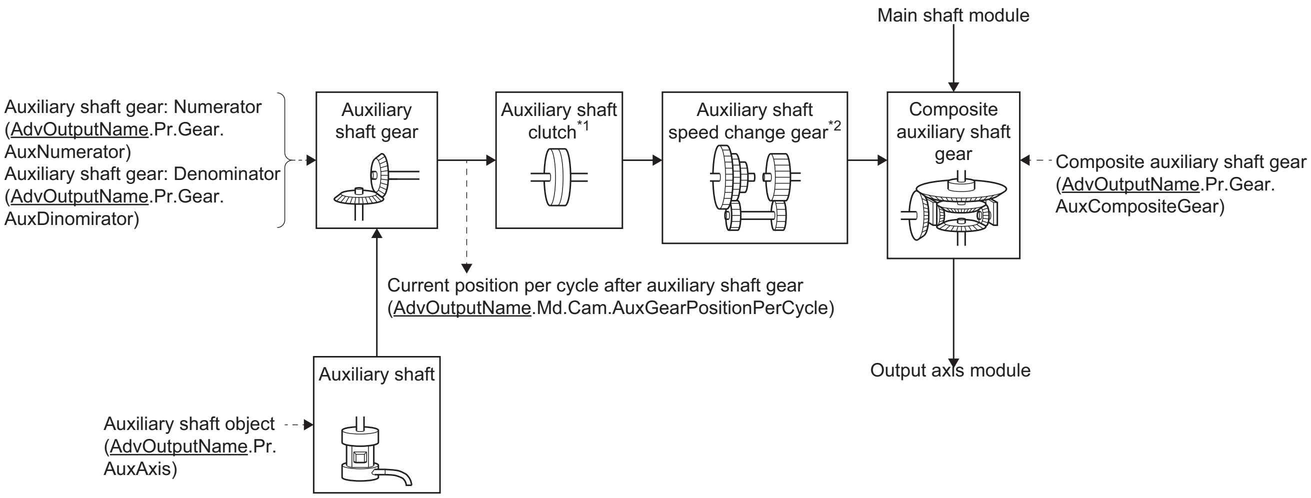
Auxiliary shaft module

The auxiliary shaft modules transmit the movement amount inputted from an input axis to the output axis.
The functions for an auxiliary shaft module include gears, clutches, and speed change gears.
The auxiliary shaft module corresponds to the axis set in Auxiliary shaft object (AdvOutPutName.Pr.AuxAxis).
Refer to the following for the setting items for gears, clutches, and speed change gears of the auxiliary shaft module.

Function	Member	Variable	Reference
Gear	Output axis parameters (AdvOutPutName.Pr.)	Gear parameter (Gear.)	☞ Page 534 Gear parameter of output axis parameters (AdvOutPutName.Pr.Gear.)
Clutch	Output axis parameters constant (AdvOutPutName.PrConst.)	—	☞ Page 535 Output axis parameters constant (AdvOutPutName.PrConst.)
	Output axis parameters (AdvOutPutName.Pr.)	—	☞ Page 535 Output axis parameters (AdvOutPutName.Pr.)
		Clutch parameter (Clutch.)	☞ Page 537 Clutch parameter of output axis parameters (AdvOutputName.Pr.Clutch.)
	Output axis monitor data (AdvOutPutName.Md.)	Clutch monitor (Clutch.)	☞ Page 543 Clutch monitor of output axis monitor data (AdvOutputName.Md.Clutch.)
	Output axis control data (AdvOutPutName.Cd.)	Clutch control data (Clutch.)	☞ Page 544 Clutch control data of output axis monitor data (AdvOutputName.Cd.Clutch.)
Speed change gear	Output axis parameters constant (AdvOutPutName.PrConst.)	—	☞ Page 560 Output axis parameters constant (AdvOutPutName.PrConst.)
	Speed change gear parameter of output axis parameters (AdvOutPutName.Pr.SpeedChangeGear.)	Speed change gear parameter (SpeedChangeGear.)	☞ Page 559 Speed change gear parameter of output axis parameters (AdvOutPutName.Pr.SpeedChangeGear.)
	Speed change gear control data of output axis parameters (AdvOutPutName.Cd.SpeedChangeGear.)	Speed change gear control data (SpeedChangeGear.)	☞ Page 560 Speed change gear control data of output axis parameters (AdvOutPutName.Cd.SpeedChangeGear.)

- Auxiliary shaft module

  The input value is generated from the auxiliary shaft. The input value can be converted by the auxiliary shaft gear that provides the deceleration ratio, rotation direction, etc. for the machine system.



*1 Clutch (☞ Page 545 Clutch)
*2 Speed change gear (☞ Page 558 Speed change gear)

Gear parameter of output axis parameters (AdvOutPutName.Pr.Gear.)

Variable name	Name	Import*1	Data type*1	Setting value	Initial value
AuxCompositeGear	Composite auxiliary shaft gear	Operation cycle (FB is starting)	WORD(HEX)	H□□□□ → Main shaft input method 0: No input 1: Input + 2: Input - → Auxiliary shaft input method 0: No input 1: Input + 2: Input -	0001H
AuxNumerator	Auxiliary shaft gear: Numerator	At start	DINT	-2147483647 to 2147483647	1
AuxDenominator	Auxiliary shaft gear: Denominator		DWORD(UDINT)	1 to 2147483647	1

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Composite auxiliary shaft gear (AdvOutputName.Pr.Gear.AuxCompositeGear)

Set the composite method for the current position after the main shaft and sub input axis composition and the current position of the auxiliary shaft.

Setting value	Description
0: No input	The input value from the input axis is calculated as 0.
1: Input +	The input value from the input axis is calculated as it is.
2: Input -	The input value from the input axis is calculated with its opposite sign.

Operation assumes "0: No input" if the value is set out of the range from 0 to 2.

Point

The composite method for the composite auxiliary shaft gear can be changed during synchronous control. It can also be used as a clutch to switch input values of the main shaft and auxiliary shafts.

■ Auxiliary shaft gear: Numerator (AdvOutputName.Pr.Gear.AuxNumerator)

Set the numerator for the auxiliary shaft gear input value conversion.
The input value is converted as follows.

$$\text{Input value after conversion} = \text{Input value before conversion} \times \frac{\text{Auxiliary shaft gear: Numerator}}{\text{Auxiliary shaft gear: Denominator}}$$

The input value direction can be reversed by setting a negative value in Auxiliary shaft gear: Numerator.

Point

Setting a reduced fraction is recommended for "Auxiliary shaft gear: Numerator/Auxiliary shaft gear: Denominator".

■ Auxiliary shaft gear: Denominator (AdvOutputName.Pr.Gear.AuxDenominator)

Set the denominator for the auxiliary shaft input value conversion.
Set together with Auxiliary shaft gear: Numerator.

Output axis parameters constant (AdvOutPutName.PrConst.)

Variable name	Name	Import*1	Data type*1	Setting value	Initial value
AuxClutchSmoothingTimeConstant	Auxiliary shaft clutch smoothing time constant	At system start	WORD(UINT)	0 to 5000 [ms]	0 [ms]

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Auxiliary shaft clutch smoothing time constant (AdvOutPutName.PrConst.AuxClutchSmoothingTimeConstant)

Set a time constant when the time constant method is set to "1: Time constant method (Exponent) (TimeConstantExponent)" or "2: Time constant method (Linear) (TimeConstantLinear)" in the Auxiliary shaft clutch smoothing method (AdvOutputName.Pr.Clutch.AuxSmoothingMethod).

The time constant setting applies for both clutch ON/ OFF.

Output axis parameters (AdvOutPutName.Pr.)

Variable name	Name	Import*1	Data type*1	Setting value	Initial value
AuxOnClutchSignal	Auxiliary shaft clutch ON signal setting	At system start	SIGNAL_SELECT	—	—
AuxOffClutchSignal	Auxiliary shaft clutch OFF signal setting			—	—

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Auxiliary shaft clutch ON signal setting (AdvOutPutName.Pr.AuxOnClutchSignal)

Set the variable and device used as the clutch external signal in SIGNAL_SELECT structure when the Auxiliary shaft clutch ON control setting (AdvOutputName.Pr.Clutch.AuxOnControl) is set to "15: I/O data specification (ClutchSignal)".

For SIGNAL_SELECT structure, refer to the following.

Page 383 SIGNAL_SELECT structure

The following shows the settings peculiar to Auxiliary shaft clutch ON signal setting (AdvOutputName.Pr.AuxOnClutchSignal).

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

Point

When setting [OBJ] for Target (Source.Target), start MCv_AdvancedSync (Advanced Synchronous Control) after the communication with device stations is established. If the communication is not established, "External Signal String Incorrect (error code: 1AB6H)" occurs.

■ Auxiliary shaft clutch OFF signal setting (AdvOutPutName.Pr.AuxOffClutchSignal)

Set the variable and device used as the clutch external signal in SIGNAL_SELECT structure when the Auxiliary shaft clutch OFF control setting (AdvOutputName.Pr.Clutch.AuxOffControl) is set to "15: I/O data specification (ClutchSignal)".

For SIGNAL_SELECT structure, refer to the following.

☞ Page 383 SIGNAL_SELECT structure

The following shows the settings peculiar to Auxiliary shaft clutch OFF signal setting (AdvOutputName.Pr.AuxOffClutchSignal).

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

Point

When setting [OBJ] for Target (Source.Target), start MCv_AdvancedSync (Advanced Synchronous Control) after the communication with device stations is established. If the communication is not established, "External Signal String Incorrect (error code: 1AB6H)" occurs.

Clutch parameter of output axis parameters (AdvOutputName.Pr.Clutch.)

Variable name	Name	Import*1	Data type*1	Setting value	Initial value
AuxOnControl	Auxiliary shaft clutch ON control setting	Operation cycle (FB is starting)	INT (MC_CLUTCH_ METHOD)	0: Invalid (NoClutch) (Direct coupled operation) 1: Clutch command (ClutchCommand) 2: Clutch command leading edge (ClutchLeadingEdge) 3: Clutch command trailing edge (ClutchTrailingEdge) 4: Address mode (ClutchAddress) 15: I/O data specification (ClutchSignal)	0: Invalid (NoClutch)
AuxOffControl	Auxiliary shaft clutch OFF control setting		INT (MC_CLUTCH_ METHOD)	0: Invalid (NoClutch) (OFF control invalid) 1: Clutch command (ClutchCommand) (One-shot operation) 2: Clutch command leading edge (ClutchLeadingEdge) 3: Clutch command trailing edge (ClutchTrailingEdge) 4: Address mode(ClutchAddress) 15: I/O data specification (ClutchSignal)	0: Invalid (NoClutch)
AuxReference	Auxiliary shaft clutch reference address setting	At start	INT (MC_CLUTCH_ REFERENCE)	0: Current position before gear (GearFrontPosition) 1: Current position per cycle after gear (GearPositionPerCycle)	0: Current position before gear (GearFrontPosition)
AuxOnAddress	Auxiliary shaft clutch ON address	Operation cycle (FB is starting)	LREAL	-10000000000.0 to 10000000000.0	0.0
AuxMovementAmountBeforeOn	Movement amount before auxiliary shaft clutch ON	At comleting clutch ON condition	LREAL	-2147483648.0 to 2147483647.0	0.0
AuxOffAddress	Auxiliary shaft clutch OFF address	Operation cycle (FB is starting)	LREAL	-10000000000.0 to 10000000000.0	0.0
AuxMovementAmountBeforeOff	Movement amount before auxiliary shaft clutch OFF	At comleting clutch ON condition	LREAL	-2147483648.0 to 2147483647.0	0.0
AuxSmoothingMethod	Auxiliary shaft clutch smoothing method	At start	INT (MC_CLUTCH_ SMOOTHING_ METHOD)	0: Direct (ClutchSmoothingDisabled) 1: Time constant method (Exponent) (TimeConstantExponent) 2: Time constant method (Linear) (TimeConstantLinear) 3: Slippage method (Exponent) (SlippageExponent) 4: Slippage method (Linear) (SlippageLinear) 5: Slippage method (Linear: Input value follow up) (SlippageLinearFollow)	0: Direct (ClutchSmoothingDisabled)
AuxOnSlippageAmount	Slippage amount at auxiliary shaft clutch ON	At turning clutch ON	LREAL	0.0 to 2147483647.0	0.0
AuxOffSlippageAmount	Slippage amount at auxiliary shaft clutch OFF	At turning clutch OFF	LREAL	0.0 to 2147483647.0	0.0

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Auxiliary shaft clutch ON control setting (AdvOutputName.Pr.Clutch.AuxOnControl)

Set the control method for auxiliary shaft clutch ON.

The Auxiliary shaft clutch ON control setting (AdvOutputName.Pr.Clutch.AuxOnControl) can be changed during synchronous control. However, changes from settings other than "0: Invalid (NoClutch)" to "0: Invalid (NoClutch)" are not possible.

Setting value	Description
0: Invalid (NoClutch) (Direct coupled operation)	Execute direct coupled operation without clutch control.
1: Clutch command (ClutchCommand)	The clutch is turned ON/OFF by the operations of the Auxiliary shaft clutch command (AdvOutputName.Cd.Clutch.AuxCommand) TRUE/FALSE. (Setting in the Auxiliary shaft clutch OFF control setting (AdvOutputName.Pr.Clutch.AuxOffControl) is not applicable in the clutch command ON/OFF mode.)
2: Clutch command leading edge (ClutchLeadingEdge)	The clutch is turned ON when the Auxiliary shaft clutch command (AdvOutputName.Cd.Clutch.AuxCommand) passes the leading edge (from FALSE to TRUE).
3: Clutch command trailing edge (ClutchTrailingEdge)	The clutch is turned ON when the Auxiliary shaft clutch command (AdvOutputName.Cd.Clutch.AuxCommand) passes the trailing edge (from TRUE to FALSE).
4: Address mode (ClutchAddress)	The clutch is turned ON when the reference address (the cumulative current position of the auxiliary shaft or the current position per cycle after auxiliary shaft gear) reaches the Auxiliary shaft clutch ON address (AdvOutputName.Pr.Clutch.AuxOnAddress). The movement amount after passing through the ON address is calculated as the output movement amount of the clutch based on the reference address passing through, thereby controlling the clutch with an accurate movement amount.
15: I/O data specification (ClutchSignal)	The clutch is turned ON when the external input signal set in the Auxiliary shaft clutch ON signal setting (AdvOutputName.Pr.AuxOnClutchSignal) is detected.

Point

- Other clutch parameters are not applicable during direct coupled operation by setting the Auxiliary shaft clutch ON control setting (AdvOutputName.Pr.Clutch.AuxOnControl) to "0: Invalid (NoClutch)". The Auxiliary shaft clutch forced OFF command (AdvOutputName.Cd.Clutch.AuxForcedOff) and the change of the clutch control setting are ignored during direct coupled operation.
- Changes from settings other than "0: Invalid (NoClutch)" to "0: Invalid (NoClutch)" are not possible. If the setting is changed in that manner, "Out of Advanced Synchronous Control Variable Range Warning (error code: 0D3FH)" occurs.

■ Auxiliary shaft clutch OFF control setting (AdvOutputName.Pr.Clutch.AuxOffControl)

Set the control method for auxiliary shaft clutch OFF.

Setting value	Description
0: Invalid (NoClutch) (OFF control invalid)	Clutch OFF control is not used. This setting is applicable only for execution with clutch ON control.
1: Clutch command (ClutchCommand) (One-shot operation)	The clutch is turned OFF after moving the distance set in the Movement amount before auxiliary shaft clutch OFF (AdvOutputName.Pr.Clutch.AuxMovementAmountBeforeOff) (One-shot operation) after the clutch command turns ON. If the Movement amount before auxiliary shaft clutch OFF (AdvOutputName.Pr.Clutch.AuxMovementAmountBeforeOff) is "0.0", the Auxiliary shaft clutch ON/OFF status (AdvOutputName.Md.Clutch.AuxOnOffStatus) does not turn to "TRUE (Clutch ON status)" in order to turn back OFF immediately.
2: Clutch command leading edge (ClutchLeadingEdge)	The clutch is turned OFF when the Auxiliary shaft clutch command (AdvOutputName.Cd.Clutch.AuxCommand) passes the leading edge (from FALSE to TRUE).
3: Clutch command trailing edge (ClutchTrailingEdge)	The clutch is turned OFF when the Auxiliary shaft clutch command (AdvOutputName.Cd.Clutch.AuxCommand) passes the trailing edge (from TRUE to FALSE).
4: Address mode (ClutchAddress)	The clutch is turned OFF when the reference address (the cumulative current position of the auxiliary shaft or the current position per cycle after auxiliary shaft gear) reaches the Auxiliary shaft clutch OFF address (AdvOutputName.Pr.Clutch.AuxOffAddress). The movement amount after passing through the OFF address is calculated as the output movement amount of the clutch based on the reference address passing through, thereby controlling the clutch with an accurate movement amount.
15: I/O data specification (ClutchSignal)	The clutch is turned OFF when the external input signal set in the Auxiliary shaft clutch OFF signal setting (AdvOutputName.Pr.AuxOffClutchSignal) is detected.

■ Auxiliary shaft clutch reference address setting (AdvOutputName.Pr.Clutch.AuxReference)

Select the address type to be used as the reference address for clutch control.

Note that the processing order of gears and clutches varies by the reference address.

Setting value	Description
0: Current position before gear (GearFrontPosition)	The clutch is controlled by using the input axis cumulative current position set in auxiliary shaft, which is before the auxiliary shaft gear conversion. The movement amount after the clutch control is converted then output through the auxiliary shaft gear. *: The clutch processing is performed for the current position of auxiliary shaft. The current position of auxiliary shaft is used as the reference address when the Auxiliary shaft clutch OFF control setting (AdvOutputName.Pr.Clutch.AuxOffControl) is set to "4: Address mode (ClutchAddress)".
1: Current position per cycle after gear (GearPositionPerCycle)	The clutch is controlled by using the current position per cycle after auxiliary shaft gear. Output after the auxiliary shaft clutch control is movement amount without conversion. *: The clutch processing is performed for the current position per cycle after auxiliary shaft gear. The current position per cycle after auxiliary shaft gear is used as the reference address when the Auxiliary shaft clutch OFF control setting (AdvOutputName.Pr.Clutch.AuxOffControl) is set to "4: Address mode (ClutchAddress)".

The unit settings of the following parameters are in position command units of the auxiliary shaft.

- Auxiliary shaft clutch ON address (AdvOutputName.Pr.Clutch.AuxOnAddress)
- Auxiliary shaft clutch OFF address (AdvOutputName.Pr.Clutch.AuxOffAddress)
- Movement amount before auxiliary shaft clutch ON (AdvOutputName.Pr.Clutch.AuxMovementAmountBeforeOn)
- Movement amount before auxiliary shaft clutch OFF (AdvOutputName.Pr.Clutch.AuxMovementAmountBeforeOff)
- Slippage amount at auxiliary shaft clutch ON (AdvOutputName.Pr.Clutch.AuxOnSlippageAmount)
- Slippage amount at auxiliary shaft clutch OFF (AdvOutputName.Pr.Clutch.AuxOffSlippageAmount)

15

■ Auxiliary shaft clutch ON address (AdvOutputName.Pr.Clutch.AuxOnAddress)

Set the clutch ON address when "4: Address mode (ClutchAddress)" is set in the Auxiliary shaft clutch ON control setting (AdvOutputName.Pr.Clutch.AuxOnControl).

When "1: Current position per cycle after gear (GearPositionPerCycle)" is set in the Auxiliary shaft clutch reference address setting (AdvOutputName.Pr.Clutch.AuxReference), the setting address is converted for control within the range from 0.0 to (length per cycle - 0.00001).

Setting value
-10000000000.0 ≤ Auxiliary shaft clutch ON address (AdvOutputName.Pr.Clutch.AuxOnAddress) < 10000000000.0

Ex.

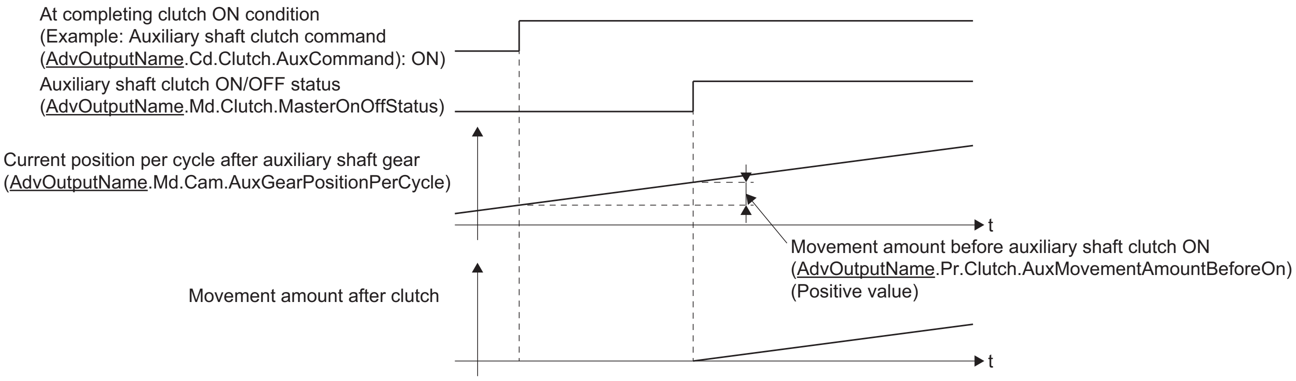
Length per cycle: 20000.0

The ON address is controlled as "19000.0" when the setting value is "-1000.0".

■ Movement amount before auxiliary shaft clutch ON (AdvOutputName.Pr.Clutch.AuxMovementAmountBeforeOn)

Set the movement amount for the reference address with a signed number for the distance between the clutch ON condition completing and the clutch closing.

Setting value	Description
0.00001 to 2147483647.0 (Positive value)	Used when the reference address is increasing in direction.
0.0	No movement (The clutch is immediately turned ON with the clutch ON condition completing.)
-2147483648.0 to -0.00001 (Negative value)	Used when the reference address is decreasing in direction.



■ Auxiliary shaft clutch OFF address (AdvOutputName.Pr.Clutch.AuxOffAddress)

Set the clutch OFF address when "4: Address mode (ClutchAddress)" is set in the Auxiliary shaft clutch OFF control setting (AdvOutputName.Pr.Clutch.AuxOffControl).

When "1: Current position per cycle after gear (GearPositionPerCycle)" is set in the Auxiliary shaft clutch reference address setting (AdvOutputName.Pr.Clutch.AuxReference), the setting address is converted for control within the range from "0.0 to (length per cycle - 0.00001)".

Setting value
-10000000000.0 ≤ Auxiliary shaft clutch OFF address (AdvOutputName.Pr.Clutch.AuxOffAddress) < 10000000000.0

Ex.

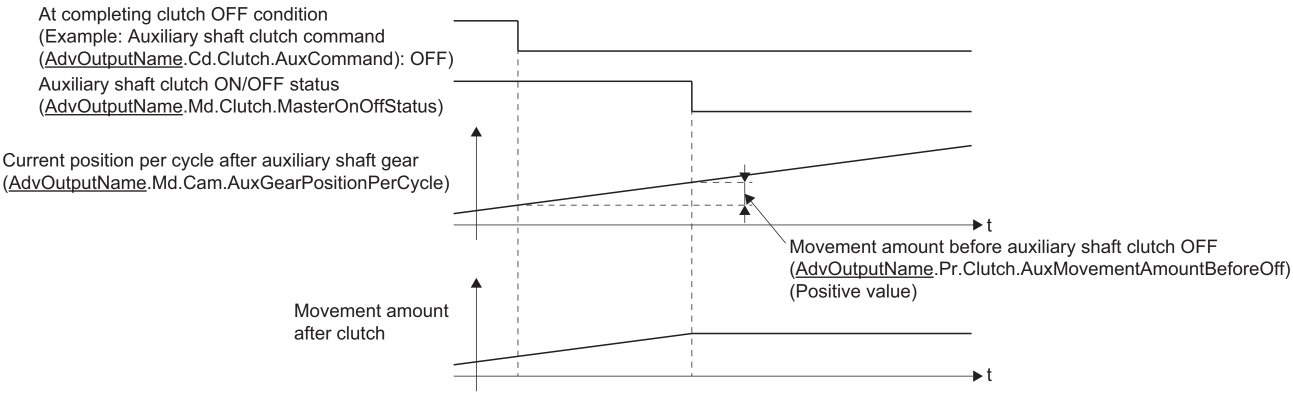
Length per cycle: 20000.0

The OFF address is controlled as "60.0" when the setting value is "40060.0".

■ Movement amount before auxiliary shaft clutch OFF (AdvOutputName.Pr.Clutch.AuxMovementAmountBeforeOff)

Set the movement amount for the reference address with a signed number for the distance between the clutch OFF condition completing and the clutch opening.

Setting value	Description
0.00001 to 2147483647.0 (Positive value)	Used when the reference address is increasing in direction.
0.0	No movement (The clutch is immediately turned OFF with the clutch OFF condition completing.)
-2147483648.0 to -0.00001 (Negative value)	Used when the reference address is decreasing in direction.



■ Auxiliary shaft clutch smoothing method (AdvOutputName.Pr.Clutch.AuxSmoothingMethod)

Set the smoothing method for clutch ON/OFF.

Setting value	Description
0: Direct (ClutchSmoothingDisabled)	No smoothing.
1: Time constant method (Exponent) (TimeConstantExponent)	Smoothing with an exponential curve based on the time constant setting.
2: Time constant method (Linear) (TimeConstantLinear)	Smoothing with linear acceleration/deceleration based on the time constant setting.
3: Slippage method (Exponent) (SlippageExponent)	Smoothing with an exponential curve based on the slippage amount setting.
4: Slippage method (Linear) (SlippageLinear)	Smoothing with linear acceleration/deceleration based on the slippage amount setting.
5: Slippage method (Linear: Input value follow up) (SlippageLinearFollow)	Smoothing with linear acceleration/deceleration (input value follow up) based on the slippage amount setting.

■ Slippage amount at auxiliary shaft clutch ON (AdvOutputName.Pr.Clutch.AuxOnSlippageAmount)

Set the slippage amount at clutch ON when any of the following is set in the Auxiliary shaft clutch smoothing method (AdvOutputName.Pr.Clutch.AuxSmoothingMethod).

- "3: Slippage method (Exponent) (SlippageExponent)"
- "4: Slippage method (Linear) (SlippageLinear) (SlippageLinear)"
- "5: Slippage method (Linear: Input value follow up) (SlippageLinearFollow)"

The slippage amount is set in units based on the setting in the Auxiliary shaft clutch reference address setting (AdvOutputName.Pr.Clutch.AuxReference).

If the set amount is negative, slippage amount at clutch ON is controlled as 0 (direct).

Point

Do not set digits after the decimal point when "3: Slippage method (Exponent) (SlippageExponent)" is set in the Auxiliary shaft clutch smoothing method (AdvOutputName.Pr.Clutch.AuxSmoothingMethod).
If set, smoothing is performed with the digits after the decimal point ignored.

■ Slippage amount at auxiliary shaft clutch OFF (AdvOutputName.Pr.Clutch.AuxOffSlippageAmount)

Set the slippage amount at clutch OFF when any of the following is set in the Auxiliary shaft clutch smoothing method (AdvOutputName.Pr.Clutch.AuxSmoothingMethod).

- "3: Slippage method (Exponent) (SlippageExponent)"
- "4: Slippage method (Linear) (SlippageLinear) (SlippageLinear)"
- "5: Slippage method (Linear: Input value follow up) (SlippageLinearFollow)"

The slippage amount is set in units based on the setting in the Auxiliary shaft clutch reference address setting (AdvOutputName.Pr.Clutch.AuxReference).

If the set amount is negative, slippage amount at clutch ON is controlled as 0 (direct).

Point

Do not set digits after the decimal point when "3: Slippage method (Exponent) (SlippageExponent)" is set in the Auxiliary shaft clutch smoothing method (AdvOutputName.Pr.Clutch.AuxSmoothingMethod).
If set, smoothing is performed with the digits after the decimal point ignored.

Clutch monitor of output axis monitor data (AdvOutputName.Md.Clutch.)

Variable name	Name	Import*1	Data type*1	Setting value
AuxOnOffStatus	Auxiliary shaft clutch ON/OFF status	Operation cycle (FB is starting)	BOOL	FALSE: Clutch OFF status TRUE: Clutch ON status
AuxSmoothingStatus	Auxiliary shaft clutch smoothing status		BOOL	FALSE: Not on clutch smoothing TRUE: On clutch smoothing
AuxCumulativeSlippage	Auxiliary shaft clutch slippage (accumulative)		LREAL	-2147483648.0 to 2147483647.0

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Auxiliary shaft clutch ON/OFF status (AdvOutputName.Md.Clutch.AuxOnOffStatus)

The auxiliary shaft clutch ON/OFF status is stored.

■ Auxiliary shaft clutch smoothing status (AdvOutputName.Md.Clutch.AuxSmoothingStatus)

The smoothing status of the auxiliary shaft clutch is stored.

The status is updated by the Auxiliary shaft clutch smoothing method (AdvOutputName.Pr.Clutch.AuxSmoothingMethod).

Method	Description
Time constant method	The status is always "TRUE: On clutch smoothing" during the clutch ON status. The status will be "FALSE: Not on clutch smoothing" when the clutch is turned OFF and smoothing is completed.
Slippage method	The status is "TRUE: On clutch smoothing" till the clutch accumulative slippage amount reaches the slippage at clutch ON when the clutch is turned ON. The status will change to "FALSE: Not on clutch smoothing" when the clutch accumulative slippage amount reaches the slippage at clutch ON. The status is "TRUE: On clutch smoothing" till the clutch accumulative slippage amount reaches 0 when the clutch is turned OFF. The status will change to "FALSE: Not on clutch smoothing" when the clutch accumulative slippage amount reaches 0.

■ Auxiliary shaft clutch slippage (accumulative) (AdvOutputName.Md.Clutch.AuxCumulativeSlippage)

The accumulative slippage amount with the slippage method is stored as a signed value.

The absolute value of the accumulative slippage increases to reach the slippage at clutch ON during clutch ON.

The absolute value of the accumulative slippage decreases to reach "0.0" during clutch OFF.

Monitoring of the accumulative slippage is used to check the smoothing progress with the slippage method.

Clutch control data of output axis monitor data (AdvOutputName.Cd.Clutch.)

Variable name	Name	Import*1	Data type*1	Setting value	Initial value
AuxCommand	Auxiliary shaft clutch command	Operation cycle (FB is starting)	BOOL	FALSE: Clutch command OFF TRUE: Clutch command ON	FALSE
AuxInvalidCommand	Auxiliary shaft clutch 0ontrol invalid command		BOOL	FALSE: Clutch control valid TRUE: Clutch control invalid	FALSE
AuxForcedOff	Auxiliary shaft clutch forced OFF command		BOOL	FALSE: Clutch normal control TRUE: Clutch forced OFF	FALSE
AuxClutchSmoothingTimeConstant	Auxiliary shaft clutch smoothing time constant change value	At start	INT	0 to 5000 [ms]	0

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Auxiliary shaft clutch command (AdvOutputName.Cd.Clutch.AuxCommand)

Sets ON/OFF for the auxiliary shaft clutch command.

This command is used when any of the following is set in the Auxiliary shaft clutch ON control setting (AdvOutputName.Pr.Clutch.AuxOnControl).

- 1: Clutch command (ClutchCommand)
- 2: Clutch command leading edge (ClutchLeadingEdge)
- 3: Clutch command trailing edge (ClutchTrailingEdge)

Status is considered as clutch command OFF just before starting synchronous control. If synchronous control is started while the clutch command is ON, the condition is established just after starting synchronous control, by setting "2: Clutch command leading edge (ClutchLeadingEdge)". The condition is not established just after starting, by setting "3: Clutch command trailing edge (ClutchTrailingEdge)".

■ Auxiliary shaft clutch control invalid command (AdvOutputName.Cd.Clutch.AuxInvalidCommand)

The auxiliary shaft clutch control is invalid if "TRUE: Clutch control invalid" is set. The previous clutch ON/OFF status remains before clutch control becomes invalid.

Clutch control will not become invalid during movement before clutch ON and during movement before clutch OFF. Instead, clutch control will become invalid after movement is completed.

■ Auxiliary shaft clutch forced OFF command (AdvOutputName.Cd.Clutch.AuxForcedOff)

Set "TRUE: Clutch forced OFF" to force the clutch OFF. The output value from the clutch becomes "0" immediately, even during clutch smoothing. The slippage (accumulative) amount is set to 0 if smoothing with a slippage method.

Reset to "FALSE: Clutch normal control" to restart the clutch control from the clutch OFF status after using the clutch forced OFF command.

■ Auxiliary shaft clutch smoothing time constant change value (AdvOutputName.Cd.Clutch.AuxClutchSmoothingTimeConstant)

The smoothing time constant of the auxiliary shaft clutch is changed. The setting is imported when MCv_AdvancedSync (Advanced Synchronous Control) is started.

Setting value	Description
When a negative value is set	No smoothing.
When 0 is set	Smoothing based on the setting value of the Auxiliary shaft clutch smoothing time constant (AdvOutPutName.PrConst.AuxClutchSmoothingTimeConstant).
When a positive value is set	Smoothing based on the setting value of the Auxiliary shaft clutch smoothing time constant change value (AdvOutPutName.Cd.Clutch.AuxClutchSmoothingTimeConstant).

Set within the following setting range.

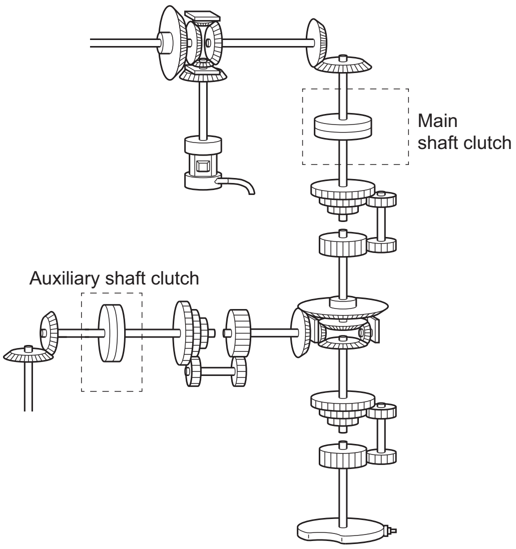
If the set value exceeds the range below, "Out of Advanced Synchronous Control Variable Range Warning (error code: 0D3FH)" occurs, and smoothing is executed based on the setting value of the Auxiliary shaft clutch smoothing time constant (AdvOutPutName.PrConst.AuxClutchSmoothingTimeConstant).

Setting range
Auxiliary shaft clutch smoothing time constant (AdvOutPutName.PrConst.AuxClutchSmoothingTimeConstant) ≤ Auxiliary shaft clutch smoothing time constant change value (AdvOutPutName.Cd.Clutch.AuxClutchSmoothingTimeConstant)

Clutch

The clutch is used to transmit/disengage command pulses from the main/auxiliary shaft input through turning the clutch ON/OFF, which controls the operation/stop of the output axis.

A clutch can be configured for the main and auxiliary shafts.



The clutch status is changed from FALSE to TRUE based on the conditions set in the Main shaft clutch ON control setting (AdvOutputName.Pr.Clutch.MasterOnControl) and Auxiliary shaft clutch ON control setting (AdvOutputName.Pr.Clutch.AuxOnControl), and changed from TRUE to FALSE based on the conditions of the Main shaft clutch OFF control setting (AdvOutputName.Pr.Clutch.MasterOffControl) and Auxiliary shaft clutch OFF control setting (AdvOutputName.Pr.Clutch.AuxOffControl).

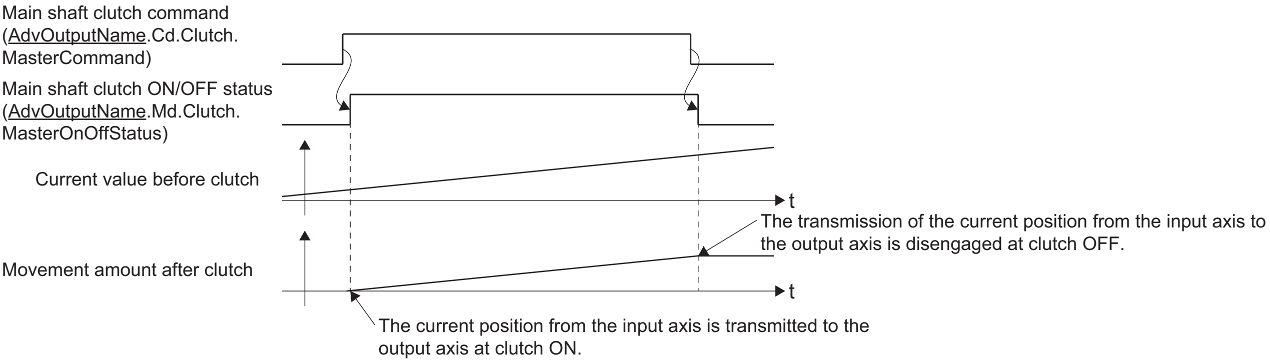
The current position of the input axis is transmitted to the output axis at the timing of the clutch ON/OFF status changing to TRUE.

The input and output axes are disengaged at the timing of the clutch ON/OFF status turning to FALSE.

The following explains the operations of clutch ON/OFF.

Ex.

When the Main shaft clutch ON control setting (AdvOutputName.Pr.Clutch.MasterOnControl) is set to"1: Clutch command (ClutchCommand)"



The movement amount of the output axis before clutch ON and OFF can be adjusted with parameters of the movement amount before clutch ON or the movement amount before clutch OFF.

The smoothing time constant can be set to smooth the operation of the output axis before and after clutch ON/OFF.

15

Clutch control setting

Set the clutch ON and OFF control methods separately. Set the ON control methods in Main shaft clutch ON control setting (AdvOutputName.Pr.Clutch.MasterOnControl) and Auxiliary shaft clutch ON control setting (AdvOutputName.Pr.Clutch.AuxOnControl), and the clutch OFF control methods in Main shaft clutch OFF control setting (AdvOutputName.Pr.Clutch.MasterOffControl) and Auxiliary shaft clutch OFF control setting (AdvOutputName.Pr.Clutch.AuxOffControl).

The clutch control setting can be changed during synchronous control. However, changes from settings other than "0: Invalid (NoClutch) (Direct coupled operation)" to "0: Invalid (NoClutch) (Direct coupled operation)" are not possible in the ON control mode.

Item	Setting item		Setting value
	Main shaft clutch	Auxiliary shaft clutch	
Clutch ON control setting	Main shaft clutch ON control setting (AdvOutputName.Pr.Clutch.MasterOnControl)	Auxiliary shaft clutch ON control setting (AdvOutputName.Pr.Clutch.AuxOnControl)	0: Invalid (NoClutch) (Direct coupled operation) 1: Clutch command (ClutchCommand) 2: Clutch command leading edge (ClutchLeadingEdge) 3: Clutch command trailing edge (ClutchTrailingEdge) 4: Address mode (ClutchAddress) 15: I/O data specification (ClutchSignal)
Clutch OFF control setting	Main shaft clutch OFF control setting (AdvOutputName.Pr.Clutch.MasterOffControl)	Auxiliary shaft clutch OFF control setting (AdvOutputName.Pr.Clutch.AuxOffControl)	0: Invalid (NoClutch) (OFF control invalid) 1: Clutch command (ClutchCommand) (Oneshot operation) 2: Clutch command leading edge (ClutchLeadingEdge) 3: Clutch command trailing edge (ClutchTrailingEdge) 4: Address mode (ClutchAddress) 15: I/O data specification (ClutchSignal)

When the clutch ON condition and the clutch OFF condition are completed simultaneously within one operation cycle, both clutch ON and OFF processing are executed within one operation cycle. Therefore, the clutch changes from OFF to ON and again to OFF at the clutch OFF status, and changes from ON to OFF and again to ON at the clutch ON status.

The clutch ON/OFF operations for clutch ON and OFF control settings are shown below.

■ Main shaft clutch ON control setting (AdvOutputName.Pr.Clutch.MasterOnControl)/Auxiliary shaft clutch ON control setting (AdvOutputName.Pr.Clutch.AuxOnControl)

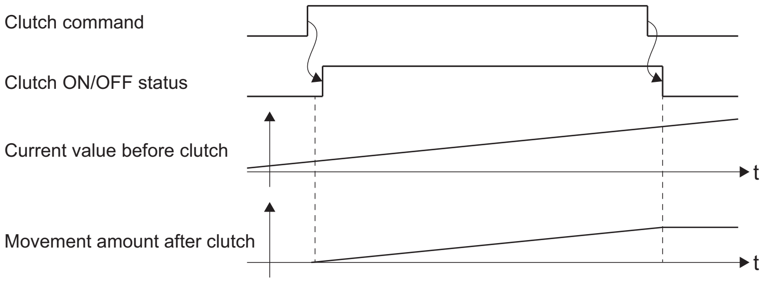
- Invalid (Direct coupled operation)

  Execute direct coupled operation without clutch control.

Point

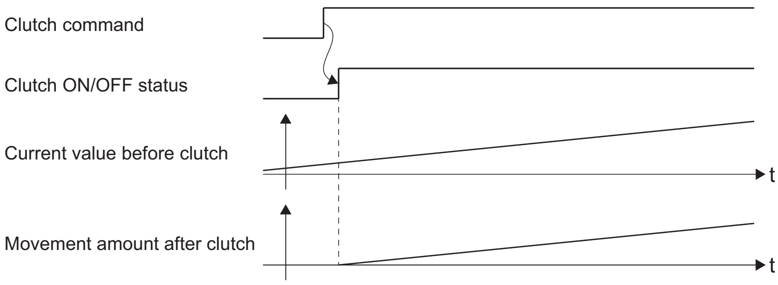
- Other clutch parameters are not applicable during direct coupled operation by setting the Main shaft clutch ON control setting (AdvOutputName.Pr.Clutch.MasterOnControl) and the Auxiliary shaft clutch ON control setting (AdvOutputName.Pr.Clutch.AuxOnControl) to "0: Invalid (NoClutch) (Direct coupled operation)". The "clutch forced OFF command" and the change of the clutch control setting are ignored during direct coupled operation.
- Changes from settings other than "0: Invalid (NoClutch) (Direct coupled operation)" to "0: Invalid (NoClutch) (Direct coupled operation)" are not possible. If the setting is changed in that manner, "Out of Advanced Synchronous Control Variable Range Warning (error code: 0D3FH)" occurs.

- Clutch command
  The clutch is turned ON/OFF by the TRUE/FALSE operation of "clutch command". (When "1: Clutch command (ClutchCommand)" is set in the Main shaft clutch ON control setting (AdvOutputName.Pr.Clutch.MasterOnControl) and the Auxiliary shaft clutch ON control setting (AdvOutputName.Pr.Clutch.AuxOnControl), the settings of the Main shaft clutch OFF control setting (AdvOutputName.Pr.Clutch.MasterOffControl) and the Auxiliary shaft clutch OFF control setting (AdvOutputName.Pr.Clutch.AuxOffControl) are not applicable.)

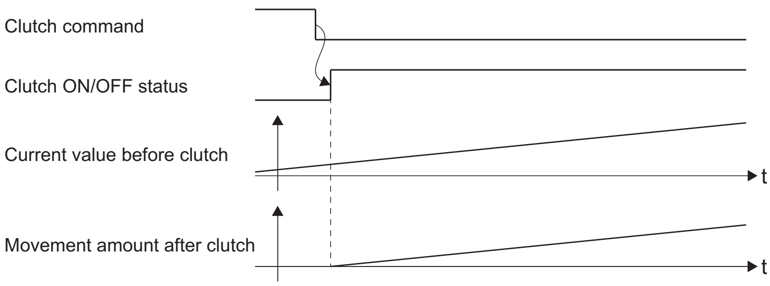


Item	Main shaft clutch	Auxiliary shaft clutch
Clutch command	Main shaft clutch command (AdvOutputName.Cd.Clutch.MasterCommand)	Auxiliary shaft clutch command (AdvOutputName.Cd.Clutch.AuxCommand)
Clutch ON/OFF status	Main shaft clutch ON/OFF status (AdvOutputName.Md.Clutch.MasterOnOffStatus)	Auxiliary shaft clutch ON/OFF status (AdvOutputName.Md.Clutch.AuxOnOffStatus)

- Clutch command leading edge
  The clutch is turned ON when the "clutch command" passes the leading edge (from FALSE to TRUE).

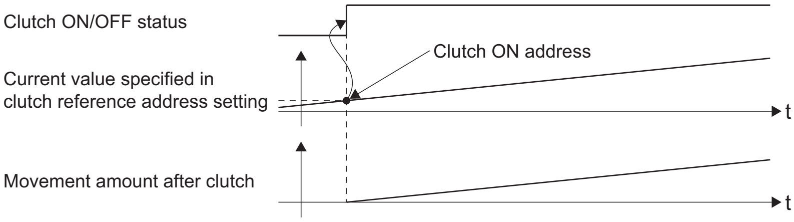


- Clutch command trailing edge
  The clutch is turned ON when the "clutch command" passes the trailing edge (from TRUE to FALSE).



- Address mode

  The clutch is turned ON when the "reference address" reaches "Clutch ON address". The movement amount after passing through the ON address is calculated as the output movement amount of the clutch based on the reference address passing through, thereby controlling the clutch with an accurate movement amount.



Item	Main shaft clutch	Auxiliary shaft clutch
Reference address	The current position specified in the Main shaft clutch reference address setting (AdvOutputName.Pr.Clutch.MasterReference) • 0: Current position before gear (GearFrontPosition)*1 • 1: Current position per cycle after gear (GearPositionPerCycle)*2	The current position specified in the Auxiliary shaft clutch reference address setting (AdvOutputName.Pr.Clutch.AuxReference) • 0: Current position before gear (GearFrontPosition)*3 • 1: Current position per cycle after gear (GearPositionPerCycle)*4
Clutch ON address	Main shaft clutch ON address (AdvOutputName.Pr.Clutch.MasterOnAddress)	Auxiliary shaft clutch ON address (AdvOutputName.Pr.Clutch.AuxOnAddress)
Clutch ON/OFF status	Main shaft clutch ON/OFF status (AdvOutputName.Md.Clutch.MasterOnOffStatus)	Auxiliary shaft clutch ON/OFF status (AdvOutputName.Md.Clutch.AuxOnOffStatus)

*1 As a monitor, the Current position after composite main shaft gear (AdvOutputName.Md.Cam.MasterCompositeGearSetPosition) is referred.

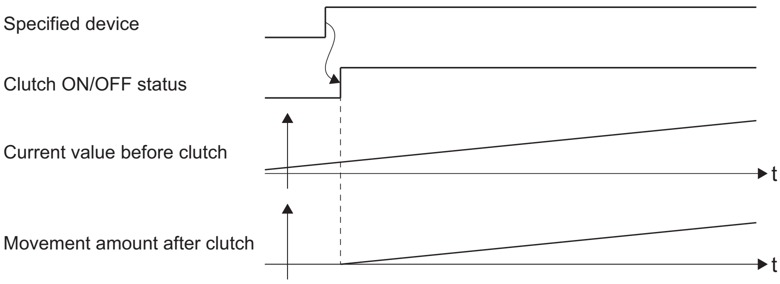
*2 As a monitor, the Current position per cycle after main shaft gear (AdvOutputName.Md.Cam.MasterGearPositionPerCycle) is referred.

*3 As a monitor, the cumulative current position of the auxiliary shaft (the Cumulative current position (AdvOutputName.Md.CumulativePosition) of the input axis set in the auxiliary shaft) is referred.

*4 As a monitor, the Current position per cycle after auxiliary shaft gear (AdvOutputName.Md.Cam.AuxGearPositionPerCycle) is referred.

- I/O data specification

  The clutch is turned ON at an input signal detection.



When using an external signal, set the following clutch ON signal settings.

Item	Main shaft	Auxiliary shaft
Clutch ON signal setting	Main shaft clutch ON signal setting (AdvOutputName.Pr.MasterOnClutchSignal)	Auxiliary shaft clutch ON signal setting (AdvOutputName.Pr.AuxOnClutchSignal)

The type for the clutch ON signal setting is SIGNAL_SELECT (Signal Select). The setting range of the clutch ON signal setting is as follows.

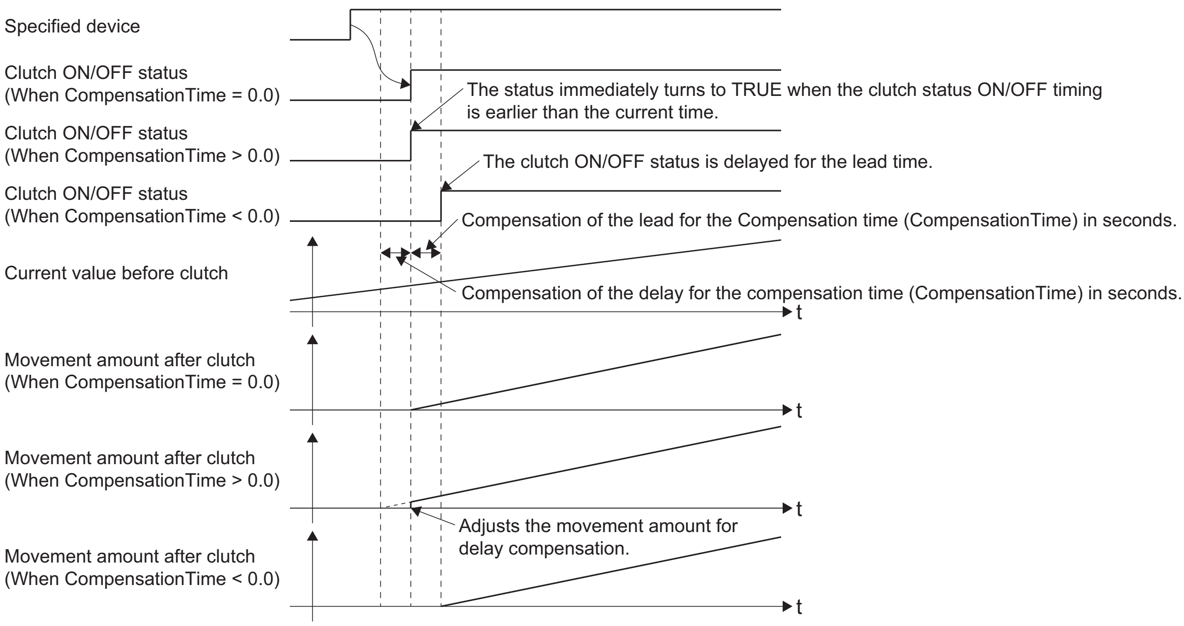
Structure	Variable name	Type	Setting range*1
SIGNAL_SELECT (Signal Select)	Source (Signal). Target (Target)	TARGET_REF	■Type • BOOL ■Data type • [OBJ]*2*3 • [VAR] • [DEV]
	Detection (Signal detection method)	INT (MC_SIGNAL_LOGIC)	• 2: Detection at FALSE → TRUE (rising edge) (RisingEdge) • 3: Detection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

*1 In the following cases, "Out of Parameter Range (Advanced Synchronous Control Output Axis) (error code: 1D86H)" occurs.
· The target modification is omitted when [OBJ] is specified as data type
· Source.Target is omitted or the target does not exist
· The set value in the Signal detection method (Detection), the Compensation time (CompensationTime), or the Filter time (FilterTime) sout of range

*2 When the referred station has the axis emulate function enabled and also has a set station address, the emulating object will be referred.

*3 The external signal high-accuracy input is available. When using the external high-accuracy input, set the Detection (Signal detection method) to "4: Detection at rising edge/falling edge (BothEdges)". For details, refer to the following.
☞ Page 381 External Signal Selection
For the high-accuracy input settings, refer to the following.
☞ Page 831 Connectable device to CC-Link IE TSN

*4 The detection time of an input signal can be compensated by setting the Compensation time (CompensationTime). Set a positive value to compensate the delay of the input signal, and set a negative value to compensate the lead. As shown below, the clutch ON/OFF status varies depending on the compensation time setting. When delay is compensated with the Compensation time (CompensationTime), the movement amount after clutch may change rapidly. Adjust so that the movement amount changes gradually with the smoothing function of the clutch and output axis.



15

■ Main shaft clutch OFF control setting (AdvOutputName.Pr.Clutch.MasterOffControl)/Auxiliary shaft clutch OFF control setting (AdvOutputName.Pr.Clutch.AuxOffControl)

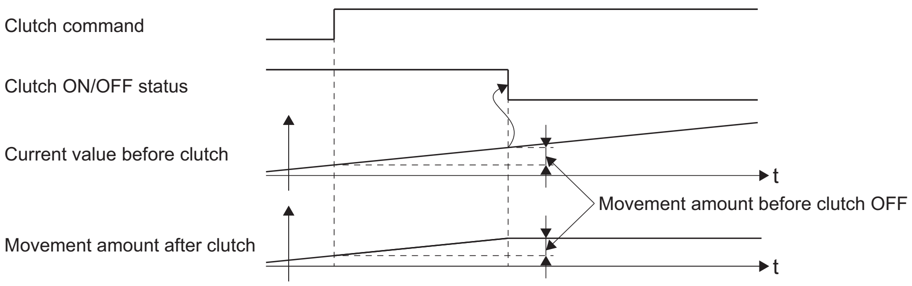
- Invalid (OFF control invalid)

  Clutch OFF control is not used. This setting is applicable only for execution with clutch ON control.

- Clutch command (Oneshot operation)

  The clutch is turned OFF after moving the distance "Movement amount before clutch OFF" after the "clutch ON/OFF status" changes from FALSE to TRUE.

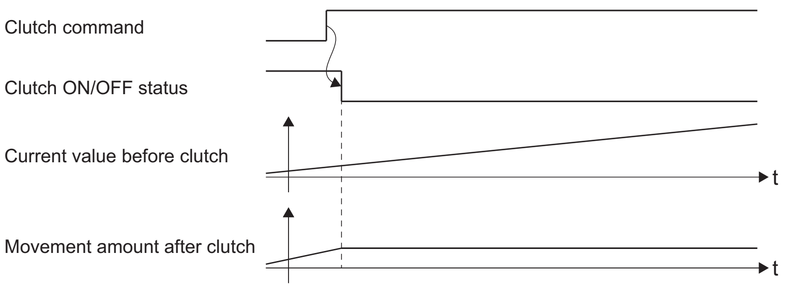
  If the "Movement amount before clutch OFF" is "0.0", the "Clutch ON/OFF status" does not turn ON in order to turn back OFF immediately.



Item	Main shaft clutch	Auxiliary shaft clutch
Clutch command	Main shaft clutch command (AdvOutputName.Cd.Clutch.MasterCommand)	Auxiliary shaft clutch command (AdvOutputName.Cd.Clutch.AuxCommand)
Clutch ON/OFF status	Main shaft clutch ON/OFF status (AdvOutputName.Md.Clutch.MasterOnOffStatus)	Auxiliary shaft clutch ON/OFF status (AdvOutputName.Md.Clutch.AuxOnOffStatus)
Movement amount before clutch OFF	Movement amount before main shaft clutch OFF (AdvOutputName.Pr.Clutch.MasterMovementAmountBeforeOff)	Movement amount before auxiliary shaft clutch OFF (AdvOutputName.Pr.Clutch.AuxMovementAmountBeforeOff)

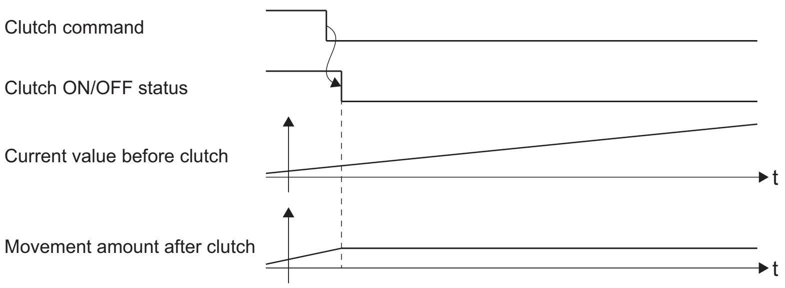
- Clutch command leading edge

  The clutch is turned OFF when the "clutch command" passes the leading edge (from FALSE to TRUE).



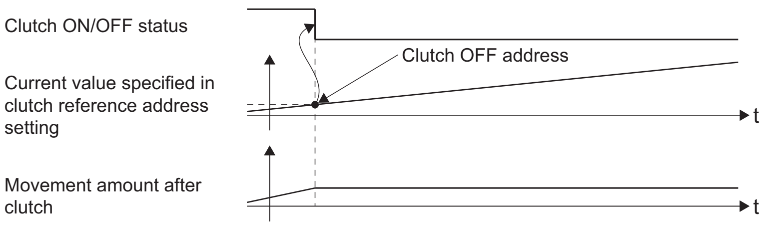
- Clutch command trailing edge

  The clutch is turned OFF when the "clutch command" passes the trailing edge (from TRUE to FALSE).



• Address mode

The clutch is turned OFF when the "reference address" reaches "Clutch OFF address". The movement amount before passing through the OFF address is calculated as the output movement amount of the clutch based on the reference address passing through, thereby controlling the clutch with an accurate movement amount.



Item	Main shaft clutch	Auxiliary shaft clutch
Reference address	The current position specified in the Main shaft clutch reference address setting (AdvOutputName.Pr.Clutch.MasterReference) • 0: Current position before gear (GearFrontPosition)*1 • 1: Current position per cycle after gear (GearPositionPerCycle)*2	The current position specified in the Auxiliary shaft clutch reference address setting (AdvOutputName.Pr.Clutch.AuxReference) • 0: Current position before gear (GearFrontPosition)*3 • 1: Current position per cycle after gear (GearPositionPerCycle)*4
Clutch OFF address	Main shaft clutch OFF address (AdvOutputName.Pr.Clutch.MasterOffAddress)	Auxiliary shaft clutch OFF address (AdvOutputName.Pr.Clutch.AuxOffAddress)
Clutch ON/OFF status	Main shaft clutch ON/OFF status (AdvOutputName.Md.Clutch.MasterOnOffStatus)	Auxiliary shaft clutch ON/OFF status (AdvOutputName.Md.Clutch.AuxOnOffStatus)

*1 As a monitor, the Current position after composite main shaft gear (AdvOutputName.Md.Cam.MasterCompositeGearSetPosition) is referred.

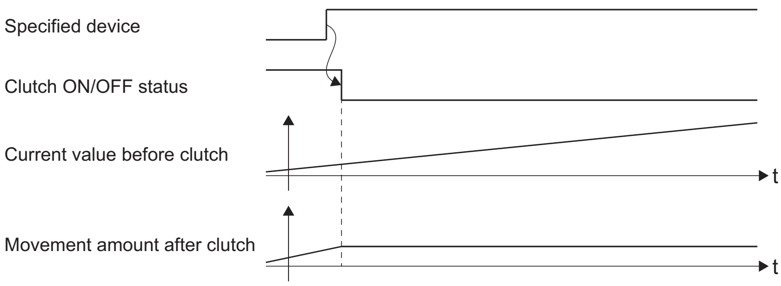
*2 As a monitor, the Current position per cycle after main shaft gear (AdvOutputName.Md.Cam.MasterGearPositionPerCycle) is referred.

*3 As a monitor, the cumulative current position of the auxiliary shaft (the Cumulative current position (AdvInputName.Md.CumulativePosition) of the input axis set in the auxiliary shaft) is referred.

*4 As a monitor, the Current position per cycle after auxiliary shaft gear (AdvOutputName.Md.Cam.AuxGearPositionPerCycle) is referred.

• I/O data specification

The clutch is turned OFF at an input signal detection.



When using an external signal for I/O data specification, set the following clutch OFF signal settings

Item	Main shaft clutch	Auxiliary shaft clutch
Clutch OFF signal setting	Main shaft clutch OFF signal setting (AdvOutputName.PrConst.MasterOffClutchSignal)	Auxiliary shaft clutch OFF signal setting (AdvOutputName.PrConst.AuxOffClutchSignal)

15

The type for the OFF clutch signal setting is SIGNAL_SELECT (Signal select). The setting range of the clutch signal setting is as follows.

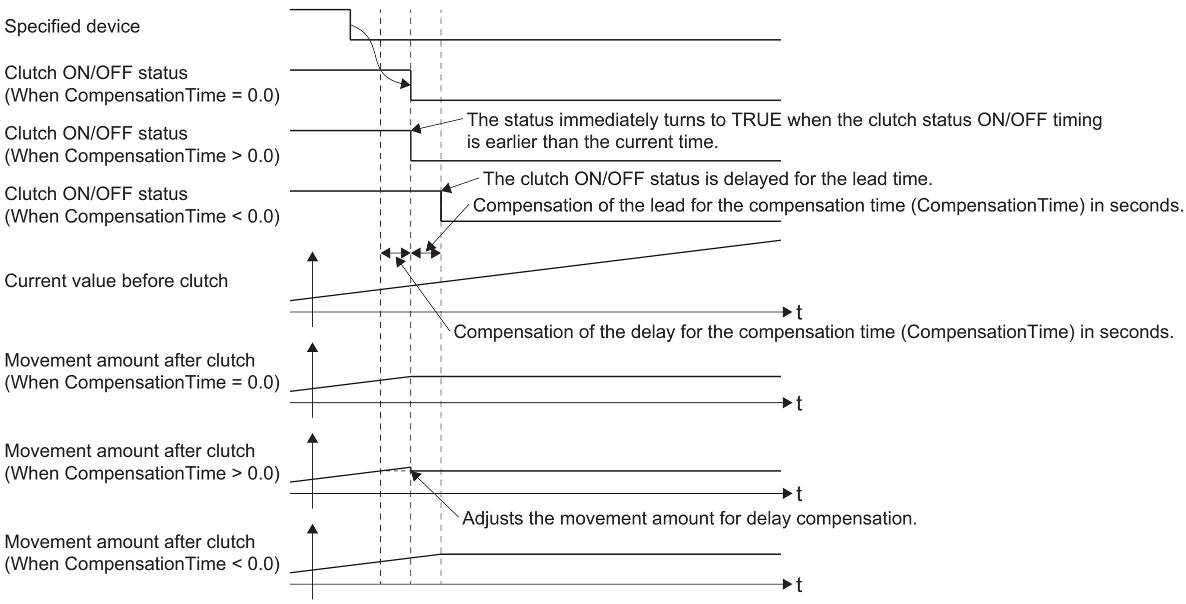
Structure	Variable name	Type	Setting range*1
SIGNAL_SELECT (Signal Select)	Source (Signal)	TARGET_REF	■Type • BOOL ■Data type • [OBJ]*2*3 • [VAR] • [DEV]
	Detection (Signal detection method)	INT (MC_SIGNAL_LOGIC)	• 2: Detection at FALSE → TRUE (rising edge) (RisingEdge) • 3: Detection at TRUE → FALSE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

*1 In the following cases, "Out of Parameter Range (Advanced Synchronous Control Output Axis) (error code: 1D86H)" occurs.
· The target modification is omitted when [OBJ] is specified as data type
· Source.Target is omitted or the target does not exist
· The set value in the Signal detection method (Detection), the Compensation time (CompensationTime), or the Filter time (FilterTime) is out of range

*2 When the referred station has the axis emulate function enabled and also has a set station address, the emulating object will be referred.

*3 The external signal high-accuracy input is available. When using the external high-accuracy input, set the Detection (Signal detection method) to "4: Detection at rising edge/falling edge (BothEdges)". For details, refer to the following.
☞ Page 381 External Signal Selection
For the high-accuracy input setting, refer to the following.
☞ Page 831 Connectable device to CC-Link IE TSN

*4 The detection time of an input signal can be compensated by setting the Compensation time (CompensationTime). Set a positive value to compensate the delay of the input signal, and set a negative value to compensate the lead. As shown below, the clutch ON/OFF status varies depending on the compensation time setting. When delay is compensated with the Compensation time (CompensationTime), the movement amount after clutch may change rapidly. Adjust so that the movement amount changes gradually with the smoothing function of the clutch and output axis.



Smoothing method for clutch

Set the clutch smoothing method in Main shaft clutch smoothing method (AdvOutputName.Pr.Clutch.MasterSmoothingMethod) and "Auxiliary shaft clutch smoothing method (AdvOutputName.Pr.Clutch.AuxSmoothingMethod)".

The following two types of clutch smoothing are available.

- Time constant method smoothing
- Slippage method smoothing

When not using clutch smoothing, set "0: Direct" in the clutch smoothing method.

Item	Setting item		Setting value
	Main shaft clutch	Auxiliary shaft clutch	
Clutch smoothing method	Main shaft clutch smoothing method (AdvOutputName.Pr.Clutch.MasterS mooth ingMethod)	Auxiliary shaft clutch smoothing method (AdvOutputName.Pr.Clutch.AuxS moothingMethod)	0: Direct (ClutchSmoothingDisabled) 1: Time constant method (Exponent) (TimeConstantExponent) 2: Time constant method (Linear) (TimeConstantLinear) 3: Slippage method (Exponent) (SlippageExponent) 4: Slippage method (Linear) (SlippageLinear) 5: Slippage method (Linear: Input value follow up) (SlippageLinearFollow)

The operation of each smoothing method is shown below.

■ Time constant method smoothing

Smoothing is executed with the time constant set in "smoothing time constant" at clutch ON/OFF. After clutch ON smoothing is complete, smoothing is executed with the time constant when the speed of the input value changes.

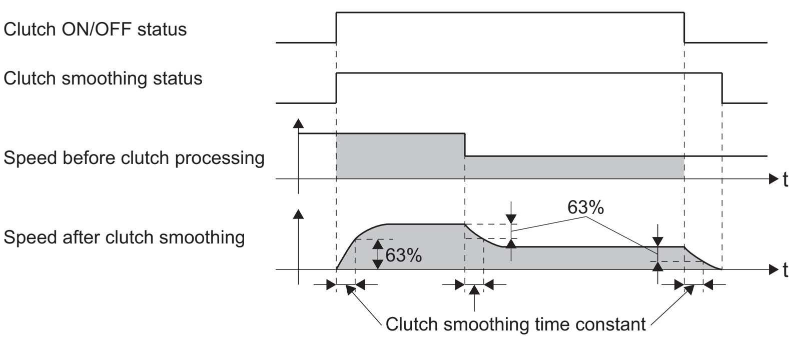
As shown below, the movement amount while the clutch changes from ON to OFF does not change after clutch smoothing.

Movement amount after clutch smoothing = Movement amount before clutch smoothing

Item	Setting item		Setting value
	Main shaft clutch	Auxiliary shaft clutch	
Clutch smoothing time constant	Main shaft clutch smoothing time constant (AdvOutputName.PrConst.MasterCl utchS moothingTimeConstant)	Auxiliary shaft clutch smoothing time constant (AdvOutputName.PrConst.AuxCl utchSmoo thingTimeConstant)	0 to 5000 [ms]

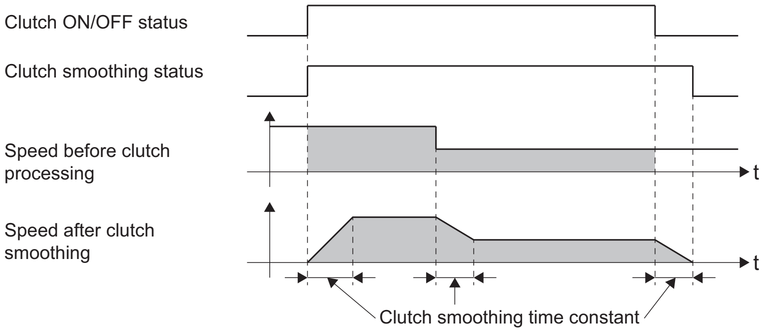
- Time constant method exponential curve smoothing

  Set "1: Time constant method (Exponential)(TimeConstantExponent)" in the "clutch smoothing method".



Item	Main shaft clutch	Auxiliary shaft clutch
Clutch ON/OFF status	Main shaft clutch ON/OFF status (AdvOutputName.Md.Clutch.MasterOnOffStatus)	Auxiliary shaft clutch ON/OFF status (AdvOutputName.Md.Clutch.AuxOnOffStatus)
Clutch smoothing status	Main shaft clutch smoothing status (AdvOutputName.Md.Clutch.MasterSmoothingStatus)	Auxiliary shaft clutch smoothing status (AdvOutputName.Md.Clutch.AuxSmoothingStatus)

- Time constant method linear acceleration/deceleration smoothing
  Set "2: Time constant method (Linear) (TimeConstantLinear)" in the "clutch smoothing method".



■ Slippage method smoothing

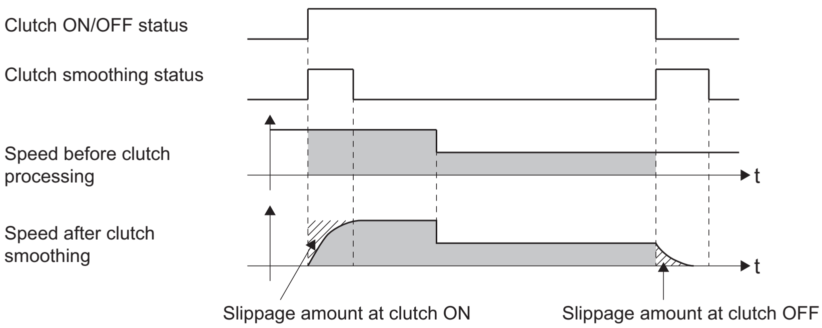
Smoothing is executed with the value in "slippage at clutch ON" when the clutch turns ON, and with "slippage at clutch OFF" when the clutch turns OFF.
Because smoothing is still executed with the set slippage amount when the input speed to the clutch changes, the position of clutch ON/OFF is not affected by speed changes and thus can be controlled.
A direct operation follows after completing clutch ON smoothing.
The movement amount while the clutch changes from ON to OFF is as follows after clutch smoothing.

Movement amount after clutch smoothing = Movement amount before clutch smoothing + (Slippage amount at OFF - Slippage amount at ON)

Item	Setting item		Setting value
	Main shaft clutch	Auxiliary shaft clutch	
Slippage amount at clutch ON	Slippage amount at main shaft clutch ON (AdvOutPutName.Pr.Clutch.MasterOnSlippageAmount)	Slippage amount at auxiliary shaft clutch ON (AdvOutPutName.Pr.Clutch.AuxOnSlippageAmount)	0 to 2147483647 [Main shaft position command units*1/auxiliary shaft position command units*2]
Slippage amount at clutch OFF	Slippage amount at main shaft clutch OFF (AdvOutPutName.Pr.Clutch.MasterOffSlippageAmount)	Slippage amount at auxiliary shaft clutch OFF (AdvOutPutName.Pr.Clutch.AuxOffSlippageAmount)	

*1 The position command unit of the axis set as the Auxiliary shaft object (AdvOutPutName.Pr.AuxAxis).
*2 position command unit of the axis set as the Auxiliary shaft object (AdvOutPutName.Pr.AuxAxis).

- Slippage method exponential curve smoothing
  Set "3: Slippage method (Exponential) (SlippageExponent)" in the "clutch smoothing method".



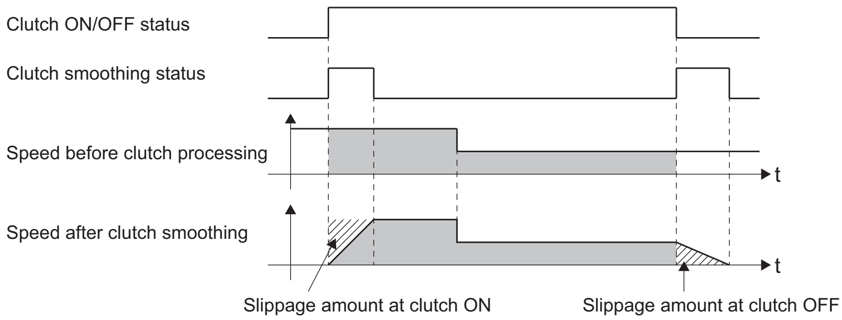
Item	Main shaft clutch	Auxiliary shaft clutch
Clutch ON/OFF status	Main shaft clutch ON/OFF status (AdvOutPutName.Md.Clutch.MasterOnOffStatus)	Auxiliary shaft clutch ON/OFF status (AdvOutPutName.Md.Clutch.AuxOnOffStatus)
Clutch smoothing status	Main shaft clutch smoothing status (AdvOutPutName.Md.Clutch.MasterSmoothingStatus)	Auxiliary shaft clutch smoothing status (AdvOutPutName.Md.Clutch.AuxSmoothingStatus)

- Slippage method linear acceleration/deceleration smoothing
  Set "4: Slippage method (Linear) (SlippageLinear)", or "5: Slippage method (Linear: Input value follow up (SlippageLinearFollow)" in the "clutch smoothing method".

Input speed during smoothing	Smoothing method	
	4: Slippage method (Linear) (SlippageLinear)	**5: Slippage method (Linear: Input value follow up) (SlippageLinearFollow)**
When input speed is fixed	No difference	
When there is a continuous small fluctuation in the input speed	The smoothing sections change.	The smoothing section is fixed.
When input speed has a large fluctuation	The change in the output speed is small. (The average speed may rise higher than before the start of smoothing)	The output speed changes with the input speed. (When re-accelerating after the input speed drops, a sudden acceleration may occur)

■When input speed is fixed

The operation for "4: Slippage method (Linear) (SlippageLinear)" and "5: Slippage method (Linear: Input value follow up) (SlippageLinearFollow)" is the same.

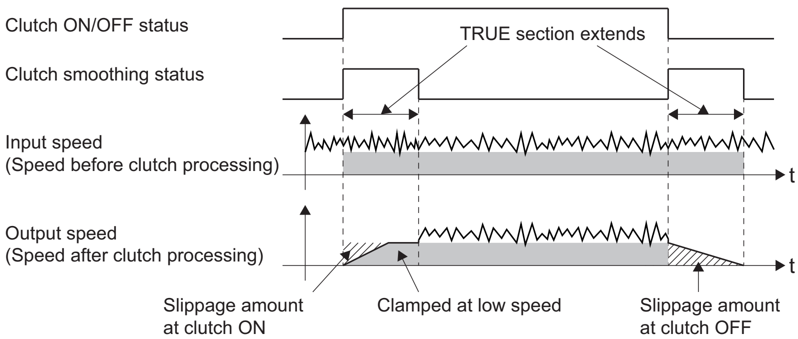


■When there is a continuous small fluctuation in the input speed

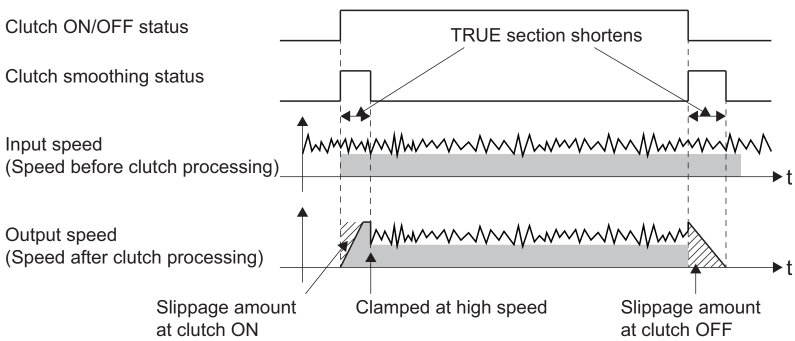
- When "4: Slippage method (Linear) (SlippageLinear)" is set

  The clutch smoothing status TRUE section changes with the fluctuation of the input speed.

  - When the clutch smoothing status TRUE section is extended

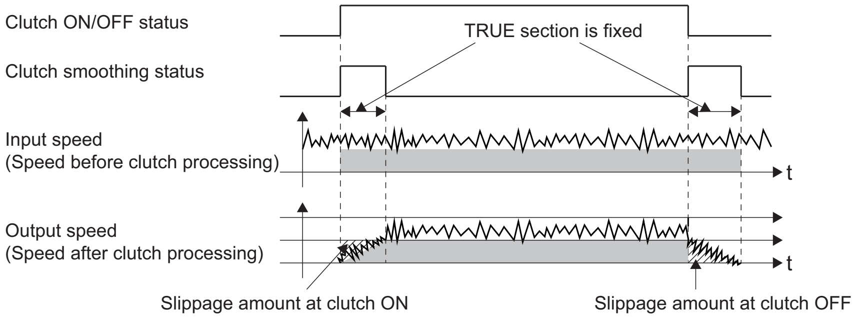


  - When the clutch smoothing status TRUE section is shortened



• When "5: Slippage method (Linear: Input value follow up) (SlippageLinearFollow)" is set

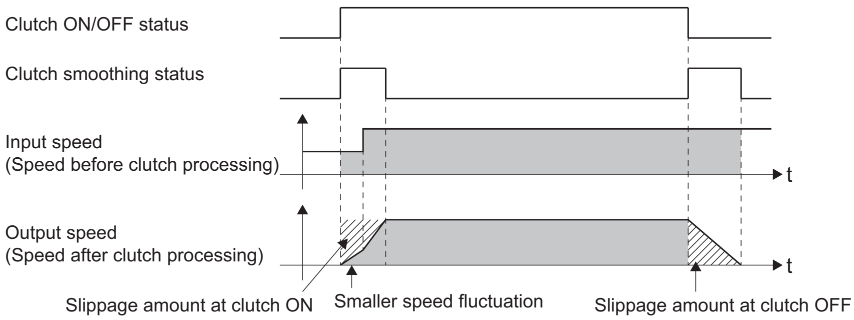
The clutch smoothing status TRUE section is fixed.



■When input speed has a large fluctuation

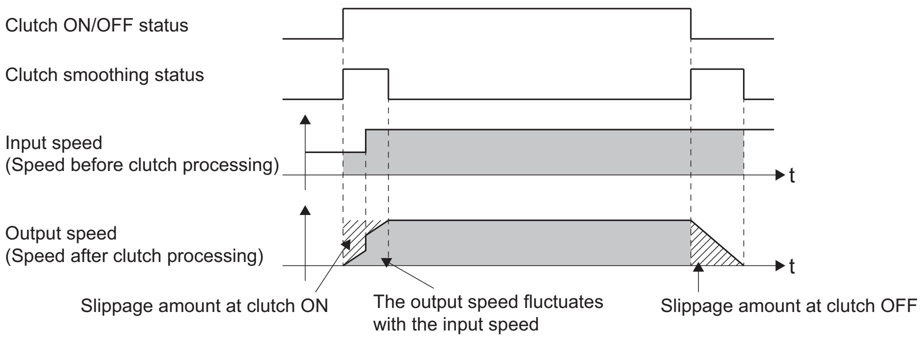
• When "4: Slippage method (Linear) SlippageLinear)" is set

The fluctuation in the output speed is smaller compared to the fluctuation in the input speed. (The average speed may rise higher than before the start of smoothing)



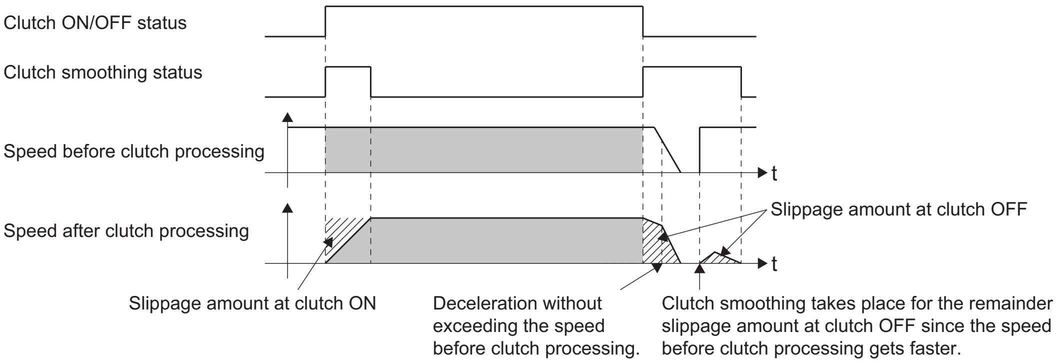
• When "5: Slippage method (Linear: Input value follow up) (SlippageLinearFollow)" is set

The output speed fluctuates with the input speed. (When re-accelerating after the input speed drops, a sudden acceleration may occur)



■ Operation at input speed deceleration during slippage method smoothing

When the speed before clutch processing decreases, the speed after clutch smoothing is controlled to keep it from exceeding the speed before clutch processing. If slippage amount remains when the speed before clutch processing becomes "0.0", the smoothing process will continue. When the speed before clutch processing gets faster than the speed after clutch smoothing again, clutch smoothing will be executed for the remainder slippage amount.



Example of clutch use

For an example of clutch use, refer to the following.

☞ Page 614 Flying cutter

15

Speed change gear

A speed change gear is used to change the input speed from the main shaft, auxiliary shaft, composite auxiliary shaft gear during operation. One speed change gear can be set to a main shaft module, an auxiliary shaft module, and an output axis module respectively.

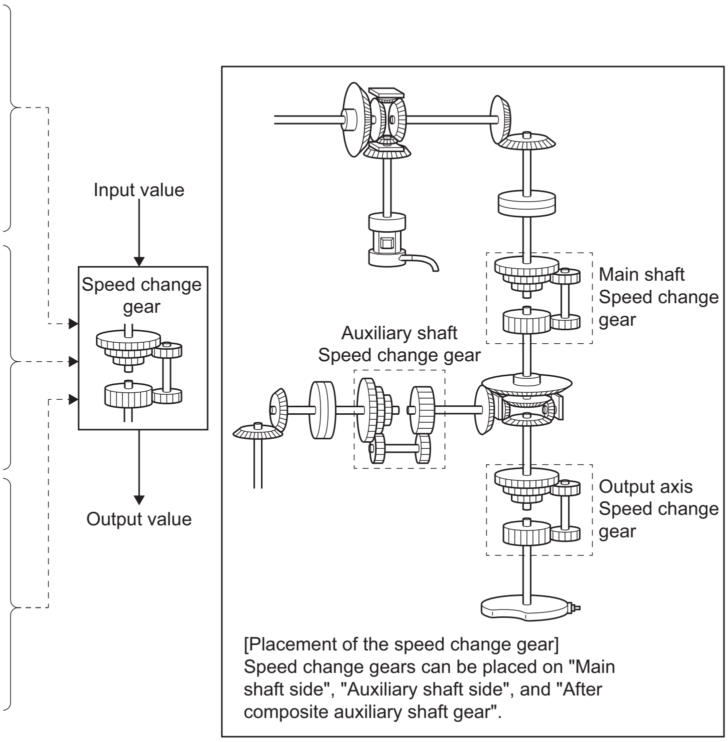
When not using a speed change gear, set "FALSE (Without speed change gear)".

Speed changes from a speed change gear are executed as linear acceleration/deceleration using the time set for the speed change gear smoothing time constant.

[Main shaft]
- Main shaft speed change gear valid setting (AdvOutPutName.Pr.SpeedChangeGear.MasterGearIn)
- Main shaft speed change gear smoothing time constant (AdvOutPutName.PrConst.MasterSpeedChangeGearSmoothingTimeConstant)
- Main shaft speed change ratio: Numerator (AdvOutPutName.Pr.SpeedChangeGear.MasterRatioNumerator)
- Main shaft speed change ratio: Denominator (AdvOutPutName.Pr.SpeedChangeGear.MasterRatioDenominator)

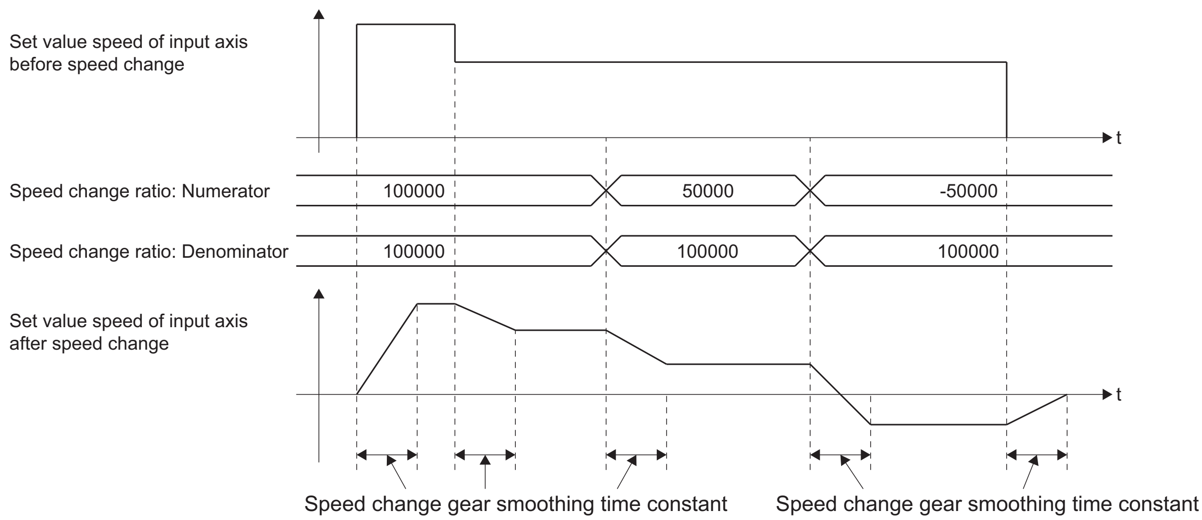
[Auxiliary shaft]
- Auxiliary shaft speed change gear valid setting (AdvOutPutName.Pr.SpeedChangeGear.AuxGearIn)
- Auxiliary shaft speed change gear smoothing time constant (AdvOutPutName.PrConst.AuxSpeedChangeGearSmoothingTimeConstant)
- Auxiliary shaft speed change ratio: Numerator (AdvOutPutName.Pr.SpeedChangeGear.AuxRatioNumerator)
- Auxiliary shaft speed change ratio: Denominator (AdvOutPutName.Pr.SpeedChangeGear.AuxRatioDenominator)

[Output axis]
- Output axis speed change gear valid setting (AdvOutPutName.Pr.SpeedChangeGear.OutGearIn)
- Output axis speed change gear smoothing time constant (AdvOutPutName.PrConst.OutSpeedChangeGearSmoothingTimeConstant)
- Output axis speed change ratio: Numerator (AdvOutPutName.Pr.SpeedChangeGear.OutRatioNumerator)
- Output axis speed change ratio: Denominator (AdvOutPutName.Pr.SpeedChangeGear.OutRatioDenominator)



Input values for speed change are processed as follows.

$$\text{Input value after conversion} = \text{Input value before conversion} \times \frac{\text{Speed change ratio: Numerator}}{\text{Speed change ratio: Denominator}}$$



Speed change gear parameter of output axis parameters (AdvOutPutName.Pr.SpeedChangeGear.)

Variable name	Name	Import[*1]	Data type[*1]	Setting value	Initial value
MasterGearIn	Main shaft speed change gear valid setting	At start	BOOL	FALSE: Without speed change gear TRUE: With speed change gear	FALSE
MasterRatioNumerator	Main shaft speed change ratio: Numerator	Operation cycle (FB is starting)	DINT	-2147483647 to 2147483647	1
MasterRatioDenominator	Main shaft speed change ratio: Denominator		DWORD(UDINT)	1 to 2147483647	1
AuxGearIn	Auxiliary shaft speed change gear valid setting	At start	BOOL	FALSE: Without speed change gear TRUE: With speed change gear	FALSE
AuxRatioNumerator	Auxiliary shaft speed change ratio: Numerator	Operation cycle (FB is starting)	DINT	-2147483647 to 2147483647	1
AuxRatioDenominator	Auxiliary shaft speed change ratio: Denominator		DWORD(UDINT)	1 to 2147483647	1
OutGearIn	Output axis speed change gear valid setting	At start	BOOL	FALSE: Without speed change gear TRUE: With speed change gear	FALSE
OutRatioNumerator	Output axis speed change ratio: Numerator	Operation cycle (FB is starting)	DINT	-2147483647 to 2147483647	1
OutRatioDenominator	Output axis speed change ratio: Denominator		DWORD(UDINT)	1 to 2147483647	1

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Main shaft speed change gear valid setting (AdvOutputName.Pr.SpeedChangeGear.MasterGearIn)/Auxiliary shaft speed change gear valid setting (AdvOutputName.Pr.SpeedChangeGear.AuxGearIn)/Output axis speed change gear valid setting (AdvOutputName.Pr.SpeedChangeGear.OutGearIn)

Set to whether to use or not use a speed change gear for each axis.

Setting value	Description
FALSE (Without speed change gear)	Speed change is not processed, and the input value is transmitted without conversion.
TRUE (With speed change gear)	Speed change is processed for the input value based on the speed change ratio settings.

■ Main shaft speed change ratio: Numerator (AdvOutputName.Pr.SpeedChangeGear.MasterRatioNumerator)/Auxiliary shaft speed change ratio: Numerator (AdvOutputName.Pr.SpeedChangeGear.AuxRatioNumerator)/Output axis speed change ratio: Numerator (AdvOutputName.Pr.SpeedChangeGear.OutRatioNumerator)

Set the numerator for the speed change ratio for each axis.

Speed change ratio: Numerator can be changed anytime during synchronous control.

Input values for speed change are processed as follows.

$$\text{Input value after conversion} = \text{Input value before conversion} \times \frac{\text{Speed change ratio: Numerator}}{\text{Speed change ratio: Denominator}}$$

The input speed direction can be reversed by setting a negative value in Speed change ratio: Numerator.

Point

Setting a reduced fraction is recommend for "Main shaft speed change ratio: Numerator/Main shaft Speed change ratio: Denominator", "Auxiliary shaft speed change ratio: Numerator/Auxiliary shaft Speed change ratio: Denominator", and "Output axis speed change ratio: Numerator/Output axis speed change ratio: Denominator".

15

■ Main shaft Speed change ratio: Denominator (AdvOutputName.Pr.SpeedChangeGear.MasterRatioDenominator)/Auxiliary shaft Speed change ratio: Denominator (AdvOutputName.Pr.SpeedChangeGear.AuxRatioDenominator)/ Output axis speed change ratio: Denominator (AdvOutputName.Pr.SpeedChangeGear.OutRatioDenominator)

Set the denominator for the speed change ratio for each axis.

Speed change ratio: Denominator can be changed anytime during synchronous control.

Set together with Speed change ratio: Numerator.

Speed change ratio: Denominator is set within the range from "1 to 2147483647".

Output axis parameters constant (AdvOutPutName.PrConst.)

Variable name	Name	Import*1	Data type*1	Setting value	Initial value
MasterSpeedChangeGearSmoothingTimeConstant	Main shaft speed change gear smoothing time constant	At system start	WORD(UNIT)	0 to 5000 [ms]	0 [ms]
AuxSpeedChangeGearSmoothingTimeConstant	Auxiliary shaft speed change gear smoothing time constant		WORD(UNIT)	0 to 5000 [ms]	0 [ms]
OutSpeedChangeGearSmoothingTimeConstant	Output axis speed change gear smoothing time constant		WORD(UNIT)	0 to 5000 [ms]	0 [ms]

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Main shaft speed change gear smoothing time constant (AdvOutputName.PrConst.MasterSpeedChangeGearSmoothingTimeConstant)/Auxiliary shaft speed change gear smoothing time constant (AdvOutputName.PrConst.AuxSpeedChangeGearSmoothingTimeConstant)/ Output axis speed change gear smoothing time constant (AdvOutputName.PrConst.OutSpeedChangeGearSmoothingTimeConstant)

Set the averaging time to execute a smoothing process for the speed change for the speed change gear.

The input value transmission is delayed depending on the time corresponding to the speed change gear smoothing time constant.

Speed is changed directly when "0" is set.

Speed change gear control data of output axis parameters (AdvOutPutName.Cd.SpeedChangeGear.)

Variable name	Name	Import*1	Data type*1	Setting value	Initial value
MasterSmoothingTimeConstant	Main shaft speed change gear smoothing time constant change value	At start	INT	0 to 5000 [ms]	0 [ms]
AuxSmoothingTimeConstant	Auxiliary shaft speed change gear smoothing time constant change value		INT	0 to 5000 [ms]	0 [ms]
OutSmoothingTimeConstant	Output axis speed change gear smoothing time constant change value		INT	0 to 5000 [ms]	0 [ms]

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Main shaft speed change gear smoothing time constant change value (AdvOutPutName.Cd.SpeedChangeGear.MasterSmoothingTimeConstant)/Auxiliary shaft speed change gear smoothing time constant change value (AdvOutPutName.Cd.SpeedChangeGear.AuxSmoothingTimeConstant)/Output axis speed change gear smoothing time constant change value (AdvOutPutName.Cd.SpeedChangeGear.OutSmoothingTimeConstant)

The speed change gear smoothing time constant of each shaft/axis is changed.

The setting is imported when MCv_AdvancedSync (Advanced Synchronous Control) is started.

Set as follows.

Setting value	Description
When a negative value is set	No smoothing.
When 0 is set	Smoothing based on the setting value of the speed change gear smoothing time constant*1.
When a positive value is set	Smoothing based on the setting value of the speed change gear smoothing time constant change value*2. Set within the following setting range. Speed change gear smoothing time constant*1 ≤ Speed change gear smoothing time constant change value*2 *: If the set value exceeds the range above, "Out of Advanced Synchronous Control Variable Range Warning (error code: 0D3FH)" occurs, and smoothing is executed based on the setting value of the speed change gear smoothing time constant*1.

*1 The speed change gear smoothing time constant is as follows.

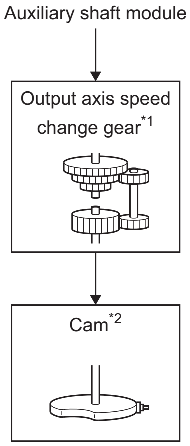
Axis	Variables
Main shaft	Main shaft speed change gear smoothing time constant (AdvOutPutName.PrConst.MasterSpeedChangeGearSmoothingTimeConstant)
Auxiliary shaft	Auxiliary shaft speed change gear smoothing time constant (AdvOutPutName.PrConst.AuxSpeedChangeGearSmoothingTimeConstant)
Output Axis	Output axis speed change gear smoothing time constant (AdvOutPutName.PrConst.OutSpeedChangeGearSmoothingTimeConstant)

*2 The speed change gear smoothing time constant change value is as follows.

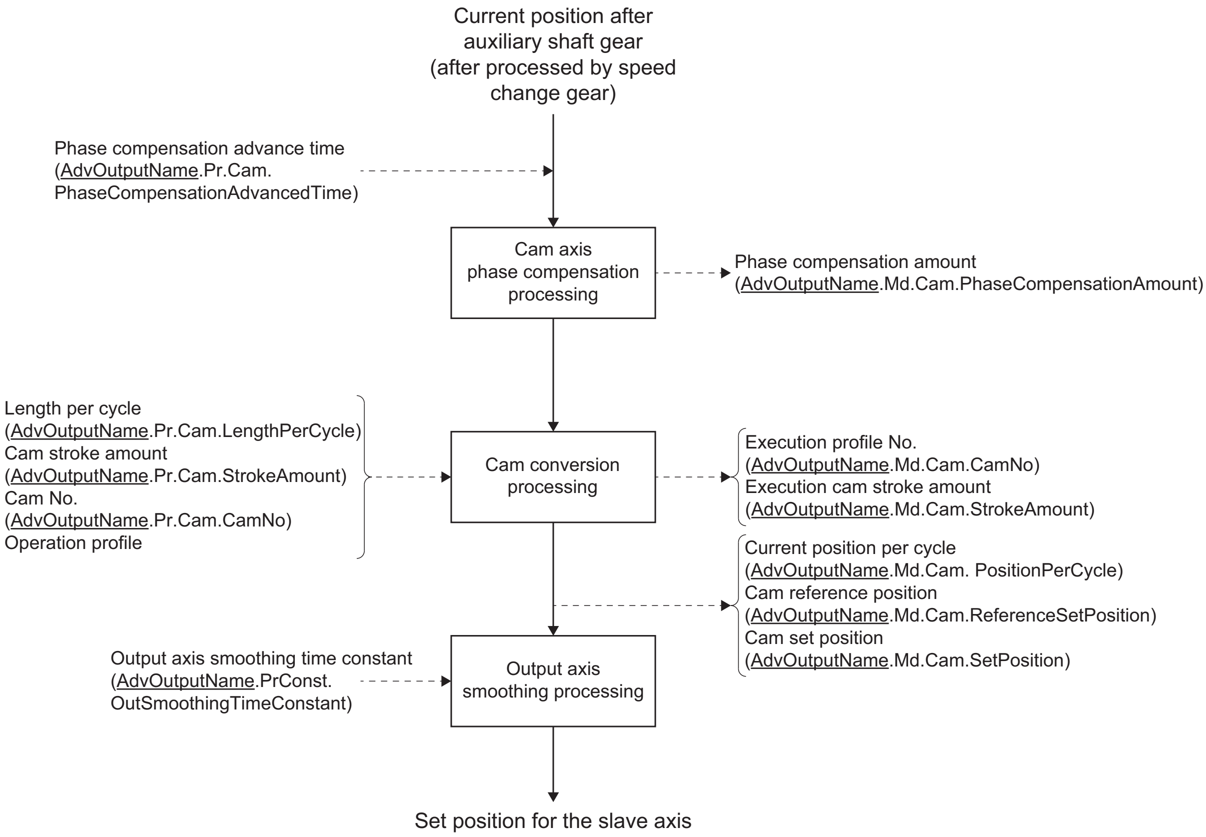
Axis	Variables
Main shaft	Main shaft speed change gear smoothing time constant change value (AdvOutPutName.Cd.SpeedChangeGear.MasterSmoothingTimeConstant)
Auxiliary shaft	Auxiliary shaft speed change gear smoothing time constant change value (AdvOutPutName.Cd.SpeedChangeGear.AuxSmoothingTimeConstant)
Output Axis	Output axis speed change gear smoothing time constant change value (AdvOutPutName.Cd.SpeedChangeGear.OutSmoothingTimeConstant)

Output axis module

For the output axis module, the current position per cycle is calculated based on the input value (the cam set position based on the current position after auxiliary shaft gear after speed change gear processing), which is then converted based on the cam data settings and output to the slave axes as the set position.
In addition, smoothing and phase compensation can be set for the output command.



*1 Speed change gear (☞ Page 558 Speed change gear)
*2 Cam function processing
The parameters and monitor data related to the cam function processing are shown below.



Cam parameter of output axis parameters (AdvOutputName.Pr.Cam.)

Variable name	Name	Import*1	Data type*1	Setting value	Initial value
LengthPerCycle	Length per cycle	At start/ At passing through the 0th point of cam data	LREAL	1.0 to 2147483647.0	1000000.0
StrokeAmount	Cam stroke amount		LREAL	-2147483648.0 to 2147483647.0	1000000.0
CamNo	Cam No.		WORD(UINT)	0 to 60000	0
StartingPoint	Cam starting point		DWORD(UDINT)	0 to 65535	0
LengthPerCycleChange	Length per cycle change setting	At start	INT (MC_LENGTH_PER_CYCLE_CHANGE)	0: Invalid (LengthChangeInvalid) 1: Valid (LengthChangeValid)	0: Invalid (LengthChangeInvalid)
PhaseCompensationAdvancedTime	Phase compensation advance time	At start/ Operation cycle (FB is starting)	DINT	-100000000 to 100000000 [µs]	0 [µs]
PhaseCompensationTimeConstant	Phase compensation time constant	At start	WORD(UINT)	0 to 65535 [ms]	10 [ms]

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

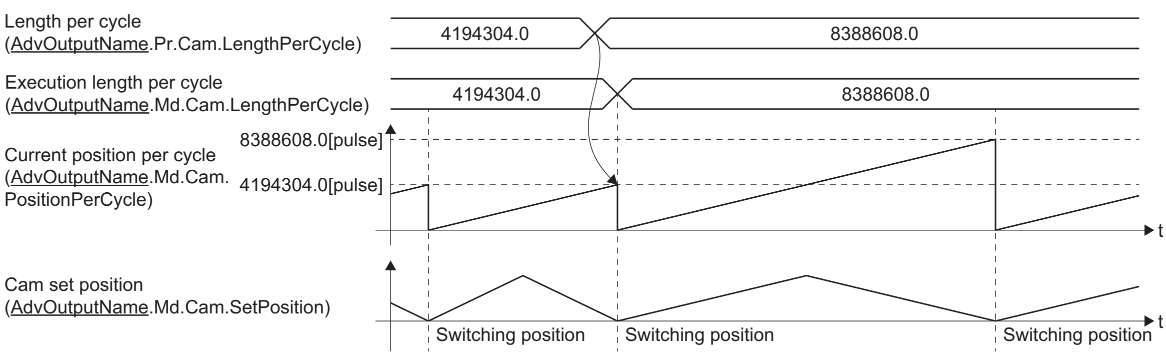
■ Length per cycle (AdvOutputName.Pr.Cam.LengthPerCycle)

Set the length per cycle of the cam axis to generate the current position per cycle.
The unit settings are in units of the input axis set as the Master axis (Master) of MCv_AdvancedSync (Advanced Synchronous Control).
Set a value within the range from "1.0 to 2147483647.0".
The length per cycle can be changed during synchronous control by setting Length per cycle change setting (AdvOutputName.Pr.Cam.LengthPerCycleChange) to "1: Valid (LengthChangeValid)". When the current position per cycle passes through the 0th point of cam data, or is at the 0th point of cam data, the value of the Length per cycle (LengthPerCycle) is imported.
When cam data is set, the waveform after the length per cycle is changed varies by the cam data format.

- For cam data (section interpolation)/cam data (spline interpolation)
  The waveform is shortened/extended so that the one cycle set in the operation profile is equal to the Length per cycle (AdvOutputName.Pr.Cam.LengthPerCycle). The maximum value of the stroke amount does not vary with the change of the length per cycle.
  An example of using cam data (section interpolation), and changing the length per cycle during synchronous control is shown below.

Ex.
A setting example of cam data (section interpolation)
· Length per cycle: 4194304.0
· Cam stroke amount: ±4194304.0
· Stroke ratio data

Section No.	Start point [pulse]	End point [pulse]	Stroke [%]
1	0.0	2097152.0	100.0
2	297152.0	4194304.0	0.0



- For cam data (linear interpolation)
  When the input value of the final coordinate is less than the length per cycle, it is controlled using a line segment calculated from the nearest two coordinates. An example of using cam data (linear interpolation), and changing the length per cycle to a value that exceeds the input value of cam data final coordinate during synchronous control is shown below.

Ex.

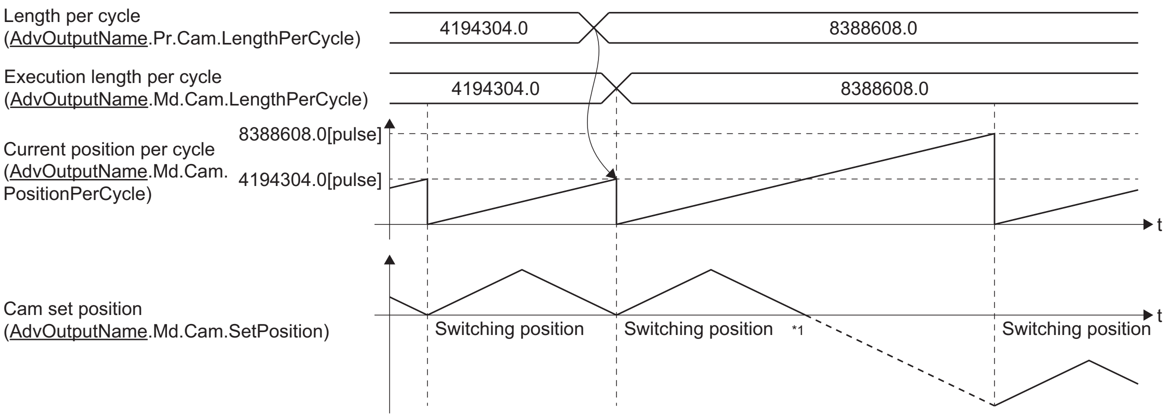
A setting example of cam data (linear interpolation)

· Length per cycle: 4194304.0

· Cam stroke amount: ±4194304.0

· Coordinate data

Point	Input value	Cam set position
1	0.0	0.0
2	2097152.0	4194304.0
3	4194304.0	0.0



*1 Because the coordinate where "input value = length per cycle" does not exist, the final coordinate is calculated from the line segment between the nearest two coordinates.

Point

- Do not set digits after the decimal point. If set, the errors will be accumulated every time the current position per cycle permissible value is exceeded.
- When cam data change and length per cycle change are executed simultaneously, the length per cycle will not be changed if cam data change fails.

■ Cam stroke amount (AdvOutputName.Pr.Cam.StrokeAmount)

When the Interpolation method specification (Interpolate) is set to "1: Section interpolation" or "2: Spline interpolation", or the data format is set to "Rotary cutter", set the cam stroke amount corresponding to a 100% stroke ratio in position units of the axis set in the output axis Axis information (AdvOutputName.Axis).

The cam stroke amount can be changed during synchronous control.

The value of Cam stroke amount (AdvOutputName.Pr.Cam.StrokeAmount) is imported when the current position per cycle passes through the 0th point of cam data, or is on the 0th point.

Point

Do not set digits after the decimal point. If set, the errors will be accumulated every time the current position per cycle permissible value is exceeded.

■ Cam No. (AdvOutputName.Pr.Cam.CamNo)

Set the cam No. used for cam control.

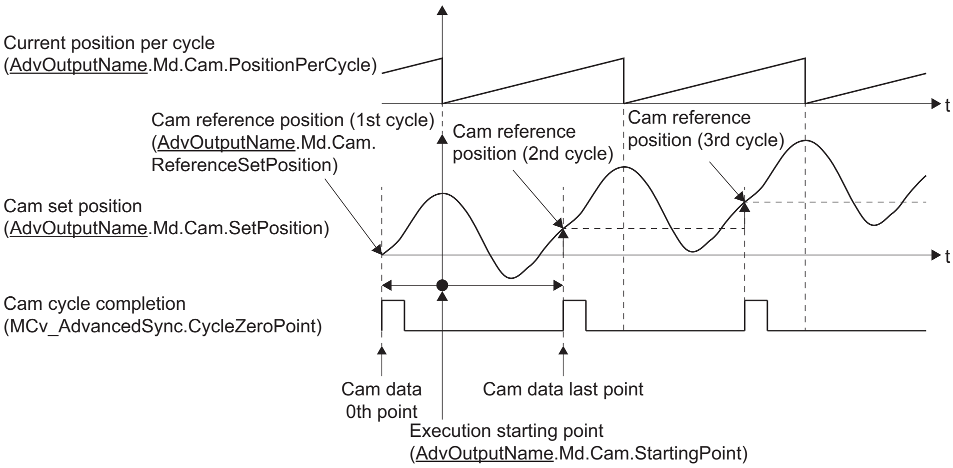
■ Cam starting point (AdvOutputName.Pr.Cam.StartingPoint)

Set the starting point of cam data in resolutions.

This setting is valid only when the Interpolation method specification (Interpolate) is set to "1: Section interpolation" or "2: Spline interpolation", or the data format is set to "Rotary cutter".

The initial value of the Cam starting point (AdvOutputName.Pr.Cam.StartingPoint) is "0". (The cam axis is controlled with cam data starting from the 0th point (stroke ratio 0%).)

When a value other than "0" is set, cam control is started from a stroke ratio other than 0%.



Point

- The cam reference position is the cam set position at the 0th point of cam data.
- The Cam starting point (AdvOutputName.Pr.Cam.StartingPoint) is imported when cam data switches. When a value exceeding the cam data resolution is set in the Cam starting point (AdvOutputName.Pr.Cam.StartingPoint), control is performed using the cam data and cam starting point before the change.

■ Length per cycle change setting (AdvOutputName.Pr.Cam.StartingPoint)

Set this when changing the Length per cycle (AdvOutputName.Pr.Cam.LengthPerCycle) during synchronous control.

This can change the length per cycle in cam No.0 (linear cam), cam data (linear interpolation), or cam data (section interpolation). However, this cannot change the length per cycle in cam data (section interpolation) that uses cam data with starting point other than 0.

Setting value	Description
0: Invalid (LengthChangeInvalid)	Cannot change the length per cycle during synchronous control.
1: Valid (LengthChangeValid)	Imports the value of the Length per cycle (AdvOutputName.Pr.Cam.LengthPerCycle) when the current position per cycle passes through the 0th point of cam data, or is on the 0th point.

15

■ Phase compensation advance time (AdvOutputName.Pr.Cam.PhaseCompensationAdvancedTime)

Set the time to advance or delay the phase of the current position per cycle in the cam control.

Setting value	Description
1 to 100000000 [μs]	Advance the phase according to the setting time.
0[μs]	Do not execute phase compensation.
-100000000 to -1 [μs]	Delay the phase according to the setting time.

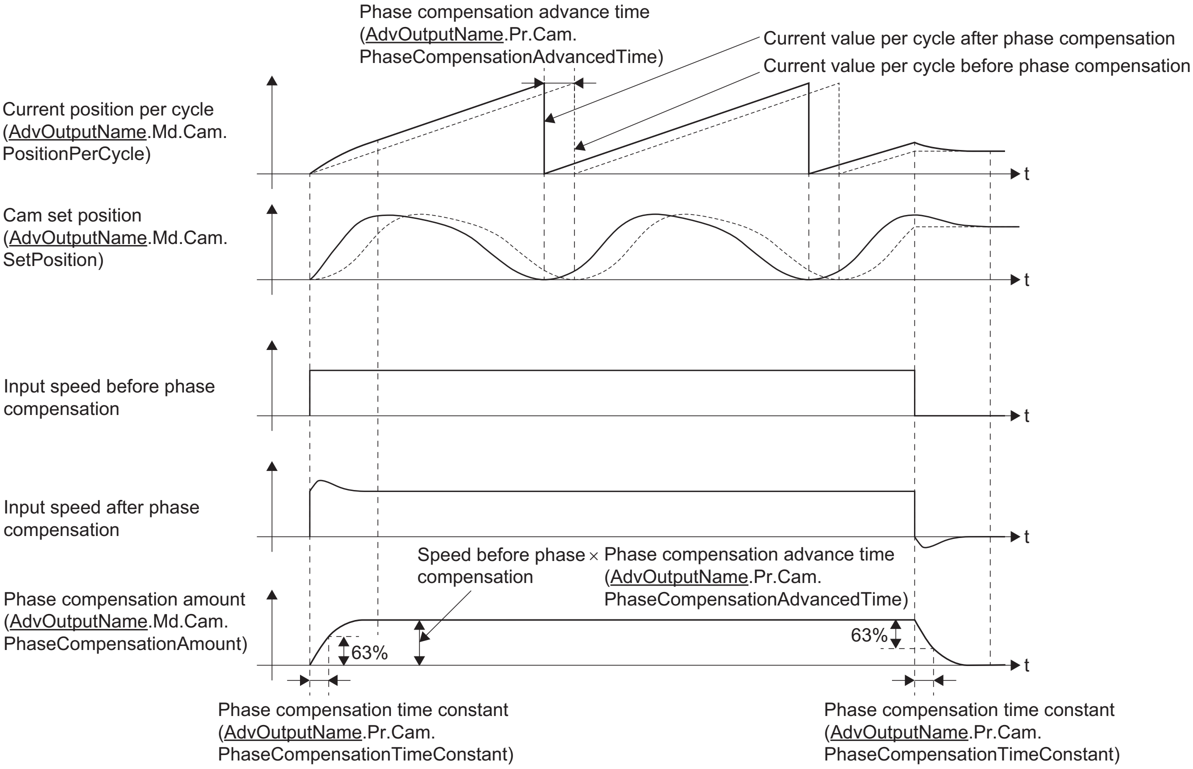
If the setting time is too long, the system experiences overshoot or undershoot at acceleration/deceleration of the input movement amount to the output axis.

In this case, set a longer time to reflect the phase compensation amount in the Phase compensation time constant (AdvOutputName.Pr.Cam.PhaseCompensationTimeConstant).

■ Phase compensation time constant (AdvOutputName.Pr.Cam.PhaseCompensationTimeConstant)

Set the time constant for reflecting the phase compensation amount for the first order delay.

63% of the phase compensation amount is reflected in the specified time constant.



Cam monitor of output axis monitor data (AdvOutputName.Md.Cam.)

Variable name	Name	Import*1	Data type*1	Monitor value
MasterCompositeGearSetPosition	Current position after composite main shaft gear	Operation cycle (FB is starting)	LREAL	-10000000000.0 to 10000000000.0
MasterGearPositionPerCycle	Current position per cycle after main shaft gear		LREAL	0.0 to (length per cycle - 0.00001)
AuxGearPositionPerCycle	Current position per cycle after auxiliary shaft gear		LREAL	0.0 to (length per cycle - 0.00001)
PhaseCompensationAmount	Phase compensation amount		LREAL	-2147483648.0 to 2147483647.0
PositionPerCycle	Length per cycle change setting		LREAL	0.0 to (length per cycle - 0.00001)
ReferenceSetPosition	Cam reference position		LREAL	-10000000000.0 to 10000000000.0
SetPosition	Cam set position		LREAL	-10000000000.0 to 10000000000.0
CamNo	Execution profile No.		WORD(UINT)	0 to 60000
StrokeAmount	Execution cam stroke amount		LREAL	-2147483648.0 to 2147483647.0
LengthPerCycle	Execution length per cycle		LREAL	1.0 to 2147483647.0
StartingPoint	Execution starting point		DWORD(UDINT)	0 to 65535

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Current position after composite main shaft gear (AdvOutputName.Md.Cam.MasterCompositeGearSetPosition)

The current position after the composite main shaft gear combines the values from the main shaft and the sub input axis is stored as an accumulative value.

The unit settings are in units of the input axis set as the Master axis (Master) of MCv_AdvancedSync (Advanced Synchronous Control).

The Current position after composite main shaft gear (AdvOutputName.Md.Cam.MasterCompositeGearSetPosition) is backed up. For details, refer to the following.

☞ Page 587 Synchronous Control Initial Position

■ Current position per cycle after main shaft gear (AdvOutputName.Md.Cam.MasterGearPositionPerCycle)

The input movement amount after the main shaft gear is stored within the range from "0.0 to (length per cycle - 0.00001)".

The unit settings are in units of the input axis set as the Master axis (Master) of MCv_AdvancedSync (Advanced Synchronous Control). The Current position per cycle after main shaft gear (AdvOutputName.Md.Cam.MasterGearPositionPerCycle) is backed up. For details, refer to the following.

☞ Page 587 Synchronous Control Initial Position

■ Current position per cycle after auxiliary shaft gear (AdvOutputName.Md.Cam.AuxGearPositionPerCycle)

The input movement amount after the auxiliary shaft gear is stored within the range from "0.0 to (length per cycle - 0.00001)".

The unit settings are in units of the input axis set as the Master axis (Master) of MCv_AdvancedSync (Advanced Synchronous Control).

The Current position per cycle after auxiliary shaft gear (AdvOutputName.Md.Cam.AuxGearPositionPerCycle) is backed up. For details, refer to the following.

☞ Page 587 Synchronous Control Initial Position

■ Phase compensation amount (AdvOutputName.Md.Cam.PhaseCompensationAmount)

The phase compensation amount after smoothing processing with the Phase compensation time constant (AdvOutputName.Pr.Cam.PhaseCompensationTimeConstant) is stored.

The unit settings of phase compensation amount are in units of the input axis set as the Master axis (Master) of MCv_AdvancedSync (Advanced Synchronous Control).

☞ Page 587 Synchronous Control Initial Position

■ Current position per cycle (AdvOutputName.Md.Cam.PositionPerCycle)

The current position per cycle is stored within the range from "0.0 to (length per cycle - 0.00001)".
The current position after phase compensation processing can be monitored. The unit settings are in units of the input axis set as the Master axis (Master) of MCv_AdvancedSync (Advanced Synchronous Control).
The Current position per cycle (AdvOutputName.Md.Cam.PositionPerCycle) is backed up. For details, refer to the following.
☞ Page 587 Synchronous Control Initial Position

■ Cam reference position (AdvOutputName.Md.Cam.ReferenceSetPosition)

The set position is stored as the cam reference position of the cam operation. The unit settings are in position units of the axis set in the output axis Axis information (AdvOutputName.Axis).
The cam reference position range is the positioning range of the axis assigned as the output axis.
The Cam reference position (AdvOutputName.Md.Cam.ReferenceSetPosition) is backed up. For details, refer to the following.
☞ Page 587 Synchronous Control Initial Position

■ Cam set position (AdvOutputName.Md.Cam.SetPosition)

- The set position of the cam axis is stored. The unit settings are in position units of the axis set in the output axis Axis information (AdvOutputName.Axis).
- When the Output axis smoothing time constant (AdvOutputName.PrConst.SmoothingTimeConstant) is "0" during the execution of synchronous control, the value of the Set position (AxisName.Md.SetPosition) of the axis set in the output axis Axis information (AdvOutputName.Axis) is equal to the value of the Cam set position (AdvOutputName.Md.Cam.SetPosition).
- When the Output axis smoothing time constant (AdvOutputName.PrConst.SmoothingTimeConstant) is other than "0" during the execution of synchronous control, the value of the Set position (AxisName.Md.SetPosition) of the axis set in the output axis Axis information (AdvOutputName.Axis) becomes equal to the value of the Cam set position (AdvOutputName.Md.Cam.SetPosition) after the smoothing is completed.
- The Cam set position (AdvOutputName.Md.Cam.SetPosition) is backed up. For details, refer to the following.
  ☞ Page 587 Synchronous Control Initial Position

■ Execution profile No. (AdvOutputName.Md.Cam.CamNo)

The executing cam No. is stored.
When the Cam No. (AdvOutputName.Pr.Cam.CamNo) is changed during synchronous control, this is updated when the controlling cam No. switches.
The same value of the Execution profile No. (AdvOutputName.Md.Cam.CamNo) is stored to the Execution profile ID (AxisName.Md.ProfileID).

■ Execution cam stroke amount (AdvOutputName.Md.Cam.StrokeAmount)

The executing cam stroke amount is stored.
When the Cam stroke amount (AdvOutputName.Pr.Cam.StrokeAmount) is changed during synchronous control, this is updated when the controlling cam stroke amount switches.

■ Execution length per cycle (AdvOutputName.Md.Cam.LengthPerCycle)

The executing cam length per cycle is stored.
When the Length per cycle (AdvOutputName.Pr.Cam.LengthPerCycle) is changed during synchronous control, this is updated when the controlling length per cycle switches.

■ Execution starting point (AdvOutputName.Md.Cam.StartingPoint)

The executing cam starting point is stored.
When the Cam starting point (AdvOutputName.Pr.Cam.StartingPoint) is changed during synchronous control, this is updated when the cam switches.

Cam control data of output axis parameters (AdvOutputName.Cd.Cam.)

Variable name	Name	Import*1	Data type*1	Setting value	Initial value
OutSmoothingTimeConstant	Output axis smoothing time constant change value	At start	INT	0 to 5000 [ms]	0

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Output axis smoothing time constant change value (AdvOutputName.Cd.Cam.OutSmoothingTimeConstant)

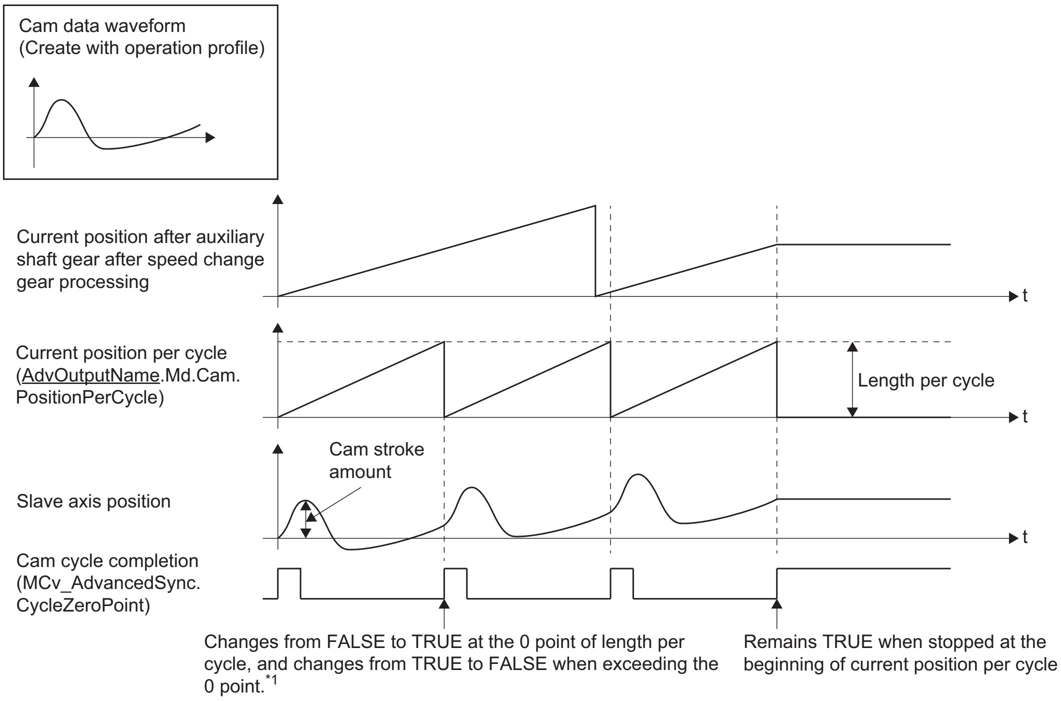
The output axis smoothing time constant is changed. The setting is imported when MCv_AdvancedSync (Advanced Synchronous Control) is started.

Setting value	Description
When a negative value is set	No smoothing.
When 0 is set	Smoothing based on the setting value of the Output axis smoothing time constant (AdvOutPutName.PrConst.SmoothingTimeConstant).
When a positive value is set	Smoothing based on the setting value of the Output axis smoothing time constant change value (AdvOutPutName.Cd.Cam.OutSmoothingTimeConstant). Set within the following setting range. Output axis smoothing time constant (AdvOutPutName.PrConst.SmoothingTimeConstant) ≤ Output axis smoothing time constant change value (AdvOutPutName.Cd.Cam.OutSmoothingTimeConstant) *: If the set value exceeds the range above, "Out of Advanced Synchronous Control Variable Range Warning (error code: 0D3FH)" occurs, and smoothing is executed based on the setting value of the Output axis smoothing time constant (AdvOutPutName.PrConst.SmoothingTimeConstant).

Cam function

Cam data sets the cam ID to Cam No. (AdvOutputName.Pr.Cam.CamNo).
To use the operation profile, set auto expand in the expand setting beforehand, or use MC_CamTableSelect (Cam Table Selection) to open the profile in cam open area.

Timing chart during cam operation



*1 When a value other than 0 is set to the cam data starting point, it becomes TRUE at the set cam data starting point.

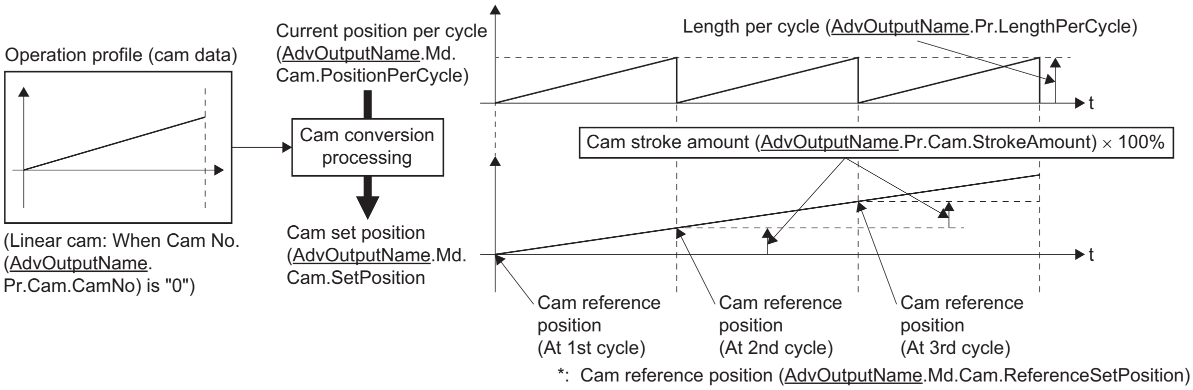
Cam data operation

Cam data for advanced synchronous control performs the following operations.

■ Linear cam

A linear operation in the cycle as the stroke ratio is 100%.

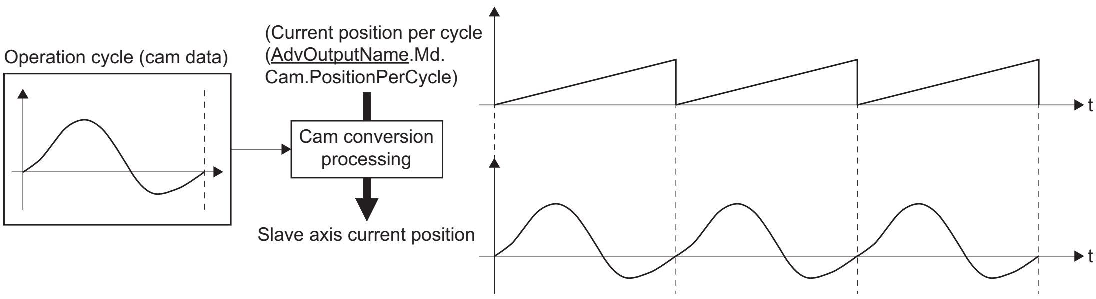
A linear operation occurs when Cam No. (AdvOutputName.Pr.Cam.CamNo) is set to "0".



■ Two-way cam

A reciprocating cam operation with a constant cam strokes range.

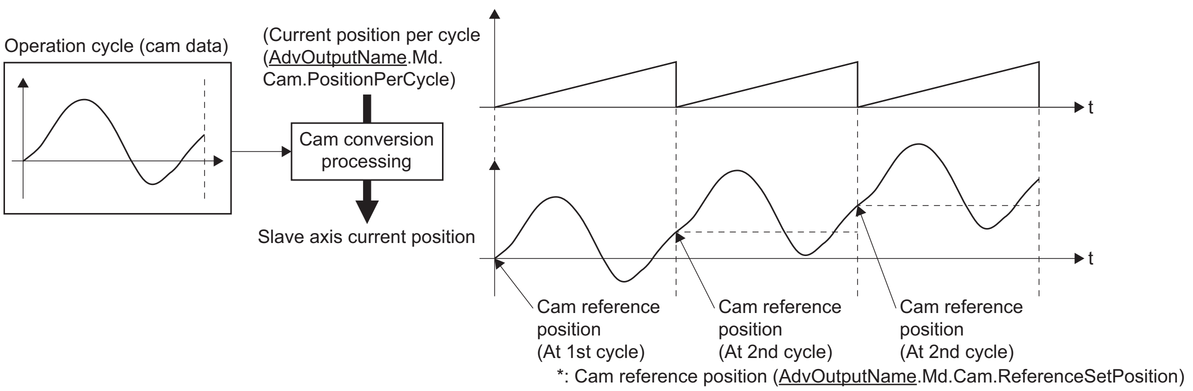
A two-way operation occurs when "start point = end point" for the cam data created with the operation profile.



■ Feed cam

A cam operation where the cam reference position is updated every cycle.

A feed cam operation occurs when "start point ≠ end point" for the cam data created with the operation profile.



15

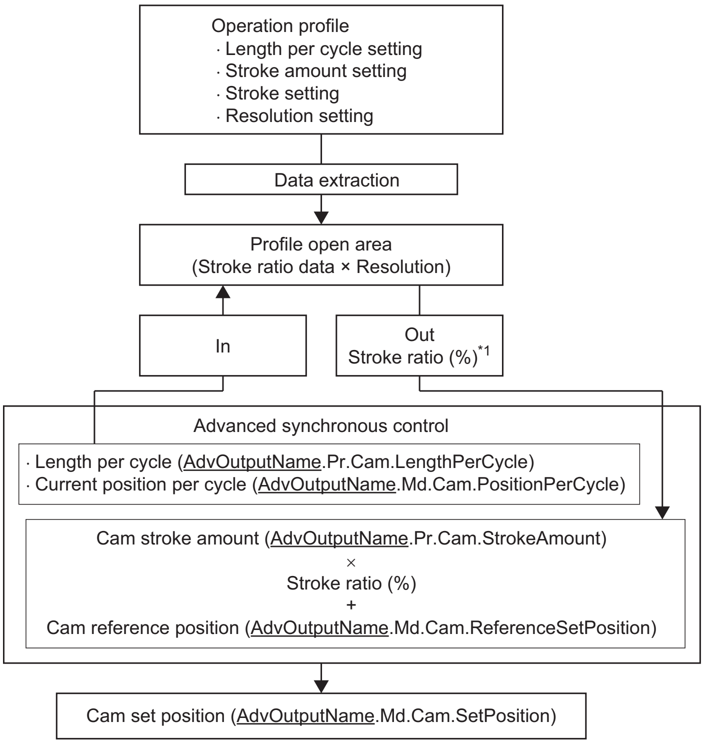
Cam control by operation profile

When using cam control in advanced synchronous control, the data transmitted to the open area varies depending on the interpolation method specification of the operation profile data and the data format.

■ When the Interpolation method specification (Interpolate) of the operation profile data is "1: Section interpolation" or "2: Spline interpolation", or the data format is set to "Rotary cutter"

The cam data is defined in equal divisions for one cam cycle based on the cam resolution, and configured with stroke ratio data from points within the cam resolution.

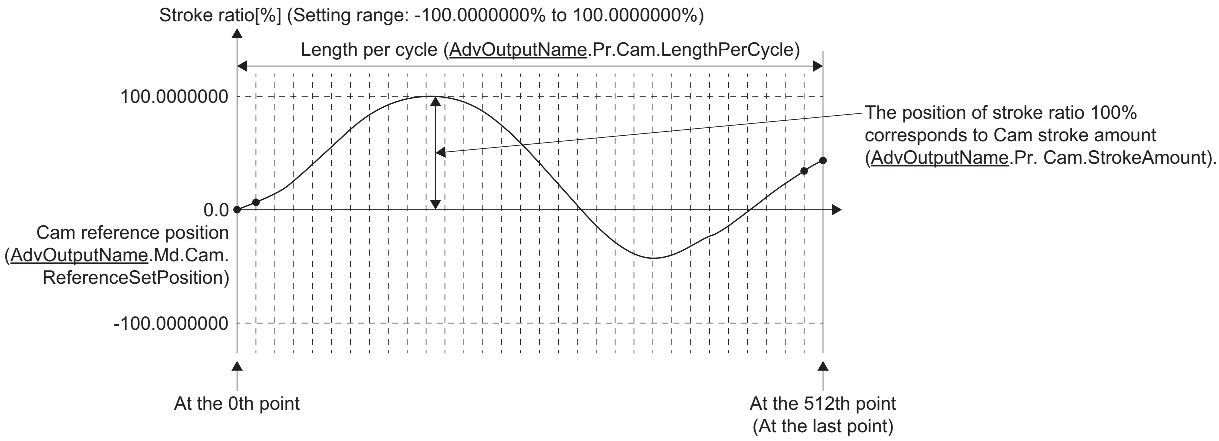
For the relation between the operation profile in this setting and Advanced synchronous control, refer to the following.



*1 Calculates the stroke ratio for Current position per cycle (AdvOutputName.Md.Cam.PositionPerCycle) using the stroke ratio and resolution of the profile open area and Length per cycle (AdvOutputName.Pr.Cam.LengthPerCycle).

Ex.

When cam resolution is 512



- Timing of applying cam control data

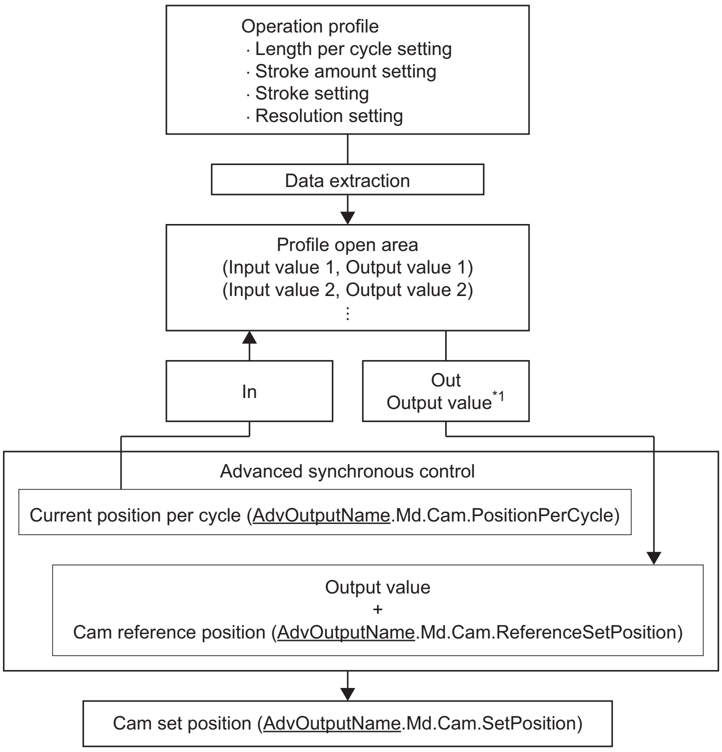
  If Length per cycle (AdvOutputName.Pr.Cam.LengthPerCycle) or Cam No. (AdvOutputName.Pr.Cam.CamNo), or Cam stroke amount (AdvOutputName.Pr.Cam.StrokeAmount) is changed, the new value is imported and applied when the Current position per cycle (AdvOutputName.Pr.Cam.PositionPerCycle) passes through the 0th point of cam data, or is on the 0th point. For "Cam starting point = 0", the cam reference position is updated when the current position per cycle exceeds the maximum permissible value. For "Cam starting point ≠ 0", the cam reference position is updated when the current position per cycle passes through the 0th point of cam data.

■ When the Interpolation method specification (Interpolate) of the operation profile data is set to "0: Linear interpolation"

The cam data defines a cam curve in two or more points of coordinate data for one cycle. The coordinate data is expressed as "(input value, output value)", with "input value = axis current position per cycle", and "output value = stroke amount from cam reference position".

The Cam stroke amount (AdvOutputName.Pr.Cam.StrokeAmount) of output axis parameter is ignored, and the output value of the I/O data becomes cam stroke amount.

For the relation between the operation profile in this setting and Advanced synchronous control, refer to the following.

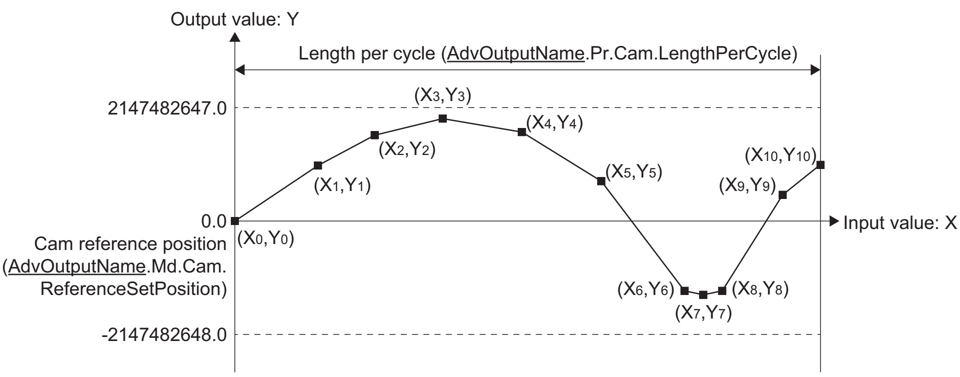


*1 The output value corresponding to the Current position per cycle (AdvOutputName.Md.Cam.PositionPerCycle) is calculated based on the input value and output value of the open area. In addition, the result of the calculation is truncated after the 5th decimal place.

Ex.

Setting example 1

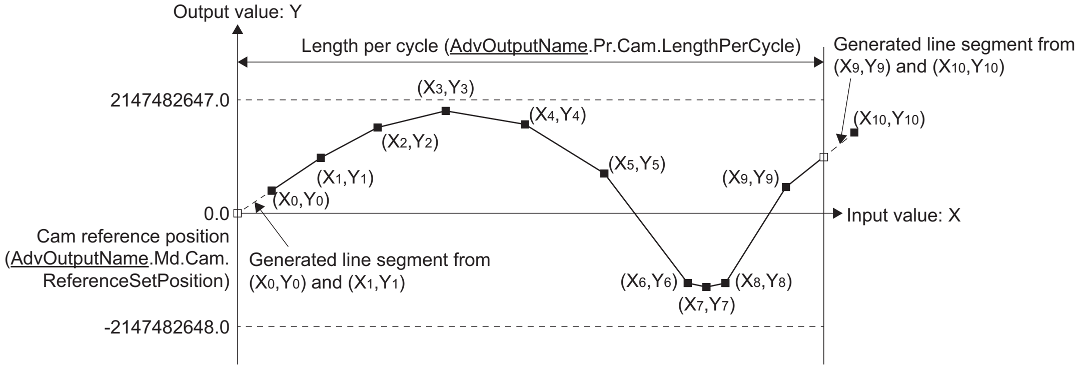
When the Length per cycle (AdvOutputName.Pr.Cam.LengthPerCycle) is equivalent to the setting value of the length per cycle setting in the operation profile



Ex.

Setting example 2

When "input value = 0" and "input value = length per cycle" coordinates do not exist in the coordinate data, control is performed using a line segment calculated from the nearest two coordinates.

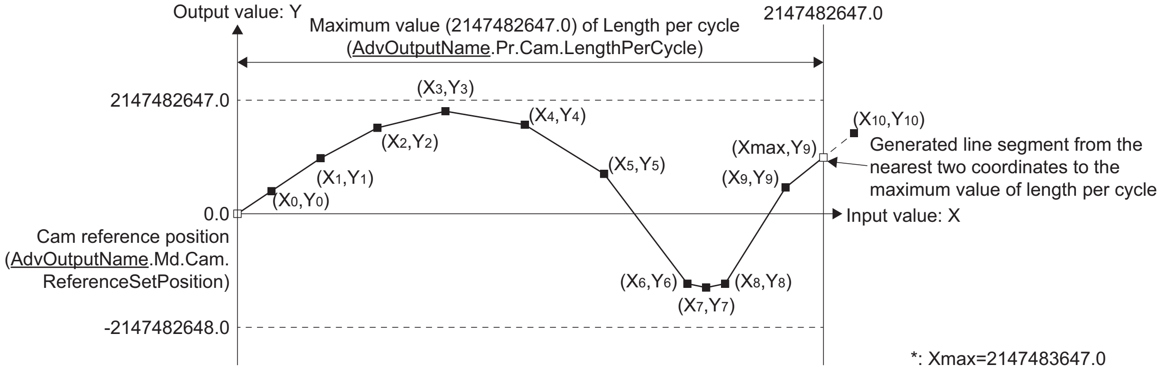


Ex.

Setting example 3

If the input value exceeds the maximum value (2147483647.0) of the Length per cycle (AdvOutputName.Pr.Cam.LengthPerCycle), the control is performed using a line segment calculated from the nearest two coordinates to the maximum value of the Length per cycle (AdvOutputName.Pr.Cam.LengthPerCycle).

In this case, the input value X at the last point of the coordinates is equivalent to the maximum value of the Length per cycle (AdvOutputName.Pr.Cam.LengthPerCycle).



- Reflection timing of the cam control data

  If the Length per cycle (AdvOutputName.Pr.Cam.LengthPerCycle) or the Cam No. (AdvOutputName.Pr.Cam.CamNo) is changed during synchronous control, the new value is imported and reflected when the Current position per cycle (AdvOutputName.Pr.Cam.PositionPerCycle) passes through "0.0", or is on the position of "0.0". The cam reference position is updated when the current position per cycle passes through "0.0".

■ Setting list for using the operation profile in advanced synchronous control

The settings for using the operation profile in advanced synchronous control are shown below.

When specifying setting values, set the specified value, or a value within the range. When a specified value or a value within the range is not set, an error, or an unintended operation may occur.

<table>
<tr><th colspan="2" rowspan="2">Setting item*1</th><th colspan="2">Interpolation Method Specification (Interpolate)</th></tr>
<tr><th>1: Section Interpolation/2: Spline Interpolation</th><th>0: Linear Interpolation</th></tr>
<tr><td colspan="2">Periodic (Options (Options): Bit 0)</td><td colspan="2">Set to "1: Periodic".
When set to other than "1: Periodic", "Operation Profile Data Settings Incorrect Under Advanced Synchronous Control (error code: 1AF8H)" will occur at FB start.</td></tr>
<tr><td colspan="2">Master axis (input) absolute coordinate (Options (Options): Bit 1)</td><td colspan="2">Set to "0: Relative coordinate".
When set to other than "0: Relative coordinate", "Operation Profile Data Settings Incorrect Under Advanced Synchronous Control (error code: 1AF8H)" will occur at FB start.</td></tr>
<tr><td colspan="2">Slave axis (output) absolute coordinate ((Options (Options): Bit 2)</td><td colspan="2">Set to "0: Relative coordinate".
When set to other than "0: Relative coordinate", "Operation Profile Data Settings Incorrect Under Advanced Synchronous Control (error code: 1AF8H)" will occur at FB start.</td></tr>
<tr><td colspan="2">Profile ID</td><td colspan="2">Set to any value within the range of 1 to 60000.</td></tr>
<tr><td colspan="2">Interpolation Method Specification (Interpolate)</td><td>Set to "1: Section Interpolation", or "2: Spline Interpolation".</td><td>Set to "0: Linear Interpolation".</td></tr>
<tr><td colspan="2">Resolution/coordinate number (Resolution)</td><td>Set the number of divisions for one cam curve.</td><td>Set the number of coordinate points in one cam cycle.
The 0th point is included in the number of coordinate points.</td></tr>
<tr><td colspan="2">Input unit character string (InputUnitString)</td><td colspan="2">Set the unit of the axis to be set as the main shaft.</td></tr>
<tr><td colspan="2">Output unit character string (OutputUnitString</td><td>Set the unit of the stroke amount.</td><td>Set the unit of the axis to be set as the output axis.</td></tr>
<tr><td colspan="2">Start point (StartPoint)</td><td colspan="2">Set to "0.0".
When set to other than "0.0", "Operation Profile Data Settings Incorrect Under Advanced Synchronous Control (error code: 1AF8H)" will occur at FB start.
When setting the cam data starting point in advanced synchronous control, set the Cam starting point (AdvOutputName.Pr.Cam.StartingPoint) of the output axis parameters.</td></tr>
<tr><td colspan="2">Initial stroke amount (StartStroke)</td><td colspan="2">Set to "0.0".
When set to other than "0.0", "Operation Profile Data Settings Incorrect Under Advanced Synchronous Control (error code: 1AF8H)" will occur at FB start.</td></tr>
<tr><td colspan="2">Initial velocity (StartVelocity)</td><td>Set this variable in the case of "1: Section Interpolation".
Set the velocity at the start point/end point.
Use when the Cam curve type (CurveType) is set to "12: 5th curve (Adj.) (FifthCurve_SpeedDesignation)".</td><td rowspan="2">It is not used in the coordinate data format.</td></tr>
<tr><td colspan="2">Initial acceleration (StartAcceleration)</td><td>Set this variable in the case of "1: Section Interpolation".
Set the acceleration at the start point/end point.
Use when the Cam curve type (CurveType) is set to "12: 5th curve (Adj.) (FifthCurve_SpeedDesignation)".</td></tr>
<tr><td rowspan="2">I/O data</td><td>Input value</td><td rowspan="2">Do not set this item in the case of "1: Section Interpolation" or "2: Spline Interpolation".</td><td>• Set the input value at each point No.
• Set the input value within the range of "0.0 to 2147483647.0". When an input value exceeding "2147483647.0" is set, the control is performed using a line segment calculated from the nearest two coordinates to the maximum value of the Length per cycle (AdvOutputName.Pr.Cam.LengthPerCycle).</td></tr>
<tr><td>Output value</td><td>• Set the output value corresponding with the input value at each point No.
• Set the output value within the range of "-2147483648.0 to 2147483647.0". If a value outside the range is set, "Operation Profile Data Settings Incorrect Under Advanced Synchronous Control (error code: 1AF8H)" will occur.</td></tr>
</table>

Setting item[*1]		Interpolation Method Specification (Interpolate)	
		1: Section Interpolation/2: Spline Interpolation	**0: Linear Interpolation**
Parameter	Cam curve type (CurveType)	Set to any type to match the cam data to be created.	It is not used in the coordinate data format.
	End point (EndPoint)	Set the end point of each section.	
	Stroke (Stroke)	Set the stroke of each section.	
	Curve applicable range (RangeP1/ RangeP2)	Set the curve applicable range (start point: P1, end point: P2) for the cam curve. (0.0 to 1.0) Set "P1 < P2". However, for "P1 = P2 = 0", "P1 = 0" and "P2 = 1" are applied. For curve applicable range, the setting can be omitted.	
	Acceleration/ deceleration range compensation (RangeL1)	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.	
	End point velocity (EndVelocity)	Set the end point velocity of the cam curve.	
	End point acceleration (EndAcceleration)	Set the end point acceleration of the cam curve.	
Length per cycle (CycleLength)		Although the setting is ignored in advanced synchronous control, set the same value to the Length per cycle (AdvOutputName.Pr.Cam.LengthPerCycle). The setting value of the Length per cycle (AdvOutputName.Pr.Cam.LengthPerCycle) is used for actual control.	
Minimum value per cycle (CycleMin)		Do not set this item in the case of "1: Section Interpolation" or "2: Spline Interpolation".	Set to "0.0". Set this variable when the Interpolation method specification (Interpolate) is set to "1: Section Interpolation".
Maximum value per cycle (CycleMax)			Set the setting value of Length per cycle (AdvOutputName.Pr.Cam. LengthPerCycle). Set this item when the Interpolation method specification (Interpolate) is set to "0: Linear Interpolation".
Time per cycle (CycleTime)		Set the time that is required for one cycle. Use when the cam curve type (CurveType) is set to "12: 5th curve (Adj.) (FifthCurve_SpeedDesignation)".	Not used in "0: Linear Interpolation".
Stroke amount (Stroke)		When the unit is "%", the setting is ignored. When the unit is other than "%", set to 100 % stroke amount. <Example> If the stroke amount is set to "200.0" when the unit is "mm", the cam data will have "200.0" as the 100% stroke ratio.	Set to a value equal to or more than the maximum value of the output value in the I/O data.
Number of sections (NumberOfSections)		Set to a value within the following ranges depending on the interpolation method specification (Interpolate). ■"1: Section Interpolation" 1 to 360 ■"2: Spline Interpolation" 3 to 360	It is not used in the coordinate data format.

*1 The variable inside the ( ) is the member name of the PROFILE_CAM_DATA data type, or the PROFILE_CAM_ELEMENT data type.

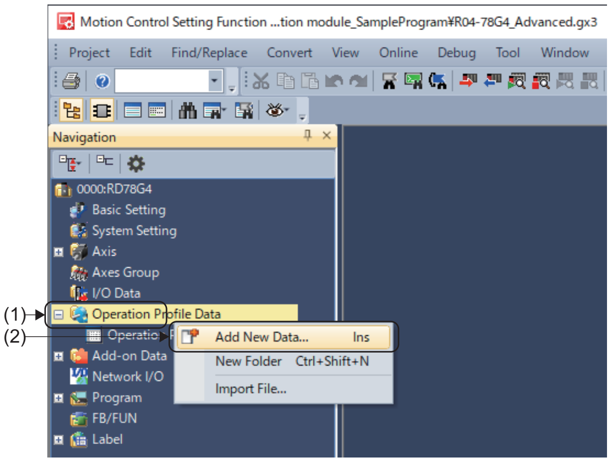
When the data format is "rotary cutter", refer to the following.

☞ Page 456 Rotary cutter

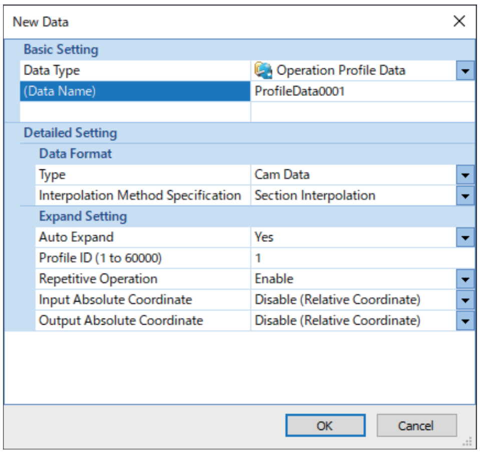
Setting method of operation profile data

This section describes the setting method for using an operation profile in advanced synchronous control.

1. In "Motion Control Setting Function", right-click Navigation window ⇨ [Operation Profile Data] (1), and select [Add New Data] (2).



2. On the "New Data" window, configure the basic setting and detailed setting, and click the [OK] button.

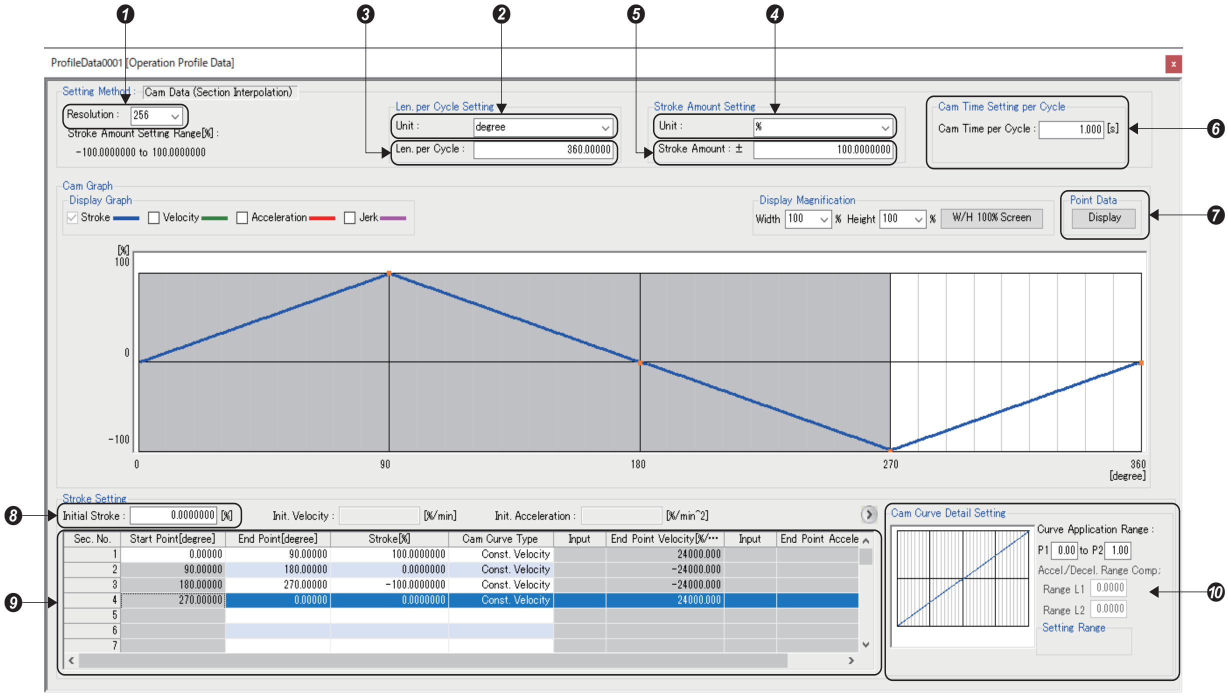


Setting item			Setting
Basic setting	Data type		Set "Operation Profile Data".
	Data name		Set any data name.
Detailed setting	Data format	Type	Set the type of data to be created. • Cam data • Rotary cutter
		Interpolation method specification	Set the interpolation method specification to match the cam to be created. • Linear interpolation • Section interpolation • Spline interpolation
	Expand setting	Auto expand	To expand the operation profile data automatically, set to "Yes". When set to "No", operation profile data must be stored in the open area by MC_CamTableSelect (Cam Table Selection) to use the operation profile data in the control.
		Profile ID	Set to any value within the range of 1 to 60000.
		Repetitive operation	Set to "Enable".
		Input absolute coordinate	Set to "Disable (Relative Coordinate)".
		Output absolute coordinate	Set to "Disable (Relative Coordinate)".

3. Set the operation profile data. Set as follows according to match the set type, and the interpolation method specification.

- Cam data

■Section interpolation



❶ Set the resolution to match the cam data to be created.

❷ Set to the same unit as that of the main shaft.*1

❸ Set to the same value as that set in the Length per cycle (AdvOutputName.Pr.Cam.LengthPerCycle).*1

❹ Set to any unit.

❺ When the unit is other than "%", set to 100% stroke amount.

❻ Use when cam curve type is set to "5th Curve (Adj.)".

❼ Click the [Display] button of the point data to check the stroke data of the open area.

❽ Set to "0.0".

❾ Set the input value and the output value to match the cam data to be created.*2

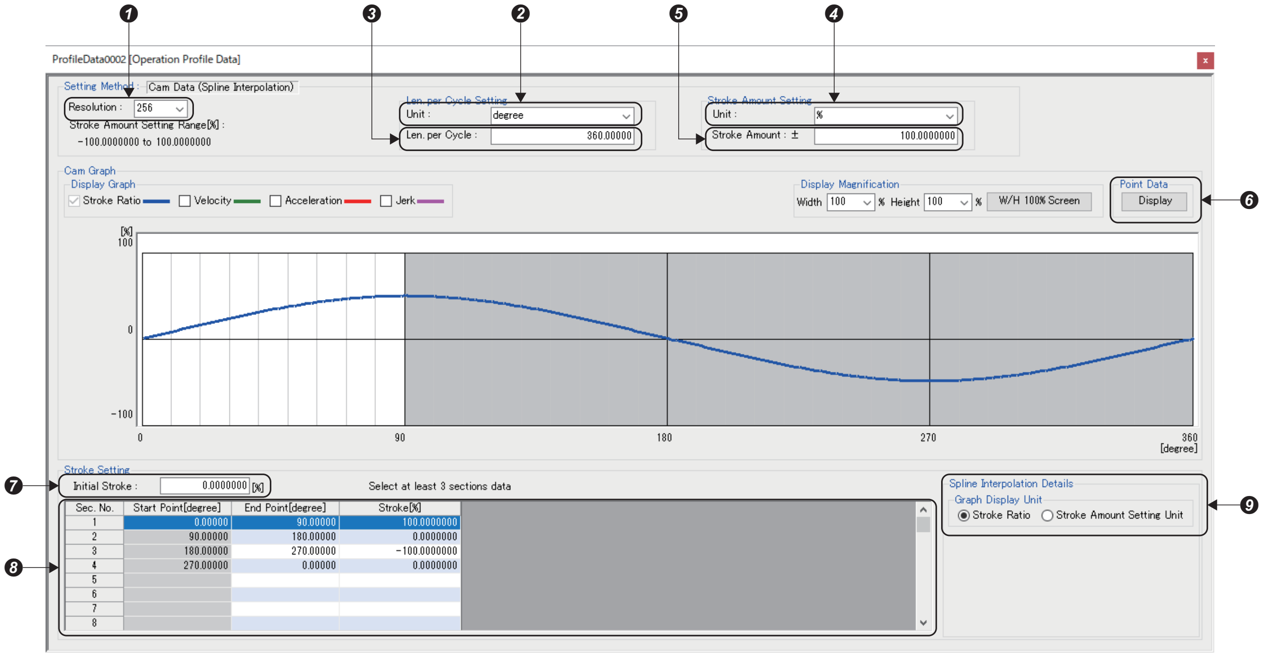
❿ Set the detailed settings to match the cam data to be created.*2

*1 Although the setting is ignored in advanced synchronous control, setting this item is recommended in order to match the setting window of the motion control setting function and the actual cam operation.

*2 For details, refer to the following.

☞ Page 446 Cam data

■Spline interpolation

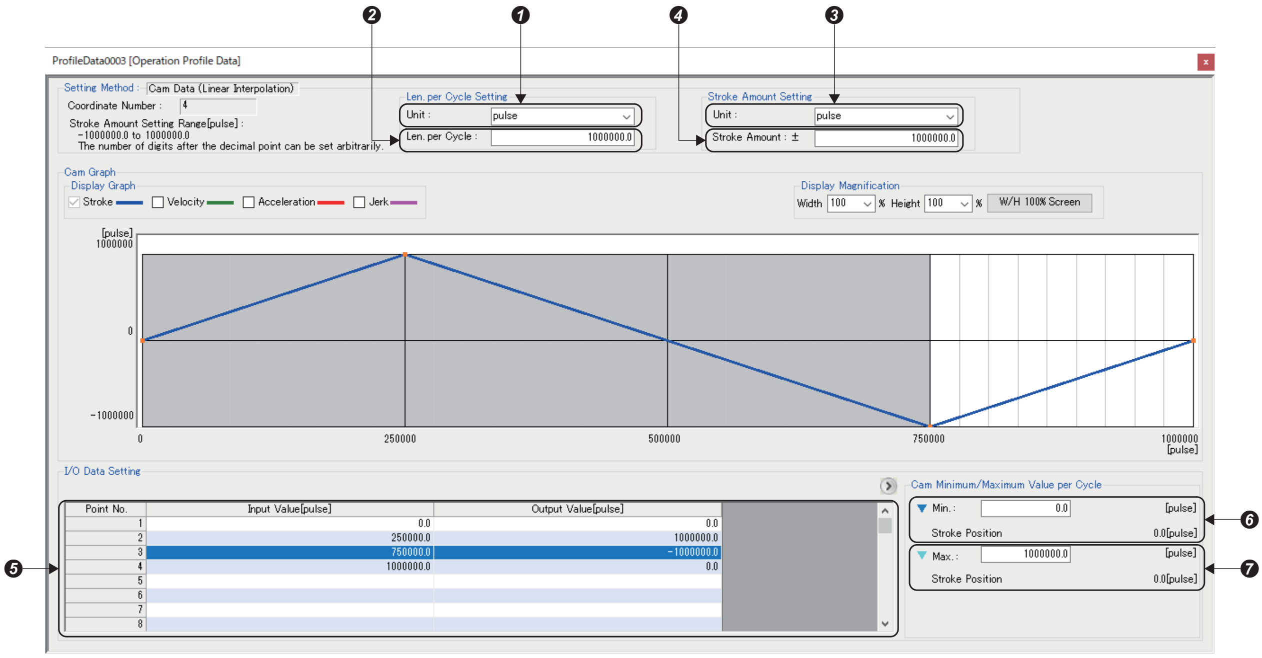


❶ Set the resolution to match the cam data to be created.
❷ Set to the same unit as that of the main shaft.*1
❸ Set to the same value as that set in the Length per cycle (AdvOutputName.Pr.Cam.LengthPerCycle).*1
❹ Set to any unit.
❺ When the unit is other than "%", set to 100 % stroke amount.
❻ Click the [Display] button of the point data to check the stroke data of the open area.
❼ Set to "0.0".
❽ Set the input value and the output value to match the cam data to be created.*2
❾ Set the detailed settings to match the cam data to be created.

*1 Although the setting is ignored in advanced synchronous control, setting this item is recommended in order to match the setting window of the motion control setting function, and the actual cam operation.
*2 For details, refer to the following.
☞ Page 446 Cam data

15

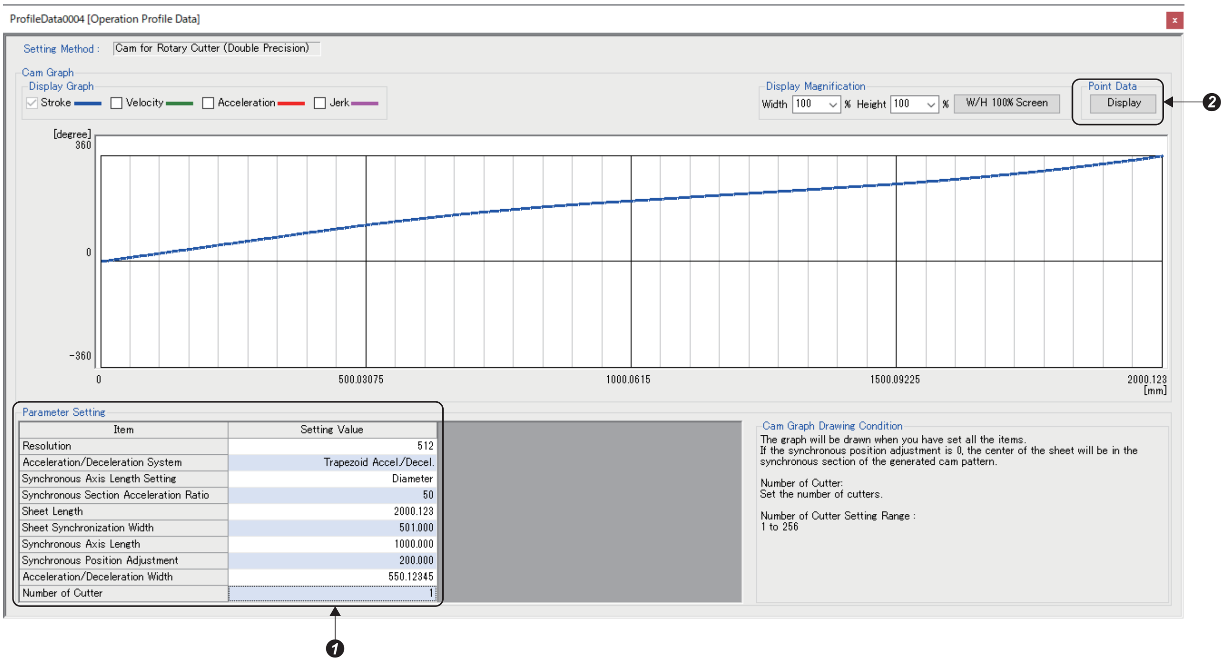
■Linear Interpolation



Point No.	Input Value[pulse]	Output Value[pulse]
1	0.0	0.0
2	250000.0	1000000.0
3	750000.0	−1000000.0
4	1000000.0	0.0
5		
6		
7		
8		

➊ Set to the same unit of the axis which is set as the main shaft.
➋ Set to the same value as that set in the Length per cycle (AdvOutputName.Pr.Cam.LengthPerCycle).
➌ Set to the same unit of the axis which is set as the output axis.
➍ Set to a value equal to or more than the maximum value of the output value in the I/O data.
➎ Set the input value and the output value to match the cam data to be created.
➏ Set to "0.0".
➐ Set to the setting value of the Length per cycle (AdvOutputName.Pr.Cam.LengthPerCycle).

- Rotary cutter



Item	Setting Value
Resolution	512
Acceleration/Deceleration System	Trapezoid Accel./Decel.
Synchronous Axis Length Setting	Diameter
Synchronous Section Acceleration Ratio	50
Sheet Length	2000.123
Sheet Synchronization Width	501.000
Synchronous Axis Length	1000.000
Synchronous Position Adjustment	200.000
Acceleration/Deceleration Width	550.12345
Number of Cutter	1

➊ Set the parameters to match the cam data to be created.
➋ Click the [Display] button of the point data to check the stroke data of the open area.

Precautions

- When rewriting the open area during control by using the Profile Write (MCv_WriteProfileData), depending on the waveform pattern the stroke value may change rapidly, causing an impact to be applied to the machine. Create operation profile data so that the waveform before and after the change does not become discontinued.
- For operation profile data format that can be used in the advanced synchronous control, refer to the following. If an incorrect profile data format is used in the advanced synchronous control, "Operation Profile Data Settings Incorrect Under Advanced Synchronous Control (error code: 1AF8H)" will occur.
  ☞ Page 446 Operation profile data types
- If the Cam No. (AdvOutputName.Pr.Cam.CamNo) is changed during advanced synchronous control in the following cases, a warning occurs, and control continues with the cam data before the change.

Description	Warning code
A value out of range for Cam No. (AdvOutputName.Pr.Cam.CamNo) was set.	Out of Advanced Synchronous Control Variable Range Warning (warning code: 0D3FH)
The cam table does not exist.	No Cam Table Warning (warning code: 0D44H)
A value that cannot be used in advanced synchronous control was set in the operation profile data.	Operation Profile Data Settings Incorrect Under Advanced Synchronous Control Warning (warning code: 0D46H)
A value which exceeds the resolution was set to the cam starting point.	Operation Profile Data Settings Incorrect Under Advanced Synchronous Control Warning (warning code: 0D46H)
Failed to acquire the add-on ProfileControl.	Add-on Acquisition Failure Warning (warning code: 0D47H)

Synchronous control change function

While MCv_AdvancedSync (Advanced Synchronous Control) is activated, this function can be used to change the cam reference position, the cam axis current value per cycle and the current value per cycle after the main/auxiliary shaft gear during the synchronous control. The following five methods exist for the synchronous control change function. (☞ Page 583 Synchronous control change command (AdvOutputName.Cd.Cam.SyncControlChangeCommand))

Synchronous control change command	Application	Output axis operation
Cam reference position movement	Adjust the cam reference position by the movement amount.	Operated
Change current position per cycle	Change the current position per cycle.	None
Change current position per cycle after main shaft gear	Change the current position per cycle after main shaft gear.	None
Change current position per cycle after auxiliary shaft gear	Change the current position per cycle after auxiliary shaft gear.	None
Current position per cycle movement	Adjust the phase of the cam set position by the movement amount.	Operated

Cam monitor of output axis parameters (AdvOutputName.Md.Cam.)

Variable name	Name	Import*1	Data type*1	Monitor value
SyncControlChangeStatus	Synchronous control change status	Operation cycle (FB is starting)	INT	-1: Synchronous control change failure 0: Synchronous control change not requested 1: Synchronous control change in progress 2: Synchronous control change complete

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Synchronous control change status (AdvOutputName.Md.Cam.SyncControlChangeStatus)

- When Synchronous control change request (AdvOutputName.Cd.Cam.SyncControlChangeRequest) is set to "TRUE (Synchronous control change requested)", "1: Synchronous control change in progress" is stored to Synchronous control change status (AdvOutputName.Md.Cam.SyncControlChangeStatus) during synchronous control change processing, and "2: Synchronous control change complete" is stored after processing.
- Synchronous control change status (AdvOutputName.Md.Cam.SyncControlChangeStatus) is "0: Synchronous control change not requested" according to the following conditions.
  - Synchronous control change request (AdvOutputName.Cd.Cam.SyncControlChangeRequest) is set to "FALSE (Synchronous control change not requested".
  - Advanced synchronous control is stopped.
- When Synchronous control change command (AdvOutputName.Cd.Cam.SyncControlChangeCommand) and Synchronous control change value (AdvOutputName.Cd.Cam.SyncControlChange) are set out of range, and Synchronous control change request (AdvOutputName.Cd.Cam.SyncControlChangeRequest) is set to "TRUE (Synchronous control change requested)", "-1: Synchronous control change failure" is stored.

Cam control data of output axis parameters (AdvOutputName.Cd.Cam.)

Variable name	Name	Import[*1]	Data type[*1]	Setting value	Initial value
SyncControlChange Request	Synchronous control change request	Operation cycle (FB is starting)	BOOL	FALSE: Synchronous control change not requested TRUE: Synchronous control change requested	FALSE
SyncControlChange Command	Synchronous control change command		INT (MC_SYNC_C HANGE_COM MAND)	0: Cam reference position movement (ReferenceSetPositionMovement) 1: Change current position per cycle (ChangeCurrentPositionPerCycle) 2: Change current position per cycle after main shaft gear (ChangeMasterGearPositionPerCycle) 3: Change current position per cycle after auxiliary shaft gear (ChangeAuxGearPositionPerCycle) 4: Current position per cycle movement (PositionPerCycleMovement)	0: Cam reference position movement (ReferenceSetP ositionMovement)
SyncControlChange	Synchronous control change value		LREAL	-2147483648.0 to 2147483647.0	0.0
SyncControlReflectio nTime	Synchronous control reflection time		WORD(UINT)	0 to 65535 [ms]	0 [ms]

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Synchronous control change request (AdvOutputName.Cd.Cam.SyncControlChangeRequest)

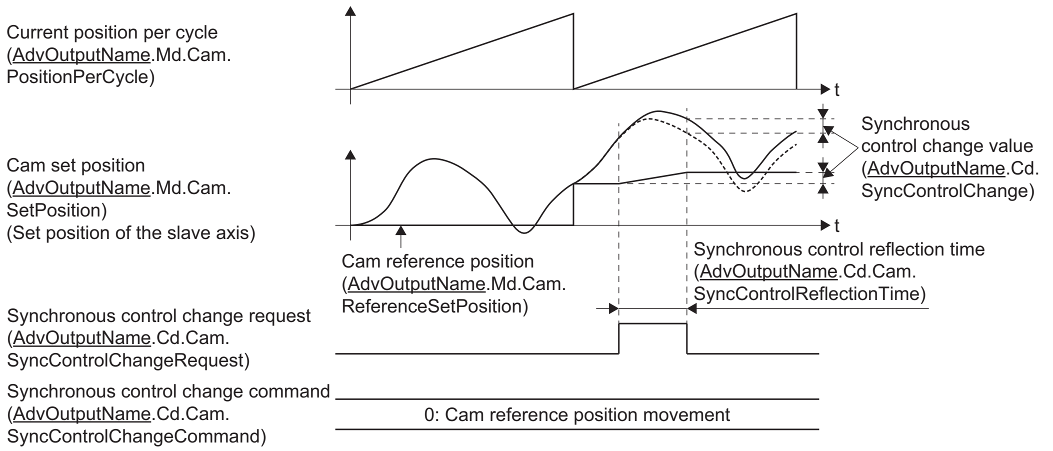
The synchronous control change is requested.

When set to "TRUE (Synchronous control change requested)", Synchronous control change command (AdvOutputName.Cd.Cam.SyncControlChangeCommand) is executed.

■ Synchronous control change command (AdvOutputName.Cd.Cam.SyncControlChangeCommand)

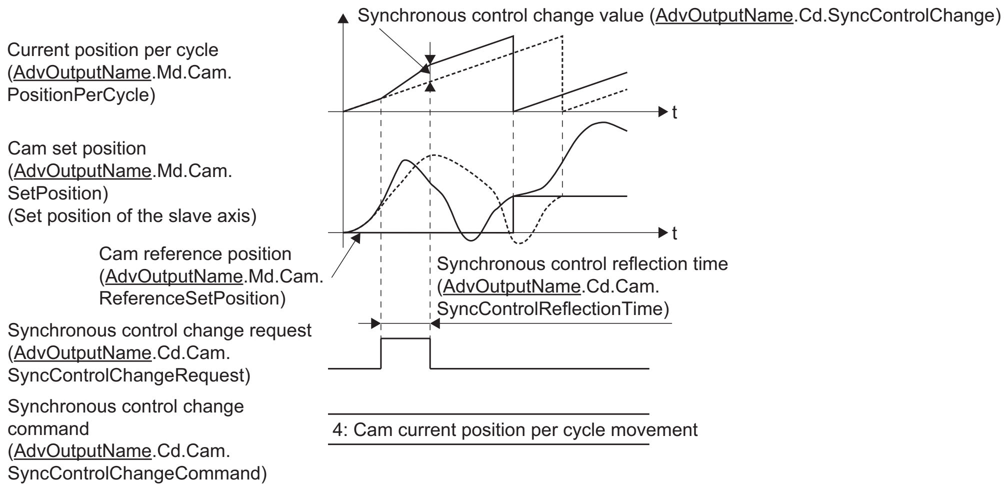
Set the synchronous control change command.

Synchronous control change command	Description
0: Cam reference position movement (ReferenceSetPosition Movement)	This command is executed to move the cam reference position through adding the setting movement amount of Synchronous control change value (AdvOutputName.Cd.Cam.SyncControlChange). The movement amount to be added is averaged in Synchronous control reflection time (AdvOutputName.Cd.Cam.SyncControlReflectionTime) for its output. Set a long reflection time when a large movement amount is used since the cam set position moves with the movement amount.



Synchronous control change command	Description
	• When Synchronous control change request (AdvOutputName.Cd.Cam.SyncControlChangeRequest) is reset to "FALSE (Synchronous control change not requested)" while executing the cam reference position movement, operation is stopped midway. If the cam reference position movement command is executed again, the remainder movement amount is not reflected, and the operation starts with Synchronous control change value (AdvOutputName.Cd.SyncControlChange) to be used again. • If synchronous control is ended while the cam reference position movement is being executed, operation stops midway. If synchronous control is restarted, the remainder movement amount is not reflected.
1: Change current position per cycle (ChangeCurrentPositio nPerCycle)	The current position per cycle is changed to Synchronous control change value (AdvOutputName.Cd.Cam.SyncControlChange). The cam reference position will be also changed to correspond to the changed current position per cycle. This operation is completed within one operation cycle.

15

Synchronous control change command	Description
2: Change current position per cycle after main shaft gear (ChangeMasterGearPositionPerCycle)	The current position per cycle after main shaft gear is changed to the value set in Synchronous control change value (AdvOutputName.Cd.Cam.SyncControlChange). This operation is completed within one operation cycle. Clutch control is not executed if the current position per cycle after main shaft gear (the position before being changed and after being changed) has already passed through the ON/OFF address in address mode.
3: Change current position per cycle after auxiliary shaft gear (ChangeAuxGearPositionPerCycle)	The current position per cycle after auxiliary shaft gear is changed to the value set in Synchronous control change value (AdvOutputName.Cd.Cam.SyncControlChange). This operation is completed within one operation cycle. Clutch control is not executed if the current position per cycle after auxiliary shaft gear (the position before being changed and after being changed) has already passed through the ON/OFF address in address mode.
4: Current position per cycle movement (PositionPerCycleMovement)	This command is executed to move the current value per cycle through adding the setting movement amount of Synchronous control change value (AdvOutputName.Cd.Cam.SyncControlChange). The movement amount to be added is averaged in Synchronous control reflection time (AdvOutputName.Cd.Cam.SyncControlReflectionTime) for its output. Set a long reflection time when a large movement amount is used since the cam set position moves with the movement amount. 

■ Synchronous control change value (AdvOutputName.Cd.Cam.SyncControlChange)

Set the change value for synchronous control change processing.

Synchronous control change command (AdvOutputName.Cd.Cam.SyncControlChangeCommand)	Synchronous control change value (AdvOutputName.Cd.Cam.SyncControlChange)		
	Setting range	Unit	Setting details
0: Cam reference position movement (ReferenceSetPositionMovement)	-2147483648.0 to 2147483647.0	Position units of the axis set in the output axis	• Set the movement amount of the cam reference position. • It moves within the range from -2147483648.0 to 2147483647.0.
1: Change current position per cycle (ChangeCurrentPositionPerCycle)			• Set the change current position per cycle. • The setting value is converted within the range from "0.0 to (length per cycle - 0.00001)".
2: Change current position per cycle after main shaft gear)". (ChangeMasterGearPositionPerCycle)			
3: Change current position per cycle after auxiliary shaft gear (ChangeAuxGearPositionPerCycle)			
4: Current position per cycle movement (PositionPerCycleMovement)			• Set the movement amount of the current position per cycle. • It moves within the range from -2147483648 to 2147483647.

■ Synchronous control reflection time (AdvOutputName.Cd.Cam.SyncControlReflectionTime)

Set the reflection time for synchronous control change processing.

Synchronous control change command (AdvOutputName.Cd.Cam.SyncControlChangeCommand)	Setting details for Synchronous control reflection time (AdvOutputName.Cd.Cam.SyncControlReflectionTime)
0: Cam reference position movement (ReferenceSetPositionMovement)	The time to reflect the movement amount to the cam reference position.
1: Change current position per cycle (ChangeCurrentPositionPerCycle)	Setting not required.
2: Change current position per cycle after main shaft gear. (ChangeMasterGearPositionPerCycle)	
3: Change current position per cycle after auxiliary shaft gear (ChangeAuxGearPositionPerCycle)	
4: Current position per cycle movement (PositionPerCycleMovement)	The time to reflect the movement amount to the current position per cycle.

Precautions

The following are precautions when operating an output axis.

- Do not set the Ring counter lower limit value (AxisName.PrConst.RingCount_LowerValue) and Ring counter upper limit value (AxisName.PrConst.RingCount_UpperValue) of the axis positioning range of the axis set in the output axis Axis information (AdvOutputName.Axis) to a decimal number. If set, an error will occur with the value output to the slave axis and the Set position (AxisName.Md.SetPosition) of the axis set to the output axis in synchronous control when the maximum permissible value is exceeded.
- When the intermediate result of the movement amount calculated by each module exceeds the range of "-2147483648.0 to 2147483647.0", "Advanced Synchronization Control Operation Overflow (error code: 1AF1H)" occurs. The function of the module in which the overflow occurred is confirmed by the detail code. The detail code of "Advanced Synchronous Control Calculation Overflow (error code: 1AF1H)" are shown below.

Function of the module in which the overflow occurred	Detail code	Action
Composite main shaft gear	0001H	Reduce the speed of the main shaft main input axis and the main shaft sub input axis.
Main shaft gear	0002H	Reduce the gear ratio (gear numerator/gear denominator).
Main shaft clutch	0003H	Reduce the speed of the input axis when a clutch is used.
Main shaft module speed change gear	0004H	Reduce the speed change ratio (speed change ratio numerator/speed change ratio denominator).
Auxiliary shaft	0005H	Reduce the command set velocity of the axis set as the auxiliary shaft.
Auxiliary shaft gear	0006H	Reduce the gear ratio (gear numerator/gear denominator).
Auxiliary shaft clutch	0007H	Reduce the speed of the input axis when a clutch is used.
Auxiliary shaft module speed change gear	0008H	Reduce the speed change ratio (speed change ratio numerator/speed change ratio denominator).
Composite auxiliary shaft gear	0009H	Reduce the input value of the main shaft and the auxiliary shaft.
Output axis speed change gear	000AH	Reduce the speed change ratio (speed change ratio numerator/speed change ratio denominator).
Output axis phase compensation	000BH	Reduce the value of phase compensation advance time or set a longer time for the phase compensation time constant.
Output axis smoothing	000CH	Review the setting value of the time constant so that the number of smoothing steps reduce or increase the setting value of the operation cycle.
Output axis synchronous control change	000DH	Reduce the synchronous control change value or set a longer time for the synchronous control reflection time.
Output axis cam calculation	000EH	Review the cam waveform so that the stroke does not vary rapidly.

Required slave object

The following shows the setting condition of the slave object when using it as the Output axis (AdvOutPutName.Axis) of MCv_AdvancedSync (Advanced Synchronous Control).

Slave (Slave axis) Slave object necessary/not necessary	
Target position	**Position actual value**
Necessary	Not required
Not necessary	Necessary
Not necessary	Not necessary

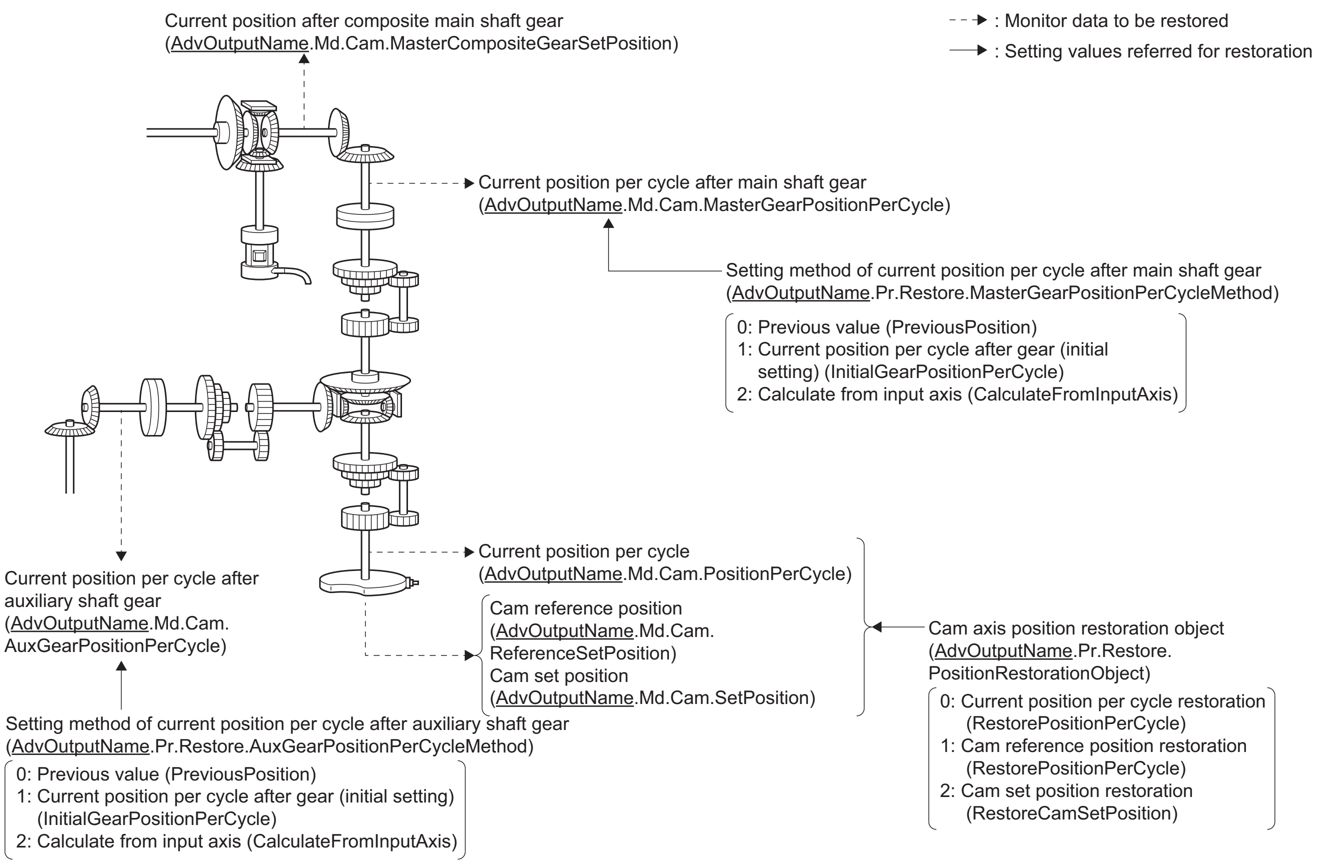
15.5 Synchronous Control Initial Position

The synchronous control monitor data can be aligned to a set position when starting MCv_AdvancedSync (Advanced Synchronous Control), as the initial position for synchronous control. Restoration is executed when MCv_AdvancedSync (Advanced Synchronous Control) is started. The alignment to a synchronous control initial position is useful for restoring a system based on the last control status along with restarting synchronous control after cancelling midway.
The following monitor data are restored to a synchronous control initial position. To restore the data, the axes set in the axis information of the input axis and output axis must be set to absolute position detection system.

Synchronous control monitor data		The position when starting synchronous control
Name	Variable name	
Current position after composite main shaft gear	AdvOutputName.Md.Cam.MasterCompositeGearSetPosition	Restored to a position based on the current position of the input axis of the main shaft.
Current position per cycle after main shaft gear	AdvOutputName.Md.Cam.MasterGearPositionPerCycle	Restored according to the Setting method of current position per cycle after main shaft gear (AdvOutputName.Pr.Restore.MasterGearPositionPerCycleMethod).
Current position per cycle after auxiliary shaft gear	AdvOutputName.Md.Cam.AuxGearPositionPerCycle	Restored according to the Setting method of current position per cycle after auxiliary shaft gear (AdvOutputName.Pr.Restore.AuxGearPositionPerCycleMethod).
Current position per cycle	AdvOutputName.Md.Cam.PositionPerCycle	Restored according to the Cam axis position restoration object (AdvOutputName.Pr.Restore.PositionRestorationObject).
Cam reference position	AdvOutputName.Md.Cam.ReferenceSetPosition	
Cam set position*1	AdvOutputName.Md.Cam.SetPosition	

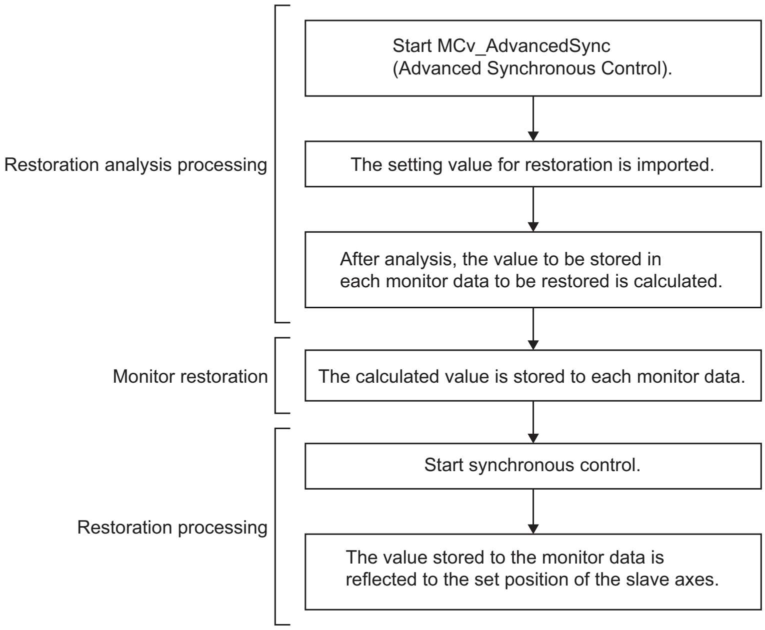
*1 The Set position (AxisName.Md.SetPosition) of the axis set as the output axis of the advanced synchronous control is calculated based on the cam set position.

The relationship between each module, the monitor data to be restored, and the setting values referred for restoration is shown below.



Restoration processing method

The restoration processing method is shown below. Restoration is executed in the following order: restoration analysis, monitor restoration, and restoration processing.



Backup data management for advanced synchronous control

For advanced synchronous control, the initial value of synchronous control monitor data and the data used to calculate and restore the synchronous control initial position are periodically saved to the latch drive as "abs_advsync.bin" file.
In order to back up data, the add-on AbsSystem must be valid.
When the axes set in the axis information of the output axis and input axis are not set to absolute position system, backup data is not generated. An "abs_advsync.bin" file of size up to the size specified in the Maximum backup RAM size (MotionControl_AdvancedSync_System.PrConst.AddonMemory.BackupRamSizeMax) can be saved. If the total size of the backup data exceeds the limit for backup data when adding the input axis and output axis, the variable that exceeds the limit will not be added.
When the total size of the backup data exceeds the limit, "Advanced Synchronous Control Backup Memory Insufficient Warning (warning code: 0F18H)" occurs.
The backup data size for an input axis and an output axis is shown below.

Axis type	Backup data size (k byte)
Input axis	0.1
Output axis	0.1

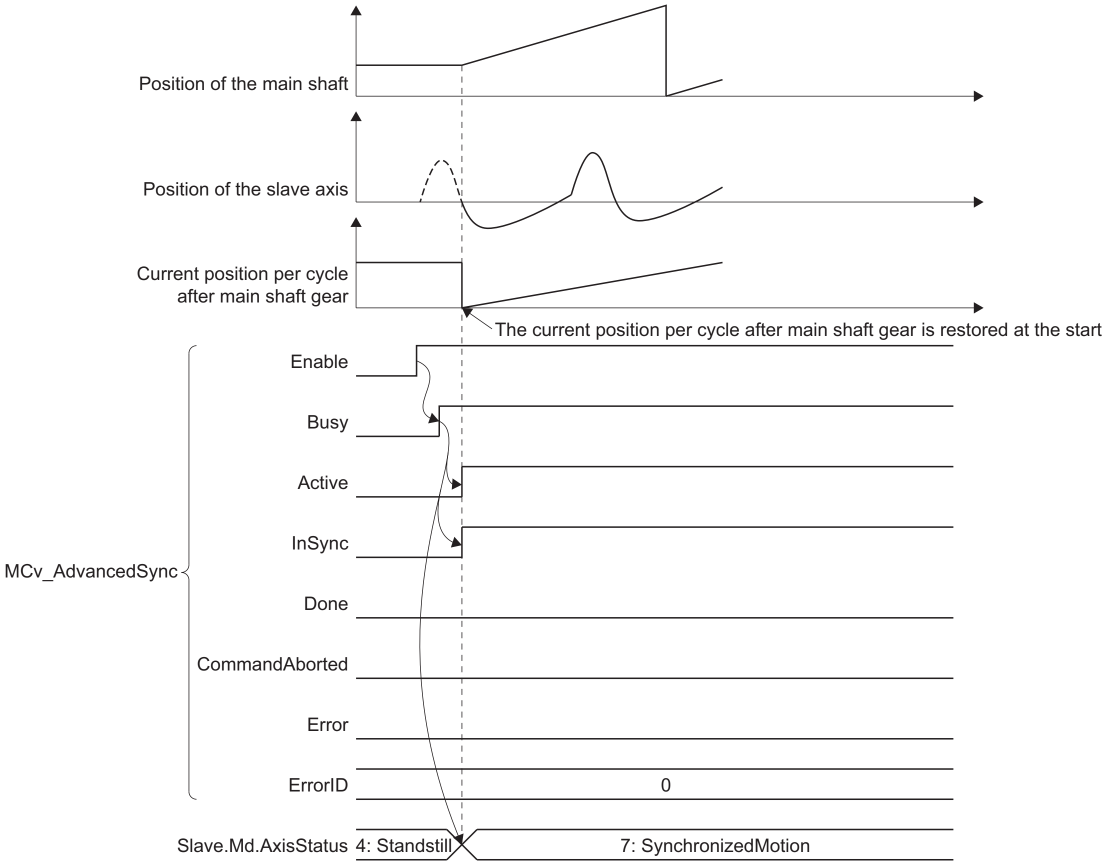
When turning the system power ON again after setting a smaller value set in the Maximum backup RAM size (MotionControl_AdvancedSync_System.PrConst.AddonMemory.BackupRamSizeMax) than the last time the system was powered ON, the "abs_advsync.bin" file will be regenerated, and all backup data will be cleared.
As the backup data is linked to the label IDs of the input axis and output axis, when the target axis No. specified in the output axis labels and input axis labels are changed before/after the system is powered ON again, restoration cannot be performed correctly because the links are broken. If restoration cannot be performed correctly, "Advanced Synchronous Control Restoration Incorrect Warning (warning code: 0D40H)" occurs.

Restoration method for current position after composite main shaft gear, current position per cycle after main shaft gear, and current position per cycle after auxiliary shaft gear at start of synchronous control

The initial position alignment and restoration of the current position after composite main shaft gear, the current position per cycle after main shaft gear, and the current position per cycle after auxiliary shaft gear are executed when MCv_AdvancedSync (Advanced Synchronous Control) is started.

Ex.

When the current position per cycle after main shaft gear is restored



Restoration method

The current position after composite main shaft gear, the current position per cycle after main shaft gear, and the current position per cycle after auxiliary shaft gear are restored as follows according to the operations executed on the axes assigned to the main shaft or auxiliary shaft before starting MCv_AdvancedSync (Advanced Synchronous Control).

Operation of the axes assigned to the main shaft and auxiliary shaft (before starting MCv_AdvancedSync (Advanced Synchronous Control))	Absolute position management setting (AxisName.PrConst.PosRestoration_AbsPosEnable) of the axes set in the axis	
	Absolute position detection system valid	Absolute position detection system invalid
Execution of MC_Home (Home Position Return)	Restoration method (1)	
Execution of MC_SetPosition (Current Position Change)	Restoration method (1)	
Transition of the Active status (AdvInputName.Md.ActiveStatus) of input axis from"0: Not connected" to "1: Input axis enabled".	Restoration method (2)	Restoration method (1)
Others	Restoration method (2)	

Precautions

- The input axis value is rounded to a value within the positioning range, and the difference between this value and the current position of the input axis when synchronous control was last conducted is used as the change amount in "Restoration method (2)".
- When the input axis movement since synchronous control was last conducted exceeds "±10000000000.0 [input axis unit]" before synchronous control starts, the change amount cannot be calculated correctly.

■ Restoration method (1)

Setting item	Restoration method	Calculation formula
Current position after composite main shaft gear	The new current position after composite main shaft gear is calculated based on the input axis cumulative current position of the main shaft.	Input direction*1 of composite main shaft gear × Cumulative current position of the input axis (main shaft)
Current position per cycle after main shaft gear	The new current position per cycle after main shaft gear is calculated based on the current position after composite main shaft gear.	Main shaft gear ratio*2 × Current position after composite main shaft gear
Current position per cycle after auxiliary shaft gear	The new current position per cycle after auxiliary shaft gear is calculated based on the input axis cumulative current position of the auxiliary shaft.	Auxiliary shaft gear ratio*2 × Cumulative current position of the input axis (auxiliary shaft)

*1 The "Input direction of composite main shaft gear" value in the formula is as follows according to the Composite main shaft gear (AdvOutputName.Pr.Gear.MasterCompositeGear).

Composite main shaft gear (AdvOutputName.Pr.Gear.MasterCompositeGear)	Setting value for "Input direction of composite main shaft gear"
No input (H□□□0)	0
Input + (H□□□1)	1
Input - (H□□□2)	-1

*2 The gear ratio of the main and auxiliary shafts is calculated by the following formula.

Item	Calculation formula
Main shaft gear ratio	$\dfrac{\text{Main shaft gear: Numerator (AdvOutputName.Pr.Gear.MasterNumerator)}}{\text{Main shaft gear: Denominator (AdvOutputName.Pr.Gear.MasterDenominator)}}$
Auxiliary shaft gear ratio	$\dfrac{\text{Auxiliary shaft gear: Numerator (AdvOutputName.Pr.Gear.AuxNumerator)}}{\text{Auxiliary shaft gear: Denominator (AdvOutputName.Pr.Gear.AuxDenominator)}}$

■ Restoration method (2)

The movement amount of the main shaft, sub input axis, or auxiliary shaft from the last synchronous control session is reflected to the current position after composite main shaft gear, the current position per cycle after main shaft gear, and the current position per cycle after auxiliary shaft gear.

Setting item	Restoration method	Calculation formula
Current position after composite main shaft gear	The movement amount of the main shaft from the last synchronous control session is reflected to the current position after composite main shaft gear.	Current position after composite main shaft gear at the last synchronous control session + Input direction of composite main shaft gear × Amount of change in the cumulative current position of input axis (main shaft) from the last synchronous control session
Current position per cycle after main shaft gear	The movement amount from the last synchronous control session is reflected to the current position per cycle after main shaft gear. The movement amount of the main shaft from the last synchronous control session is reflected to the current position after composite main shaft gear and the current position per cycle after main shaft gear.	Current position per cycle after main shaft gear at the last synchronous control session + Main shaft gear ratio × Amount of change in the current position after composite main shaft gear from the last synchronous control session
Current position per cycle after auxiliary shaft gear	The movement amount from the last synchronous control session is reflected to the current position per cycle after auxiliary shaft gear. The movement amount of the auxiliary shaft from the last synchronous control session is reflected to the current position per cycle after auxiliary shaft gear.	Current position per cycle after auxiliary shaft gear at the last synchronous control session + Auxiliary shaft gear ratio × Amount of change in the cumulative current position of input axis (auxiliary shaft) from the last synchronous control session

Point

"The last synchronous control session" indicates the status immediately before MCv_AdvancedSync (Advanced Synchronous Control) was stopped and the Synchronization status (AdvOutputName.Md.SyncStatus) is changed to "0: Not synchronizing" as follows.

- Immediately before MCv_AdvancedSync (Advanced Synchronous Control) is stopped by a stop command, error, etc.
- Immediately before the power supply of the Motion system was turned OFF

Cam axis position at synchronous control start

The cam axis position is composed of the relationship of the three positions "Current position per cycle", "Cam reference position", and "Cam set position". The remaining position can be restored by determining two positions when starting MCv_AdvancedSync (Advanced Synchronous Control).

Determine which position to restore by selecting one of the three settings below in the Cam axis position restoration object (AdvOutputName.Pr.Restore.PositionRestorationObject).

- 0: Current position per cycle restoration (RestorePositionPerCycle)
- 1: Cam reference position restoration (RestoreReferenceSetPosition)
- 2: Cam set position restoration (RestoreCamSetPosition)

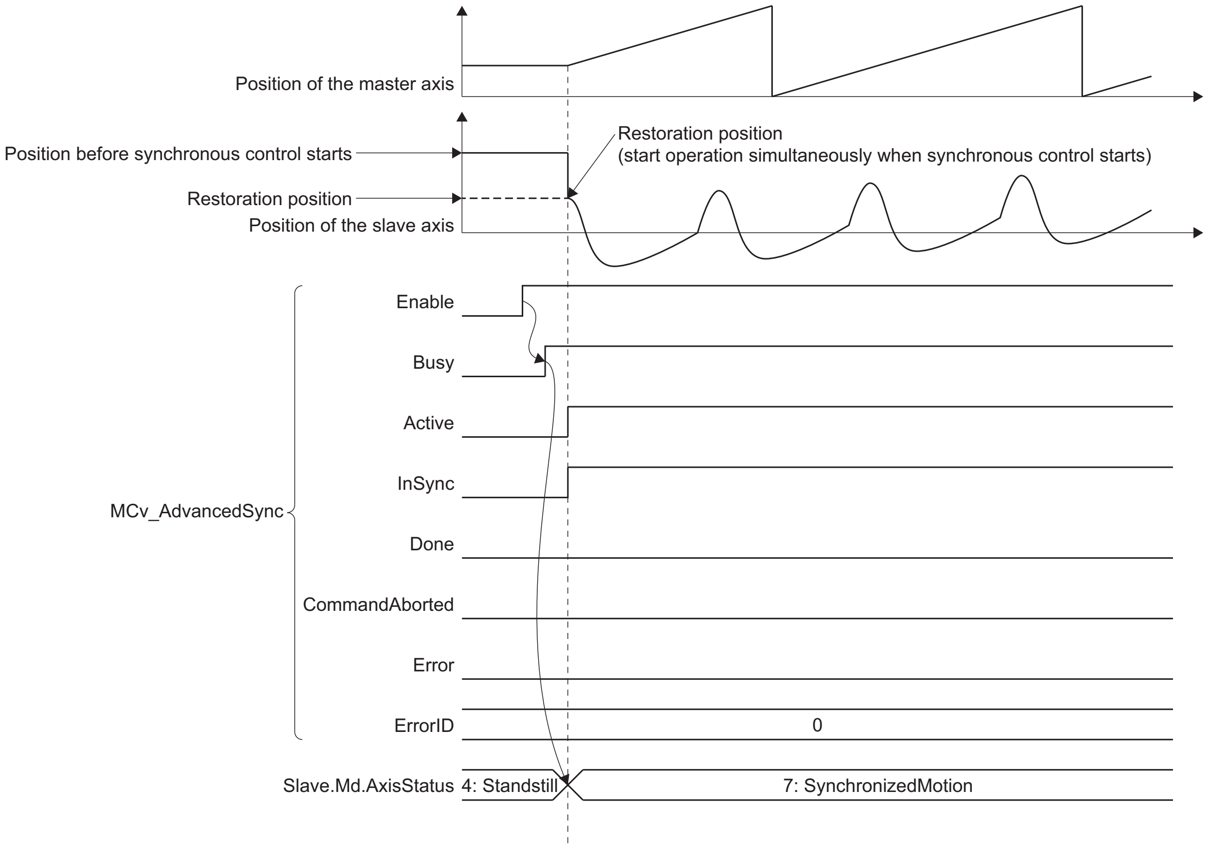
The parameters that need to be set for the cam axis position restoration are shown below. (For details of the settings, refer to synchronous control initial position parameter.) (☞ Page 602 Variables related to synchronous control initial position)

○: Required, △: Required when using initial setting value, —: Not required

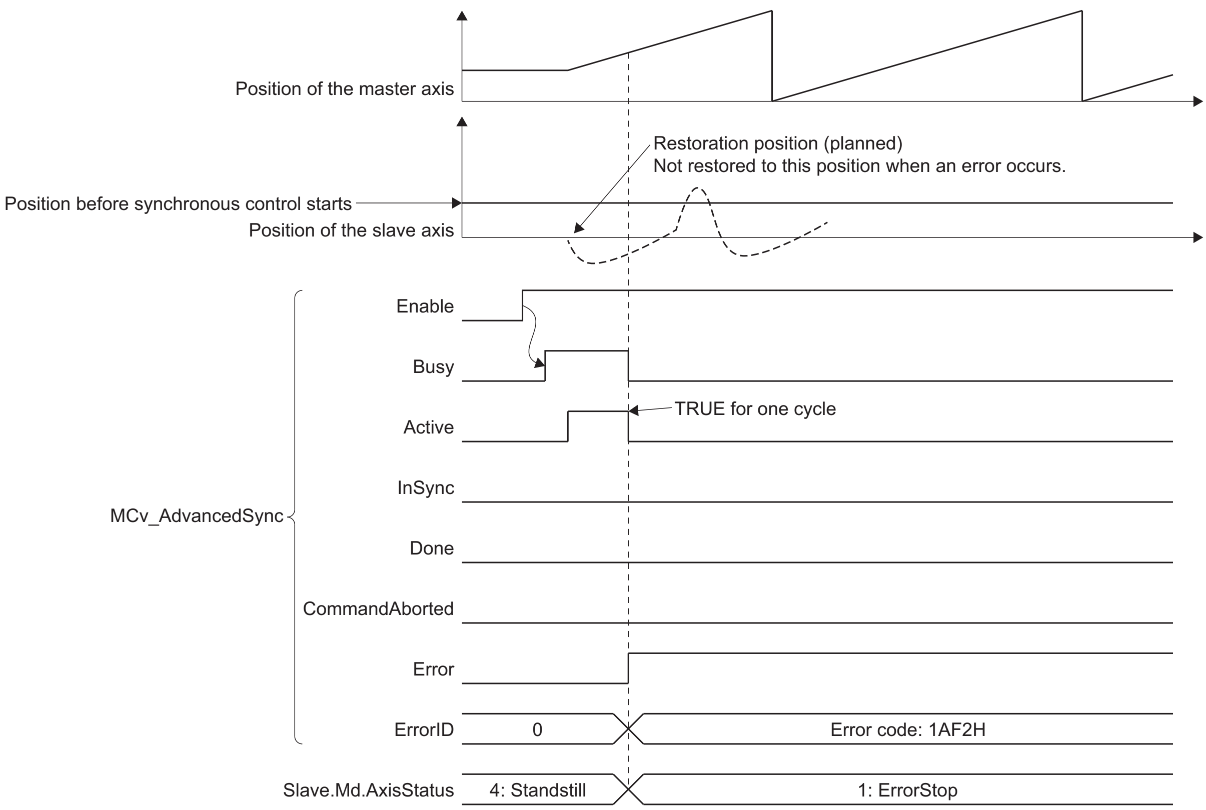
Cam axis position restoration object (AdvOutputName.Pr.Restore.PositionRestorationObject)	Setting method of cam reference position (AdvOutputName.Pr.Restore.ReferenceSetPositionMethod)	Cam reference position (initial setting) (AdvOutputName.Pr.Restore.InitialReferenceSetPosition)	Setting method of current position per cycle (AdvOutputName.Pr.Restore.PositionPerCycleMethod)	Current position per cycle (initial setting) (AdvOutputName.Pr.Restore.InitialPositionPerCycle)	Restoration processing details
0: Current position per cycle restoration (RestorePositionPerCycle)	○	△	—	○ (Used as search starting position)	Restores "current position per cycle" based on "cam reference position" and "cam set position".
1: Cam reference (RestoreReferenceSetPosition)	—	—	○	△	Restores "cam reference position" based on "current position per cycle" and "cam set position".
2: Cam set position restoration (RestoreCamSetPosition)	○	△	○	△	Restores "cam set position" based on "current position per cycle" and "cam reference position".

The initial position alignment and restoration of the cam axis position are executed when MCv_AdvancedSync (Advanced Synchronous Control) is started.

- A timing chart when cam restoration is normally executed



- A timing chart when an error occurs in cam output axis restoration

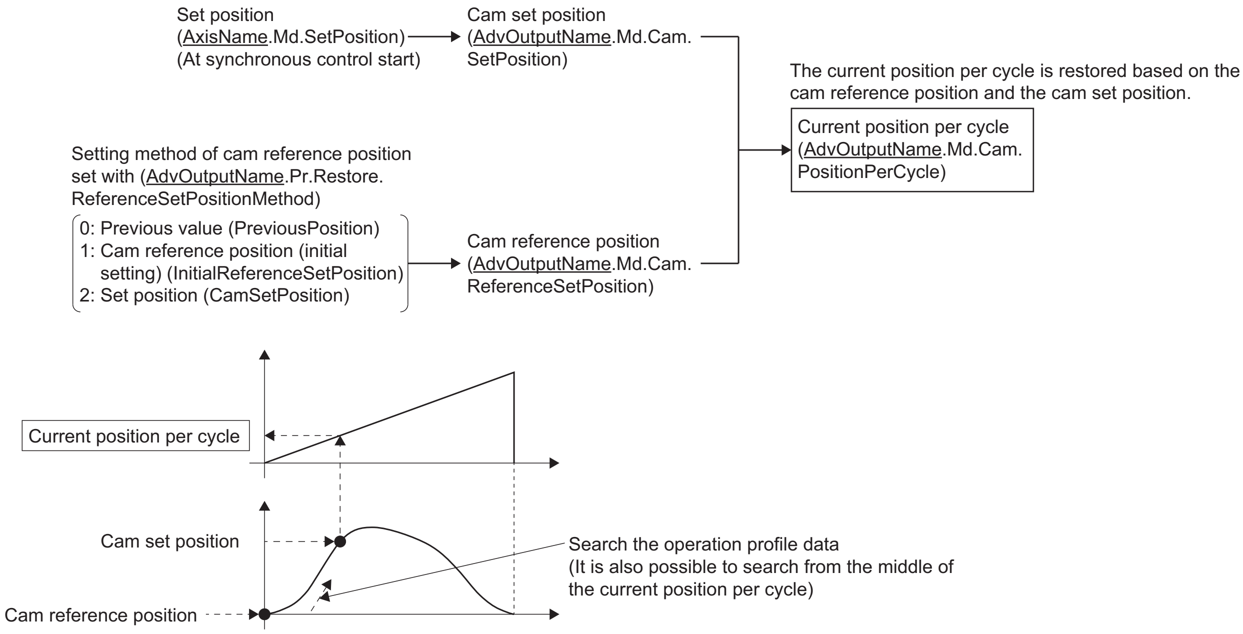


Current position per cycle restoration

If Cam axis position restoration object (AdvOutputName.Pr.Restore.PositionRestorationObject) is set to "0: Current position per cycle restoration (RestorePositionPerCycle)" when starting synchronous control, the current position per cycle is restored based on the cam reference position and the cam set position, and MCv_AdvancedSync (Advanced Synchronous Control) starts.

Set the cam reference position to be used for restoration with a parameter. The cam set position uses the set position of the output axis set to the slave axis when MCv_AdvancedSync (Advanced Synchronous Control) starts.

The current position per cycle restoration is calculated by searching from the beginning to the end of the cam pattern for the corresponding current position per cycle. The starting position for searching the cam pattern is set with the Current position per cycle (initial setting) (AdvOutputName.Pr.Restore.InitialPositionPerCycle). (It is also possible to search the return path in a two-way cam pattern operation.)



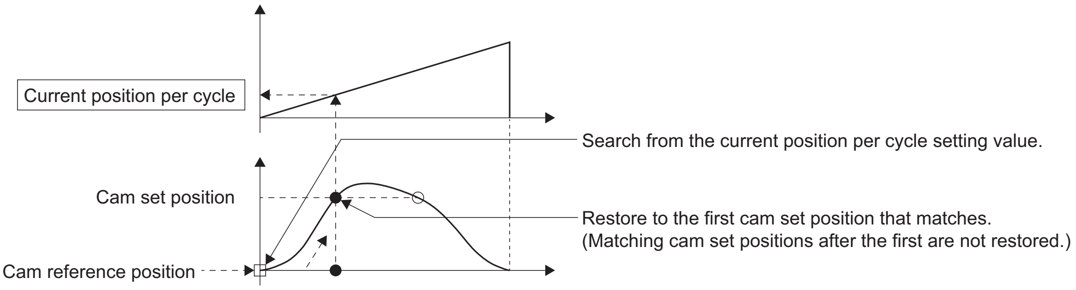
*1 The current position per cycle indicated by [ ] is restored.

- With two-way cam pattern operation, if the corresponding current position per cycle is not found, "Advanced Synchronous Control Restoration Disabled (error code: 1AF2H)" occurs, and MCv_AdvancedSync (Advanced Synchronous Control) cannot start.
- With a feed operation cam pattern, if the corresponding current position per cycle is not found on the first cycle, the cam reference position is changed automatically and the pattern is searched again.
- If the cam resolution is large, search processing may take a long time when starting synchronous control. For an approximate guide of the searching time, refer to the following.
☞ Page 605 Restoration time of the current position per cycle
- When the Ring counter lower limit value (AxisName.PrConst.RingCount_LowerValue) of the output axis positioning range is "0.0", the current position is not restored in a cam stroke range that crosses over "0.0", and "Advanced Synchronous Control Restoration Disabled (error code: 1AF2H)" occurs. Execute the current position per cycle restoration within a cam stroke range between 0 and the Ring counter upper limit value (AxisName.PrConst.RingCount_UpperValue) of the positioning range.
- When the difference between the cam reference position and the cam set position exceeds "10000000000.0" at restoration, the current position per cycle cannot be restored correctly. Adjust the values so that the difference between the cam reference position and the cam set position is smaller than "10000000000.0" for restoration.

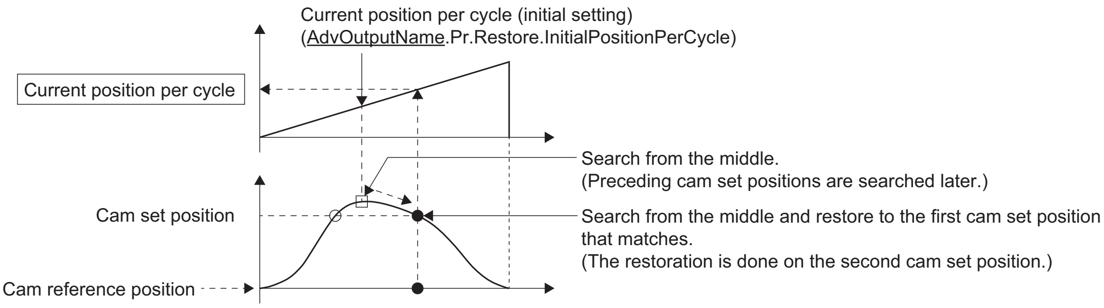
The following shows details of the current position per cycle restoration operation.

■ Current position per cycle restoration operation with a "two-way cam pattern operation"

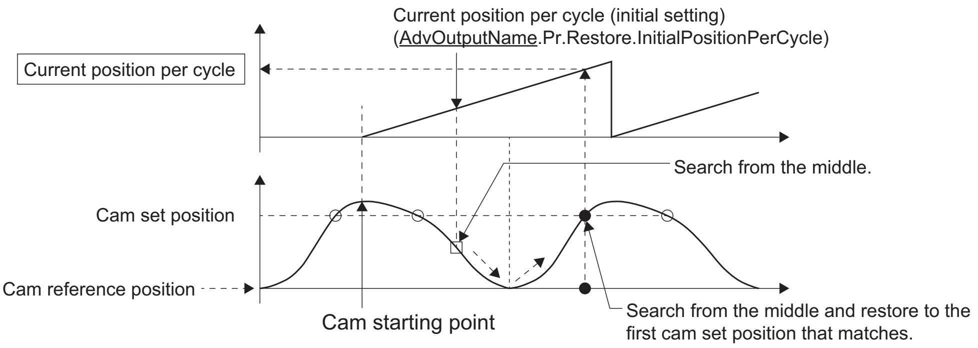
- Search from "Current position per cycle = 0" (Cam starting point = 0)



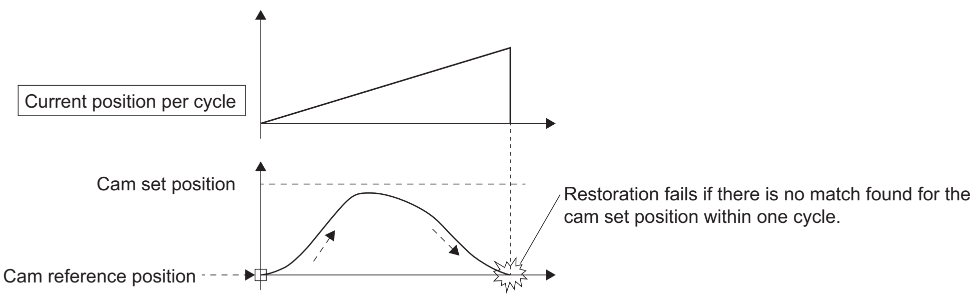
- Search from the middle of the current position per cycle (Cam starting point = 0)



- Search from the middle of the current position per cycle (Cam starting point ≠ 0)

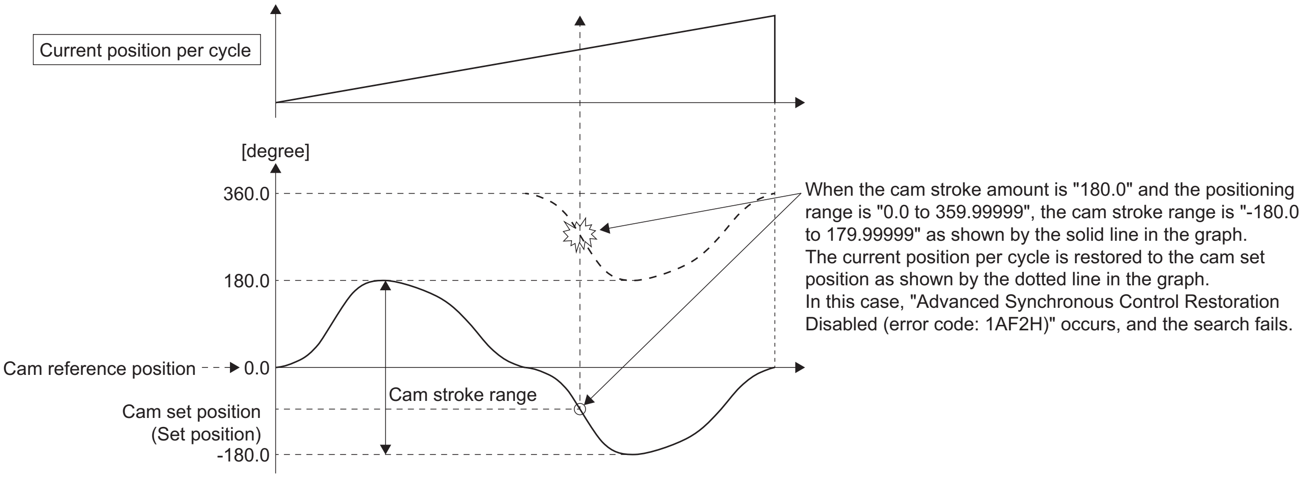


- The search fails
  - When a matching cam set position is not found

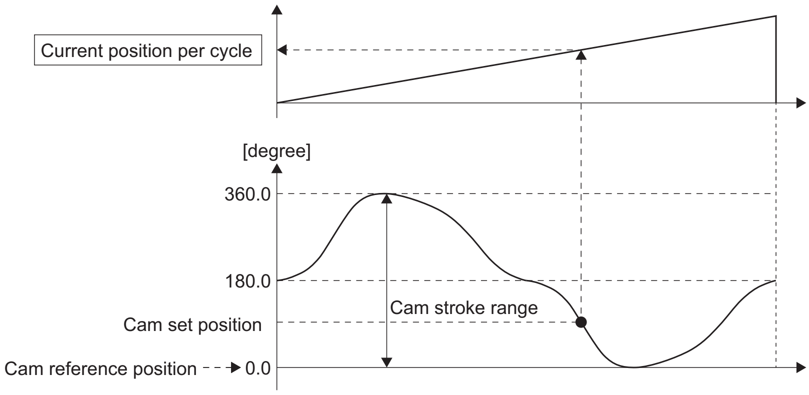


15

• When the current position restoration is executed in a cam stroke range that crosses over "0.0" (the positioning range is "0.0 to 360.0degrees")

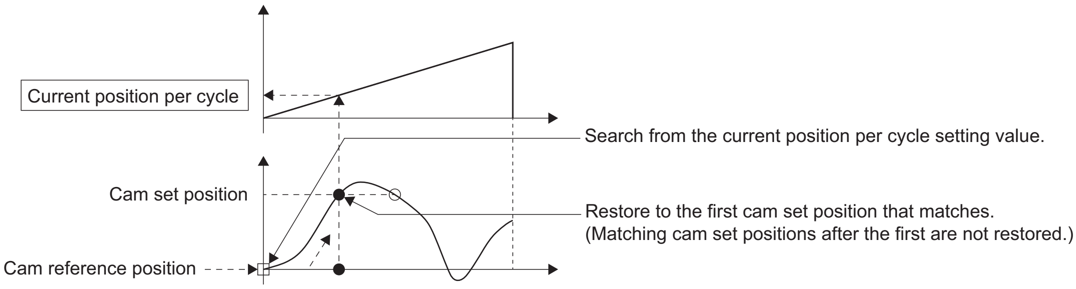


• In the case above, execute current position restoration within a cam stroke range of the positioning range (0.0 to 360.0 degrees).

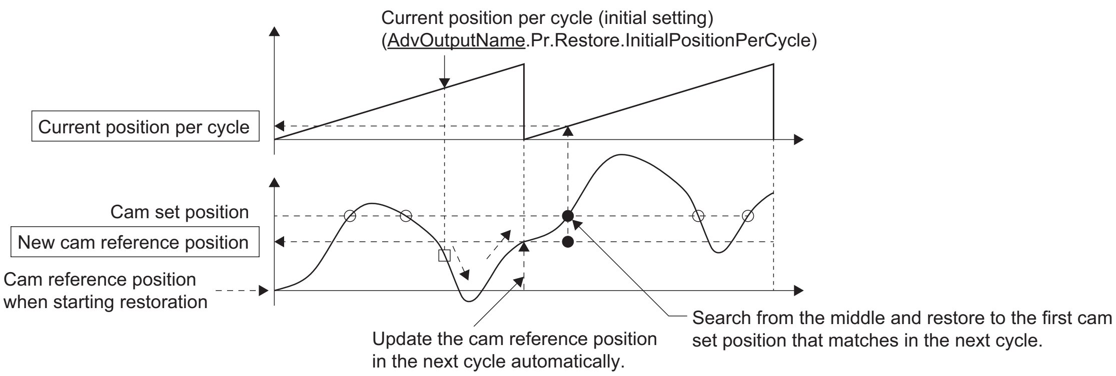


■ Current position per cycle restoration operation with a "feed operation cam pattern"

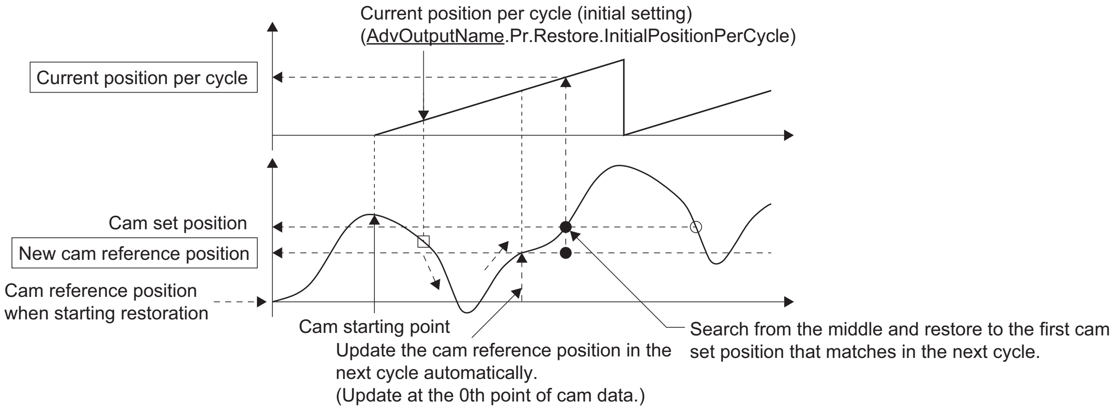
• Search from "Current position per cycle = 0" (Cam starting point = 0)



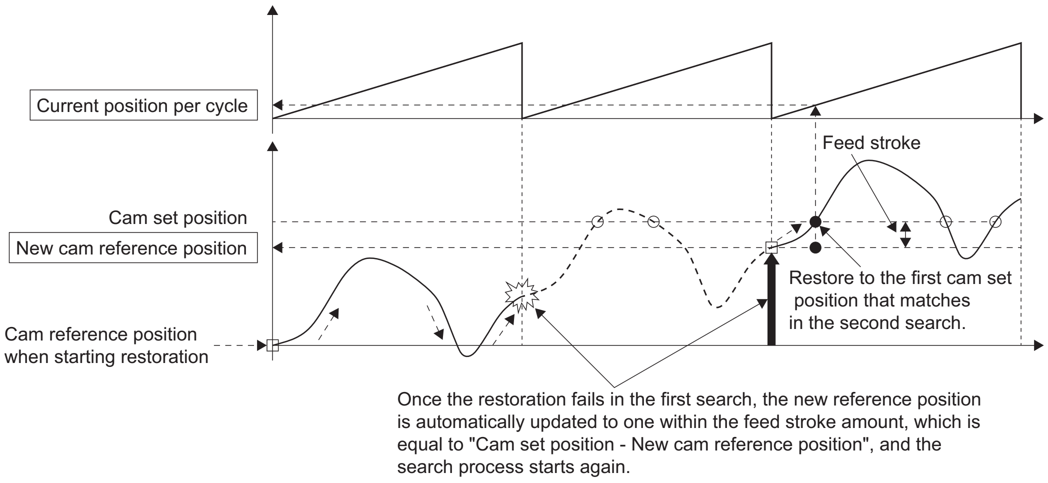
• Search from the middle of the current position per cycle (Cam starting point = 0)



- Search from the middle of the current position per cycle (Cam starting point ≠ 0)



- The first search fails and a search begins for the second time



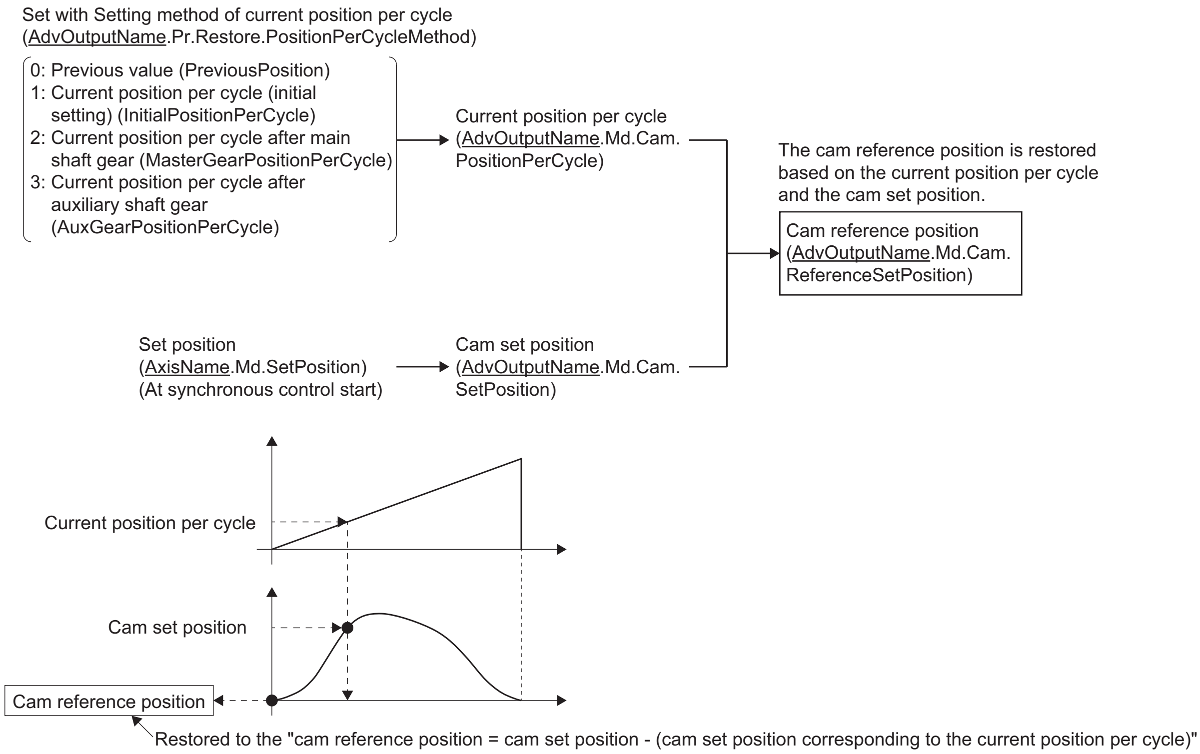
Point

If the first search fails, a second search may not be processed on the next cycle for a cam pattern with a feed stroke that is smaller than 100% of the stroke as above.
The intended current position per cycle can be found in the first search, by setting or positioning the cam reference position in advance.

Cam reference position restoration

If Cam axis position restoration object (AdvOutputName.Pr.Restore. PositionRestorationObject) is set to "1: Cam reference position restoration (RestorePositionPerCycle)" when starting synchronous control, the cam reference position is restored based on the current position per cycle and the cam command set position.

Set the current position per cycle to be used for restoration with the parameter.

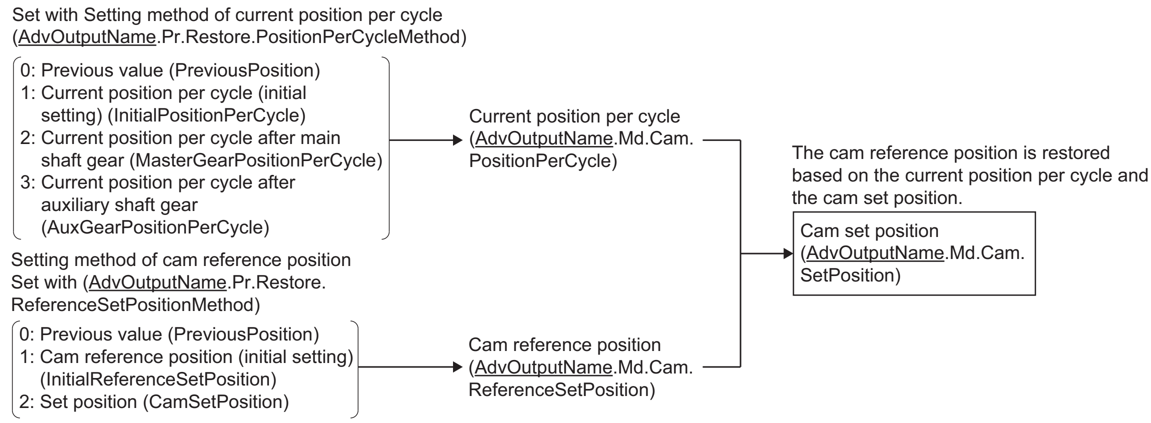


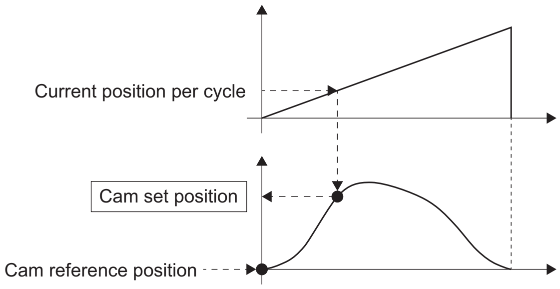
*1 The cam reference position indicated by ☐ is restored.

Cam set position restoration

If Cam axis position restoration object (AdvOutputName.Pr.Restore. PositionRestorationObject) is set to "2: Cam set position restoration (RestoreCamSetPosition)" when starting synchronous control, the cam set postition is restored based on the current position per cycle and the cam reference position.

Set the current position per cycle and cam reference position to be used for restoration with the parameter.





*1 The cam set position indicated by ☐ is restored.

Point

The cam moves to its restored cam set position immediately after starting synchronous control when the restored cam set position is different from the set position at synchronous control start.

■ Cam set position restoration allowable movement setting value

When MCv_AdvancedSync (Advanced Synchronous Control) is started, if the difference between the restored cam set position and the set position of the slave axis is larger than the cam set position restoration allowable movement setting value, "Advanced Synchronous Control Restoration Disabled (error code: 1AF2H)" occurs, and synchronous control cannot start.

When using cam set position restoration, before starting MCv_AdvancedSync (Advanced Synchronous Control) calculate the corresponding cam set position with MCv_AdvPositionPerCycleCalc (Advanced Synchronous Control Current Position Per Cycle Calculation) or with MCv_AdvCamSetPositionCalc (Advanced Synchronous Control Cam Set Position Calculation) or with synchronous control analysis mode. Start synchronous control after positioning to the correct cam set position.

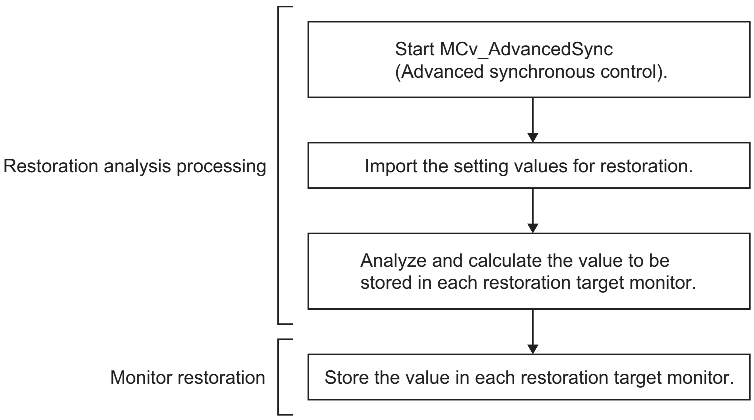
Synchronous control analysis mode

With synchronous control analysis mode, synchronous control restoration is only analyzed when there is a command to start MCv_AdvancedSync (Advanced Synchronous Control). This mode is used to confirm the synchronous positions of the output axes in order to align axes with position control before starting MCv_AdvancedSync (Advanced Synchronous Control).
After starting MCv_AdvancedSync (Advanced Synchronous Control), when synchronous control analysis mode setting (Options (Options): Bit 16) of MCv_AdvancedSync (Advanced Synchronous Control) is "TRUE (Enabled)", operation enters synchronous control analysis mode.
When analysis is completed, Execution completion (Done) of MCv_AdvancedSync (Advanced Synchronous Control) turns to "TRUE", and the Output axis monitor data (AdvOutputName.Md.□) that are not synchronized in Synchronization status (AdvOutputName.Md.SyncStatus) are all updated. The following Output axis monitor data (AdvOutputName.Md.□) are also updated to the restored values. In this case, synchronous control analysis mode setting (Options (Options): Bit 16) of MCv_AdvancedSync (Advanced Synchronous Control) will not turn to "FALSE (Disabled)". When starting synchronous control without using the synchronous control analysis mode, turn synchronous control analysis mode setting (Options (Options): Bit 16) of MCv_AdvancedSync (Advanced Synchronous Control) to "FALSE (Disabled)".

Output axis monitor data (AdvOutputName.Md.□)
• Current position after composite main shaft gear (AdvOutputName.Md.Cam.MasterCompositeGearSetPosition) • Current position per cycle after main shaft gear (AdvOutputName.Md.Cam.MasterGearPositionPerCycle) • Current position per cycle after auxiliary shaft gear (AdvOutputName.Md.Cam.AuxGearPositionPerCycle) • Current position per cycle (AdvOutputName.Md.Cam.PositionPerCycle) • Cam reference position (AdvOutputName.Md.ReferenceSetPosition) • Cam set position (AdvOutputName.Md.Cam.SetPosition)

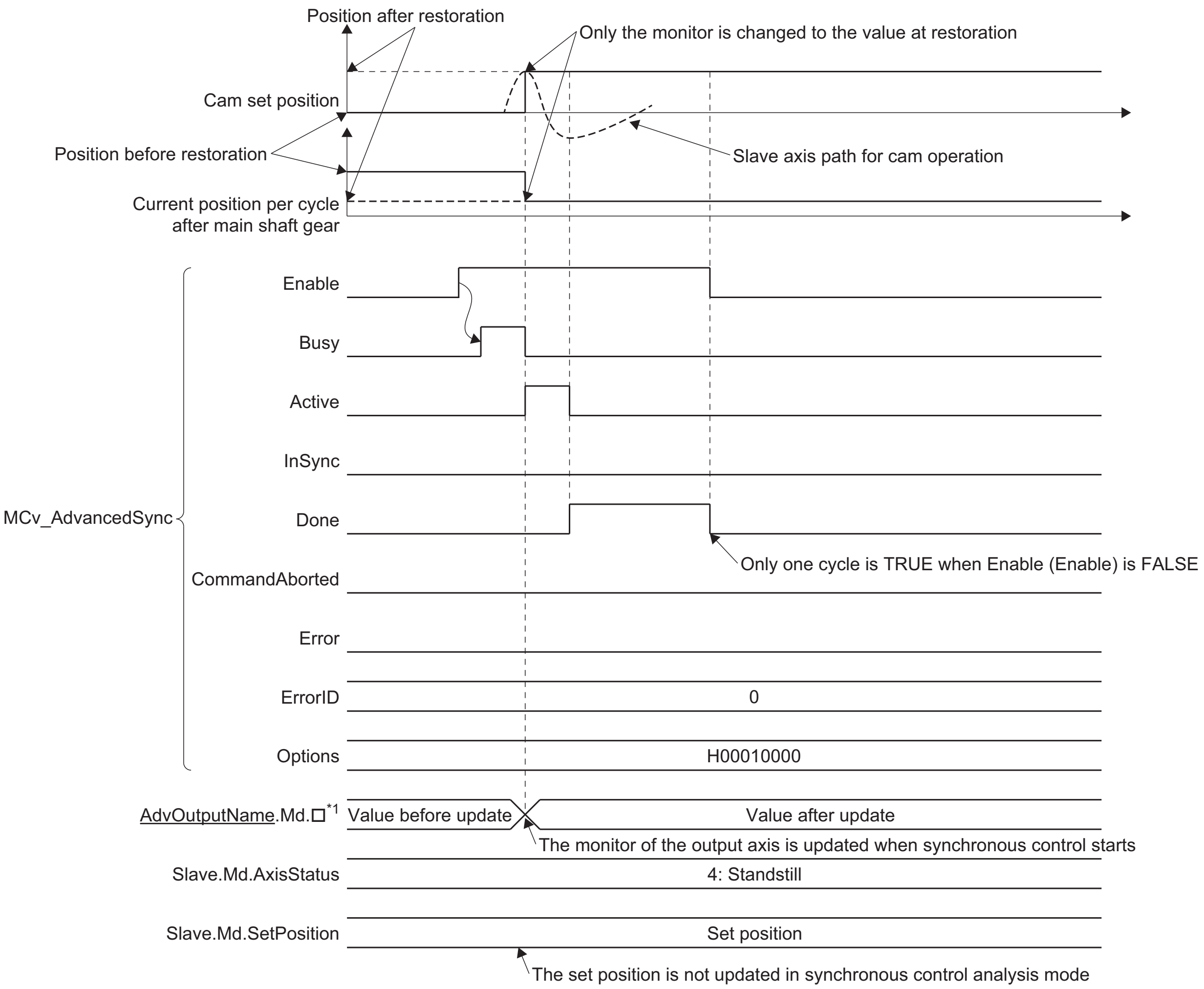
Processing in synchronous control analysis mode

In synchronous control analysis mode, unlike normal restoration, processing is executed until monitor restoration.



Synchronous control analysis mode will not operate if the Axis status (AxisName.Md.AxisStatus) of the Slave axis (Slave) of MCv_AdvancedSync (Advanced synchronous control) is other than "4: Standby (Standstill)".
When MCv_AdvancedSync (Advanced synchronous control) is started in synchronous control analysis mode, if the cam current position per cycle cannot be restored, "Advanced Synchronous Control Restoration Disabled (error code: 1AF2H)" occurs.

Timing chart of synchronous control analysis mode



*1 Synchronization status (AdvOutputName.Md.SyncStatus) of the Output axis monitor data remains "0: Not synchronizing" and is not updated.

15

Variables related to synchronous control initial position

Synchronous control initial position parameter (AdvOutputName.Pr.Restore.)

Variable name	Name	Import[*1]	Data type[*1]	Setting value	Initial value
MasterGearPositionPerCycleMethod	Setting method of current position per cycle after main shaft gear	At start	INT (MC_GEAR_RESTORE_METHOD)	0: Previous value (PreviousPosition) 1: Current position per cycle after main shaft gear (initial setting) (InitialGearPositionPerCycle) 2: Calculate from input axis (CalculateFromInputAxis)	0: Previous value (PreviousPosition)
AuxGearPositionPerCycleMethod	Setting method of current position per cycle after auxiliary shaft gear		INT (MC_GEAR_RESTORE_METHOD)	0: Previous value (PreviousPosition) 1: Current position per cycle after auxiliary shaft gear (initial setting) (InitialGearPositionPerCycle) 2: Calculate from input axis (CalculateFromInputAxis)	0: Previous value (PreviousPosition)
PositionRestorationObject	Cam axis position restoration object		INT (MC_CAM_RESTORE_METHOD)	0: Current position per cycle restoration (RestorePositionPerCycle) 1: Cam reference position restoration (RestoreReferenceSetPosition) 2: Cam set position restoration (RestoreCamSetPosition)	0: Current positionper cycle restoration (RestorePositionPerCycle)
ReferenceSetPositionMethod	Setting method of cam reference position		INT (MC_CAM_REFERENCE_METHOD)	0: Previous value (PreviousPosition) 1: Cam reference position (initial setting) (InitialReferenceSetPosition) 2: Set position (CamSetPosition)	0: Previous value (PreviousPosition)
PositionPerCycleMethod	Setting method of current position per cycle		INT (MC_CAM_CYCLE_METHOD)	0: Previous value (PreviousPosition) 1: Current position per cycle (initial setting) (InitialPositionPerCycle) 2: Current position per cycle after main shaft gear (MasterGearPositionPerCycle) 3: Current position per cycle after auxiliary shaft gear (AuxGearPositionPerCycle)	0: Previous value (PreviousPosition)
MasterGearInitialPositionPerCycle	Current position per cycle after main shaft gear (initial setting)		LREAL	0.0 to 2147483647.0	0.0
AuxGearInitialPositionPerCycle	Current position per cycle after auxiliary shaft gear (initial setting)		LREAL	0.0 to 2147483647.0	0.0
InitialReferenceSetPosition	Cam reference position (initial setting)		LREAL	-10000000000.0 to 10000000000.0	0.0
InitialPositionPerCycle	Current position per cycle (initial setting)		LREAL	0.0 to 2147483646.0	0.0
RestorationAllowablePosition	Cam set position restoration: Allowable movement amount setting		DWORD(UDINT)	0 to 2147483647 [pulse]	25600

*1 For details on importing and data types, refer to "List of Variables" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Setting method of current position per cycle after main shaft gear (AdvOutputName.Pr.Restore.MasterGearPositionPerCycleMethod)

Select tha setting method of Current position per cycle after main shaft gear (AdvOutputName.Md.Cam.MasterGearPositionPerCycle) when starting MCv_AdvancedSync (Advanced Synchronous Control).

Setting value	Description
0: Previous value (PreviousPosition)	The current position per cycle after main shaft gear from the previous synchronous control is stored.
1: Current position per cycle after main shaft gear (initial setting) (InitialGearPositionPerCycle)	The value set in Current position per cycle after main shaft gear (initial setting) (AdvOutputName.Pr.Restore.MasterGearInitialPositionPerCycle) is stored.
2: Calculate from input axis (CalculateFromInputAxis)	The value calculated based on the current position after composite main shaft gear is stored.

■ Setting method of current position per cycle after auxiliary shaft gear (AdvOutputName.Pr.Restore.AuxGearPositionPerCycleMethod)

Select tha Setting method of for current position per cycle after auxiliary shaft gear (AdvOutputName.Md.Cam.AuxGearPositionPerCycle) when starting MCv_AdvancedSync (Advanced Synchronous Control).

Setting value	Description
0: Previous value (PreviousPosition)	The current position per cycle after auxiliary shaft gear from the previous synchronous control is stored.
1: Current position per cycle after auxiliary shaft gear (initial setting) (InitialGearPositionPerCycle)	The value set in Current position per cycle after auxiliary shaft gear (initial setting) (AdvOutputName.Pr.Restore.AuxGearInitialPositionPerCycle) is stored.
2: Calculate from input axis (CalculateFromInputAxis)	The value calculated based on the current position of the auxiliary shaft is stored.

■ Cam axis position restoration object (AdvOutputName.Pr.Restore.PositionRestorationObject)

Select the restoration object from "Current position per cycle" or "Cam reference position" or "Cam set position" when starting synchronous control.

Setting value	Description
0: Current position per cycle restoration (RestorePositionPerCycle)	The current position per cycle from "Cam reference position" and "Cam set position" is restored.
1: Cam reference position restoration (RestoreReferenceSetPosition)	The cam reference position from "Current position per cycle" and "Cam set position" is restored.
2: Cam set position restoration (RestoreCamSetPosition)	The cam set position from "Current position per cycle" and "Cam reference position" is restored.

■ Setting method of cam reference position (AdvOutputName.Pr.Restore.ReferenceSetPositionMethod)

Select the setting method for the cam reference position to be used for restoration when Cam axis position restoration object (AdvOutputName.Pr.Restore.PositionRestorationObject) is set to "0: Current position per cycle restoration (RestorePositionPerCycle)" or "2: Cam set position restoration (RestoreCamSetPosition)."

Setting value	Description
0: Previous value (PreviousPosition)	The cam reference position from the previous synchronous control is stored. The set position is stored when the cam reference position from the previous synchronous control is not saved.
1: Cam reference position (initial setting) (InitialReferenceSetPosition)	The value set in Cam reference position (initial setting) (AdvOutputName.Pr.Restore.InitialReferenceSetPosition) is stored.
2: Set position (CamSetPosition)	The value set in Set position (AxisName.Md.SetPosition) of the slave axis is stored.

■ Setting method of current position per cycle (AdvOutputName.Pr.Restore.PositionPerCycleMethod)

Set the setting method for the current position per cycle to be used for restoration when Cam axis position restoration object (AdvOutputName.Pr.Restore.PositionRestorationObject) is set to "1: Cam reference position restoration (RestorePositionPerCycle)" or "2: Cam set position restoration (RestoreCamSetPosition)".

Setting value	Description
0: Previous value (PreviousPosition)	The current position per cycle from the previous synchronous control is stored as is.
1: Current position per cycle (initial setting) (InitialPositionPerCycle)	The value set in Current position per cycle (initial setting) (AdvOutputName.Pr.Restore.InitialPositionPerCycle) is stored.
2: Current position per cycle after main shaft gear (MasterGearPositionPerCycle)	Current position per cycle after main shaft gear (AdvOutputName.Md.Cam.MasterGearPositionPerCycle) is stored.
3: Current position per cycle after auxiliary shaft gear (AuxGearPositionPerCycle)	Current position per cycle after auxiliary shaft gear (AdvOutputName.Md.Cam.AuxGearPositionPerCycle) is stored.

■ Current position per cycle after main shaft gear (initial setting) (AdvOutputName.Pr.Restore.MasterGearInitialPositionPerCycle)

Set the initial setting value of the current position per cycle after main shaft gear when Setting method of current position per cycle after main shaft gear (AdvOutputName.Pr.Restore.MasterGearPositionPerCycleMethod) is set to "1: Current position per cycle after gear (Initial setting) (InitialGearPositionPerCycle)".
The setting unit is the unit of the input axis set to Master axis (Master) of MCv_AdvancedSync (Advanced Synchronous Control). Set within the range from "0.0 to (Length per cycle - 0.00001)".

■ Current position per cycle after auxiliary shaft gear (initial setting) (AdvOutputName.Pr.Restore.AuxGearInitialPositionPerCycle)

Set the initial setting value of the current position per cycle after auxiliary shaft gear when Setting method of current position per cycle after auxiliary shaft gear (AdvOutputName.Pr.Restore.AuxGearPositionPerCycleMethod) is set to "1: Current position per cycle after gear (initial setting) (InitialGearPositionPerCycle)".
The setting unit is the unit of the input axis set to Master axis (Master) of MCv_AdvancedSync (Advanced Synchronous Control). Set within the range from "0.0 to (Length per cycle - 0.00001)".

■ Cam reference position(initial setting) (AdvOutputName.Pr.Restore.InitialReferenceSetPosition)

Set the initial value of the cam reference position in the units of the axis set to the output axis Axis information (AdvOutputName.Pr.Restore.ReferenceSetPositionMethod) is set to "1: Cam reference position (initial setting) (InitialReferenceSetPosition)".

■ Current position per cycle (initial setting) (AdvOutputName.Pr.Restore.InitialPositionPerCycle)

Set a value according to the setting for Cam axis position restoration object (AdvOutputName.Pr.Restore.PositionRestorationObject).
The setting unit is the unit of the input axis set to Master axis (Master) of MCv_AdvancedSync (Advanced Synchronous Control). Set within the range from "0.0 to (Length per cycle - 0.00001)".

Cam axis position restoration object (AdvOutputName.Pr.Restore.PositionRestorationObject)	Description
0: Current position per cycle restoration (RestorePositionPerCycle)	Set the starting point for search processing to restore the current position per cycle. Set to restore the position on the return path of a two-way operation cam pattern. For details on search processing, refer to the following. ☞ Page 594 Current position per cycle restoration
1: Cam reference position restoration (RestoreReferenceSetPosition)	Set the initial setting value for the current position per cycle when Setting method of current position per cycle (AdvOutputName.Pr.Restore.PositionPerCycleMethod) is set to "1: Current position per cycle (initial setting) (InitialPositionPerCycle)".
2: Cam set position restoration (RestoreCamSetPosition)	

■ Cam set position restoration: Allowable movement amount setting (AdvOutputName.Pr.Restore.RestorationAllowablePosition)

Set the allowable value of the difference between the restored cam set position and the set position in units of the servo command value of the output axis when Cam axis position restoration object (AdvOutputName.Pr.Restore.PositionRestorationObject) is set to "2: Cam set position restoration (RestoreCamSetPosition).
If the setting value is large, a rapid operation may occur when starting synchronous control.

Point

The MR-J5(W)-G servo parameter "In-position range (PA10)" initial value (25600) is set as the initial value.

Precautions

Accuracy at restoration

Since accuracy of the advanced synchronous control output axis is guaranteed for up to 5 decimal places, the calculations during restoration are also performed with this accuracy.
Therefore, depending on the position of the output axis, the fourth decimal place of the current position of the output axis may move slightly, but this does not result in position displacement.
Depending on parameter settings and programs, the axis may move significantly at restoration. For safety reasons, check the movement amount by restoration in synchronous control analysis mode before starting synchronous control.

15

Current position per cycle restoration for coordinate data

- When the final point of cam data is larger than the length per cycle, the entire range of cam data is searched regardless of the set value of the length per cycle, which may result in restoration to a position different from the path of synchronous control.
- When the final point of cam data is smaller than the length per cycle, the current position per cycle may become larger than the length per cycle depending on the relation between the cam set position and the cam reference position.
- When using the coordinate data, it is recommended to match the final point of cam data with the length per cycle.

Restoration time of the current position per cycle

The current position per cycle restoration completion time varies depending on the resolution, coordinate number, and the restoration settings.
An approximate guide of the restoration time is shown below.

■ Cam data (section interpolation/spline interpolation)/rotary cutter

Resolution/Coordinate number	256			65535		
Operation cycle [ms]	0.25	0.5	1.0	0.25	0.5	1.0
Restoration time [ms]	0.4	0.3	0.4	5.2	4.7	4.5

■ Cam data (section interpolation

Resolution/Coordinate number	256			65535		
Operation cycle[ms]	0.25	0.5	1.0	0.25	0.5	1.0
Restoration time[ms]	0.4	0.3	0.4	5.8	4.7	4.9

Method to restart synchronous control

The relationship of the synchronous position for synchronous control is always saved in the Motion module. Synchronous control can be restarted without returning all axes to their starting points by restoring the synchronized relationship through the synchronous control initial position parameters. (☞ Page 602 Synchronous control initial position parameter (AdvOutputName.Pr.Restore.))
For an example, refer to the following.
☞ Page 627 Filling machine

15.6 Cam Position Calculation Function

The cam position calculation function calculates the current position per cycle and the cam set position with the following Motion control FBs.

- MCv_AdvPositionPerCycleCalc (Advanced Synchronous Control Current Position Per Cycle Calculation)
- MCv_AdvCamSetPositionCalc (Advanced Synchronous Control Cam Set Position Calculation)

This function is used to calculate the synchronous position of the output axis for the synchronous position alignment before starting advanced synchronous control.

With the synchronous control analysis mode, the output axis whose position is to be calculated must be in the standby status (the Axis status (AxisName.Md.AxisStatus) is "4: Standby (Standstill)"). However, with the cam position calculation function, the cam position calculation can be executed regardless of the output axis status, thus it is used to calculate the synchronous position of the next control in advance while the output axis is operating.

Control details

Setting the ring counter

Set the Ring counter upper limit value (AxisName.PrConst.RingCount_UpperValue) of the output axis to the Ring counter upper limit (RingCountUpperValue) of MCv_AdvCamSetPositionCalc (Advanced Synchronous Control Cam Set Position Calculation) or MCv_AdvPositionPerCycleCalc (Advanced Synchronous Control Current Position Per Cycle Calculation).

When a value other than "0.0" is set to Ring counter upper limit (RingCountUpperValue), the ring counter lower limit iscalculated as "0.0".

Setting the cam starting point

- Set the cam position calculation Cam starting point (StartingPoint) of MCv_AdvCamSetPositionCalc (Advanced Synchronous Control Cam Set Position Calculation) or MCv_AdvPositionPerCycleCalc (Advanced Synchronous Control Current Position Per Cycle Calculation) when taking into account the cam starting point for calculations.
- The cam position calculation Cam starting point (StartingPoint) sets the position to start cam data in resolution. This is valid only in the following cases.
  · When the Interpolation method specification (Interpolate) of the operation profile data is "1: Section interpolation" or "2: Spline interpolation"
  · When the data type is set to "Rotary cutter"

Current position per cycle calculation (MCv_AdvPositionPerCycleCalc)

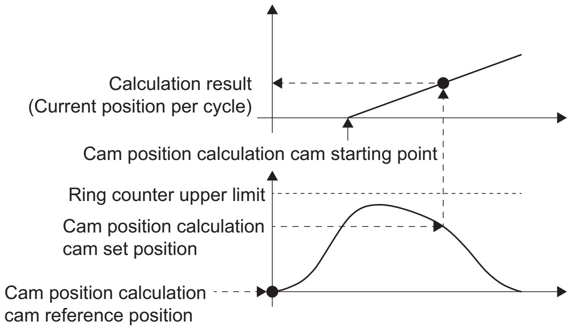
When the Execute command (Execute) is changed from FALSE to TRUE, the current position per cycle is calculated based on the cam position calculation Cam reference position (ReferenceSetPosition) and the cam position calculation Cam set position (SetPosition) from the cam data. The calculation result is outputted to the Calculation result (CauculatioinResult).

■ When the ring counter upper limit is set

The cam position calculation cam reference position and the cam position calculation cam set position are rounded to the ring counter upper limit and then calculated when the cam position calculation Ring counter upper limit (RingCountUpperValue) is set.

■ When the cam starting point is set

The cam starting point is taken into account for the current position per cycle calculation when the cam position calculation Cam starting point (StartingPoint) is set.



■ Searching the current position per cycle

When calculating the current position per cycle from the cam data, the position corresponding to the cam position calculation cam set position is searched from the cam data based on the position set in the cam position calculation Current position per cycle (PositionPerCycle).

The cam position calculation cam set position is searched with the following procedure.

- Cam data (when the Interpolation method specification (Interpolate) of the operation profile data is "1: Section interpolation" or "2: Spline interpolation", or when the data type is set to "Rotary cutter")

■When "nth point of cam data ≤ Cam position calculation Current position per cycle (PositionPerCycle) < Cam data n + First point"

- The position corresponding to the cam position calculation Cam set position (SetPosition) is searched from the nth point of cam data.

■When the cam position calculation current position per cycle (PositionPerCycle) is in the middle of cam data and there is no corresponding position until the last point of cam data

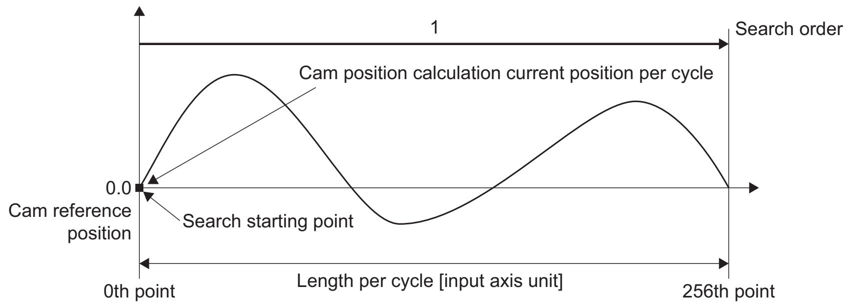
- The search returns to the 0th point of cam data and searches to the position where the search started.

■When there is no corresponding position after searching the entire range of cam data

- For a two-way cam, "Unable to Calculate Cam Position Under Advanced Synchronous Control (error code: 34AFH)" occurs.
- For a feed cam, the cam position calculation Cam set position (SetPosition) is calculated from the stroke difference, and the entire range is searched from the 0th point. When there is no corresponding position after researching, "Unable to Calculate Cam Position Under Advanced Synchronous Control (error code: 34AFH)" occurs.

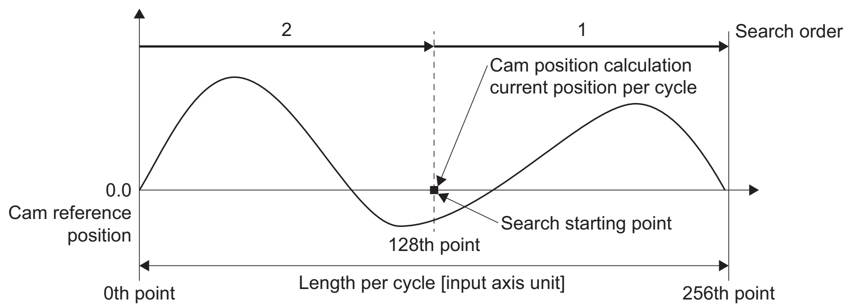
<Example 1>

When the cam position calculation Current position per cycle (PositionPerCycle) corresponds to the 0th point of cam data



<Example 2>

When the cam position calculation Current position per cycle (PositionPerCycle) corresponds to the 128th point of cam data



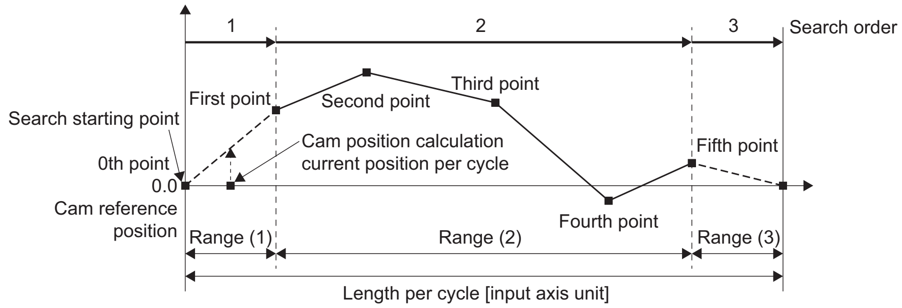
- Cam data (when the Interpolation method specification (Interpolate) of the operation profile data is "0: Linear interpolation")

■Range before the first point of cam data

- When the first point of cam data is larger than 0 and "Cam position calculation Current position per cycle (PositionPerCycle) < First point of cam data" is satisfied, the position corresponding to the cam position calculation Cam set position (SetPosition) is searched from the range before the first point of cam data.
- When there is no corresponding position in range (1), range (2) is searched. When there is also no corresponding position in range (2), range (3) is searched.
- When there is no corresponding position after searching range (1) to (3), "Unable to Calculate Cam Position Under Advanced Synchronous Control (error code: 34AFH)" occurs for a two-way cam.
- For a feed cam, the cam position calculation Cam set position (SetPosition) is calculated from the stroke difference, and the entire range is searched from the 0th point. When there is no corresponding position even after researching, "Unable to Calculate Cam Position Under Advanced Synchronous Control (error code: 34AFH)" occurs.

<Example>

When the cam position calculation current position per cycle is before the first point of cam data



*: Search from range (1).

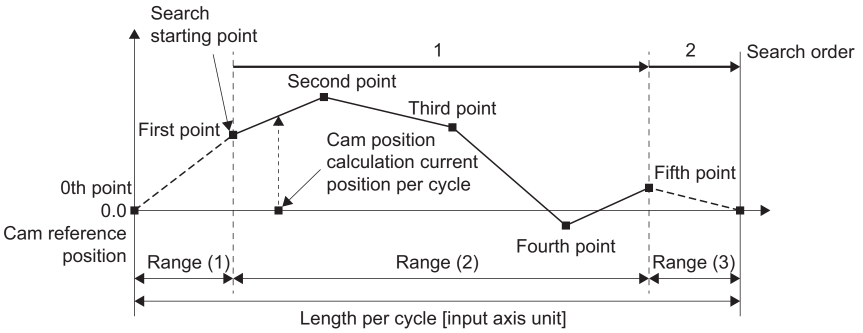
*: When there is no corresponding position in range (1), search in order from the first point of cam data in range (2).

■Within the cam data range

- When "Cam position calculation Current position per cycle (PositionPerCycle) < Last point of cam data" is satisfied, the position corresponding to the cam position calculation Cam set position (SetPosition) is searched from within the cam data range.
- When "nth point of cam data ≤ Cam position calculation current position per cycle < Cam data n + First point" is satisfied, the position corresponding to the cam position calculation cam set position is searched from the nth point of cam data.
- When the cam position calculation current position per cycle is in the middle of cam data and there is no corresponding position until the last point of cam data, the search returns to the first point of cam data and searches to the position where the search started.
- When there is no corresponding position in range (2), range (3) is searched.
- When there is no corresponding position after searching range (2) to (3), "Unable to Calculate Cam Position Under Advanced Synchronous Control (error code: 34AFH)" occurs for a twoway cam.
- For a feed cam, the cam position calculation cam set position is calculated from the stroke difference, and the entire range is searched from the 0th point. When there is no corresponding position even after researching, "Unable to Calculate Cam Position Under Advanced Synchronous Control (error code: 34AFH)" occurs.

<Example 1>

When the cam position calculation current position per cycle is before the first point of cam data

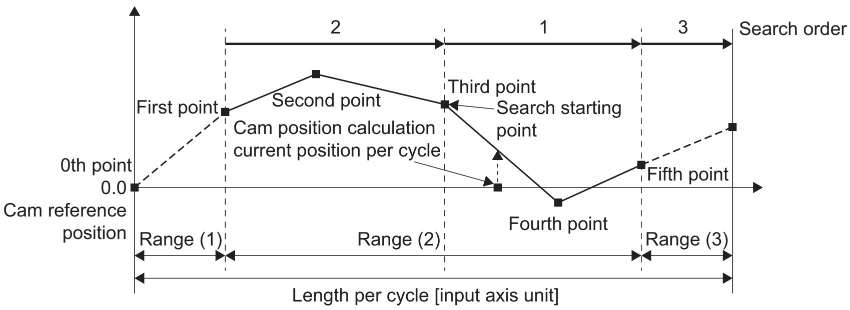


❶ Search from cam data first point to second point, second point to third point, and up to fourth point to fifth point (last point) in order.
❷ If there is no corresponding position until the last point of cam data, search from range (3).

15

<Example 2>

When the cam position calculation current position per cycle corresponds to the third point of cam data



❶ Search in the order from cam data third point to fourth point, fourth point to fifth point (last point).
❷ If there is no corresponding position until the last point of cam data, search from the first point of cam data.
❸ If there is no corresponding position in the cam data first point to second point and second point to third point, search from range (3).

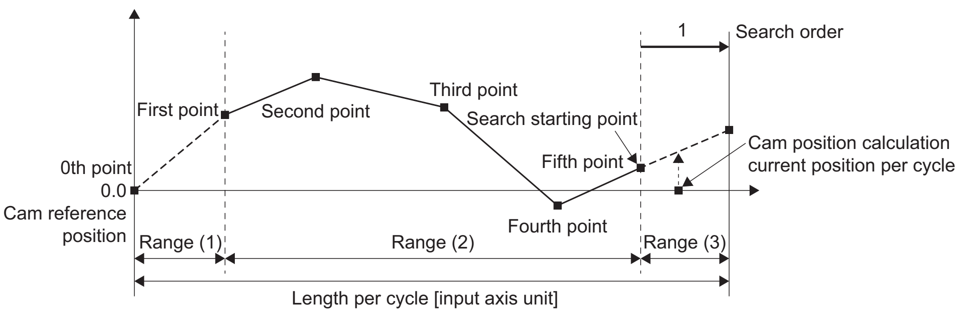
■Range from the cam data last point to the length per cycle

When "Last point of cam data ≤ Cam position calculation Current position per cycle (PositionPerCycle) < Length per cycle" is satisfied, the position corresponding to the cam position calculation Cam set position (MCv_AdvPositionPerCycleCalc.SetPosition) is searched from the range after the last point of cam data.

If there is no corresponding position after searching range (3), "Unable to Calculate Cam Position Under Advanced Synchronous Control (error code: 34AFH)" occurs for a twoway cam. For a feed cam, the cam position calculation cam set position is calculated from the stroke difference, and the entire range is searched from the 0th point. When there is no corresponding position even after researching, "Unable to Calculate Cam Position Under Advanced Synchronous Control (error code: 34AFH)" occurs.

<Example>

When the cam position calculation current position per cycle corresponds to the last point of cam data



❶ Search from range (3).

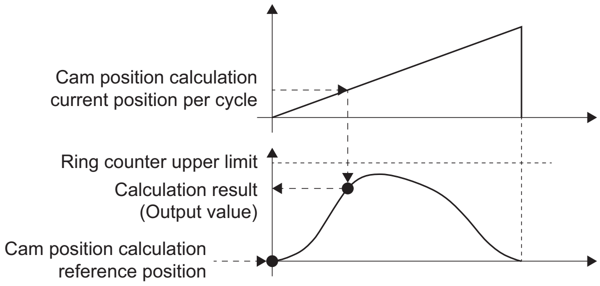
Cam set position calculation (MCv_AdvCamSetPositionCalc)

When the Execute command (Execute) is changed from "FALSE" to "TRUE", the cam set position is calculated based on the cam position calculation Current position per cycle (PositionPerCycle) and the cam position calculation Cam reference position (ReferenceSetPosition). The calculation result is outputted to the Calculation result (CauculatioinResult).

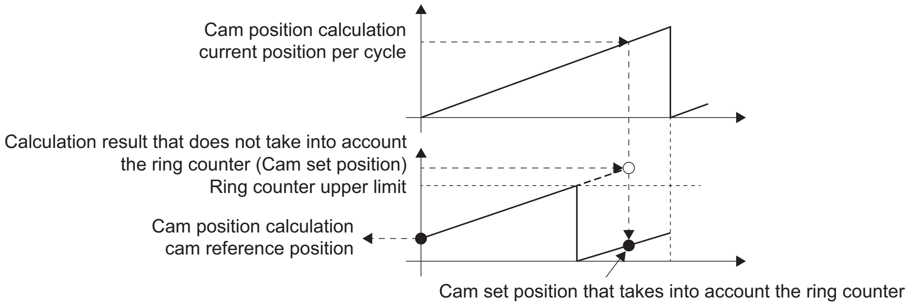
■ When the ring counter upper limit is set

The calculation result is rounded to a value that takes into account the ring counter when the cam position calculation Ring counter upper limit (RingCountUpperValue) is set.

- When the calculation result does not exceed the ring counter upper limit

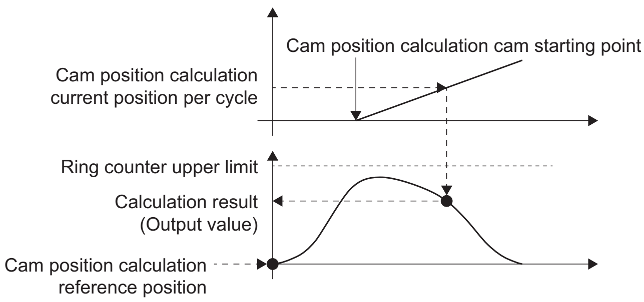


- When the calculation result exceeds the ring counter upper limit



■ When the cam starting point is set

The cam set position is calculated with the current position per cycle that takes into account the cam starting point when the cam position calculation Cam starting point (StartingPoint) is set.



Precautions

- Set the cam data to expand automatically in the expand setting in advance, or send it to the open area with MC_CamTableSelect (Cam Table Selection) for use.
- Apply the calculation results of the Current position per cycle calculation (MCv_AdvPositionPerCycleCalc) and the Cam set position calculation (MCv_AdvCamSetPositionCalc) only to MCv_AdvancedSync (Advanced Synchronous Control). If they are applied to Motion FBs (such as MC_CamIn) other than MCv_AdvancedSync (Advanced Synchronous Control), the calculation result and the actual operation will not match.
- The current position per cycle calculation of the coordinate data may result as follows depending on the length per cycle of the last point of cam data.

Last point of cam data	Description
Larger than the length per cycle	The entire range of cam data is searched regardless of the set value of the length per cycle, which may result in a different calculation result from the path of synchronous control.
Smaller than the length per cycle	The current position per cycle may become larger than the length per cycle depending on the relation between the cam set position and the cam reference position. When using the coordinate data, it is recommended to match the final point of cam data with the length per cycle.

15

15.7 Applied Functions

Phase compensation function

In synchronous control, delays in progresses, etc. cause the phase to deviate at the output axis motor shaft end with respect to the input axis (real drive, real encoder axis, virtual encoder axis). The phase compensation function compensates in this case so that the phase does not deviate.
Phase compensation can be set for the input and the output axis. It is possible to compensate using the delay time peculiar to the system based on the input axis type on the input axis side. It is also possible to use a compensation delay time equivalent to the position deviation for each driver device on the output axis side.

Phase compensation on delay time of the input axis

Set delay time peculiar to the system in the phase compensation advance time of the input axis (AdvInputName.Pr.PhaseCompensationAdvanceTime). For reference, the delay time peculiar to a system with MR-J5(W)-G connected to the driver device is shown below. As the delay time will change depending on the actual equipment configuration, make adjustments from reference values below.

■ Delay time peculiar to the system for a real drive axis (MR-J5(W)-G)*1

The delay time depending on the setting of data source selection (AdvInputName.PrConst.SourceValue) is as follows.

- When set to "1: Set value (mcSetValue)" or "101: Latest set value (mcLatestSetValue)"
  · Delay time: 0 [μs]
- When set to "2: Actual value (mcActualValue)" or "102: Latest actual value (mcLatestActualValue)"
  The delay time is calculated using the following formula.

Motion Synchronization Station Send/Receive Data Refresh Method (System.PrConst.Link_MotionStationRefreshType)	
0: Response preferred method (EmphasisResponse)	1: Operation cycle preferred method (EmphasisOperationCycle)
Operation cycle × 5 + 750 [μs]	Operation cycle × 7 + 750 [μs]

*1 The delay time of "1: Set value (mcSetValue)" and "2: Actual value (mcActualValue)" is the delay time of "101: Latest set value (mcLatestSetValue)" and "102: Latest actual value (mcLatestActualValue)" + 1 operation cycle.
When compensating a delay, set the value obtained by adding 1 operation cycle to the delay time peculiar to the system as the compensation time.

■ Delay time peculiar to the system for a real encoder axis*1

The delay time is calculated using the following formula and does not depend on the setting of data source selection (AdvInputName.PrConst.SourceValue).

Motion Synchronization Station Send/Receive Data Refresh Method (System.PrConst.Link_MotionStationRefreshType)	
0: Response preferred method (EmphasisResponse)	1: Operation cycle preferred method (EmphasisOperationCycle)
Operation cycle × 5 + 750 [μs]	Operation cycle × 7 + 750 [μs]

*1 The delay time of "1: Set value (mcSetValue)" and "2: Actual value (mcActualValue)" is the delay time of "101: Latest set value (mcLatestSetValue)" and "102: Latest actual value (mcLatestActualValue)" + 1 operation cycle.
When compensating a delay, set the value obtained by adding 1 operation cycle to the delay time peculiar to the system as the compensation time.

■ Delay time peculiar to the system for a virtual encoder axis

The delay time is calculated using the following formula.

- Delay time peculiar to the input target + Virtual encoder input value refresh time [μs]

Ex.

For the following

Example	Description
1	■When setting the MR-J5(W)-G object data_PosActualValue (AxisName.Md.Io_PosActualValue) to the virtual encoder axis slave object data (AxisName.PrConst.SlaveObject.PosActualValue) settings • Delay time peculiar to the real drive axis*1 + Refresh cycle of AxisName.Md.Io_PosActualValue (1 operation cycle)
2	■When updating the virtual encoder axis encoder input value (AxisName.Cd.Encoder_InputValue) with a counter value of another unit in an ST program • Delay time peculiar to the counter + ST program scan time
3	■When using the encoder value of an encoder connected with the driver For the setting example, refer to the following. ☞ Page 52 When using the encoder value of an encoder connected with the driver A rough standard for the delay time peculiar to the system is the same value as the delay time peculiar to the system for a real encoder axis. For the delay time peculiar to the system for a real encoder axis, refer to the following. ☞ Page 612 Delay time peculiar to the system for a real encoder axis*1

*1 "102: Latest actual value (mcLatestActualValue)" of data source selection (AdvInputName.PrConst.SourceValue)

■ Delay time peculiar to the system for a virtual drive axis/virtual linked axis

The delay time is "0 [μs]" and does not depend on the setting of data source selection (AdvInputName.PrConst.SourceValue).

15

Phase compensation of delay time of the output axis

Set delay time equivalent to the position deviation on the drive unit in the output axis phase compensation advance time (AdvOutputName.Pr.Cam.PhaseCompensationAdvancedTime). The delay time equivalent to position deviation of the drive unit is calculated using the following formula.

<When using MR-J5(W)-G>

Delay time [μs] = 1000000 ÷ Servo parameter "Model loop gain (PB07)"

When the servo parameter "Feed forward gain (PB04)" is set, the delay time is set to a smaller value than the value listed above.

The model loop gain will change when "Gain adjustment mode selection (PA08.0)" is set to "1: Auto tuning mode 1" or "2: Auto tuning mode 2". For axes executing phase compensation, set "3: Manual mode" or "0: 2 gain adjustment mode 1 (interpolation mode)" to prevent model loop gain from changing.

Setting example

When the output axis is synchronized with a synchronous encoder axis, the phase compensation advance time is set as follows.

- When model loop gain of the output axis is 80

Setting item		Setting value
Input axis	Phase compensation advance time (AdvInputName.Pr.PhaseCompensationAdvanceTime)	Set the delay time peculiar to the system for a real encoder axis. Refer to the following for delay time peculiar to the system for a real encoder axis. ☞ Page 612 Delay time peculiar to the system for a real encoder axis*1
Output axis	Phase compensation advance time (AdvOutputName.Pr.Cam.PhaseCompensationAdvancedTime)	1000000 ÷ 80 = 12500 [μs]

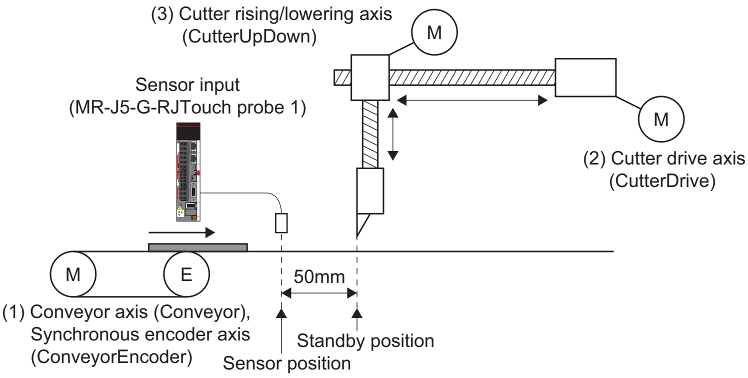
When the slave axis set velocity overshoots or undershoots during acceleration/deceleration, set a longer time for the phase compensation time constant.

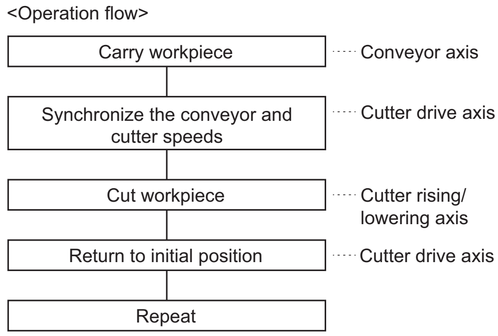
15.8 Example Programs

Example programs for advanced synchronous control are shown below.

Flying cutter

This section describes the example program and how to use a clutch operation that uses an advanced synchronous control FB sensor input based on a flying cutter system.





Setting item	(1) Conveyor axis	(1) Synchronous encoder axis	(2) Cutter drive axis	(3) Cutter rising/ lowering axis	Virtual drive axis for cutter drive axis highspeed return operation
Axis name	Conveyor	ConveyorEncoder	CutterDrive	CutterUpDown	CutterDriveBack
Axis type	Real drive axis	Real encoder axis	Real drive axis	Real drive axis	Virtual drive axis
Real encoder axis type setting	—	Via drive unit	—	—	—
Station address	192.168.3.1	192.168.3.1	192.168.3.2	192.168.3.3	—
Position command unit	μm	μm	μm	μm	μm

Operation overview

1. Advanced synchronous control FB starts with the conveyor axis as the "input axis" and the cutter drive axis as the "output axis".

2. The sensor input starts the clutch ON operation of the main shaft clutch.

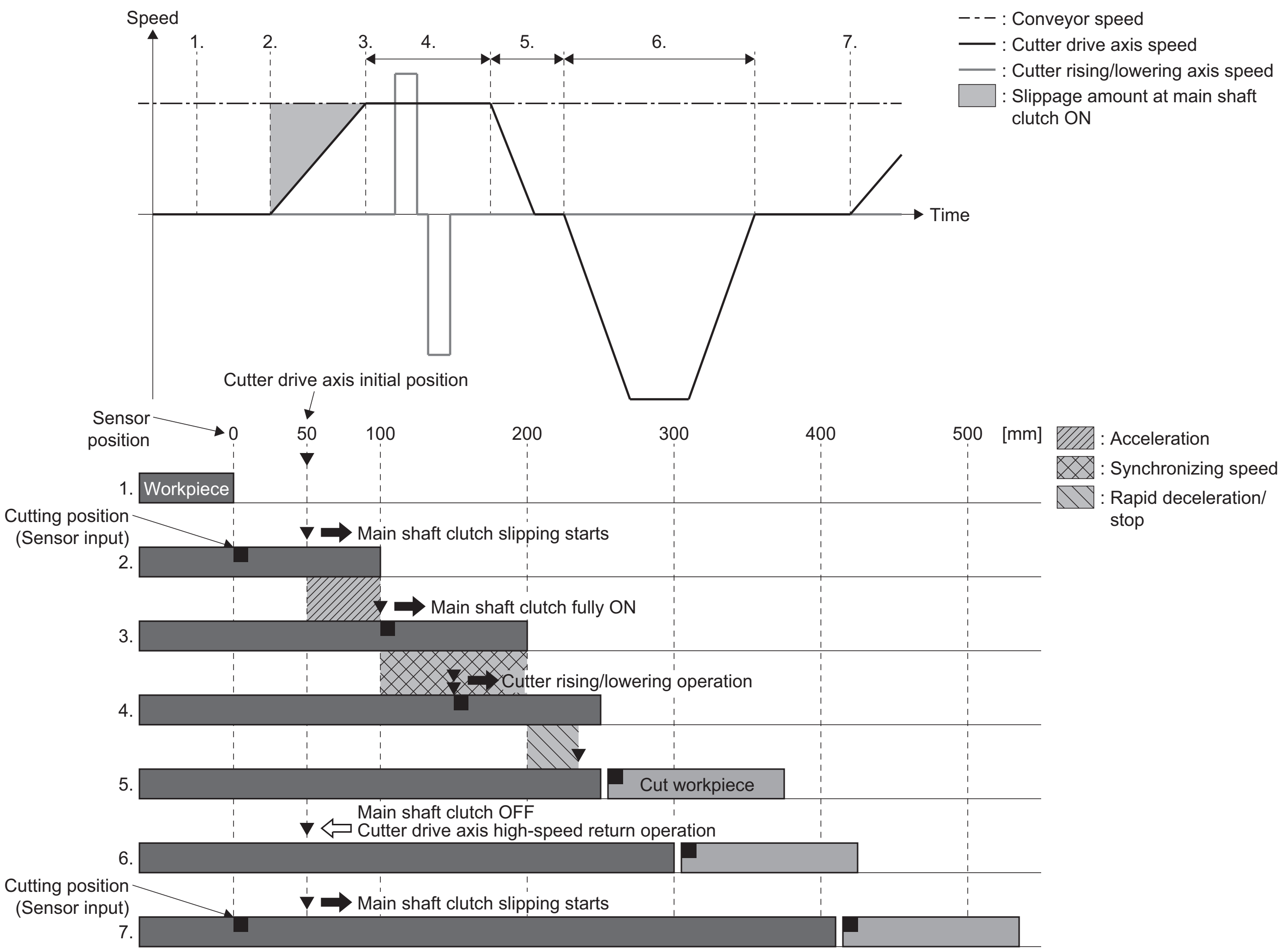
3. After completing main shaft clutch ON smoothing, the speed of the conveyor axis and the cutter drive axis are the same.

4. The cutter rising/lowering axis is started by positioning control when the speed of the conveyor axis and the cutter drive axis become the same, and it cuts the workpiece.

5. After completing the cutting of the workpiece (completion of positioning control of the cutter rising/lowering axis), the main shaft clutch turns OFF and the cutter drive axis is stopped.

6. After stopping the cutter drive axis, positioning of the virtual drive axis that is set as the auxiliary shaft returns the cutter drive axis to its initial position.

7. Step 2. to 6. are repeated.



15

Parameter settings

■ Setting of the real encoder axis (conveyor axis) to be used as the input axis of advanced synchronous control

In the "CC-Link IE TSN Configuration" screen under Network Configuration Settings in GX Works3, add the required objects for the real encoder axis in the "PDO Mapping Setting" of the MR-J5(W)-G(-RJ) that uses a synchronous encoder in [PDO Mapping Setting] ⇨ "<Detail Setting>" ⇨ "PDO Mapping Setting" screen.

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ [Target module] ⇨ [Module Parameter(Network)] ⇨ "Parameter editor" screen ⇨ [Basic Settings] ⇨ [Network Configuration Settings] ⇨ [Detailed Setting]

- RPDO mapping
  None
- TPDO mapping

Index	Sub-index	Entry name	Data type
2d36	00	Scale cycle counter	UNSIGNED32
2d37	00	Scale ABS counter	INTEGER32
2d3c	00	Scale measurement encoder recept	UNSIGNED32

Also, when using a synchronous encoder, the scale measurement function must be enabled in MR-J5(W)-G(-RJ). Set the following servo parameter.

No.	Symbol	Name	Setting value
PA22.3	**PCS	Scale measurement function selection	Set 1 or 2. • 1: Use with absolute position detection system • 2: Use with incremental system

For the settings of the real encoder axis, refer to the following.

☞ Page 42 Real encoder axis

■ Setting of the sensor input (MR-J5(W)-G(-RJ) touch probe function)

In the "CC-Link IE TSN Configuration" screen under Network Configuration Settings in GX Works3, add the required objects for the touch probe function in the "PDO Mapping Setting" of the MR-J5(W)-G(-RJ) that uses a touch probe in [PDO Mapping Setting] ⇨ "<Detail Setting>" ⇨ "PDO Mapping Setting" screen.

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ [Target module] ⇨ [Module Parameter (Network)] ⇨ "Parameter editor" screen ⇨ [Basic Settings] ⇨ [Network Configuration Settings] ⇨ [Detailed Setting]

- RPDO mapping

Index	Sub-index	Entry name	Data type
60d8	00	Touch probe function	UNSIGNED16

- TPDO mapping

Index	Sub-index	Entry name	Data type
60b9	00	Touch probe status	UNSIGNED16
60d1	00	Touch probe time stamp 1 positiv	UNSIGNED32
60d2	00	Touch probe time stamp 1 negativ	UNSIGNED32

In the "Network I/O" screen of Motion Control Setting Function of GX Works3, make the Touch probe function set in the RPDO mapping and TPDO mapping into a slave label.

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ [Target module] ⇨ [Module Extended Parameter] ⇨ [Motion Control Setting Function] ⇨ Navigation window ⇨ [Network I/O]

For touch probe function settings, refer to the following.

☞ Page 858 External signal high-accuracy input

For how to make slave labels, refer to the following.

☞ Page 831 How to use the slave label

■ Advanced synchronous control parameter settings

Set the parameters of the axes to be used as the input/output of the advanced synchronous control FB.

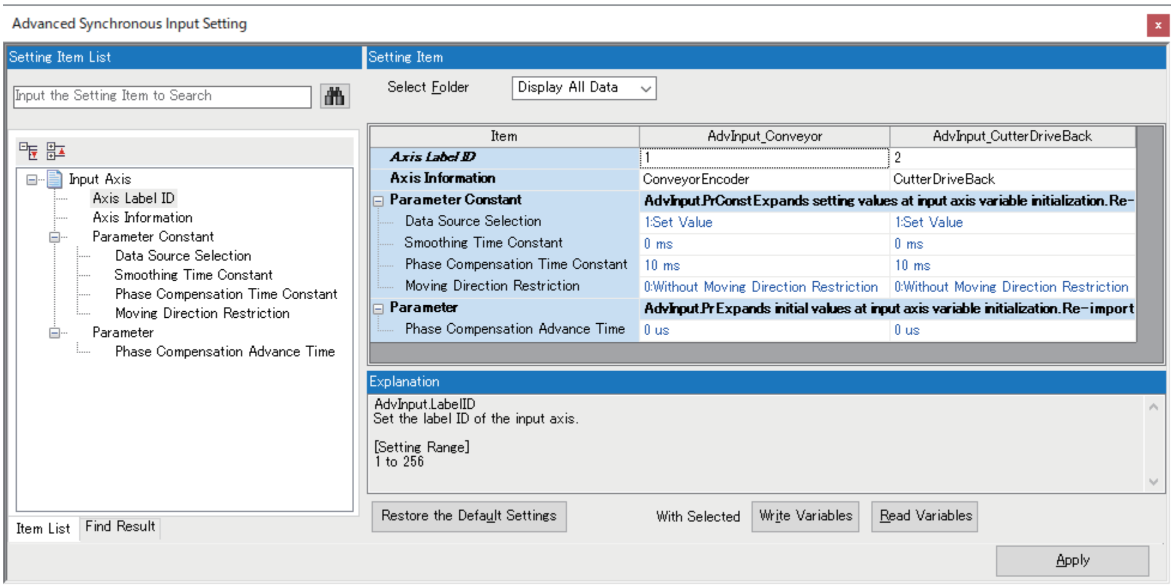
Parameter	Axis used
Input axis parameter	• Conveyor axis • Virtual drive axis for cutter drive axis highspeed return operation
Output axis parameter	• Cutter drive axis

- Input axis parameter settings

  In the "Advanced Synchronous Input Setting" screen of Motion Control Setting Function, configure the input axis settings of the main shaft and auxiliary shaft used by the advanced synchronous control FB.

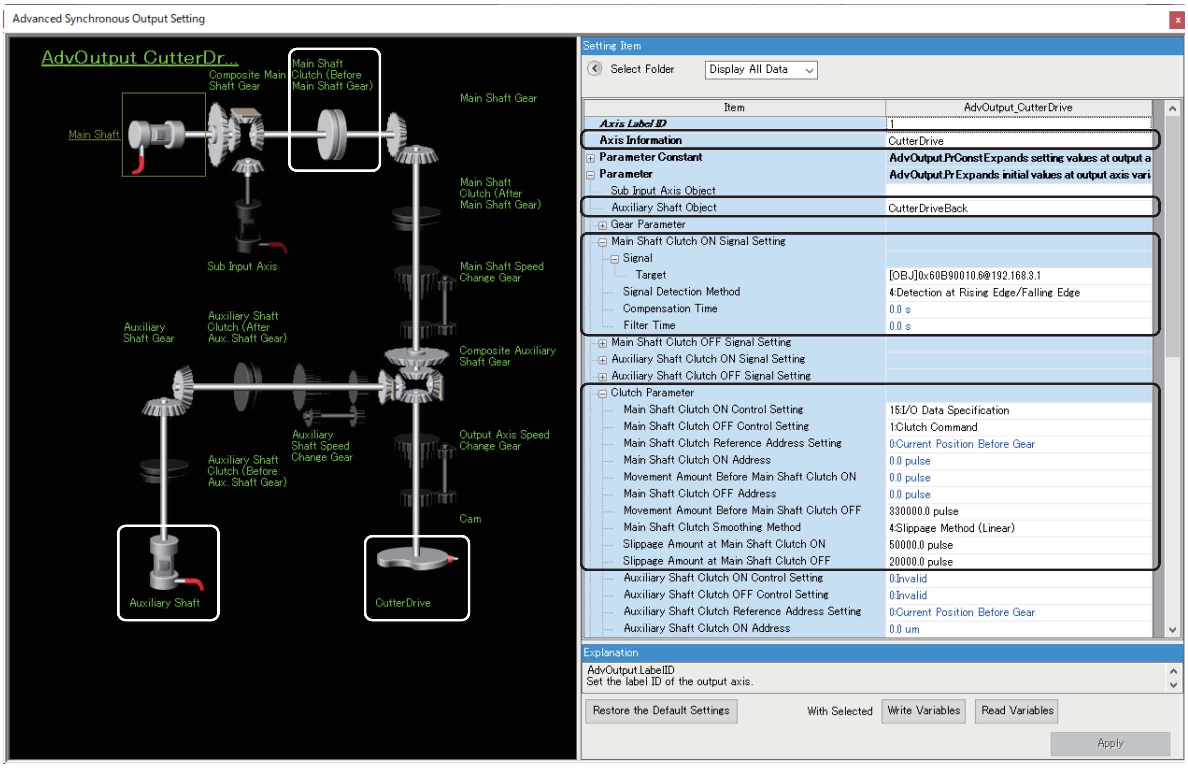
Data name	Setting details
AdvInput_Conveyor	Real encoder axis (ConveyorEncoder) of the conveyor axis
AdvInput_CutterDriveBack	Virtual drive axis (CutterDriveBack) used for cutter drive axis highspeed return operation

Motion Control Setting Function ⇨ Navigation window ⇨ [Add-on Data] ⇨ [Advanced Synchronous Control] ⇨ [Input Setting]



- Output axis parameter settings
  In the "Advanced Synchronous Output Setting" screen of Motion Control Setting Function, configure the output axis settings of the output axes used by the advanced synchronous control FB.

Motion Control Setting Function ⇨ Navigation window ⇨ [Addon Data] ⇨ [Advanced Synchronous Control] ⇨ [Output Setting]



■Settings of axes used by output axis and auxiliary shaft

Setting item	Setting value
Axis information (AdvOutputName.Axis)	CutterDrive (the axis used as the cutter drive axis)
Auxiliary shaft target (AdvOutputName.Pr.AuxAxis)	CutterDriveBack (the axis used for the cutter axis highspeed return operation)

■Main shaft clutch setting

Setting item	Setting value
Main shaft clutch ON control setting (AdvOutputName.Pr.Clutch.MasterOnControl)	15: I/O data specification (ClutchSignal)
Main shaft clutch OFF control setting (AdvOutputName.Pr.Clutch.MasterOffControl)	1: Clutch command (ClutchCommand)
Main shaft clutch ON signal setting (Signal target) (AdvOutputName.Pr.MasterOnClutchSignal.Source.Target)	[OBJ]0x60B90010.6@192.168.3.1 (Touch probe 1 rising edge latch complete toggle status)
Main shaft clutch ON signal setting (Signal detection method) (AdvOutputName.Pr.MasterOnClutchSignal.Detection)	4: Detection at rising edge/falling edge (BothEdges)
Main shaft clutch ON signal setting (Compensation time) (AdvOutputName.Pr.MasterOnClutchSignal.CompensationTime)	0.0 [s]
Main shaft clutch ON signal setting (Filter time) (AdvOutputName.Pr.MasterOnClutchSignal.FilterTime)	0.0 [s]
Main shaft clutch reference address setting (AdvOutputName.Pr.Clutch.MasterReference)	0: Current position before gear (GearFrontPosition)
Movement amount before main shaft clutch ON (AdvOutputName.Pr.Clutch.MasterMovementAmountBeforeOn)	0.0 [mm]
Movement amount before main shaft clutch OFF (AdvOutputName.Pr.Clutch.MasterMovementAmountBeforeOff)	330.0 [mm]
Main shaft clutch smoothing method (AdvOutputName.Pr.ClutchMasterSmoothingMethod)	4: Slippage method (linear) (SlippageLinear))
Slippage amount at main shaft clutch ON (AdvOutputName.Pr.Clutch.MasterOnSlippageAmount)	50.0 [mm] (Distance between the sensor position and the cutter drive axis initial position)
Slippage amount at main shaft clutch OFF (AdvOutputName.Pr.Clutch.MasterOffSlippageAmount)	20.0 [mm]

Program creation

■ Enabling the touch probe function of MR-J5(W)-G(-RJ)

Use the label created in "Setting of the sensor input (MR-J5(W)-G(-RJ) touch probe function)" to enable the touch probe function.

ST program

```
// TouchProbe Enable
MR_J5_G_RJ_001_TouchProbeFunction:=UINT#16#0013;
```

- Labels used

  None

■ Starting advanced synchronous control

Confirm that the input axis is enabled (AdvInput_Conveyor.Md.ActiveStatus is "1: Input axis enabled") and the output axis is on standby (CutterDrive.Md.AxisStatus is "4: Standby (Standstill)") before starting the advanced synchronous control FB.

- Example program

ST program

```
IF gb_SystemStart THEN
   // AdvSync Start
   IF  AdvInput_Conveyor.Md.ActiveStatus=1 AND
       CutterDrive.Md.AxisStatus=MC_AXIS_STATUS__Standstill THEN
       gb_AdvSync_Drive_Enable:=TRUE;
       END_IF;
END_IF;

MCv_AdvancedSync_Drive(
   Master:= AdvInput_Conveyor.Axis,
   Slave:=  AdvOutput_CutterDrive.Axis,
   Enable:= gb_AdvSync_Drive_Enable,
   InSync=> gb_AdvSync_Drive_InSync);
```

- Labels used

Label name	Data type	Comment
gb_SystemStart	Bit	System start command
MCv_AdvancedSync_Drive	MCv_AdvancedSync	Advanced synchronous control FB (cutter drive axis)
gb_AdvSync_Drive_Enable	Bit	Advanced synchronous control FB start command
gb_AdvSync_Drive_InSync	Bit	Advanced synchronous control FB synchronizing

■ Starting the conveyor axis (JOG operation)

Confirm that the output axis is synchronized (AdvOutput_CutterDrive.Md.SyncStatus is"1: In synchronization") and start JOG operation on the conveyor axis.

- Example program

ST program

```
IF gb_SystemStart THEN
   // ConveyorAxis Move
   IF AdvOutput_CutterDrive.Md.SyncStatus=1 THEN
       gl_SetSpeedComveyor:=40000000.0;
       gb_MoveConveyorReq:=TRUE;
       END_IF;
   END_IF;

MCv_Jog_Conveyor(
   Axis:= Conveyor.AxisRef,
   JogForward:= gb_MoveConveyorReq,
   Velocity:= gl_SetSpeedComveyor,
   Acceleration:= 0.5,
   Deceleration:= 0.5,
   Options:= H0001);
```

• Labels used

Label name	Data type	Comment
gb_SystemStart	Bit	System start command
MCv_Jog_Conveyor	MCv_Jog	JOG operation FB (conveyor axis)
gb_MoveConveyorReq	Bit	Positive direction JOG operation command
gl_SetSpeedComveyor	Double-precision real number	Command speed of JOG operation

■ Positioning control of the cutter rising/lowering axis

Confirm that smoothing is complete for output axis main shaft clutch ON (Main shaft clutch ON/OFF status (AdvOutput_CutterDrive.Md.Clutch.MasterOnOffStatus) is TRUE and main shaft clutch smoothing status (AdvOutput_CutterDrive.Md.Clutch.MasterSmoothingStatus) is FALSE), before starting positioning control of the rising/lowering cutter axis. (An example of a return operation conducted by multiple start with Buffered.)

• Example program

ST program

```
IF AdvOutput_CutterDrive.Md.Clutch.MasterOnOffStatus AND
   NOT(AdvOutput_CutterDrive.Md.Clutch.MasterSmoothingStatus) AND
   NOT(lb_CutterMoveComp) THEN

   MC_MoveAbsolute_1.Execute:=TRUE;
   IF MC_MoveAbsolute_1.Busy THEN
      MC_MoveAbsolute_2.Execute:=TRUE;
   END_IF;
END_IF;

IF MC_MoveAbsolute_1.Done AND
   MC_MoveAbsolute_2.Done THEN
   MC_MoveAbsolute_1.Execute:=FALSE;
   MC_MoveAbsolute_2.Execute:=FALSE;
   lb_CutterMoveComp:=TRUE;
END_IF;

MC_MoveAbsolute_1(
   Axis:= CutterUpDown.AxisRef,
   Position:= 30000.0 ,//30.0[mm]
   Velocity:= 3000000.0 ,//3000000.0[mm/min]
   Acceleration:= 0.1,
   Deceleration:= 0.1,
   Direction:= MC_DIRECTION__mcShortestWay,
   Options:= H0001);

MC_MoveAbsolute_2(
   Axis:= CutterUpDown.AxisRef,
   Position:= 0.0,
   Velocity:= 3000000.0 ,
   Acceleration:= 0.1,
   Deceleration:= 0.1,
   Direction:= MC_DIRECTION__mcShortestWay ,
   BufferMode:= MC_BUFFER_MODE__mcBuffered,
   Options:= H0001);
```

• Labels used

Label name	Data type	Comment
MC_MoveAbsolute_1	MC_MoveAbsolute	Absolute positioning (cutting operation)
MC_MoveAbsolute_2	MC_MoveAbsolute	Absolute positioning (retracting operation)
lb_CutterMoveComp	Bit	Cutter operation complete flag

■ Cutter drive axis high-speed return operation (positioning control of the virtual drive axis set to auxiliary shaft)

Confirm that smoothing is complete for output axis main shaft clutch ON (Main shaft clutch ON/OFF status (AdvOutput_CutterDrive.Md.Clutch.MasterOnOffStatus) is TRUE and main shaft clutch smoothing status (AdvOutput_CutterDrive.Md.Clutch.MasterSmoothingStatus) is FALSE), and positioning control of the cutter rising/lowering axis is complete before starting positioning control of the virtual drive axis (CutterDriveBack) set to auxiliary shaft.

- Example program

ST program

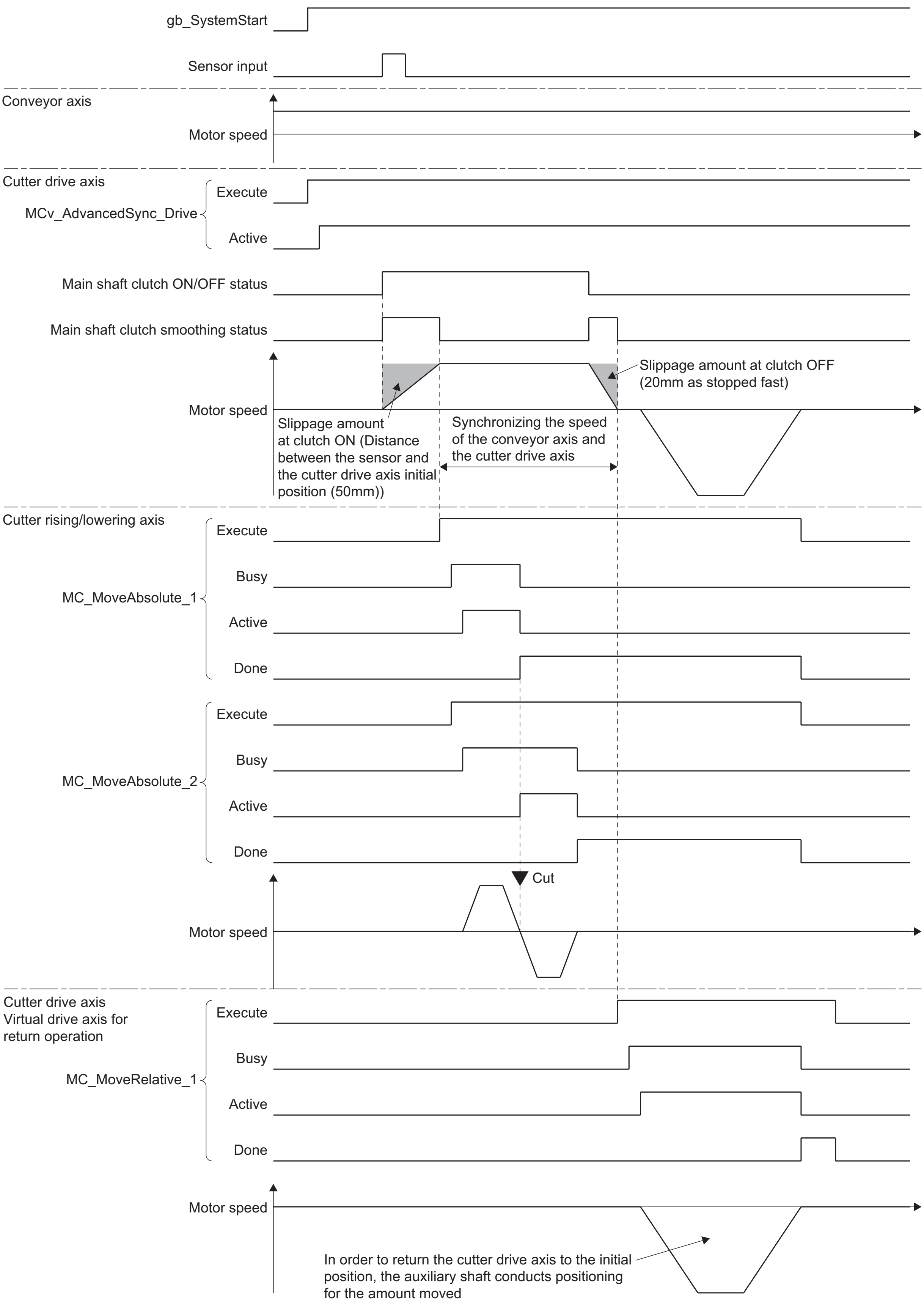
```
IF NOT(AdvOutput_CutterDrive.Md.Clutch.MasterOnOffStatus) AND
   NOT(AdvOutput_CutterDrive.Md.Clutch.MasterSmoothingStatus) AND
   lb_CutterMoveComp THEN
   MC_MoveRelative_1.Execute:=TRUE;
END_IF;

IF MC_MoveRelative_1.Done THEN
   MC_MoveRelative_1.Execute:=FALSE;
   lb_CutterMoveComp:=FALSE;
END_IF;

MC_MoveRelative_1(
   Axis:= CutterDriveBack.AxisRef,
   Distance:= -250000.0,
   Velocity:= 3000000.0,
   Acceleration:= 0.5,
   Deceleration:= 0.5,
   Options:= H0001);
```

- Labels used

Label name	Data type	Comment
MC_MoveRelative_1	MC_MoveRelative	Relative positioning (cutter drive axis return operation)
lb_CutterMoveComp	Bit	Cutter operation complete flag

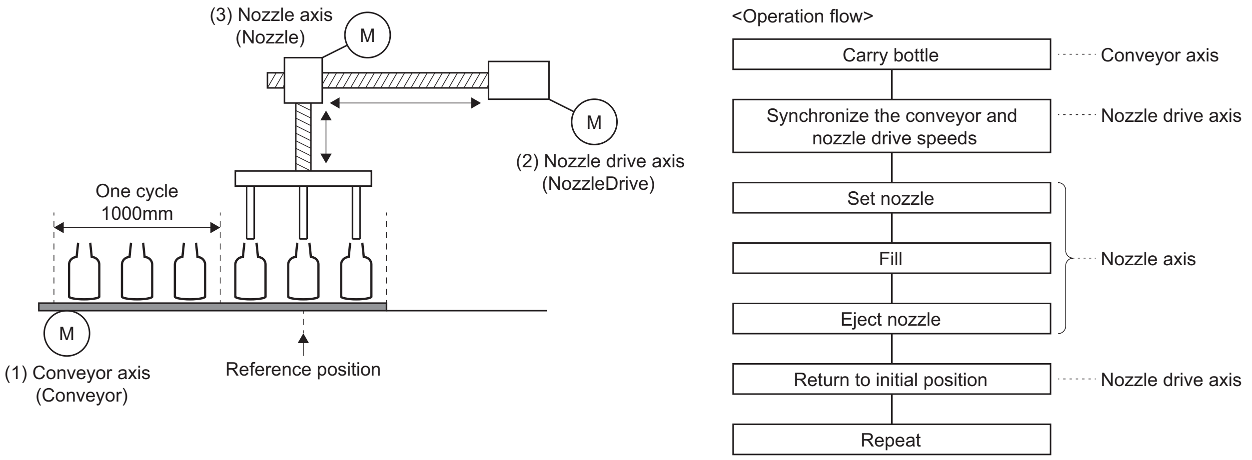
Timing chart

The timing chart for the operation that uses the program example is shown below.



Filling machine

This section describes the example program and how to use synchronous control of multiple axes that use an advanced synchronous control FB operation profile (cam data), and how to restart synchronous control based on a filling machine system.



Setting item	(1) Conveyor axis	(2) Nozzle drive axis	(3) Nozzle axis
Axis name	Conveyor	NozzleDrive	Nozzle
Axis type	Real drive axis	Real drive axis	Real drive axis
Station address	192.168.3.1	192.168.3.2	192.168.3.3
Position command unit	μm	μm	μm

Operation overview

1. An advanced synchronous control FB starts with the conveyor axis as the input axis and the nozzle drive axis as the output axis. Additionally, an advanced synchronous control FB starts with the conveyor axis as the input axis and the nozzle axis as the output axis.

2. The conveyor axis starts. The nozzle drive axis and the nozzle axis synchronize with the conveyor axis according to the set length per cycle and operation profile (cam data) and execute the operations in step 3. to 6.

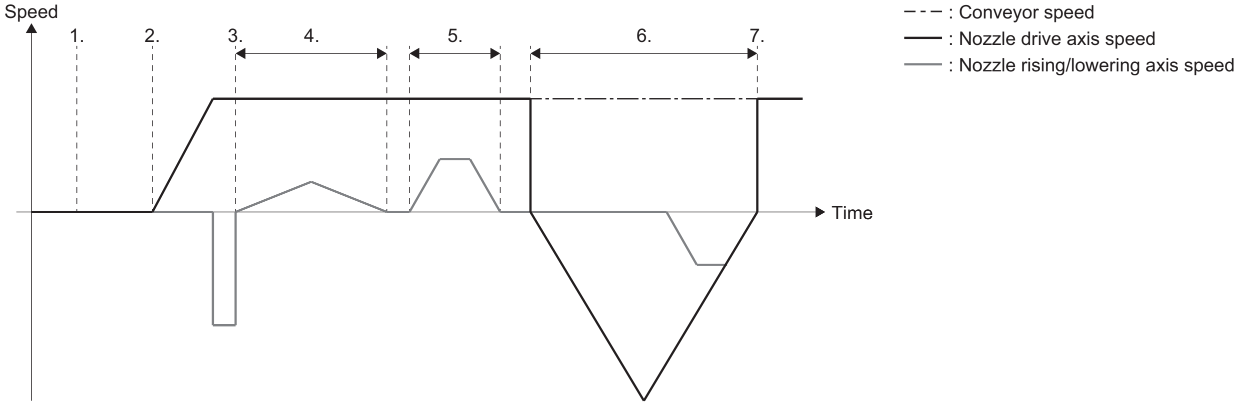
3. The nozzle is set on the nozzle axis.

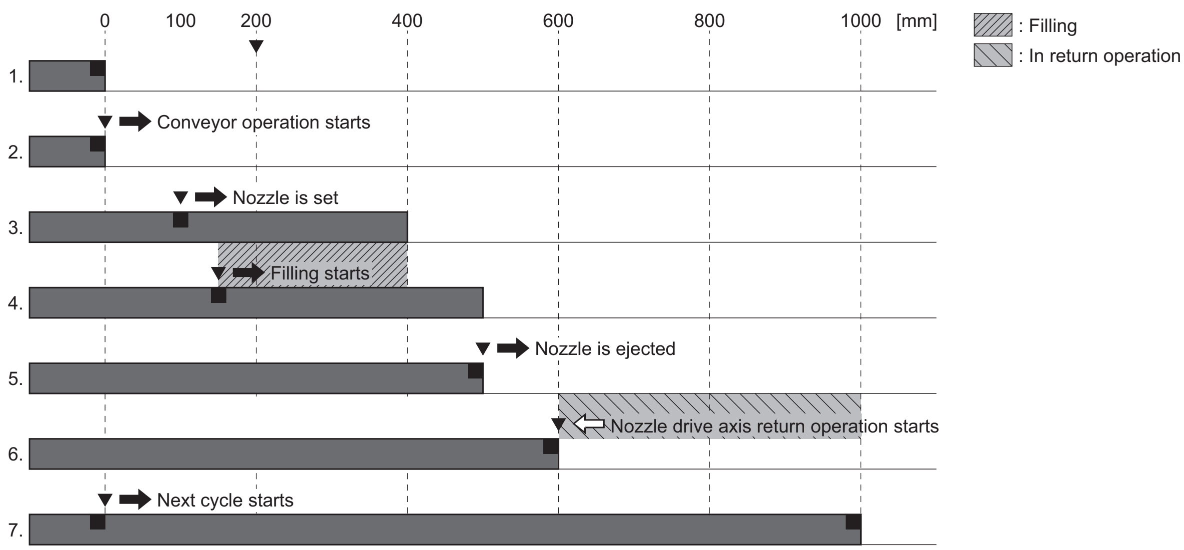
4. The bottle is filled.

5. The nozzle on the nozzle axis is ejected.

6. The nozzle drive axis and nozzle axis return to the reference position.

7. Step 3. to 6. are repeated.





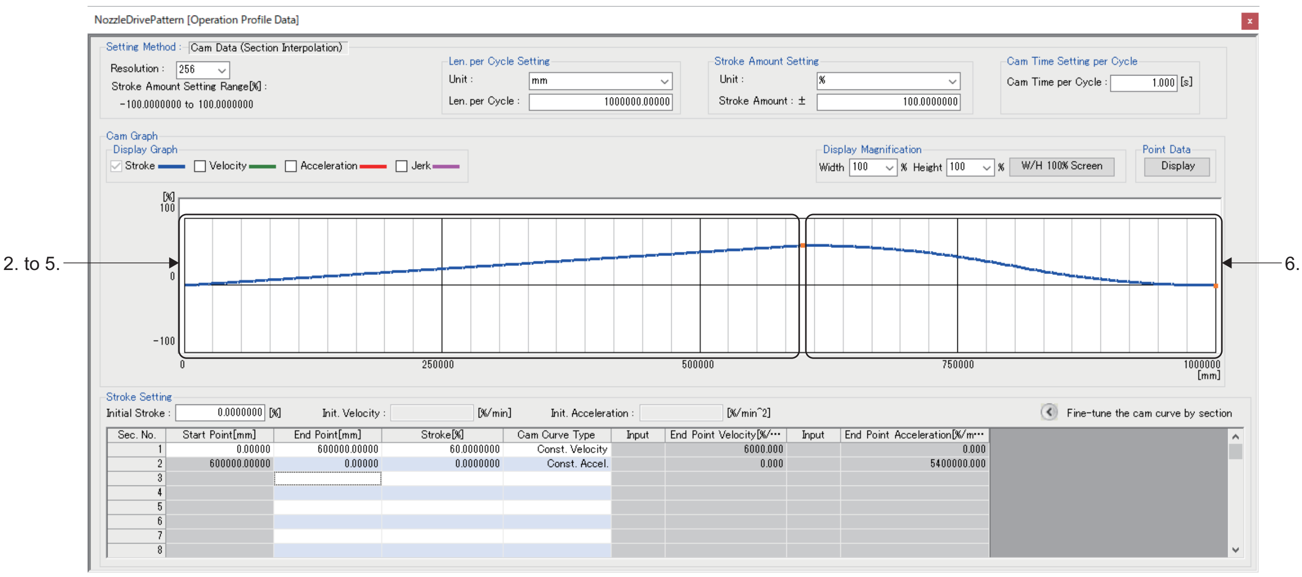
Creating an operation profile (cam data)

The following describes how to create the operation profile (cam data) used by the nozzle drive axis and nozzle axis output axes.

■ Creating the nozzle drive axis operation profile (cam data)

Create the "NozzleDrivePattern" cam data below with "Profile ID: 1" as the the operation profile (cam data).

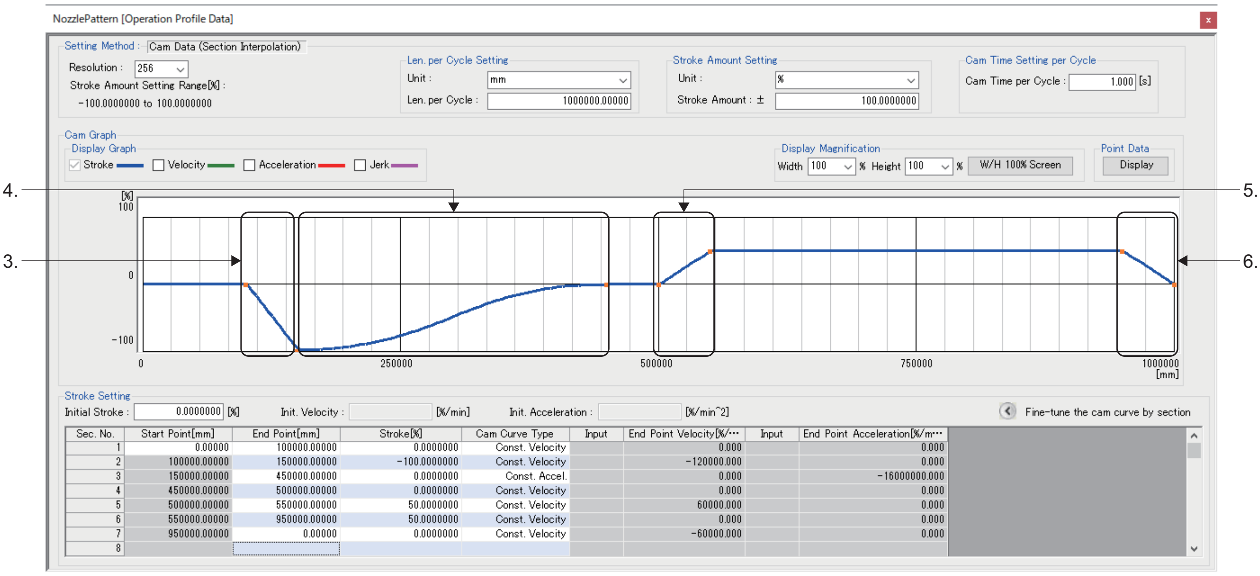
Step	Cam pattern details of each step	Setting details
2. to 5.	Cam pattern that synchronizes the speed with the conveyor axis	• Current position per cycle: 0.0 [mm] → 600000.0 [mm] • Stroke amount: 0.0 [%] → 60.0 [%]
6.	Cam pattern that returns the nozzle drive axis to the reference position	• Current position per cycle: 600000.0 [mm] → 1000000.0 [mm] (0.0 [mm]) • Stroke amount: 60.0 [%] → 0.0 [%]



■ Creating the nozzle axis operation profile (cam data)

Create the "NozzlePattern" cam data below with "Profile ID: 2" as the the operation profile (cam data).

Step	Cam pattern details of each step	Setting details
3.	Cam pattern for the nozzle setting operation	• Current position per cycle:100000.0 [mm] → 150000.0 [mm] • Stroke amount: 0.0 [%] → -100.0 [%]
4.	Cam pattern while filling (operation that gradually raises the nozzle)	• Current position per cycle:150000.0 [mm] → 450000.0 [mm] • Stroke amount: 0.0 [%] → -100.0 [%] → 0.0 [%]
5.	Cam pattern for the nozzle eject operation	• Current position per cycle:500000.0 [mm] → 550000.0 [mm] • Stroke amount: 0.0 [%] → 0.0 [%] → 50.0 [%]
6.	Cam pattern that returns the nozzle to the reference position	• Current position per cycle:950000.0 [mm] → 1000000 [mm] (0.0 [mm]) • Stroke amount: 0.0 [%] →50.0 [%] → 0.0 [%]



Sec. No.	Start Point[mm]	End Point[mm]	Stroke[%]	Cam Curve Type	Input	End Point Velocity[%/···	Input	End Point Acceleration[%/m···
1	0.00000	100000.00000	0.0000000	Const. Velocity		0.000		0.000
2	100000.00000	150000.00000	-100.0000000	Const. Velocity		-120000.000		0.000
3	150000.00000	450000.00000	0.0000000	Const. Accel.		0.000		-16000000.000
4	450000.00000	500000.00000	0.0000000	Const. Velocity		0.000		0.000
5	500000.00000	550000.00000	50.0000000	Const. Velocity		60000.000		0.000
6	550000.00000	950000.00000	50.0000000	Const. Velocity		0.000		0.000
7	950000.00000	0.00000	0.0000000	Const. Velocity		-60000.000		0.000
8								

Parameter settings

Set the input axis parameters for the axis (conveyor axis) to be used as the input for the advanced synchronous control FB and the output axis parameters for the axes (nozzle drive axis and nozzle axis) to be used as the output for the advanced synchronous control FB.

Set the parameters for the axes used as the input/output for the advanced synchronous control FB.

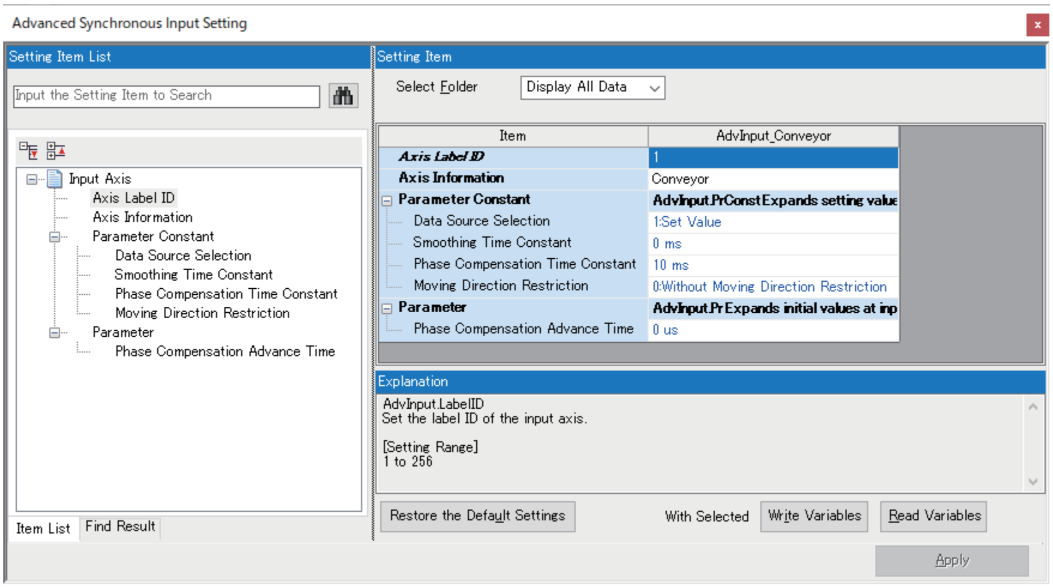
Parameter	Axis used
Input axis parameter	• Conveyor axis
Output axis parameter	• Nozzle drive axis • Nozzle axis

■ Advanced synchronous control input axis parameter settings

In the "Advanced Synchronous Input Setting" screen of Motion Control Setting Function, configure the input axis settings of the main shaft and auxiliary shaft used by the advanced synchronous control FB.

Data name	Setting details
AdvInput_Conveyor	Real drive axis (Conveyor) of the conveyor axis

Motion Control Setting Function ⇨ Navigation window ⇨ [Add-on Data] ⇨ [Advanced Synchronous Control] ⇨ [Input Setting]

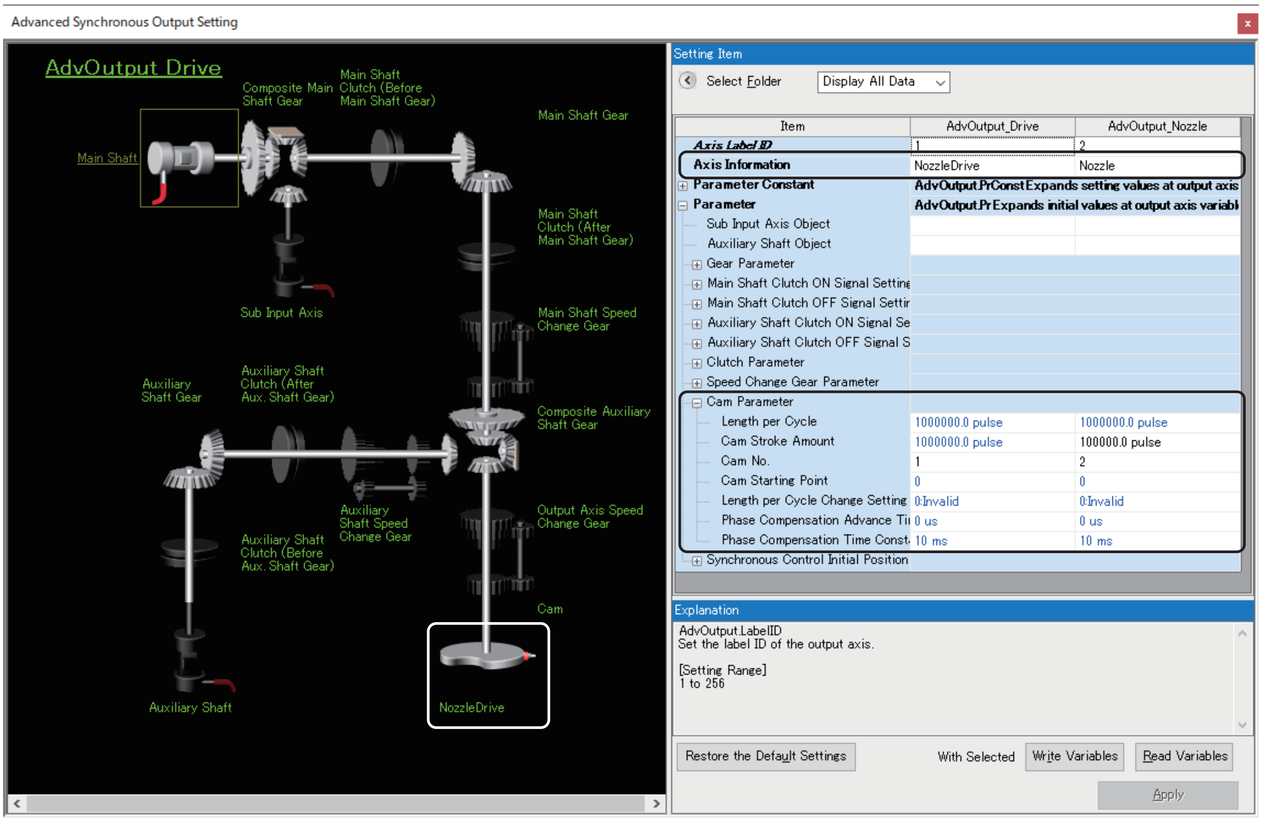


15

■ Advanced synchronous control FB output axis parameter settings

In the "Advanced Synchronous Output Setting" screen of Motion Control Setting Function, configure the output axis settings of the output axes used by the advanced synchronous control FB.

Motion Control Setting Function ⇨ Navigation window ⇨ [Addon Data] ⇨ [Advanced Synchronous Control] ⇨ [Output Setting]



- Settings of axes used by output axis and auxiliary shaft

Seting item	Setting value	
	Nozzle drive axis	Nozzle axis
Axis information (AdvOutputName.Axis)	NozzleDrive	Nozzle

- Cam parameter setting

Seting item	Setting value	
	Nozzle drive axis	Nozzle axis
Length per cycle	1000000.0 (Match with the length per cycle set in the operation profile)	1000000.0 (Match with the length per cycle set in the operation profile)
Cam stroke amount	1000000.0 (Same as the length per cycle)	100000.0 (Set the stroke amount of the output axis when the stroke amount[%] set in the operation profile is 100[%].)
Cam No.	1 (The profile ID set in the operation profile (NozzleDrivePattern))	2 (The profile ID set in the operation profile (NozzlePattern))

Program creation

■ Starting advanced synchronous control (initial start)

After setting the positions of the input axis and output axes by homing, etc., confirm that the input axis is enabled (AdvInput_Conveyor.Md.ActiveStatus is "1: Input axis enabled") and the output axes are on standby (NozzleDrive.Md.AxisStatus is "4: Standby (Standstill)" and Nozzle.Md.AxisStatus is "4: Standby (Standstill)") before starting the advanced synchronous control FB.

This section also describes the procedures for restarting synchronous control using cam set position restoration (restoring the input axis position as reference).

For restarting synchronous control using cam set position restoration (restoring the input axis position as reference), refer to the following.

☞ Page 633 Starting advanced synchronous control (restart)

Set the positions of the input axis (Conveyor) and the output axes (NozzleDrive and Nozzle), set the synchronous control initial position parameters as shown below, and start the advanced synchronous control FB.

Setting item	Setting value
Setting method of current position per cycle after main shaft gear • AdvOutput_NozzleDrive.Pr.Restore.MasterGearPositionPerCycleMethod • AdvOutput_Nozzle.Pr.Restore.MasterGearPositionPerCycleMethod	2: Calculate from input axis (CalculateFromInputAxis)
Cam axis position restoration object • AdvOutput_NozzleDrive.Pr.Restore.PositionRestorationObject • AdvOutput_Nozzle.Pr.Restore.PositionRestorationObject	0: Current position per cycle restoration (RestorePositionPerCycle)
Setting method of cam reference position • AdvOutput_NozzleDrive.Pr.Restore.ReferenceSetPositionMethod • AdvOutput_Nozzle.Pr.Restore.ReferenceSetPositionMethod	2: Set position (CamSetPosition)
Current position per cycle (initial setting) • AdvOutput_NozzleDrive.Pr.Restore.InitialPositionPerCycle • AdvOutput_Nozzle.Pr.Restore.InitialPositionPerCycle	0.0

- Example program

■Nozzle drive axis

ST program

```
IF NOT(gb_SystemStart) THEN
   gb_PositionAdjustComp_Drive:=FALSE;
   END_IF;

//----------For initial start (from here)----------
IF gb_SystemStart AND NOT(gb_SyncRestart) THEN
   // AdvSync Start(NozzleDrive)
   // Set Restore Parameter
   AdvOutput_Drive.Pr.Restore.MasterGearPositionPerCycleMethod:=MC_GEAR_RESTORE_METHOD__CalculateFromInputAxis;
   AdvOutput_Drive.Pr.Restore.PositionRestorationObject:=MC_CAM_RESTORE_METHOD__RestorePositionPerCycle;
   AdvOutput_Drive.Pr.Restore.ReferenceSetPositionMethod:=MC_CAM_REFERENCE_METHOD__CamSetPosition;
   AdvOutput_Drive.Pr.Restore.InitialPositionPerCycle:=0.0;

   IF AdvInput_Conveyor.Md.ActiveStatus=1 AND
      NozzleDrive.Md.AxisStatus=MC_AXIS_STATUS__Standstill THEN
      gdw_AdvSync_Drive_Options:=H00000000;
      gb_AdvSync_Drive_Enable:=TRUE;
   END_IF;
END_IF;
//----------For initial start (until here)----------

//----------For restart (from here)----------
IF gb_SystemStart AND gb_SyncRestart THEN
   // AdvSync Restart(NozzleDrive)

   // Select For Restore Target
   CASE gw_SelectRestoreMode OF
   RESTORE_POSITION_PER_CYCLE_FOR_PREPROMODE :
<Omission>*1
   RESTORE_POSITION_PER_CYCLE_FOR_CAMCALCFUNC :
<Omission>*1
   RESTORE_SETPOSITION_FOR_PREPROMODE :
<Omission>*1
   RESTORE_SETPOSITION_FOR_CAMCALCFUNC :
<Omission>*1

   ELSE
      ;
   END_CASE;
END_IF;
//----------For restart (until here)----------

MC_MoveAbsolute_Drive(
   Axis:= NozzleDrive.AxisRef,
   Execute:= gb_MoveAbsolute_Drive_Execute,
   Position:= gl_MoveAbsolute_Drive_Position,
   Velocity:= 10000000.0,
   Acceleration:= 0.5,
   Deceleration:= 0.5,
   Direction:= MC_DIRECTION__mcShortestWay,
   Options:= H0001);

MCv_AdvancedSync_Drive(
   Master:= AdvInput_Conveyor.Axis,
   Slave:=  AdvOutput_Drive.Axis,
   Options:= gdw_AdvSync_Drive_Options,
   Enable:= gb_AdvSync_Drive_Enable,
   InSync=> gb_AdvSync_Drive_InSync);

MCv_AdvCamSetPositionCalc_Drive(
   Execute:= gb_AdvCamSetPositionCalc_Drive_Execute,
   CamTableID:= g_CamTableID_Drive,
   LengthPercycle:= gl_AdvCamSetPositionCalc_Drive_LengthPercycle,
   StrokeAmount:= gl_AdvCamSetPositionCalc_Drive_StrokeAmount,
   ReferenceSetPosition:= gl_AdvCamSetPositionCalc_Drive_ReferenceSetPosition,
   PositionPerCycle:= gl_AdvCamSetPositionCalc_Drive_PositionPerCycle);
```

*1 For an example program when restarting advanced synchronous control, refer to the following.

☞ Page 633 Starting advanced synchronous control (restart)

■Nozzle axis

ST program

```
IF NOT(gb_SystemStart) THEN
    gb_PositionAdjustComp_Nozzle:=FALSE;
END_IF;

//----------For initial start (from here)----------
IF gb_SystemStart AND NOT(gb_SyncRestart) THEN
    // AdvSync Start(Nozzle)
    // Set Restore Parameter
    AdvOutput_Nozzle.Pr.Restore.MasterGearPositionPerCycleMethod:=MC_GEAR_RESTORE_METHOD__CalculateFromInputAxis;
    AdvOutput_Nozzle.Pr.Restore.PositionRestorationObject:=MC_CAM_RESTORE_METHOD__RestorePositionPerCycle;
    AdvOutput_Nozzle.Pr.Restore.ReferenceSetPositionMethod:=MC_CAM_REFERENCE_METHOD__CamSetPosition;
    AdvOutput_Nozzle.Pr.Restore.InitialPositionPerCycle:=0.0;

    IF AdvInput_Conveyor.Md.ActiveStatus=1 AND
        Nozzle.Md.AxisStatus=MC_AXIS_STATUS__Standstill THEN
        gdw_AdvSync_Nozzle_Options:=H00000000;
        gb_AdvSync_Nozzle_Enable:=TRUE;
    END_IF;
END_IF;
//----------For initial start (until here)----------

//----------For restart (from here)----------
IF gb_SystemStart AND gb_SyncRestart THEN
    // AdvSync Restart(Nozzle)

    // Select For Restore Target
    CASE gw_SelectRestoreMode OF
    RESTORE_POSITION_PER_CYCLE_FOR_PREPROMODE :
<Omission>*1
    RESTORE_POSITION_PER_CYCLE_FOR_CAMCALCFUNC :
<Omission>*1
    RESTORE_SETPOSITION_FOR_PREPROMODE :
<Omission>*1
    RESTORE_SETPOSITION_FOR_CAMCALCFUNC :
<Omission>*1

    ELSE
        ;
    END_CASE;
END_IF;
//----------For restart (until here)----------

MC_MoveAbsolute_Nozzle(
    Axis:= Nozzle.AxisRef,
    Execute:= gb_MoveAbsolute_Nozzle_Execute,
    Position:= gl_MoveAbsolute_Nozzle_Position,
    Velocity:= 10000000.0,
    Acceleration:= 0.5,
    Deceleration:= 0.5,
    Direction:= MC_DIRECTION__mcShortestWay,
    Options:= H0001);

MCv_AdvancedSync_Nozzle(
    Master:= AdvInput_Conveyor.Axis,
    Slave:=  AdvOutput_Nozzle.Axis,
    Options:= gdw_AdvSync_Nozzle_Options,
    Enable:= gb_AdvSync_Nozzle_Enable,
    Deceleration:= 2000000.0,
    InSync=> gb_AdvSync_Nozzle_InSync);

MCv_AdvCamSetPositionCalc_Nozzle(
    Execute:= gb_AdvCamSetPositionCalc_Nozzle_Execute,
    CamTableID:= g_CamTableID_Nozzle,
    LengthPercycle:= gl_AdvCamSetPositionCalc_Nozzle_LengthPercycle,
    StrokeAmount:= gl_AdvCamSetPositionCalc_Nozzle_StrokeAmount,
    ReferenceSetPosition:= gl_AdvCamSetPositionCalc_Nozzle_ReferenceSetPosition,
    PositionPerCycle:= gl_AdvCamSetPositionCalc_Nozzle_PositionPerCycle);

//-----Continued on next page-----
```

ST program

//-----continued from previous page-----
 	
MCv_AdvPositionPerCycleCalc_Nozzle(
    Execute:= gb_AdvPositionPerCycleCalc_Nozzle_Execute,
    CamTableID:= g_CamTableID_Nozzle,
    LengthPercycle:= gl_AdvPositionPerCycleCalc_Nozzle_LengthPercycle,
    StrokeAmount:= gl_AdvPositionPerCycleCalc_Nozzle_StrokeAmount,
    ReferenceSetPosition:= gl_AdvPositionPerCycleCalc_Nozzle_ReferenceSetPosition,
    SetPosition:= gl_AdvPositionPerCycleCalc_Nozzle_SetPosition,
    PositionPerCycle:= gl_AdvPositionPerCycleCalc_Nozzle_PositionPerCycle);

*1 For an example program when restarting advanced synchronous control, refer to the following.
☞ Page 633 Starting advanced synchronous control (restart)

• Labels used

Label name	Data type	Comment
gb_SystemStart	Bit	System start command
gb_SyncRestart	Bit	System restart command
MC_MoveAbsolute_Drive	MC_MoveAbsolute	Absolute value positioning FB (Nozzle drive axis positioning)
gb_MoveAbsolute_Drive_Execute	Bit	Absolute value positioning FB start command (nozzle drive axis)
gl_MoveAbsolute_Drive_Position	Double-precision real number	Absolute value positioning FB target position (nozzle drive axis)
gb_PositionAdjustComp_Drive	Bit	Nozzle drive axis positioning complete flag
MCv_AdvancedSync_Drive	MCv_AdvancedSync	Advanced synchronous control FB (nozzle drive axis)
gb_AdvSync_Drive_Enable	Bit	Advanced synchronous control FB start command (nozzle drive axis)
gdw_AdvSync_Drive_Options	Double word [unsigned]/bit string [32-bit]	Advanced synchronous control FB option (nozzle drive axis)
gb_AdvSync_Drive_InSync	Bit	Advanced synchronous control FB synchronizing (nozzle drive axis)
MC_MoveAbsolute_Nozzle	MC_MoveAbsolute	Absolute value positioning FB (nozzle axis positioning)
gb_MoveAbsolute_Nozzle_Execute	Bit	Absolute value positioning FB start command (nozzle axis)
gl_MoveAbsolute_Nozzle_Position	Double-precision real number	Absolute value positioning FB target position (nozzle axis)
gb_PositionAdjustComp_Nozzle	Bit	Nozzle axis positioning complete flag
MCv_AdvancedSync_Nozzle	MCv_AdvancedSync	Advanced synchronous control FB (nozzle axis)
gb_AdvSync_Nozzle_Enable	Bit	Advanced synchronous control FB start command (nozzle axis)
gdw_AdvSync_Nozzle_Options	Double word [unsigned]/bit string [32-bit]	Advanced synchronous control FB option (nozzle axis)
gb_AdvSync_Nozzle_InSync	Bit	Advanced synchronous control FB synchronizing (nozzle axis)
gw_SelectRestoreMode	Word [signed]	Restoration mode selection at system restart
g_CamTableID_Drive	MC_CAM_ID	Cam ID (nozzle axis)
g_CamTableID_Nozzle	MC_CAM_ID	Cam ID (nozzle drive axis)
MCv_AdvCamSetPositionCalc_Nozzle	MCv_AdvCamSetPositionCalc	Advanced synchronous control cam set position calculation FB (nozzle axis)
gb_AdvCamSetPositionCalc_Nozzle_Execute	Bit	Advanced synchronous control cam set position calculation FB execute command (nozzle axis)
gl_AdvCamSetPositionCalc_Nozzle_PositionPerCycle	Double-precision real number	Advanced synchronous control cam set position calculation FB current position per cycle (nozzle axis)
gl_AdvCamSetPositionCalc_Nozzle_LengthPercycle	Double-precision real number	Advanced synchronous control cam set position calculation FB length per cycle (nozzle axis)
gl_AdvCamSetPositionCalc_Nozzle_StrokeAmount	Double-precision real number	Advanced synchronous control cam set position calculation FB stroke amount (nozzle axis)
gl_AdvCamSetPositionCalc_Nozzle_ReferenceSetPosition	Double-precision real number	Advanced synchronous control cam set position calculation FB cam reference position (nozzle axis)
MCv_AdvCamSetPositionCalc_Drive	MCv_AdvCamSetPositionCalc	Advanced synchronous control cam set position calculation FB (nozzle drive axis)
gb_AdvCamSetPositionCalc_Drive_Execute	Bit	Advanced synchronous control cam set position calculation FB execute command (nozzle drive axis)
gl_AdvCamSetPositionCalc_Drive_PositionPerCycle	Double-precision real number	Advanced synchronous control cam set position calculation FB current position per cycle (nozzle drive axis)
gl_AdvCamSetPositionCalc_Drive_LengthPercycle	Double-precision real number	Advanced synchronous control cam set position calculation FB length per cycle (nozzle drive axis)

Label name	Data type	Comment
gl_AdvCamSetPositionCalc_Drive_StrokeAmount	Double-precision real number	Advanced synchronous control cam set position calculation FB stroke amount (nozzle drive axis)
gl_AdvCamSetPositionCalc_Drive_ReferenceSetPosition	Double-precision real number	Advanced synchronous control cam set position calculation FB cam reference position (nozzle drive axis)
MCv_AdvPositionPerCycleCalc_Nozzle	MCv_AdvPositionPerCycleCalc	Advanced synchronous control current position per cycle calculation FB (nozzle axis)
gb_AdvPositionPerCycleCalc_Nozzle_Execute	Bit	Advanced synchronous control current position per cycle calculation FB current position per cycle (nozzle axis)
gl_AdvPositionPerCycleCalc_Nozzle_PositionPerCycle	Double-precision real number	Advanced synchronous control current position per cycle calculation FB current position per cycle (nozzle axis)
gl_AdvPositionPerCycleCalc_Nozzle_LengthPercycle	Double-precision real number	Advanced synchronous control current position per cycle calculation FB length per cycle (nozzle axis)
gl_AdvPositionPerCycleCalc_Nozzle_StrokeAmount	Double-precision real number	Advanced synchronous control current position per cycle calculation FB stroke amount (nozzle axis)
gl_AdvPositionPerCycleCalc_Nozzle_ReferenceSetPosition	Double-precision real number	Advanced synchronous control current position per cycle calculation FB cam reference position (nozzle axis)
gl_AdvPositionPerCycleCalc_Nozzle_SetPosition	Double-precision real number	Advanced synchronous control current position per cycle calculation FB cam set position (nozzle axis)
gdw_AdvInput_Conveyor_CumulativePosition	Double word [signed]	Cumulative current position (conveyor axis) A temporary variable for rounding the cumulative current position calculation to the current position per cycle range
gdw_AdvOutput_Nozzle_LengthPerCycle	Double word [signed]	Current position per cycle (nozzle axis) A temporary variable for rounding the cumulative current position calculation to the current position per cycle range
gdw_AdvOutput_Drive_LengthPerCycle	Double word [signed]	Current position per cycle (nozzle drive axis) A temporary variable for rounding the cumulative current position calculation to the current position per cycle range
RESTORE_POSITION_PER_CYCLE_FOR_PREPROMODE	Word [signed]	Restoration mode selection at system restart: Current position per cycle restoration (analysis mode) (constant=1)
RESTORE_POSITION_PER_CYCLE_FOR_CAMCALCFUNC	Word [signed]	Restoration mode selection at system restart: Current position per cycle restoration (cam position calculation function) (constant=2)
RESTORE_SETPOSITION_FOR_PREPROMODE	Word [signed]	Restoration mode selection at system restart: Set position (analysis mode) (constant=3)
RESTORE_SETPOSITION_FOR_CAMCALCFUNC	Word [signed]	Restoration mode selection at system restart: Set position (cam position calculation function) (constant=4)

■ Starting advanced synchronous control (restart)

When restarting advanced synchronous control select a restoration method suitable for the configuration and application of the equipment and start the advanced synchronous control FB. Additionally, use "synchronous control analysis mode" or "cam position calculation function" when recalculating to synchronize the input axis and output axis in order to restart advanced synchronous control.

An example of restoration procedures when using "synchronous control analysis mode" or "cam position calculation function" is shown below.

No.	Restoration method	Using method	Application
1	Current position per cycle restoration	Synchronous control analysis mode	Restart synchronization after moving the nozzle axis to the retract point.
2		Cam position calculation function (MCv_AdvCamSetPositionCalc (Advanced synchronous control current position per cycle calculation))	
3	Cam set position restoration	Synchronous control analysis mode	Restart with the phases of input axis cumulative current position and the output axis current position per cycle matched. (When the workpiece has not been set)
4		Cam position calculation function (MCv_AdvPositionPerCycleCalc (Advanced synchronous control cam set position calculation))	

Point

For restoration after turning ON the motion system power again, the input axis and output axis must be set to absolute position detection system.

15

Operating procedure

■ No.1 "Restart synchronization after moving the nozzle axis to the retract point" (when using synchronous control analysis mode)

1. Position the output axis (Nozzle) to the retract point (position of stroke amount "50%").

2. Set the synchronous control initial position parameters of the output axis (Nozzle) as follows, and execute the advanced synchronous control FB with the synchronous control analysis mode setting (Options (Options): Bit 16) set to "1: Enabled".

Setting item	Setting value
Setting method of current position per cycle after main shaft gear • AdvOutput_Nozzle.Pr.Restore.MasterGearPositionPerCycleMethod	2: Calculate from input axis (CalculateFromInputAxis)
Cam axis position restoration object • AdvOutput_Nozzle.Pr.Restore.PositionRestorationObject	0: Current position per cycle restoration (RestorePositionPerCycle)
Setting method of cam reference position • AdvOutput_Nozzle.Pr.Restore.ReferenceSetPositionMethod	0: Previous value(PreviousPosition)
Current position per cycle (initial setting) • AdvOutput_NozzleDrive.Pr.Restore.InitialPositionPerCycle	0.0

3. To calculate the Cam set position (AdvOutput_Drive.Md.Cam.SetPosition) at the start of advanced synchronous control of the output axis (NozzleDrive) based on the position per cycle calculated in step 2., set the synchronous control initial position parameters of the output axis (NozzleDrive) as follows, and execute the advanced synchronous control FB with the synchronous control analysis mode setting (Options (Options): Bit 16) set to "1: Enabled".

Setting item	Setting value
Setting method of current position per cycle after main shaft gear • AdvOutput_NozzleDrive.Pr.Restore.MasterGearPositionPerCycleMethod	2: Calculate from input axis (CalculateFromInputAxis)
Cam axis position restoration object • AdvOutput_NozzleDrive.Pr.Restore.PositionRestorationObject	2: Cam set position restoration (RestoreCamSetPosition)
Setting method of cam reference position • AdvOutput_NozzleDrive.Pr.Restore.ReferenceSetPositionMethod	0: Previous value (PreviousPosition)
Setting method of current position per cycle • AdvOutput_NozzleDrive.Pr.Restore.PositionPerCycleMethod	1: Current position per cycle (InitialPositionPerCycle)
Current position per cycle (initial setting) • AdvOutput_NozzleDrive.Pr.Restore.InitialPositionPerCycle	The current position per cycle calculated in step 2. • AdvOutput_Nozzle.Md.Cam.PositionPerCycle

4. After synchronous control analysis mode execution completion (MCv_AdvancedSync.Done is TRUE), position the nozzle drive axis (NozzleDrive) to the position of the updated Cam set position (AdvOutput_Drive.Md.Cam.SetPosition).

5. Set the synchronous control initial position parameters of the output axes (NozzleDrive and Nozzle) as follows, and start (change Execute from FALSE to TRUE) the advanced synchronous control FB.

Setting item	Setting value
Setting method of current position per cycle after main shaft gear • AdvOutput_NozzleDrive.Pr.Restore.MasterGearPositionPerCycleMethod • AdvOutput_Nozzle.Pr.Restore.MasterGearPositionPerCycleMethod	2: Calculate from input axis (CalculateFromInputAxis)
Cam axis position restoration object • AdvOutput_NozzleDrive.Pr.Restore.PositionRestorationObject • AdvOutput_Nozzle.Pr.Restore.PositionRestorationObject	0: Current position per cycle restoration (RestorePositionPerCycle)
Setting method of cam reference position • AdvOutput_NozzleDrive.Pr.Restore.ReferenceSetPositionMethod • AdvOutput_Nozzle.Pr.Restore.ReferenceSetPositionMethod	0: Previous value (PreviousPosition)
Current position per cycle (initial setting) • AdvOutput_NozzleDrive.Pr.Restore.InitialPositionPerCycle • AdvOutput_Nozzle.Pr.Restore.InitialPositionPerCycle	The current position per cycle calculated in step 2. • AdvOutput_Nozzle.Md.Cam.PositionPerCycle

• Example program

■Nozzle drive axis

ST program[*1]

```
// Select For Restore Target
CASE gw_SelectRestoreMode OF
RESTORE_POSITION_PER_CYCLE_FOR_PREPROMODE :
   // Set Restore Parameter
   AdvOutput_Drive.Pr.Restore.MasterGearPositionPerCycleMethod:=MC_GEAR_RESTORE_METHOD__CalculateFromInputAxis;
   AdvOutput_Drive.Pr.Restore.PositionRestorationObject:=MC_CAM_RESTORE_METHOD__RestoreCamSetPosition;
   AdvOutput_Drive.Pr.Restore.ReferenceSetPositionMethod:=MC_CAM_REFERENCE_METHOD__PreviousPosition;
   AdvOutput_Drive.Pr.Restore.PositionPerCycleMethod:=MC_CAM_CYCLE_METHOD__InitialPositionPerCycle;
   AdvOutput_Drive.Pr.Restore.InitialPositionPerCycle:=gl_Restored_Nozzle_PositionPerCycle;

   // Positionadjustment
   IF gb_PositionAdjustComp_Nozzle THEN
      IF NOT(gb_PositionAdjustComp_Drive) THEN
         IF AdvInput_Conveyor.Md.ActiveStatus=1 THEN
            gdw_AdvSync_Drive_Options:=H00010000;
            gb_AdvSync_Drive_Enable:=TRUE;

            IF MCv_AdvancedSync_Drive.Done THEN
               gl_MoveAbsolute_Drive_Position:=AdvOutput_Drive.Md.Cam.SetPosition;
               gb_MoveAbsolute_Drive_Execute:=TRUE;
            END_IF;

            IF MC_MoveAbsolute_Drive.Done THEN
               gb_AdvSync_Drive_Enable:=FALSE;
               gb_MoveAbsolute_Drive_Execute:=FALSE;
               gb_PositionAdjustComp_Drive:=TRUE;
            END_IF;
         END_IF;
         ELSE
         IF AdvInput_Conveyor.Md.ActiveStatus=1 AND
            NozzleDrive.Md.AxisStatus=MC_AXIS_STATUS__Standstill AND
            NOT(MCv_AdvancedSync_Drive.Done) THEN
            // Set Restore Parameter
            AdvOutput_Drive.Pr.Restore.PositionRestorationObject:=MC_CAM_RESTORE_METHOD__RestorePositionPerCycle;
            AdvOutput_Drive.Pr.Restore.PositionPerCycleMethod:=MC_CAM_CYCLE_METHOD__InitialPositionPerCycle;
            AdvOutput_Drive.Pr.Restore.InitialPositionPerCycle:=gl_Restored_Nozzle_PositionPerCycle;
            gdw_AdvSync_Drive_Options:=H00000000;
            gb_AdvSync_Drive_Enable:=TRUE;
         END_IF;
      END_IF;
   ND_IF;
```

*1 For the entire program, refer to the following.

☞ Page 629 Starting advanced synchronous control (initial start)

■Nozzle axis

ST program[*1]

```
// Select For Restore Target
CASE gw_SelectRestoreMode OF
RESTORE_POSITION_PER_CYCLE_FOR_PREPROMODE :
  // AdvSync Restart(Nozzle)
  AdvOutput_Nozzle.Pr.Restore.MasterGearPositionPerCycleMethod:=MC_GEAR_RESTORE_METHOD__CalculateFromInputAxis;
  AdvOutput_Nozzle.Pr.Restore.PositionRestorationObject:=MC_CAM_RESTORE_METHOD__RestorePositionPerCycle;
  AdvOutput_Nozzle.Pr.Restore.ReferenceSetPositionMethod:=MC_CAM_REFERENCE_METHOD__PreviousPosition;
  AdvOutput_Nozzle.Pr.Restore.InitialPositionPerCycle:=0.0;
  // Positionadjustment
  IF NOT(gb_PositionAdjustComp_Nozzle) THEN
    IF AdvInput_Conveyor.Md.ActiveStatus=1 THEN
        gl_MoveAbsolute_Nozzle_Position:=AdvOutput_Nozzle.Pr.Cam.StrokeAmount * 0.5;
        gb_MoveAbsolute_Nozzle_Execute:=TRUE;
      IF MC_MoveAbsolute_Nozzle.Done THEN
        gdw_AdvSync_Nozzle_Options:=H00010000;
        gb_AdvSync_Nozzle_Enable:=TRUE;
      END_IF;

      IF MCv_AdvancedSync_Nozzle.Done THEN
        gl_Restored_Nozzle_PositionPerCycle:=AdvOutput_Nozzle.Md.Cam.PositionPerCycle;
        gb_AdvSync_Nozzle_Enable:=FALSE;
        gb_PositionAdjustComp_Nozzle:=TRUE;
      END_IF;
    END_IF;
    ELSE
    IF AdvInput_Conveyor.Md.ActiveStatus=1 AND
      NozzleDrive.Md.AxisStatus=MC_AXIS_STATUS__Standstill AND
      NOT(MCv_AdvancedSync_Nozzle.Done) THEN
      AdvOutput_Drive.Pr.Restore.PositionPerCycleMethod:=MC_CAM_CYCLE_METHOD__InitialPositionPerCycle;
      AdvOutput_Drive.Pr.Restore.InitialPositionPerCycle:=gl_Restored_Nozzle_PositionPerCycle;
      gdw_AdvSync_Nozzle_Options:=H00000000;
      gb_AdvSync_Nozzle_Enable:=TRUE;
    END_IF;
  END_IF;
```

*1 For the entire program, refer to the following.
☞ Page 629 Starting advanced synchronous control (initial start)

- Labels used

For the labels used in the program, refer to the following.
☞ Page 629 Starting advanced synchronous control (initial start)

■ No.2 "Restart synchronization after moving the nozzle axis to the retract point" (when using cam position calculation function (MCv_AdvCamSetPositionCalc (Advanced Synchronous Control Current Position Per Cycle Calculation)))

1. Position the output axis (Nozzle) to the retract point (position of stroke amount "50%").

2. Calculate the current position per cycle based on the set position of the output axis (in this case output axis (Nozzle)) which keeps the set position.

Setting item	Setting value
Cam table ID • MCv_AdvPositionPerCycleCalc_Nozzle.CamTableID	Cam No. of output axis (Nozzle) • AdvOutput_Nozzle.Pr.Cam.CamNo
Cam position length per cycle • MCv_AdvPositionPerCycleCalc_Nozzle.LengthPerCycle	Length per cycle of output axis (Nozzle) • AdvOutput_Nozzle.Pr.Cam.LengthPerCycle
Cam position calculation stroke amount • MCv_AdvPositionPerCycleCalc_Nozzle.StrokeAmount	Stroke amount of output axis (Nozzle) • AdvOutput_Nozzle.Pr.Cam.StrokeAmount
Cam position calculation set position • MCv_AdvPositionPerCycleCalc_Nozzle.CamSetPosition	Set position of output axis (Nozzle) • AdvOutput_Nozzle.Md.SetPosition
Cam position calculation cam reference position • MCv_AdvPositionPerCycleCalc_Nozzle.ReferenceSetPosition	Cam reference position of output axis (Nozzle) • AdvOutput_Nozzle.Md.Cam.ReferenceSetPosition
Cam position calculation current position per cycle • MCv_AdvPositionPerCycleCalc_Nozzle.PositionPerCycle	Current position per cycle of output axis (Nozzle) • AdvOutput_Nozzle.Md.Cam.PositionPerCycle

3. Calculate the set position of the output axis (NozzleDrive) based on the current position per cycle calculated in step 2.

Setting item	Setting value
Cam table ID • MCv_AdvCamSetPositionCalc_Drive.CamTableID	Cam No. of output axis (NozzleDrive) • AdvOutput_Drive.Pr.Cam.CamNo
Cam position length per cycle • MCv_AdvCamSetPositionCalc_Drive.LengthPerCycle	Length per cycle of output axis (NozzleDrive) • AdvOutput_Drive.Pr.Cam.LengthPerCycle
Cam position calculation stroke amount • MCv_AdvCamSetPositionCalc_Drive.StrokeAmount	Stroke amount of output axis (NozzleDrive) • AdvOutput_Drive.Pr.Cam.StrokeAmount
Cam position calculation cam reference position • MCv_AdvCamSetPositionCalc_Drive.ReferenceSetPosition	Cam reference position of output axis (NozzleDrive) • AdvOutput_Drive.Md.Cam.ReferenceSetPosition
Cam position calculation current position per cycle • MCv_AdvCamSetPositionCalc_Drive.PositionPerCycle	Current position per cycle calculated in step 2. • MCv_AdvPositionPerCycleCalc.CalculationResult

4. Position the output axis (NozzleDrive) to the set position calculated in step 3., and match the current position per cycle of the output axis (Drive) and output axis (NozzleDrive).

5. Set the synchronous control initial position parameters of the output axes (NozzleDrive and Nozzle) as follows, and start (change Execute from FALSE to TRUE) the advanced synchronous control FB.

Setting item	Setting value
Setting method of current position per cycle after main shaft gear • AdvOutput_NozzleDrive.Pr.Restore.MasterGearPositionPerCycleMethod • AdvOutput_Nozzle.Pr.Restore.MasterGearPositionPerCycleMethod	2: Calculate from input axis (CalculateFromInputAxis)
Cam axis position restoration object • AdvOutput_NozzleDrive.Pr.Restore.PositionRestorationObject • AdvOutput_Nozzle.Pr.Restore.PositionRestorationObject	0: Current position per cycle restoration (RestorePositionPerCycle)
Setting method of cam reference position • AdvOutput_NozzleDrive.Pr.Restore.ReferenceSetPositionMethod • AdvOutput_Nozzle.Pr.Restore.ReferenceSetPositionMethod	0: Previous value (PreviousPosition)
Setting method of current position per cycle • AdvOutput_NozzleDrive.Pr.Restore.PositionPerCycleMethod • AdvOutput_Nozzle.Pr.Restore.PositionPerCycleMethod	1: Current position per cycle (initial setting) (InitialPositionPerCycle)
Current position per cycle (initial setting) • AdvOutput_NozzleDrive.Pr.Restore.InitialPositionPerCycle • AdvOutput_Nozzle.Pr.Restore.InitialPositionPerCycle	Current position per cycle calculated in step 2. • MCv_AdvPositionPerCycleCalc.CalculationResult

• Example program

■Nozzle drive axis

ST program[*1]

```
IF gb_SystemStart AND gb_SyncRestart THEN
  // AdvSync Restart(NozzleDrive)

  // Select For Restore Target
  CASE gw_SelectRestoreMode OF
<Omission>
  RESTORE_POSITION_PER_CYCLE_FOR_CAMCALCFUNC :
    // Set Restore Parameter
    AdvOutput_Drive.Pr.Restore.MasterGearPositionPerCycleMethod:=MC_GEAR_RESTORE_METHOD__CalculateFromInputAxis;
    AdvOutput_Drive.Pr.Restore.PositionRestorationObject:=MC_CAM_RESTORE_METHOD__RestorePositionPerCycle;
    AdvOutput_Drive.Pr.Restore.ReferenceSetPositionMethod:=MC_CAM_REFERENCE_METHOD__PreviousPosition;
    AdvOutput_Drive.Pr.Restore.PositionPerCycleMethod:=MC_CAM_CYCLE_METHOD__InitialPositionPerCycle;
    AdvOutput_Drive.Pr.Restore.InitialPositionPerCycle:=gl_Restored_Nozzle_PositionPerCycle;

    // Positionadjustment
    IF gb_PositionAdjustComp_Nozzle THEN
      IF NOT(gb_PositionAdjustComp_Drive) THEN
        IF AdvInput_Conveyor.Md.ActiveStatus=1 THEN
          g_CamTableID_Drive.ProfileID.Number := AdvOutput_Drive.Pr.Cam.CamNo;
          gl_AdvCamSetPositionCalc_Drive_LengthPercycle := AdvOutput_Drive.Pr.Cam.LengthPerCycle;
          gl_AdvCamSetPositionCalc_Drive_StrokeAmount := AdvOutput_Drive.Pr.Cam.StrokeAmount;
          gl_AdvCamSetPositionCalc_Drive_ReferenceSetPosition := AdvOutput_Drive.Md.Cam.ReferenceSetPosition;
          gl_AdvCamSetPositionCalc_Drive_PositionPerCycle := gl_Restored_Nozzle_PositionPerCycle;
          gb_AdvCamSetPositionCalc_Drive_Execute := TRUE;

          IF MCv_AdvCamSetPositionCalc_Drive.Done THEN
            gl_MoveAbsolute_Drive_Position:=MCv_AdvCamSetPositionCalc_Drive.CalculationResult;
            gb_MoveAbsolute_Drive_Execute:=TRUE;
          END_IF;

          IF MC_MoveAbsolute_Drive.Done THEN
            gb_AdvSync_Drive_Enable:=FALSE;
            gb_MoveAbsolute_Drive_Execute:=FALSE;
            gb_PositionAdjustComp_Drive:=TRUE;
          END_IF;
        END_IF;
      ELSE
        IF AdvInput_Conveyor.Md.ActiveStatus=1 AND
          NozzleDrive.Md.AxisStatus=MC_AXIS_STATUS__Standstill AND
          NOT(MCv_AdvancedSync_Drive.Done) THEN
          gdw_AdvSync_Drive_Options:=H00000000;
          gb_AdvSync_Drive_Enable:=TRUE;
        END_IF;
      END_IF;
    END_IF;
```

*1 For the entire program, refer to the following.

☞ Page 629 Starting advanced synchronous control (initial start)

■Nozzle axis

ST program[*1]

```
IF gb_SystemStart AND gb_SyncRestart THEN
   // AdvSync Restart(Nozzle)

   // Select For Restore Target
   CASE gw_SelectRestoreMode OF
<Omission>
   RESTORE_POSITION_PER_CYCLE_FOR_CAMCALCFUNC :
      // Set Restore Parameter
      AdvOutput_Nozzle.Pr.Restore.MasterGearPositionPerCycleMethod:=MC_GEAR_RESTORE_METHOD__CalculateFromInputAxis;
      AdvOutput_Nozzle.Pr.Restore.PositionRestorationObject:=MC_CAM_RESTORE_METHOD__RestorePositionPerCycle;
      AdvOutput_Nozzle.Pr.Restore.ReferenceSetPositionMethod:=MC_CAM_REFERENCE_METHOD__PreviousPosition;
      AdvOutput_Nozzle.Pr.Restore.PositionPerCycleMethod:=MC_CAM_CYCLE_METHOD__InitialPositionPerCycle;
      AdvOutput_Nozzle.Pr.Restore.InitialPositionPerCycle:=gl_Restored_Nozzle_PositionPerCycle;

      // Positionadjustment
      IF NOT(gb_PositionAdjustComp_Nozzle) THEN
         IF AdvInput_Conveyor.Md.ActiveStatus=1 THEN
            gl_MoveAbsolute_Nozzle_Position:=AdvOutput_Nozzle.Pr.Cam.StrokeAmount * 0.5;
            gb_MoveAbsolute_Nozzle_Execute:=TRUE;
            IF MC_MoveAbsolute_Nozzle.Done THEN
               g_CamTableID_Nozzle.ProfileID.Number := AdvOutput_Nozzle.Pr.Cam.CamNo;
               gl_AdvPositionPerCycleCalc_Nozzle_LengthPercycle := AdvOutput_Nozzle.Pr.Cam.LengthPerCycle;
               gl_AdvPositionPerCycleCalc_Nozzle_StrokeAmount := AdvOutput_Nozzle.Pr.Cam.StrokeAmount;
               gl_AdvPositionPerCycleCalc_Nozzle_ReferenceSetPosition := AdvOutput_Nozzle.Md.Cam.ReferenceSetPosition;
               gl_AdvPositionPerCycleCalc_Nozzle_SetPosition := Nozzle.Md.SetPosition;
               gl_AdvPositionPerCycleCalc_Nozzle_PositionPerCycle := 0.0;
               gb_AdvPositionPerCycleCalc_Nozzle_Execute := TRUE;
            END_IF;
            IF MCv_AdvPositionPerCycleCalc_Nozzle.Done THEN
               gl_Restored_Nozzle_PositionPerCycle := MCv_AdvPositionPerCycleCalc_Nozzle.CalculationResult;
               gb_AdvSync_Nozzle_Enable:=FALSE;
               gb_PositionAdjustComp_Nozzle:=TRUE;
            END_IF;
         END_IF;
      ELSE
         IF AdvInput_Conveyor.Md.ActiveStatus=1 AND
            NozzleDrive.Md.AxisStatus=MC_AXIS_STATUS__Standstill AND
            NOT(MCv_AdvancedSync_Nozzle.Done) THEN
            gdw_AdvSync_Nozzle_Options:=H00000000;
            gb_AdvSync_Nozzle_Enable:=TRUE;
         END_IF;
      END_IF;
```

*1 For the entire program, refer to the following.
☞ Page 629 Starting advanced synchronous control (initial start)

- Labels used

For the labels used in the program, refer to the following.
☞ Page 629 Starting advanced synchronous control (initial start)

■ No.3 "Restart with the phases of input axis cumulative current position and the output axis current position per cycle matched" (when using synchronous control analysis mode)

1. Set the synchronous control initial position parameters of the output axes (NozzleDrive, Nozzle) as follows, and execute the advanced synchronous control FB with the synchronous control analysis mode setting (Options (Options): Bit 16) set to "1: Enabled".
2. After synchronous control analysis mode execution completion (MCv_AdvancedSync.Done is TRUE), use the updated Set position (AdvOutput_Drive.Md.Cam.SetPosition, AdvOutput_Nozzle.Md.Cam.SetPosition) to position the nozzle drive axis (NozzleDrive) and nozzle axis (Nozzle) to the position of the updated set position.
3. Set the synchronous control analysis mode setting (Options (Options): Bit 16) set to "1: Disabled" and start (change Execute from FALSE to TRUE) the advanced synchronous control FB again.

Setting item	Setting value
Setting method of current position per cycle after main shaft gear • AdvOutput_NozzleDrive.Pr.Restore.MasterGearPositionPerCycleMethod • AdvOutput_Nozzle.Pr.Restore.MasterGearPositionPerCycleMethod	2: Calculate from input axis (CalculateFromInputAxis)
Cam axis position restoration object • AdvOutput_NozzleDrive.Pr.Restore.PositionRestorationObject • AdvOutput_Nozzle.Pr.Restore.PositionRestorationObject	2: Cam set position restoration (RestoreCamSetPosition)
Setting method of cam reference position • AdvOutput_NozzleDrive.Pr.Restore.ReferenceSetPositionMethod • AdvOutput_Nozzle.Pr.Restore.ReferenceSetPositionMethod	0: Previous value (PreviousPosition)
Setting method of current position per cycle • AdvOutput_NozzleDrive.Pr.Restore.PositionPerCycleMethod • AdvOutput_Nozzle.Pr.Restore.PositionPerCycleMethod	2: Current position per cycle after main shaft gear (MasterGearPositionPerCycle)

- Example program

■Nozzle drive axis

ST program[*1]

```
IF gb_SystemStart AND gb_SyncRestart THEN
    // AdvSync Restart(NozzleDrive)

    // Select For Restore Target
    CASE gw_SelectRestoreMode OF
<Omission>
    RESTORE_SETPOSITION_FOR_PREPROMODE :
        // Set Restore Parameter
        AdvOutput_Drive.Pr.Restore.MasterGearPositionPerCycleMethod:=MC_GEAR_RESTORE_METHOD__CalculateFromInputAxis;
        AdvOutput_Drive.Pr.Restore.PositionRestorationObject:=MC_CAM_RESTORE_METHOD__RestoreCamSetPosition;
        AdvOutput_Drive.Pr.Restore.ReferenceSetPositionMethod:=MC_CAM_REFERENCE_METHOD__PreviousPosition;
        AdvOutput_Drive.Pr.Restore.PositionPerCycleMethod:=MC_CAM_CYCLE_METHOD__MasterGearPositionPerCycle;

        // Positionadjustment
        IF NOT(gb_PositionAdjustComp_Drive) THEN
            IF AdvInput_Conveyor.Md.ActiveStatus=1 THEN
                gdw_AdvSync_Drive_Options:=H00010000;
                gb_AdvSync_Drive_Enable:=TRUE;

                IF MCv_AdvancedSync_Drive.Done THEN
                    gl_MoveAbsolute_Drive_Position:=AdvOutput_Drive.Md.Cam.SetPosition;
                    gb_MoveAbsolute_Drive_Execute:=TRUE;
                END_IF;

                IF MC_MoveAbsolute_Drive.Done THEN
                    gb_AdvSync_Drive_Enable:=FALSE;
                    gb_MoveAbsolute_Drive_Execute:=FALSE;
                    gb_PositionAdjustComp_Drive:=TRUE;
                END_IF;
            END_IF;
            ELSE
            IF AdvInput_Conveyor.Md.ActiveStatus=1 AND
                NozzleDrive.Md.AxisStatus=MC_AXIS_STATUS__Standstill AND
                NOT(MCv_AdvancedSync_Drive.Done) THEN
                gdw_AdvSync_Drive_Options:=H00000000;
                gb_AdvSync_Drive_Enable:=TRUE;
            END_IF;
        END_IF;
```

*1 For the entire program, refer to the following.

☞ Page 629 Starting advanced synchronous control (initial start)

15

■Nozzle axis

ST program[*1]

```
IF gb_SystemStart AND gb_SyncRestart THEN
  // AdvSync Restart(Nozzle)

  // Select For Restore Target
  CASE gw_SelectRestoreMode OF
<Omission>
  RESTORE_SETPOSITION_FOR_PREPROMODE :
    // Set Restore Parameter
    AdvOutput_Nozzle.Pr.Restore.MasterGearPositionPerCycleMethod:=MC_GEAR_RESTORE_METHOD__CalculateFromInputAxis;
    AdvOutput_Nozzle.Pr.Restore.PositionRestorationObject:=MC_CAM_RESTORE_METHOD__RestoreCamSetPosition;
    AdvOutput_Nozzle.Pr.Restore.ReferenceSetPositionMethod:=MC_CAM_REFERENCE_METHOD__PreviousPosition;
    AdvOutput_Nozzle.Pr.Restore.PositionPerCycleMethod:=MC_CAM_CYCLE_METHOD__MasterGearPositionPerCycle;

    // Positionadjustment
    IF NOT(gb_PositionAdjustComp_Nozzle) THEN
      IF AdvInput_Conveyor.Md.ActiveStatus=1 THEN
        gdw_AdvSync_Nozzle_Options:=H00010000;
        gb_AdvSync_Nozzle_Enable:=TRUE;

        IF MCv_AdvancedSync_Nozzle.Done THEN
          gl_MoveAbsolute_Nozzle_Position:=AdvOutput_Nozzle.Md.Cam.SetPosition;
          gb_MoveAbsolute_Nozzle_Execute:=TRUE;
        END_IF;

        IF MC_MoveAbsolute_Nozzle.Done THEN
          gb_AdvSync_Nozzle_Enable:=FALSE;
          gb_MoveAbsolute_Nozzle_Execute:=FALSE;
          gb_PositionAdjustComp_Nozzle:=TRUE;
        END_IF;
      END_IF;
      ELSE
      IF AdvInput_Conveyor.Md.ActiveStatus=1 AND
        Nozzle.Md.AxisStatus=MC_AXIS_STATUS__Standstill AND
        NOT(MCv_AdvancedSync_Nozzle.Done) THEN
        gdw_AdvSync_Nozzle_Options:=H00000000;
        gb_AdvSync_Nozzle_Enable:=TRUE;
      END_IF;
    END_IF;
```

*1 For the entire program, refer to the following.
☞ Page 629 Starting advanced synchronous control (initial start)

- Labels used

For the labels used in the program, refer to the following.

☞ Page 629 Starting advanced synchronous control (initial start)

■ No.4 "Restart with the phases of input axis cumulative current position and the output axis current position per cycle matched" (when using cam position calculation function (MCv_AdvPositionPerCyclecalc (Advanced Synchronous Control Set Position Calculation)))

1. Use the cumulative current position of the input axis (Conveyor) to calculate the set position with the following settings and calculate set position of the output axis (NozzleDrive and Nozzle).

Setting item	Setting value	
	Output axis (NozzleDrive)	Output axis (Nozzle)
Cam table ID • MCv_AdvCamSetPositionCalc_Drive.CamTableID • MCv_AdvCamSetPositionCalc_Nozzle.CamTableID	Cam No. of output axis (NozzleDrive) • AdvOutput_Drive.Pr.Cam.CamNo	Cam No. of output axis (Nozzle) • AdvOutput_Nozzle.Pr.Cam.CamNo
Cam position length per cycle • MCv_AdvCamSetPositionCalc_Drive.LengthPerCycle • MCv_AdvCamSetPositionCalc_Nozzle.LengthPerCycle	Length per cycle of output axis (NozzleDrive) • AdvOutput_Drive.Pr.Cam.LengthPerCycle	Length per cycle of output axis (Nozzle) • AdvOutput_Nozzle.Pr.Cam.LengthPerCycle
Cam position calculation stroke amount • MCv_AdvCamSetPositionCalc_Drive.StrokeAmount • MCv_AdvCamSetPositionCalc_Nozzle.StrokeAmount	Stroke amount of output axis (NozzleDrive) • AdvOutput_Drive.Pr.Cam.StrokeAmount	Stroke amount of output axis (Nozzle) • AdvOutput_Nozzle.Pr.Cam.StrokeAmount
Cam position calculation cam reference position • MCv_AdvCamSetPositionCalc_Drive.ReferenceSetPosition • MCv_AdvCamSetPositionCalc_Nozzle.ReferenceSetPosition	Cam reference position of output axis (NozzleDrive) • AdvOutput_Drive.Md.Cam.ReferenceSetPosition	Cam reference position of output axis (Nozzle) • AdvOutput_Nozzle.Md.Cam.ReferenceSetPosition
Cam position calculation current position per cycle • MCv_AdvCamSetPositionCalc_Drive.PositionPerCycle • MCv_AdvCamSetPositionCalc_Nozzle.PositionPerCycle	The value of the input axis (Conveyor) cumulative current position rounded within the cam position length per cycle*1	The value of the input axis (Conveyor) cumulative current position rounded within the cam position length per cycle*1

*1 When main shaft/auxiliary shaft gear or main shaft/auxiliary shaft speed change gear are set, the gear ratio and speed change gear ratio for the movement amount from synchronization stop until restart need to be taken into account of in the calculation.
When setting gears and speed change gears, using synchronous control analysis mode is recommended.

2. Position to the position of the set position (MCv_AdvCamSetPositionCalc.CalculationResult) calculated with the nozzle drive axis (NozzleDrive) and nozzle axis (Nozzle) in step 1.

3. Set the synchronous control initial position parameters of the output axes (NozzleDrive and Nozzle) as follows, and start (change Execute from FALSE to TRUE) the advanced synchronous control FB.

Setting item	Setting value
Setting method of current position per cycle after main shaft gear • AdvOutput_NozzleDrive.Pr.Restore.MasterGearPositionPerCycleMethod • AdvOutput_Nozzle.Pr.Restore.MasterGearPositionPerCycleMethod	2: Calculate from input axis (CalculateFromInputAxis)
Cam axis position restoration object • AdvOutput_NozzleDrive.Pr.Restore.PositionRestorationObject • AdvOutput_Nozzle.Pr.Restore.PositionRestorationObject	2: Cam set position restoration (RestoreCamSetPosition)
Setting method of cam reference position • AdvOutput_NozzleDrive.Pr.Restore.ReferenceSetPositionMethod • AdvOutput_Nozzle.Pr.Restore.ReferenceSetPositionMethod	0: Previous value (PreviousPosition)
Setting method of current position per cycle • AdvOutput_NozzleDrive.Pr.Restore.PositionPerCycleMethod • AdvOutput_Nozzle.Pr.Restore.PositionPerCycleMethod	2: Current position per cycle after main shaft gear (MasterGearPositionPerCycle)

• Example program

■Nozzle drive axis

ST program*1

```
IF gb_SystemStart AND gb_SyncRestart THEN
    // AdvSync Restart(NozzleDrive)

    // Select For Restore Target
    CASE gw_SelectRestoreMode OF
<Omission>
    RESTORE_SETPOSITION_FOR_CAMCALCFUNC :
        // Set Restore Parameter
        AdvOutput_Drive.Pr.Restore.MasterGearPositionPerCycleMethod:=MC_GEAR_RESTORE_METHOD__CalculateFromInputAxis;
        AdvOutput_Drive.Pr.Restore.PositionRestorationObject:=MC_CAM_RESTORE_METHOD__RestoreCamSetPosition;
        AdvOutput_Drive.Pr.Restore.ReferenceSetPositionMethod:=MC_CAM_REFERENCE_METHOD__PreviousPosition;
        AdvOutput_Drive.Pr.Restore.PositionPerCycleMethod:=MC_CAM_CYCLE_METHOD__MasterGearPositionPerCycle;

        // Positionadjustment
        IF NOT(gb_PositionAdjustComp_Drive) THEN
            IF AdvInput_Conveyor.Md.ActiveStatus=1 THEN
                g_CamTableID_Drive.ProfileID.Number := AdvOutput_Drive.Pr.Cam.CamNo;
                gl_AdvCamSetPositionCalc_Drive_LengthPercycle := AdvOutput_Drive.Pr.Cam.LengthPerCycle;
                gl_AdvCamSetPositionCalc_Drive_StrokeAmount := AdvOutput_Drive.Pr.Cam.StrokeAmount;
                gl_AdvCamSetPositionCalc_Drive_ReferenceSetPosition := AdvOutput_Drive.Md.Cam.ReferenceSetPosition;

                //Rounding PositionPerCycle
                gdw_AdvInput_Conveyor_CumulativePosition := LREAL_TO_DINT(AdvInput_Conveyor.Md.CumulativePosition);
                gdw_AdvOutput_Drive_LengthPerCycle := LREAL_TO_DINT(AdvOutput_Drive.Pr.Cam.LengthPerCycle);
                IF AdvInput_Conveyor.Md.CumulativePosition < 0.0 THEN
                    gl_AdvCamSetPositionCalc_Drive_PositionPerCycle := AdvInput_Conveyor.Md.CumulativePosition -
                                                (DINT_TO_LREAL(DIV((gdw_AdvInput_Conveyor_CumulativePosition + 1) ,
                                                gdw_AdvOutput_Drive_LengthPerCycle) - 1) * AdvOutput_Drive.Pr.Cam.LengthPerCycle);
                    ELSE
                    gl_AdvCamSetPositionCalc_Drive_PositionPerCycle := AdvInput_Conveyor.Md.CumulativePosition -
                                                (DINT_TO_LREAL(DIV(gdw_AdvInput_Conveyor_CumulativePosition ,
                                                gdw_AdvOutput_Drive_LengthPerCycle)) * AdvOutput_Drive.Pr.Cam.LengthPerCycle);
                END_IF;
                gb_AdvCamSetPositionCalc_Drive_Execute := TRUE;

                IF MCv_AdvCamSetPositionCalc_Drive.Done THEN
                    gl_MoveAbsolute_Drive_Position:=MCv_AdvCamSetPositionCalc_Drive.CalculationResult;
                    gb_MoveAbsolute_Drive_Execute:=TRUE;
                END_IF;

                IF MC_MoveAbsolute_Drive.Done THEN
                    gb_AdvSync_Drive_Enable:=FALSE;
                    gb_MoveAbsolute_Drive_Execute:=FALSE;
                    gb_PositionAdjustComp_Drive:=TRUE;
                END_IF;
            END_IF;
            ELSE
            IF AdvInput_Conveyor.Md.ActiveStatus=1 AND
                NozzleDrive.Md.AxisStatus=MC_AXIS_STATUS__Standstill AND
                NOT(MCv_AdvancedSync_Drive.Done) THEN
                gdw_AdvSync_Drive_Options:=H00000000;
                gb_AdvSync_Drive_Enable:=TRUE;
            END_IF;
        END_IF;

    ELSE
        ;
    END_CASE;
END_IF;
```

*1 For the entire program, refer to the following.

☞ Page 629 Starting advanced synchronous control (initial start)

■Nozzle axis

ST program[*1]

```
IF gb_SystemStart AND gb_SyncRestart THEN
    // AdvSync Restart(Nozzle)

    // Select For Restore Target
    CASE gw_SelectRestoreMode OF
<Omission>
    RESTORE_SETPOSITION_FOR_CAMCALCFUNC :
        // Set Restore Parameter
        AdvOutput_Nozzle.Pr.Restore.MasterGearPositionPerCycleMethod:=MC_GEAR_RESTORE_METHOD__CalculateFromInputAxis;
        AdvOutput_Nozzle.Pr.Restore.PositionRestorationObject:=MC_CAM_RESTORE_METHOD__RestoreCamSetPosition;
        AdvOutput_Nozzle.Pr.Restore.ReferenceSetPositionMethod:=MC_CAM_REFERENCE_METHOD__PreviousPosition;
        AdvOutput_Nozzle.Pr.Restore.PositionPerCycleMethod:=MC_CAM_CYCLE_METHOD__MasterGearPositionPerCycle;

        // Positionadjustment
        IF NOT(gb_PositionAdjustComp_Nozzle) THEN
            IF AdvInput_Conveyor.Md.ActiveStatus=1 THEN
                g_CamTableID_Nozzle.ProfileID.Number := AdvOutput_Nozzle.Pr.Cam.CamNo;
                gb_AdvCamSetPositionCalc_Nozzle_Execute := TRUE;
                gl_AdvCamSetPositionCalc_Nozzle_LengthPercycle := AdvOutput_Nozzle.Pr.Cam.LengthPerCycle;
                gl_AdvCamSetPositionCalc_Nozzle_StrokeAmount := AdvOutput_Nozzle.Pr.Cam.StrokeAmount;
                gl_AdvCamSetPositionCalc_Nozzle_ReferenceSetPosition := AdvOutput_Nozzle.Md.Cam.ReferenceSetPosition;

                //rounding
                gdw_AdvInput_Conveyor_CumulativePosition := LREAL_TO_DINT(AdvInput_Conveyor.Md.CumulativePosition);
                gdw_AdvOutput_Nozzle_LengthPerCycle := LREAL_TO_DINT(AdvOutput_Nozzle.Pr.Cam.LengthPerCycle);
                IF AdvInput_Conveyor.Md.CumulativePosition < 0.0 THEN
                    gl_AdvCamSetPositionCalc_Nozzle_PositionPerCycle := AdvInput_Conveyor.Md.CumulativePosition -
                                                                (DINT_TO_LREAL(DIV((gdw_AdvInput_Conveyor_CumulativePosition + 1) ,
                                                                gdw_AdvOutput_Nozzle_LengthPerCycle) - 1) * AdvOutput_Nozzle.Pr.Cam.LengthPerCycle);
                    ELSE
                    gl_AdvCamSetPositionCalc_Nozzle_PositionPerCycle := AdvInput_Conveyor.Md.CumulativePosition -
                                                                (DINT_TO_LREAL(DIV(gdw_AdvInput_Conveyor_CumulativePosition ,
                                                                 gdw_AdvOutput_Nozzle_LengthPerCycle)) * AdvOutput_Nozzle.Pr.Cam.LengthPerCycle);
                END_IF;
                IF MCv_AdvCamSetPositionCalc_Nozzle.Done THEN
                    gl_MoveAbsolute_Nozzle_Position:=MCv_AdvCamSetPositionCalc_Nozzle.CalculationResult;
                    gb_MoveAbsolute_Nozzle_Execute:=TRUE;
                END_IF;

                IF MC_MoveAbsolute_Nozzle.Done THEN
                    gb_AdvSync_Nozzle_Enable:=FALSE;
                    gb_MoveAbsolute_Nozzle_Execute:=FALSE;
                    gb_PositionAdjustComp_Nozzle:=TRUE;
                END_IF;
            END_IF;
            ELSE
            IF AdvInput_Conveyor.Md.ActiveStatus=1 AND
                Nozzle.Md.AxisStatus=MC_AXIS_STATUS__Standstill AND
                NOT(MCv_AdvancedSync_Nozzle.Done) THEN
                gdw_AdvSync_Nozzle_Options:=H00000000;
                gb_AdvSync_Nozzle_Enable:=TRUE;
            END_IF;
        END_IF;

    ELSE
        ;
    END_CASE;
END_IF;
```

*1 For the entire program, refer to the following

☞ Page 629 Starting advanced synchronous control (initial start)

- Labels used

For the labels used in the program, refer to the following.

☞ Page 629 Starting advanced synchronous control (initial start)

■ Starting the conveyor axis (JOG operation)

Confirm that the output axes are synchronized (AdvOutput_Drive.Md.SyncStatus is "1: In synchronization" and AdvOutput_Nozzle.Md.SyncStatus is "1: In synchronization") and start the conveyor axis (Conveyor).

- Example program

ST program

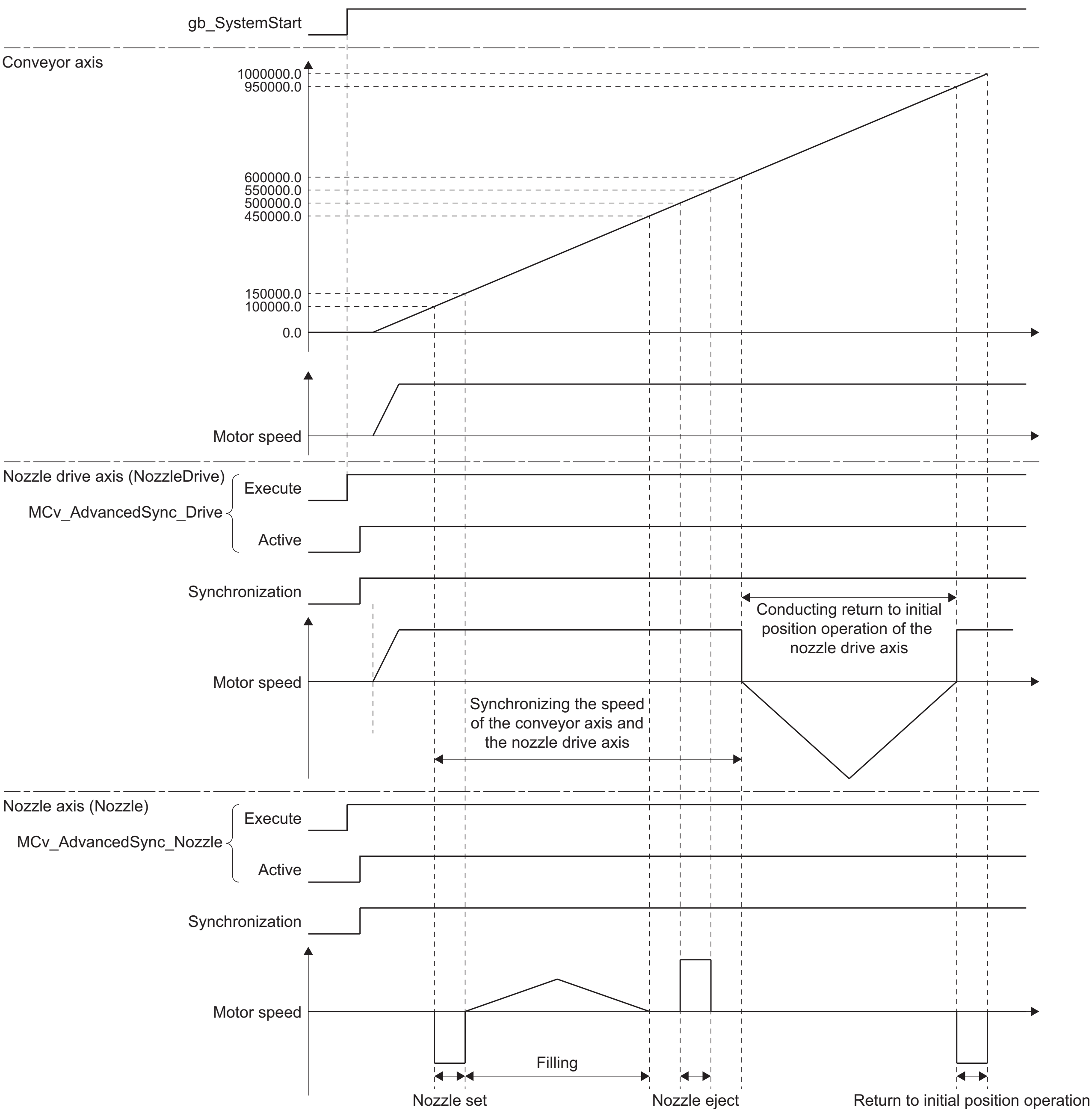
```
IF gb_SystemStart THEN
   // Conveyor Move
   IF AdvOutput_Drive.Md.SyncStatus=1 AND AdvOutput_Nozzle.Md.SyncStatus=1 THEN
      gl_SetSpeedComveyor:=10000000.0;
      gb_MoveConveyorReq:=TRUE;
   END_IF;
END_IF;

MCv_Jog_Conveyor(
   Axis:= Conveyor.AxisRef,
   JogForward:= gb_MoveConveyorReq,
   Velocity:= gl_SetSpeedComveyor,
   Acceleration:= 0.5,
   Deceleration:= 0.5,
   Options:= H0001);
```

- Labels used

Label name	Data type	Comment
gb_SystemStart	Bit	System start command
MCv_Jog_Conveyor	MCv_Jog	JOG operation FB (conveyor axis)
gb_MoveConveyorReq	Bit	Positive direction JOG operation command
gl_SetSpeedComveyor	Double-precision real number	Command speed of JOG operation

Timing chart

The timing chart for the operation that uses the program example is shown below.



MEMO

PART 3 MOTION CONTROL

This part consists of the following chapters.

16 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.

16.1 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

System parameter constant (System.PrConst.)

Variable/Structure name	Name	Details
Logging_AutoLoggingEnable	Auto Logging Enabled	Sets "Allow" or "Do not allow" of auto logging function when the logging setting file is stored in the SD memory card. • FALSE: Disabled • TRUE: Enabled

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

The elements [1] to [10] of the "System.LoggingRef[1..10].Md." array correspond to logging settings 1 to 10.

Variable/Structure name	Name	Details
SettingPath	Logging Setting File	Displays the path (file name) of the logging setting file during the logging.
LogDataPath	Logging Data File Path	Displays the path (file name) of the logging data file.
LoggingStatus	Logging Status	Indicates the logging execution status. • FALSE: Stop • TRUE: Executing
Triggered	Triggered	Displays the triggered status. • FALSE: Not triggered • TRUE: Triggered
TriggerCount	Trigger Detection Count	Displays the trigger detection count.
TriggerCountIgnored	Trigger Ignored Count	A trigger occurs again during collecting data after trigger and it indicates the trigger ignored count.
LogDataSavedCount	Number of Saved Files	Displays the number of saved data logging files.
TriggerStatus[1..32]	Trigger Status	Displays the status of each trigger condition as a bit (b0 to b31). Indicates the status of each trigger condition. • FALSE: Trigger condition not established • TRUE: Trigger condition established
TriggerStatusword	Trigger Status Word	Reflects the same status as Trigger status (TriggerStatus[1..32]) variable to a bit (b0 to b31).
SaveStatus	Save Status	Displays the save status to files of collected data. • 0: Unsaved (UnSave) • 1: Saving (Saving) • 2: Saved (Saved) • 3: All Files Saved (FullySaved)
LogType	Logging Type	Displays the logging type in progress. • 0: None (_None) • 1: Continuous logging (ContinuousLogging) • 2: Trigger logging (TriggerLogging)
ErrStatus	Error Status	Stores the error code related to logging.
SamplingTime[0..1]	Data Collection Interval	Displays the data collection interval in a unit of ns. • [0] Sampling interval lower 32 bits • [1] Sampling interval upper 32 bits

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

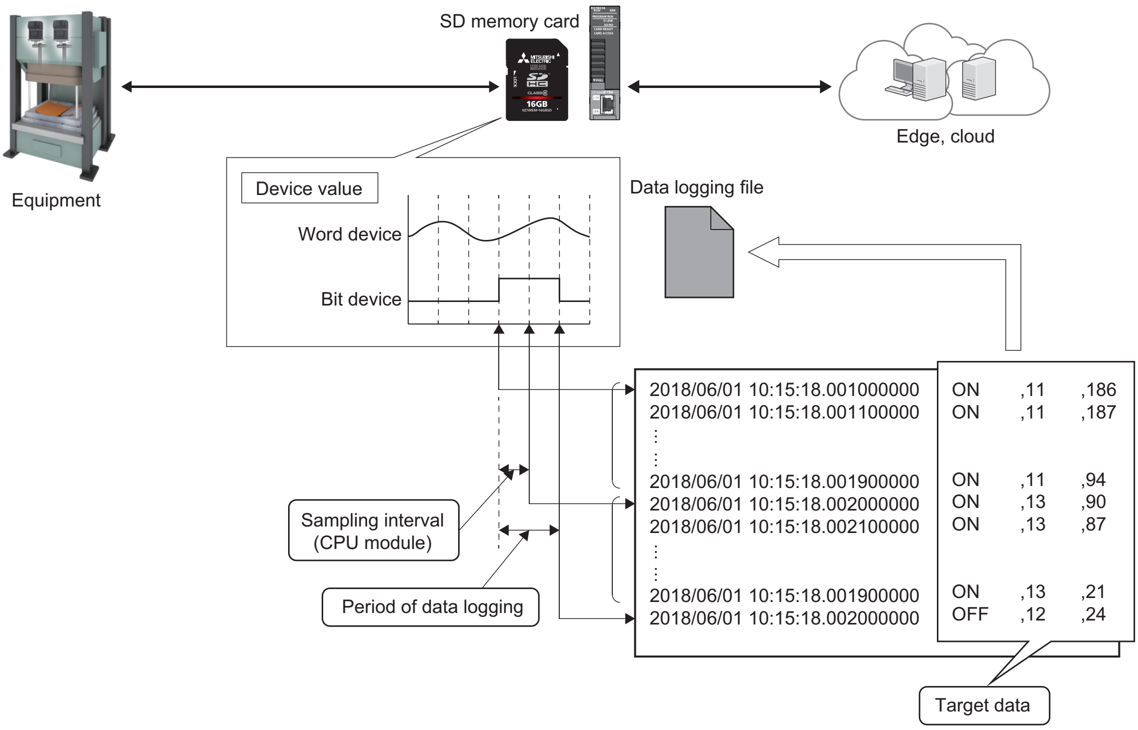
The elements [1] to [10] of the "System.LoggingRef[1..10].Cd." array correspond to logging settings 1 to 10.

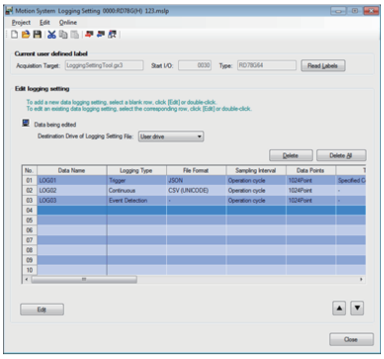
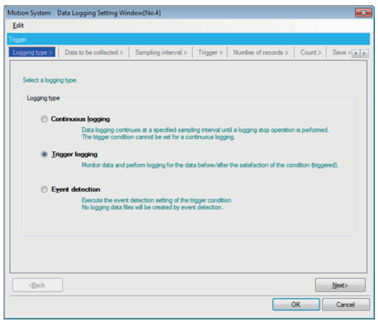
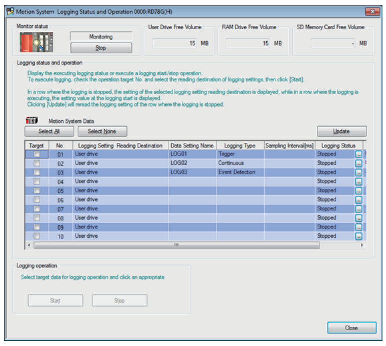
Variable/Structure name	Name	Details
SettingPath	Logging Setting Path	Sets the storage location path (folder name/file name) of logging setting.
StartLog	Logging Request	Requests the execution of logging. • FALSE: STOP • TRUE: RUN
Trigger	Trigger Request	Requests the trigger when "Trigger command" is set for a trigger condition. • FALSE: No request • TRUE: Request

16

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.

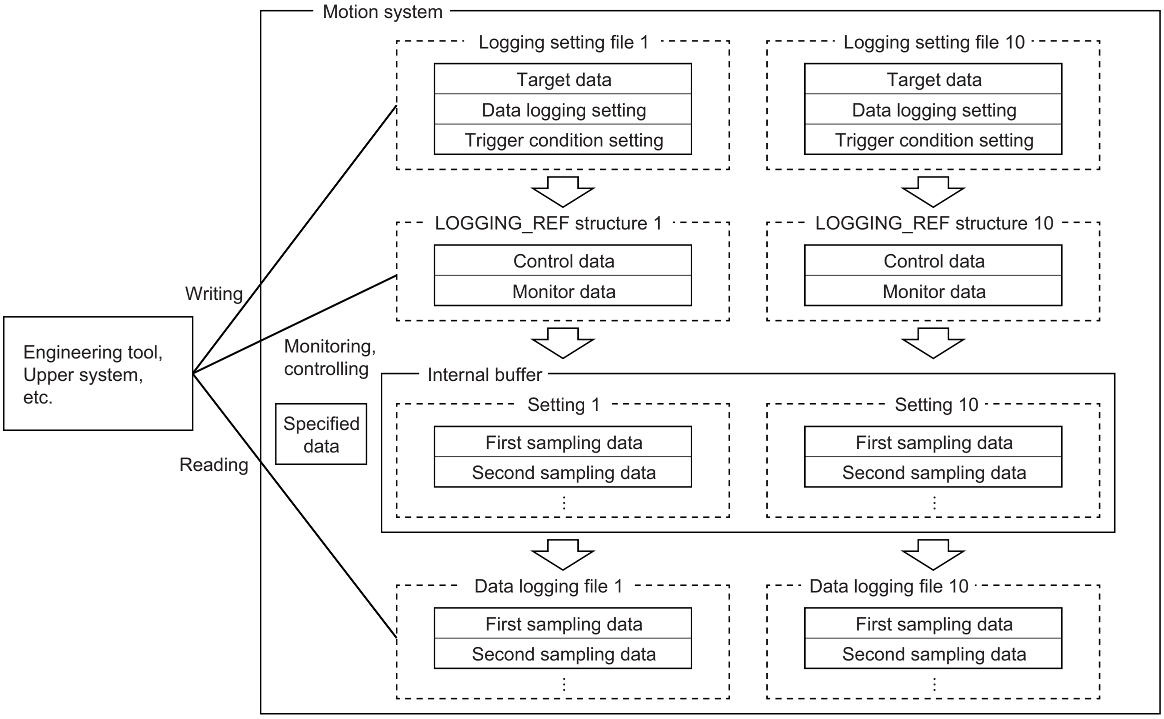
Steps until the collected data is saved

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.



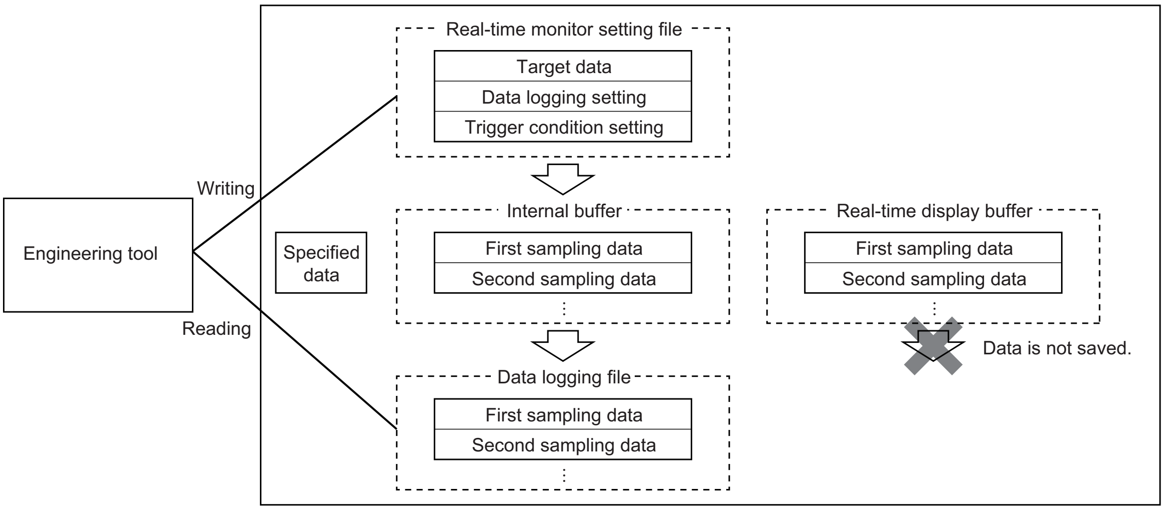
16

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.

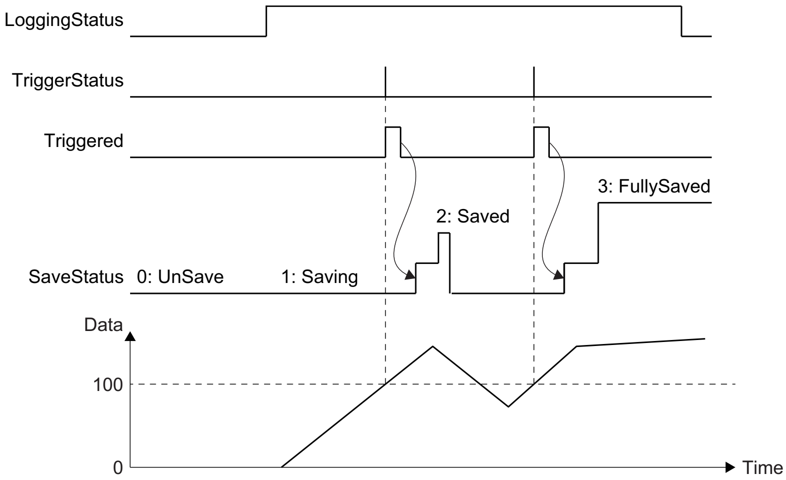


Status of logging

The status of logging in progress can be checked by the following.

- Logging Status (System.LoggingRef[1..10].Md.LoggingStatus)
- Trigger Status (System.LoggingRef[1..10].Md.TriggerStatus[1..32])
- Triggered (System.LoggingRef[1..10].Md.Triggered
- Save Status (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 becomes TRUE while trigger condition is processed for Trigger Status (System.LoggingRef[1..10].Md.TriggerStatus[1..32]), and after trigger condition is processed for Triggered (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 Save Status (System.LoggingRef[1..10].Md.SaveStatus) as follows.

Save Status (System.LoggingRef[1..10].Md.SaveStatus)	Description
3: All Files Saved (FullySaved)	• When the all records specified the number in "Number of file switching records" by continuous logging are saved to files • When the all records specified the number in "Number of trigger records after trigger" by trigger logging are saved to files and the only files specified the number in "Number of trigger logging"
2: Saved (Saved)	When the files are saved with the condition which is not met the condition of "3: All Files Saved (FullySaved)"

However, after the status becomes "2: Saved (Saved)" or "3: All Files Saved (FullySaved)", the status moves to be "0: Unsaved (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 / 1024

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. <Example> When the setting order is four-word data, one-word data, four word data. • The number of appropriative words is "12". (The number of 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 is Number of file switching records (SAVSWICTMNRECNUM) ■In trigger logging R is 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 as follows • Within 1 [s]: B is 4 • Within 1 [s] to 5 [s]: B is 3 • Other than the above: B is 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.

Point

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 385 TARGET_REF structure

To collect a mapped object to cyclic data, assign the object to the device label and set it. For how to assign the device label, refer to the following.

☞ Page 831 Connectable device to CC-Link IE TSN

■ 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.
When the logging is stopped, the data in buffer at the time is output to the file.

Logging start conditions	Description
Starts automatically	When the motion system starts[*1], Logging Request (System.LoggingRef[1..10].Cd.StartLog) becomes TRUE and logging starts automatically. Logging can start and stop after the system starts by Logging Request (System.LoggingRef[1..10].Cd.StartLog).
Stop disabled after automatically start	When the motion system starts[*1], Logging Request (System.LoggingRef[1..10].Cd.StartLog) becomes TRUE and logging starts automatically. Logging cannot stop manually (Logging Request (System.LoggingRef[1..10].Cd.StartLog) changes to FALSE).
Starts by user operation	Logging can start and stop by Logging Request (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 664 Auto logging

16

■ Collection interval (SMPHSPDTYPE, SMPHSPDTIME)

Set the collection timing of the target data.

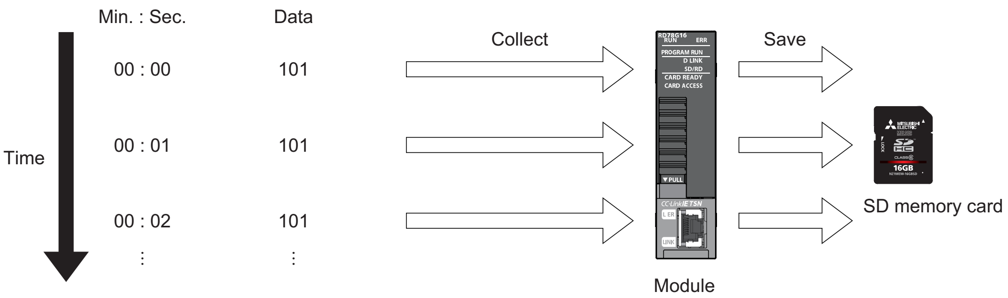
Data collection timing	Description
Operation cycle	Collects data in operation cycle interval. From the first operation cycles are available.
Time specification	Collects data at specified time interval ([ms]). (from 1 to 86400 [ms]) The closest cycle among cycles obtained by multiplying the first operation cycles by an integer becomes the actual collection interval. (When there are multiple options of the cycle, the longer cycle is obtained) <Example> When the first operation is 4.0 [ms] • When specifying 6.0 [ms], the actual collection interval is 8.0 [ms] (the first operation cycle × 2)

■ Logging type (TFFLGTP)

Set the logging type to be executed. The continuous logging and trigger logging methods of data collection are available.

- Continuous logging

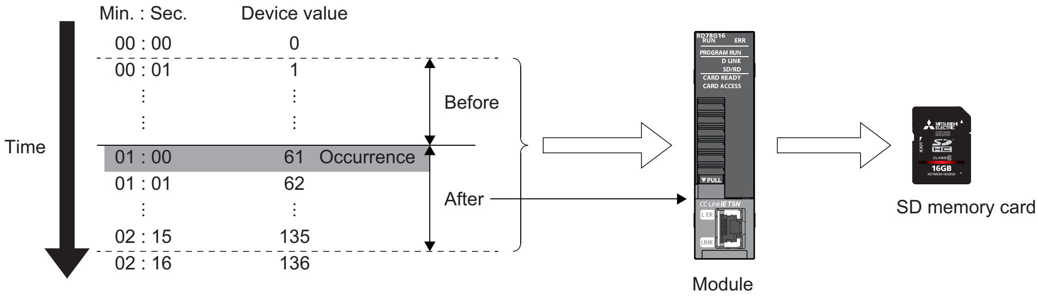
  In continuous logging, the module collects specified data continuously and keeps saving the data logging file at file switching timing.



<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 Logging Request (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 Triggered (System.LoggingRef[1..10].Md.Triggered) and Trigger Status (System.LoggingRef[1..10].Md.TriggerStatus[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. Save Status (System.LoggingRef[1..10].Md.SaveStatus) changes to "2: Saved (Saved)" and "3: All Files Saved (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 Logging Request (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
Number of total records	Number of total records of number of records before and after trigger. Set the number of records before trigger/the number of records after trigger not to exceed the setting range.	1 to 1000000

Point

- When setting from the engineering tool, the setting range is as follows.
  - Number of records before trigger: 0 to 299999
  - Number of records after trigger: 1 to 300000
  - Number of total records: 1 to 300000
- 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 665 Precautions

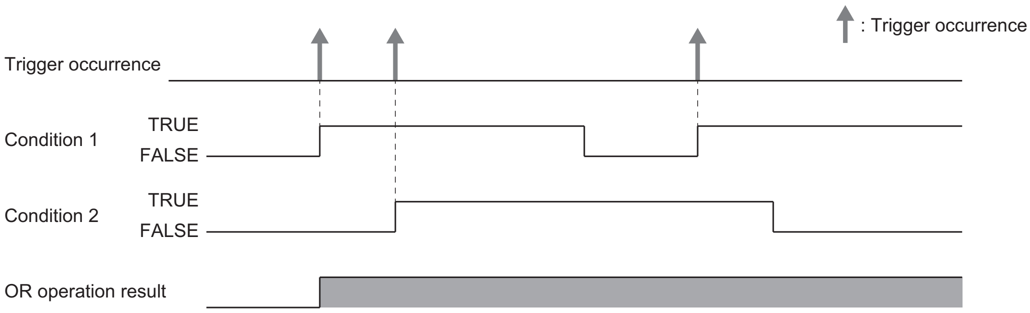
■ Trigger type (TRGCOMPTYPE)

Set the combination of multiple trigger conditions. One trigger type can be set for one setting.

<table>
<tr><th>Trigger type</th><th>Number of trigger conditions can be set</th><th>Trigger conditions can be specified</th></tr>
<tr><td>OR Combination</td><td>Up to 32</td><td rowspan="2">■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) (Condition is satisfied only at the changed timing)
• Fixed cycle (Condition is satisfied only at the fixed cycle timing)
■The following conditions are regarded as an AND/OR condition is satisfied after a condition is satisfied once.
• System start
• At trigger command execution
• Event history</td></tr>
<tr><td>AND Combination</td><td>Up to 32</td></tr>
</table>

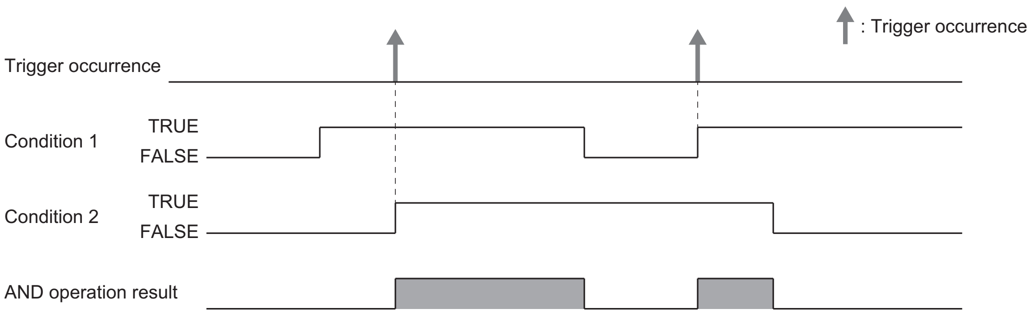
- OR Combination

  When any of the set conditions is satisfied, the trigger occurs.



- AND Combination

  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 specifying the location in this setting and executing multiple logging settings simultaneously.

■ File format (TFFFILE)

"CSV (Unicode)" and "JSON" 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 668 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 Number of Saved Files (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, Number of Saved Files (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 Saved Files (System.LoggingRef[1..10].Md.LogDataSavedCount).

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.

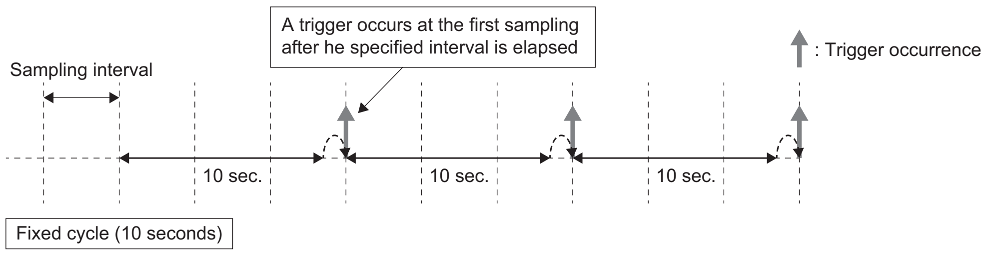
When the creation interval of the data logging files is short, and the error BUSY occurs, increase this setting value. When the setting value is "0", the number of buffers is determined automatically according to the file creation interval. For details, refer to the following.

☞ Page 656 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)

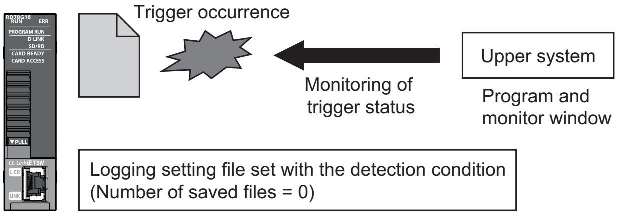
Condition		Description
Data condition	Comparison	Data is compared with data or constant and a trigger occurs when a condition is satisfied. (=, ≠, ≥, >, ≤, <) • =: When the current value of monitored data is equal to the comparison value • ≠: When the current value of monitored data is not equal to the 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	A trigger occurs when a value is changed. (↑, ↓, at change) • ↑: When the monitored data becomes FALSE to TRUE • ↓: When the monitored data becomes TRUE to FALSE • At change: When the current value of the monitored data changes
Fixed cycle		A trigger occurs in the specified interval (specified cycle [ms]) 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 that can be specified: 1 to 86400000 [ms] <Example> 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 Trigger Request (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). <Example> When setting event codes from 0x80001000 to 0x80001010 as a trigger condition • 0x80001000-0x80001010 <Example> When setting all event codes as a trigger condition • 0x00000000-0xFFFFFFFF

Application function

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.



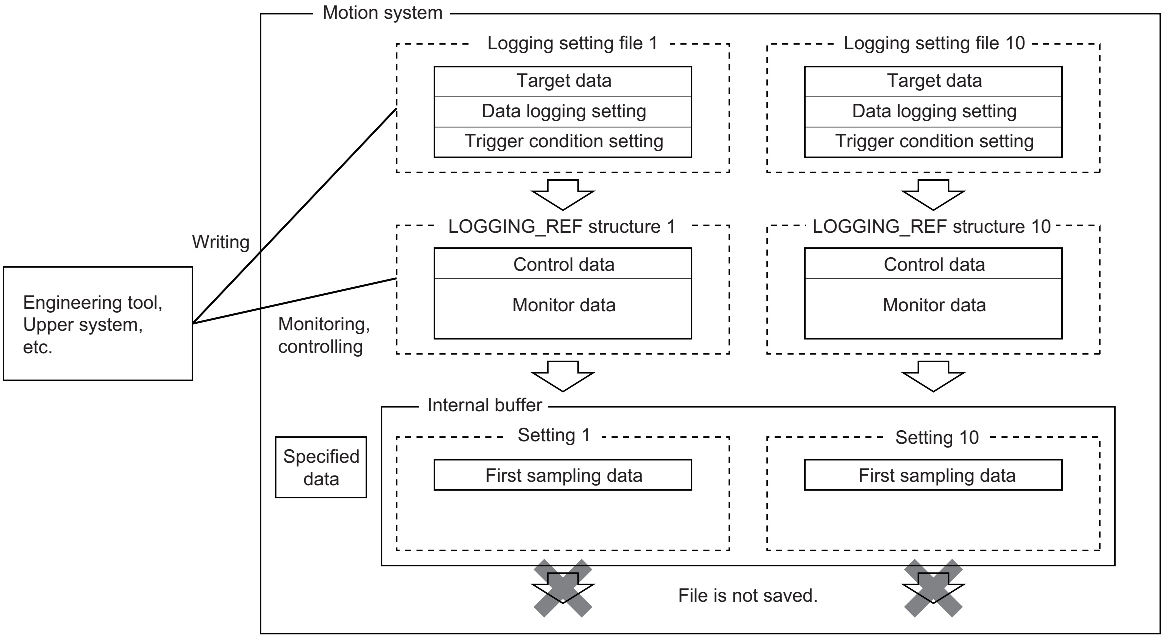
■ Procedures of event detection

1. Set as follows in the logging setting and start logging.

Item	Setting details
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, Trigger Request (System.LoggingRef[1..10].Md.TriggerCount) turns 1. This signal is used to control. (From the upper, used for monitoring, program execution condition, etc.)

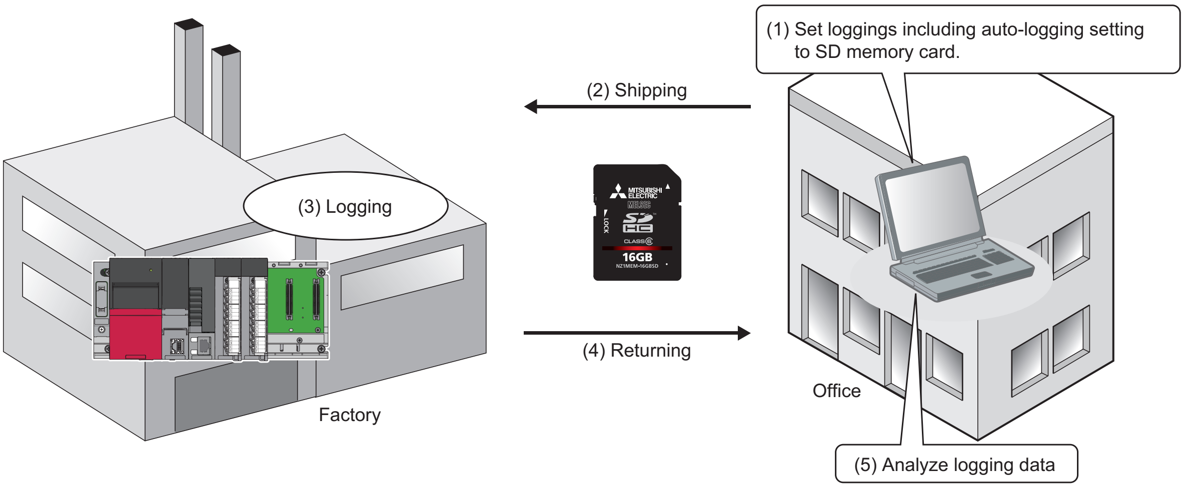
3. To detect an event continuously, set Logging Request (System.LoggingRef[1..10].Cd.StartLog) to FALSE once after a trigger is detected and set to TRUE again.



16

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 Auto Logging Enabled (System.PrConst.Logging_AutoLoggingEnable) to "1: Enabled" 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. Logging Setting Path (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, Logging Request (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.

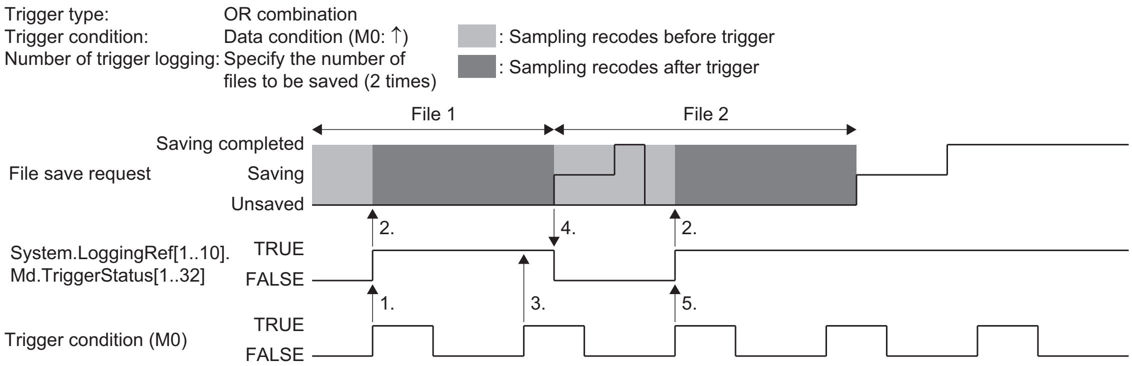
☞ Page 689 Servo System Recorder

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. (** supports variables included in System.LoggingRef[1..10].Cd)
- 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 of operation is shown below.

1. Trigger Status (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, Trigger Status (System.LoggingRef[1..10].Md.TriggerStatus[1..32]) becomes FALSE. (Preparation for the next logging operation)
5. Trigger Status (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.
  The example of operation is shown below.

Ex.

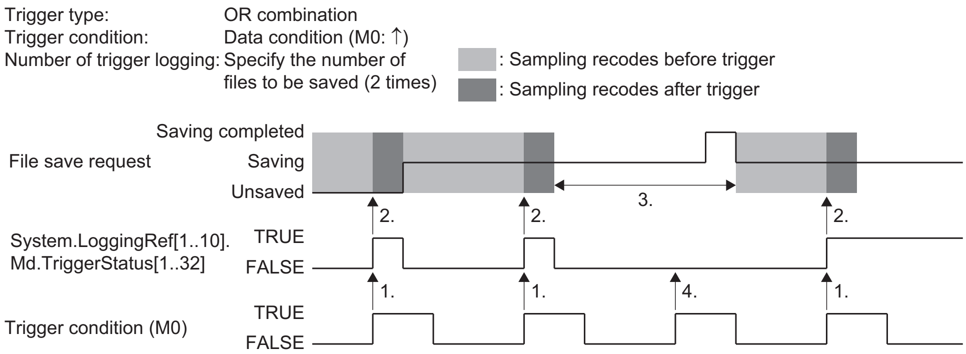
When "Number of buffers (NUMBUFF) = 2".

1. Trigger Status (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 720 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 828 List of Logging Error Codes

Relevant add-ons

The following add-on is required to use this function.

- Logging
- SignalIO
- MotionEventHist[*1]

*1 For using "Event history" as the trigger conditions

System memory capacity

■ RAM Usage

Refer to the following.

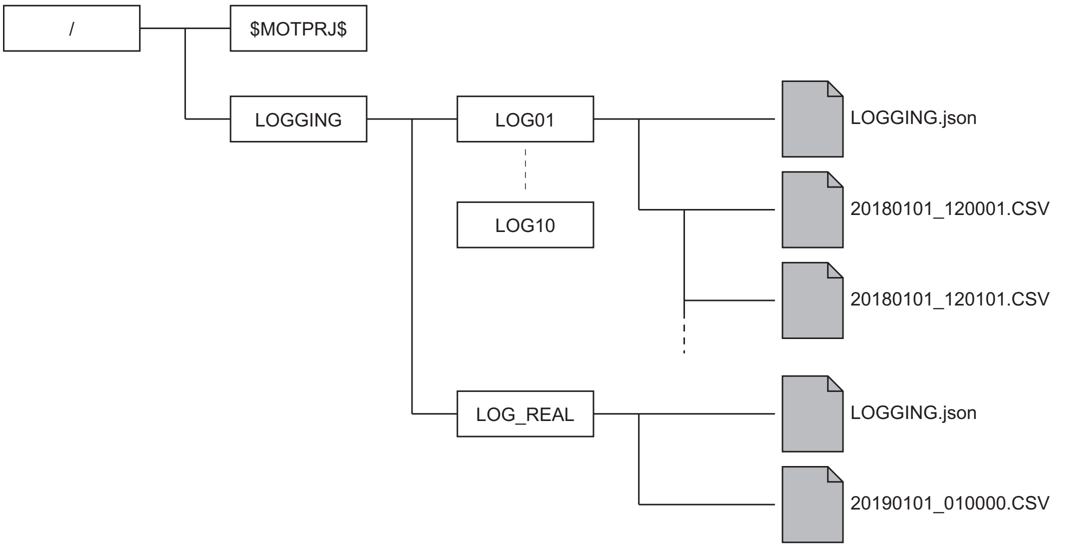
☞ Page 656 Internal buffer

■ Backup RAM Usage

Backup RAM is not used.

Folder configuration

The default folder configuration of logging is shown below.



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]

16.2 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 657 Logging setting
☞ Page 670 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.sssssss]"*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 .sssssss 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.

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]".*1
Double word [unsigned]	ULONG	Outputs in the format of "ULONG[DEC.0]".*1
Word [signed]	SHORT	Outputs in the format of "SHORT[DEC.0]".*1
Double word [signed]	LONG	Outputs in the format of "LONG[DEC.0]".*1
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 [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.
*2 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.

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. The numerical values starting from 1 and increased in ascending order are output. When the value exceeds the upper limit value 4294967295, it returns to 0, and the value increases within the range of 0 to 4294967295 again. The values are taken over from file to file when performing continuous logging. (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 689 Servo System Recorder
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 FALSE: Not save TRUE: Automatically decelerating
Number of trigger logging[*2]	TRIGGERTIME	number	0: Number of save files specification 1 to 32767: Number of times specification
Number of buffer	NUMBUFF	number	0: Automatic setting 2 to 10: Specified number of buffer

*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).

16.3 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]

```
"FileType" : "LOGGING",                                  <<object>>
"Ver" : "1.00",                                          Common head item
"Creator" : "RD78G(H)",
"Comment" : "Comment",
```

```
"Record" :                                               <<object>>
{                                                        Sampling
 "MotionSettingInfo"                                     Data part
 {
          "Logging" : {...}
          "Trigger" : [...]
          "Data" : [...]
 }

"Datainfo" :
[
         {"Type" : "Time","TimeType" : "ABS_TIME","DataFormat" : "YYYY/MM/DD hh:mm:ss.ssssss","Order": "0"},
         {"Type" : "TRIGGER","Mark" : "*","Order" : "1"},
         {"Type" : "DATA","DataType" : "INT[DEC.0]","DataName" : "Velocity","Unit": "r/min","Order" : "2"}
         {"Type" : "DATA","DataType" : "LREAL[DEC.9]","DataName" : "Torque","Order" : "3"}
         {"Type" : "DATA","DataType" : "BIT[1;0]","DataName" : "Start accept","Order" : "4"}
 ]

"Data" : [
        [2000/01/08 04:12:35.480000,0,0,2,0,...],
        [2000/01/08 04:12:35.481000,0,2,5,0,...],
        [2000/01/08 04:12:35.482000,1,44.2,0,...],
        [2000/01/08 04:12:35.483000,0,73.9,0,...],
        [2000/01/08 04:12:35.485000,625,108.1,0,...] ]
}
```

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 Waveform data (Record)

Item	The maximum number
Number of data points (Data column information (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 674 Waveform data (Record)	Waveform data
Status bar color	Possible	StatusBarColor	Color object	0x ➊➊➋➋➌➌ ➊➊: Red, ➋➋: Green, ➌➌: 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 676 Device dependent information	Refer to the specification document of each device.
Data column information	Not possible	DataInfo	☞ Page 675 Data column information (DataInfo)	Array of data column information
Data	Not possible	Data	☞ Page 676 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)

■ Common

Item name	Omission	Key name	Data type	Description
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

Item name	Omission	Key name	Data type	Description
Time information type	Not possible	TimeType	Character string	ABS_TIME: Absolute value

- TimeType is ABS_TIME

Item name	Omission	Key name	Data type	Description
Time format	Not possible	DateFormat	Character string	Only when TimeType is absolute value <Example> YYYY/MM/DD hh:mm:ss.s

■ When the type is triggered flag information

Item name	Omission	Key name	Data type	Description
Trigger signal	Not possible	Mark	Character	<Example> "*"

■ When the type is index information (Used for determination of missing data)

Item name	Omission	Key name	Data type	Description
—	—	—	—	—

■ When the type is sampling data

Item name	Omission	Key name	Data type	Description
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	Depends 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 (DataInfo)

■ The type of Data column information (DataInfo) is "DATA"

The range for each data type is as follows.

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 "NaN" (outputs in string type), "Inf" (outputs in string type), "-Inf" (outputs in string type)
	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 "NaN" (outputs in string type), "Inf" (outputs in string type), "-Inf" (outputs in string type)
	LREAL[EXP.(number of digits after the decimal point)]	
Character string	STRING	—
	WSTRING	—

■ The type of Data column information (DataInfo) is "INDEX"

The numerical values counted up from 1 in ascending order are output.

When the count exceeds 4294967295 which is the upper limit, it returns to 0 and starts a new count within the range from 0 to 4294967295.

If missing occurs in correcting data, index will be reassigned from 1 again.

When this value is continuing in 2 files, the graphs are drawn without missing at the joint when performing "Show Next/Previous Graph" (combining graphs).

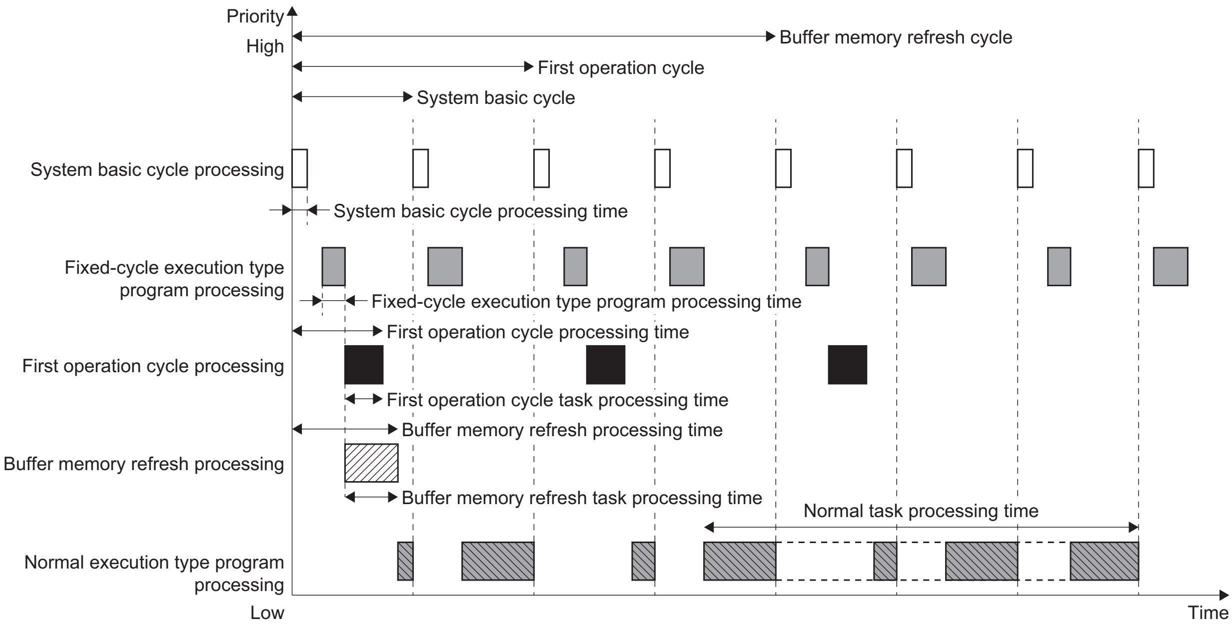
Device dependent information

For the motion system, the key name is MotionSettingInfo.

17 RAS FUNCTIONS

17.1 Execution Time Monitor

This function is used to monitor the operation cycle processing and the normal task execution time.



Operation of this function for each system status

○: Possible

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

Relevant variables

System parameter constant (System.PrConst.)

Variable/Structure name	Name	Details
OperationCycle[1]	Operation Cycle Setting	Sets each setting value of first operation cycle. Operation Cycle Setting (OperationCycle[1]) is a CYCLE_PARAM structure.
BuffermemoryRefreshCycle	Buffer Memory Refresh Cycle Setting	Sets each setting value of buffer memory refresh cycle. Buffer Memory Refresh Cycle Setting (BuffermemoryRefreshCycle) is a CYCLE_PARAM structure.

System monitor data (System.Md.)

Variable/Structure name	Name	Details
SystemBaseCycle	System Basic Cycle Monitor	Stores the monitor value of system basic cycle. System Basic Cycle Monitor (SystemBaseCycle) is a CYCLE_MONI structure.
OperationCycle[1]	Operation Cycle Monitor	Stores the monitor value of first operation cycle. Operation Cycle Monitor (OperationCycle[1]) is a CYCLE_MONI structure.
BuffermemoryRefreshCycle	Buffer Memory Refresh Cycle Monitor	Stores each monitor value of buffer memory refresh cycle. Buffer Memory Refresh Cycle Monitor (BuffermemoryRefreshCycle) is a CYCLE_MONI structure.
Program_NormalTaskCycle	Normal Task Processing Time Monitor	Stores the monitor value of normal task. Normal Task Processing Time Monitor (Program_NormalTaskCycle) is a CYCLE_MONI structure. The output specific to Normal Task Processing Time Monitor (Program_NormalTaskCycle) is shown below. ■Cycle Setting (Cycle) "0" is always stored because the cycle setting is not existed. ■Cycle Over (CycleOver) FALSE is always set because the cycle over detection is not executed.
WDTerror	WDT Error	Becomes TRUE when WDT error is detected. • FALSE: No WDT error • TRUE: WDT error present
SystemBaseCycle_Counter	System Basic Cycle Counter	Stores the system basic cycle counter.

CYCLE_PARAM

Variable/Structure name	Name	Details
Cycle	Cycle Setting	For details, refer to the following. ☞ Page 105 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. • 0 to 65535
NumOfCycleOverErrDetectTimes	Number of Cycle Over Error Detections	The number of cycles until the cycle over error of each cycle is detected can be set. When "0" is set, it will operate as "5 (initial value)". • 0 to 65535
CycleOverErrorType	Cycle Over Error Selection	An assignment for the over error of each cycle can be selected. • 2: Minor Error (MinorError) • 3: Moderate Error (ModerateError)

CYCLE_MONI

Variable/Structure name	Name	Details
ProcessingTime	Processing Time	Stores the processing time [ns] of each cycle.
MaximumProcessingTime	Maximum Processing Time	Stores the maximum processing time [ns] of each cycle.
Cycle	Setting Cycle	Stores the cycle setting [ns] of each cycle.
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 105 Operation Cycle

■ System basic cycle setting

Stores the system basic cycle setting in Cycle Setting (System.Md.SystemBaseCycle.Cycle) with a unit of nanosecond [ns].

■ System basic cycle processing time

Stores the system basic cycle processing time in Processing Time (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 Maximum Processing Time (System.Md.SystemBaseCycle.MaximumProcessingTime) with a unit of nanosecond [ns].

■ System basic cycle over

For details, refer to the following.

☞ Page 681 System basic cycle over check

Normal task

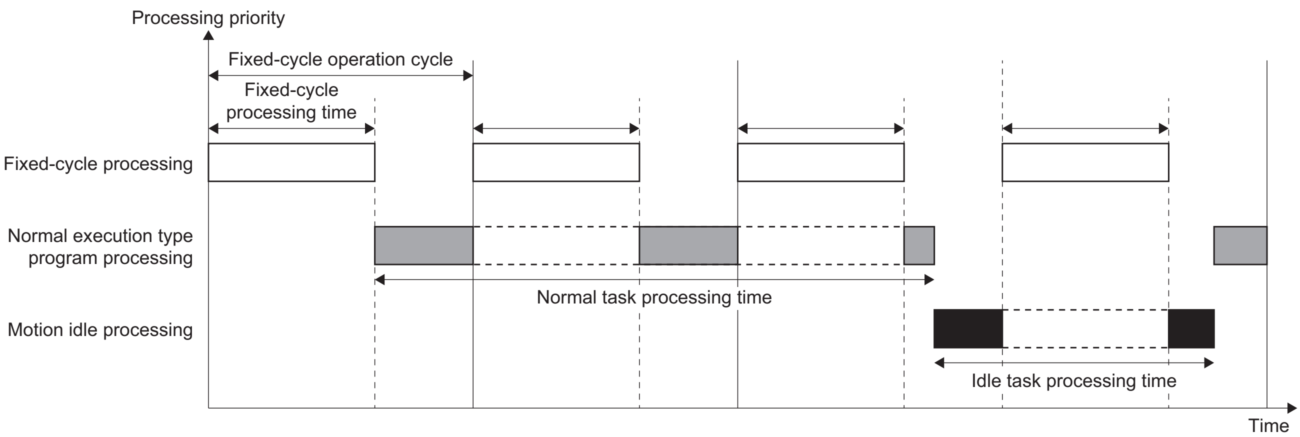
■ Normal task processing time

Stores the normal task processing time in Processing Time (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 Maximum Processing Time (System.Md.Program_NormalTaskCycle.MaximumProcessingTime) with a unit of nanosecond [ns].

First operation cycle

For details, refer to the following.

☞ Page 105 Operation Cycle

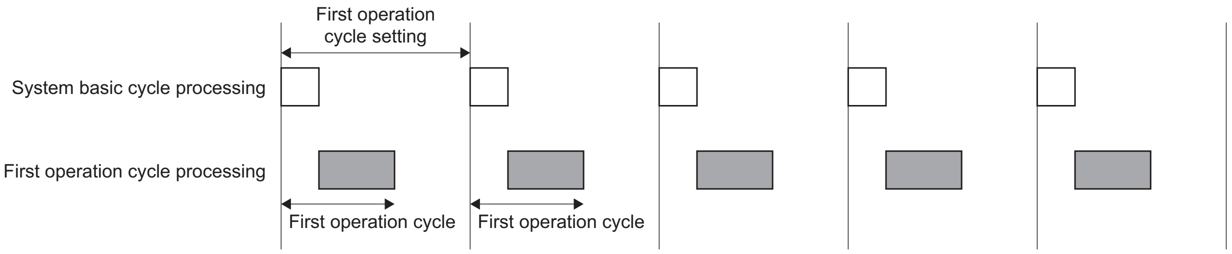
■ First operation cycle setting

Stores the first operation cycle setting in Setting Cycle (System.Md.OperationCycle[1].Cycle) with a unit of nanosecond [ns].

■ First operation cycle processing time

Stores the first operation cycle processing time in Processing Time (System.Md.OperationCycle[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 Maximum Processing Time (System.Md.OperationCycle[1].MaximumProcessingTime).

■ First operation cycle over

For details, refer to the following.

☞ Page 681 Operation cycle over check

Buffer memory refresh

For details, refer to the following.

☞ Page 105 Operation Cycle

■ Buffer memory refresh cycle setting

Stores the buffer memory refresh cycle setting in Setting Cycle (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 Processing Time (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 Maximum Processing Time (System.Md.BufferMemoryRefreshCycle.MaximumProcessingTime) with a unit of nanosecond [ns].

■ Buffer memory refresh cycle over

For details, refer to the following.

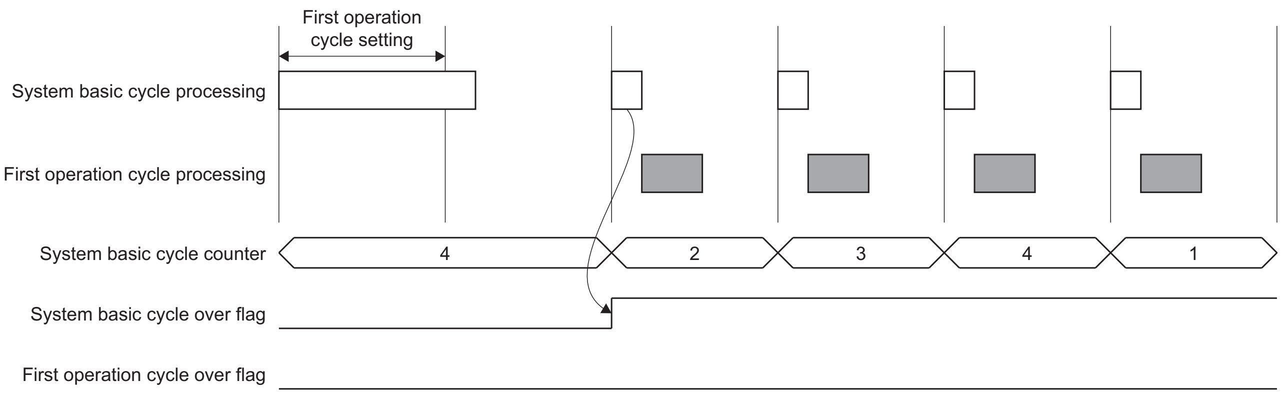
☞ Page 682 Buffer memory refresh cycle over check

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.

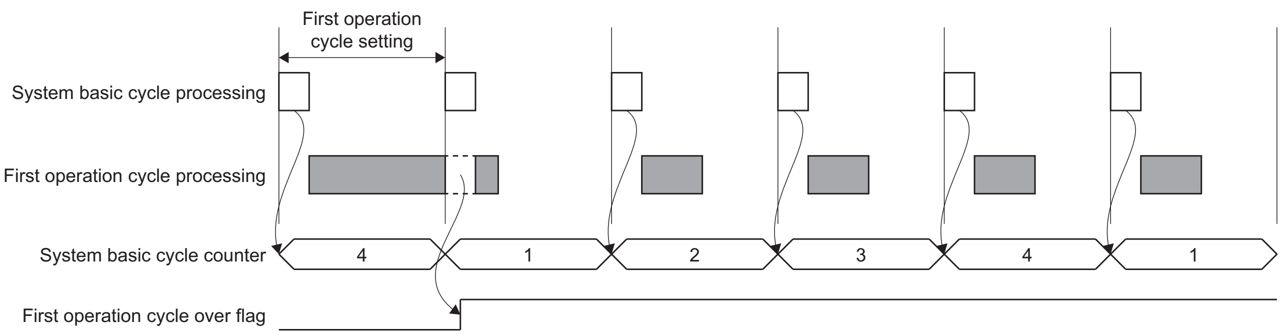
- Power OFF → ON and reset
- Reset by user program



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

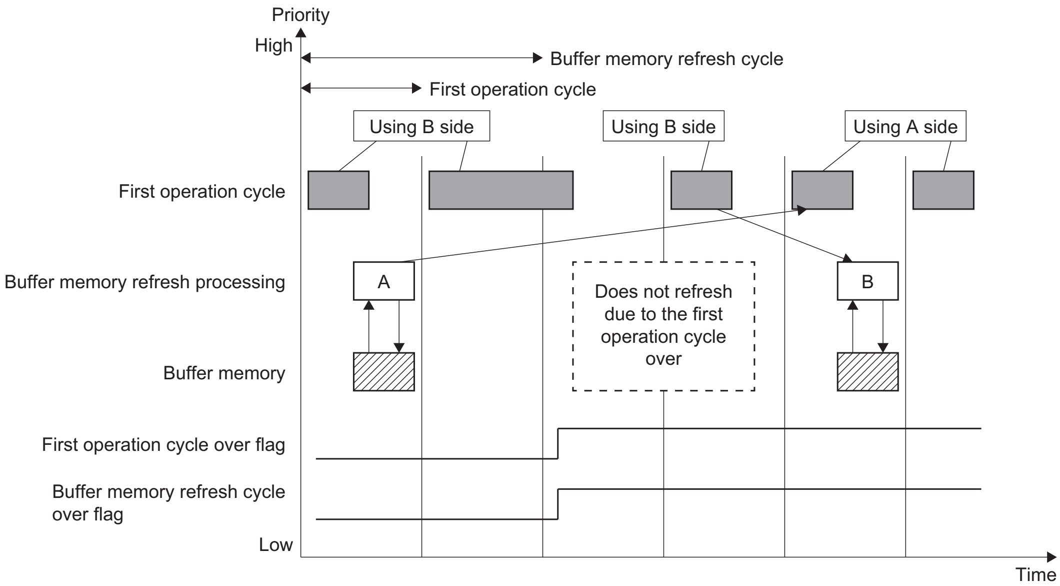
When the buffer memory refresh cycle is not completed before the start of the next cycle, the buffer memory refresh cycle over flag becomes TRUE.

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



Output of errors

WDT error detection

When the processing is stopped because the operation cycle or other processing is error, a WDT error occurs 1 [s] later.

When the WDT error is detected, the motion system stops following the process selected by Stop Selection at All Axes Stop Cause Occurrence (System.Pr.StopMode_All), 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. When "0" is set to the detection number of the cycle over warning, a warning does not occur.

When "0" is set to the detection number of the cycle over error, it will operate as "5 (initial value)".

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 motion system stops following the process selected by Stop Selection at All Axes Stop Cause Occurrence (System.Pr.StopMode_All).

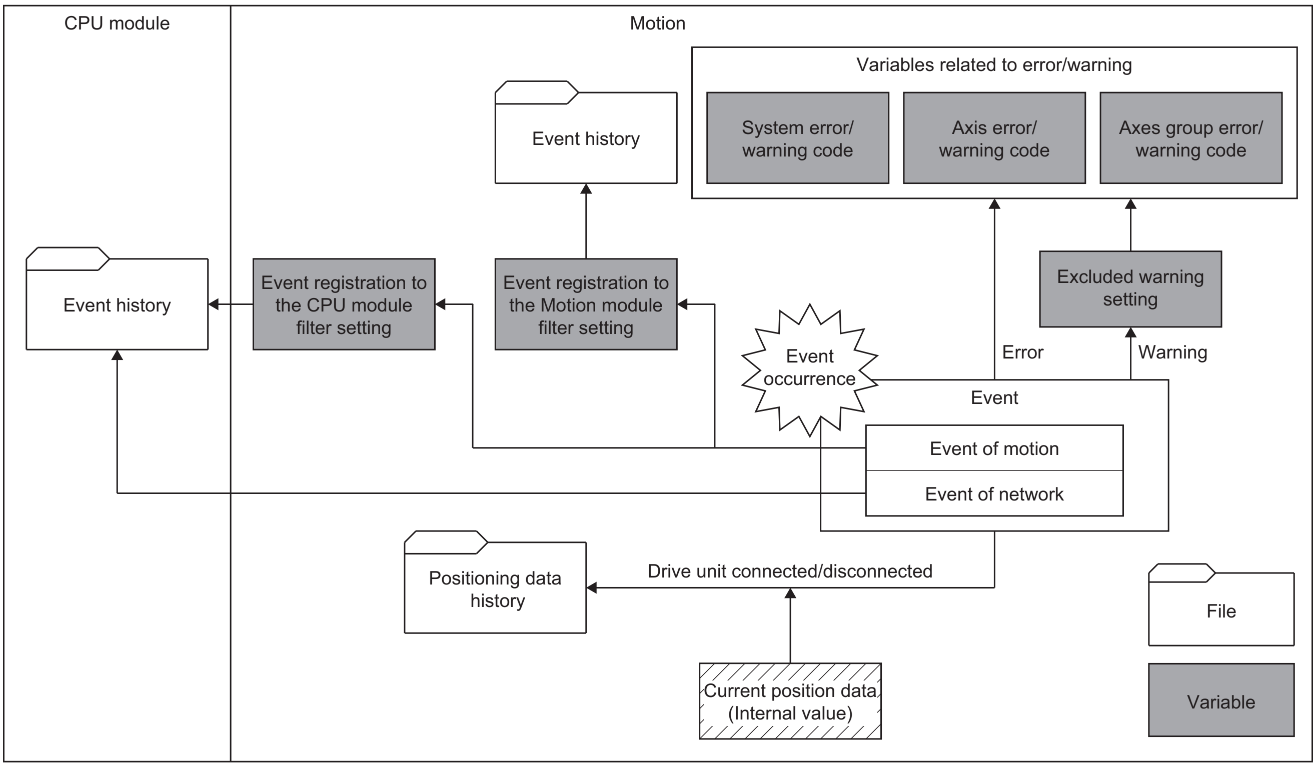
⚠CAUTION

- 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.
- Depending on the system status, the axis cannot decelerate to stop at WDT error detection and may stop immediately.

17.2 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



Operation of this function for each system status

○: Possible

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

Relevant variables

Event history filter setting

■ System parameter constant (System.PrConst.)

Variable/Structure name	Name	Details
EventHistoryMotion_Exclude	Event Excluding Motion Event History	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. • 1 to 2048 [k bytes]
EventHistoryCpu_Exclude	Event Excluding CPU Event History	Sets a event code which does not register an event to the CPU module.

■ System monitor data (System.Md.)

Variable/Structure name	Name	Details
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: Waiting for Execution Request (Ready) • 1: Executing (Executing) • 2: Execution Completed (Done) • -1: Error Occurrence (Error_)

■ System control data (System.Cd.)

Variable/Structure name	Name	Details
EventHistoryMotion_Clear	Motion Event History Clear Execution Request	Carries out execution requests of the event history clear. • FALSE: Not executed • TRUE: Executes the clear

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 789 List of Error 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 829 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 830 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 830 Operation	20000H to 27FFFH

■ Motion control FB execution and stop event

- The Motion control FB 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 Motion Event History Path (System.PrConst.EventHistoryMotion_Path). The file capacity is specified in Motion Event History Capacity (System.PrConst.EventHistoryMotion_Capacity) in k bytes.
  When Motion Event History Path (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 Motion Event History Capacity (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 Motion Event History Path (System.PrConst.EventHistoryMotion_Path) and Motion Event History Capacity (System.PrConst.EventHistoryMotion_Capacity) are changed, the event registered before the change cannot refer.
- When saving the event history file fails, "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 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

The following methods are available for clearing the event history.

Event history clearing method	Description
Clearing by the engineering tool	The event history can be cleared by 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 Motion Event History Clear Execution Request (System.Cd.EventHistoryMotion_Clear) to TRUE. • Motion Event History Clear Execution Status (System.Md.EventHistoryMotion_ClearStatus) turns "1: Executing (Executing)" during the event history clear execution. • Motion Event History Clear Execution Status (System.Md.EventHistoryMotion_ClearStatus) turns "2: Execution Completed (Done)" as the event history clear is completed. When executing the event history clear again, set Motion Event History Clear Execution Request (System.Cd.EventHistoryMotion_Clear) to FALSE and execute the clearing after confirming that Motion Event History Clear Execution Request (System.Cd.EventHistoryMotion_Clear) turns "0: Waiting for Execution Request (Ready)". • When the event history clear is failed, Motion Event History Clear Execution Status (System.Md.EventHistoryMotion_ClearStatus) turns "-1: Error Occurrence (Error_)". When turning to "-1: Error Occurrence (Error_)", Motion Event History Clear Execution Request (System.Cd.EventHistoryMotion_Clear) set to FALSE temporarily and after a while, carry out the event history clear again after confirming that Motion Event History Clear Execution Status (System.Md.EventHistoryMotion_ClearStatus) turns "0: Waiting for Execution Request (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 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

The event history can be cleared by 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.

■ Filter type

The following two settings are available for filter setting.

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 setting can be set by the following.
  - Event Excluding Motion Event History (System.PrConst.EventHistoryMotion_Exclude)
  - Event Excluding CPU Event History
- The filter can be set for each event code.The range can be specified by tying event codes using "-" (hyphen). A setting example is shown below.

Setting example	Setting value
For the setting when the event codes 0x80001000 and 0x80001010 are not to be registered in event history	0x80001000,0x80001010
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 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)
- The positioning data history file is updated periodically. When the power of the motion system is turned OFF or the system is reset before saving positioning data to the positioning data history file, the unsaved positioning data is erased.

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, "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, "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.

Relevant add-ons

The following add-on is required to use this function.

- MotionEventHist

System memory capacity

■ Backup RAM Usage

Refer to the following.

☞ Page 686 Event history file of the Motion module side

17.3 Servo System Recorder

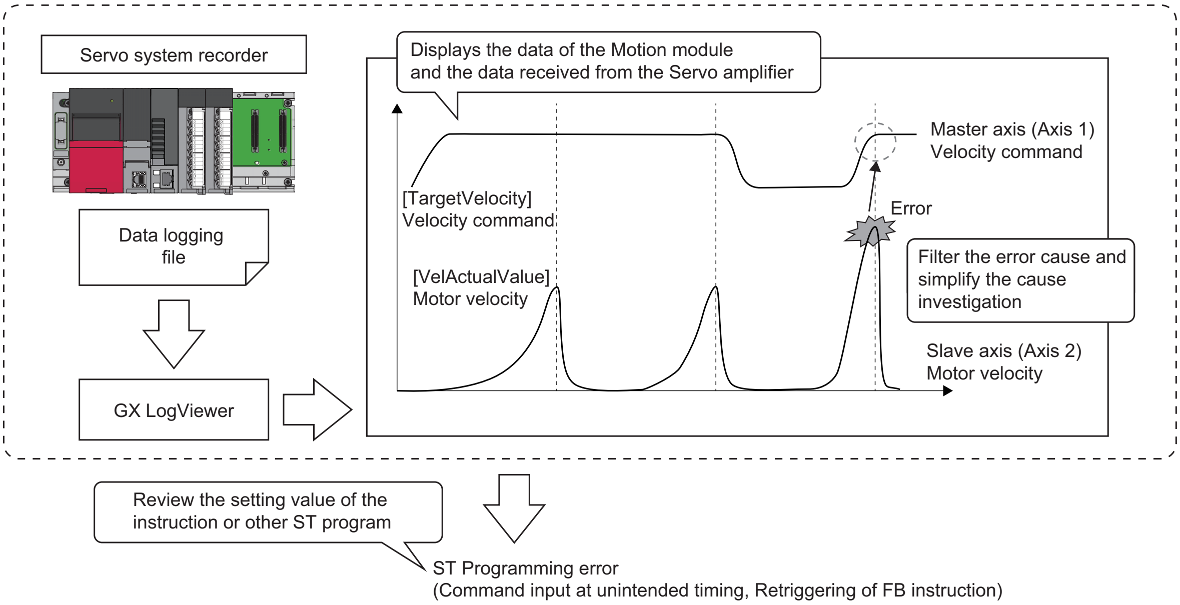
The servo system recorder generates a logging setting file which is optimal for analyzing of the error factors when Add-on ServoSystemRecorder Monitor is enabled. The logging setting file continuously monitors the error status of the motion system and supported device stations. When an error is detected, 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 device stations which are connected to the slave axis in cam control.



Operation of this function for each system status

○: Possible, △: Possible (restricted), ×: Not possible

Status	Operation availability
STOP	○
RUN	○
Moderate error	△ (Not possible if READY does not become TRUE)
Major error	×

Relevant variables

System parameter constant (System.PrConst.)

Variable/Structure name	Name	Details
Addon_ServoSystemRecorder	Add-on ServoSystemRecorder Parameter	Sets the maximum capacity for the memory used in Add-on ServoSystemRecorder parameter (Addon_ServoSystemRecorder). Add-on ServoSystemRecorder parameter (Addon_ServoSystemRecorder) is a CYCLE_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. • FALSE: Disabled • TRUE: Enabled For details on the auto logging, refer to the following. ☞ Page 664 Auto logging
CoRecordingEnable	Co-recording Setting	Sets whether to enable/disable co-recording. • FALSE: Not used • TRUE: Used

System monitor data (System.Md.)

Variable/Structure name	Name	Details
Addon_ServoSystemRecorder	Add-on ServoSystemRecorder Monitor	Sets the maximum capacity for the memory used in Add-on ServoSystemRecorder parameter (Addon_ServoSystemRecorder). Add-on ServoSystemRecorder parameter (Addon_ServoSystemRecorder) is a ADDON_MONI structure.
ServoSystemRecorder_Enable	Servo System Recorder Enabled	Displays whether logging collection by the servo system recorder is enabled or disabled. Becomes TRUE when the logging collection is operating normally. • FALSE: Disabled • TRUE: Enabled
CoRecordingStatus	Co-recording Operation Status	Displays the operation status of co-recording. • 0: Stopped (CoRecordingStopping) • 1: In operating (CoRecordingEnable)

ADDON_PARAM

Variable/Structure name	Name	Details
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

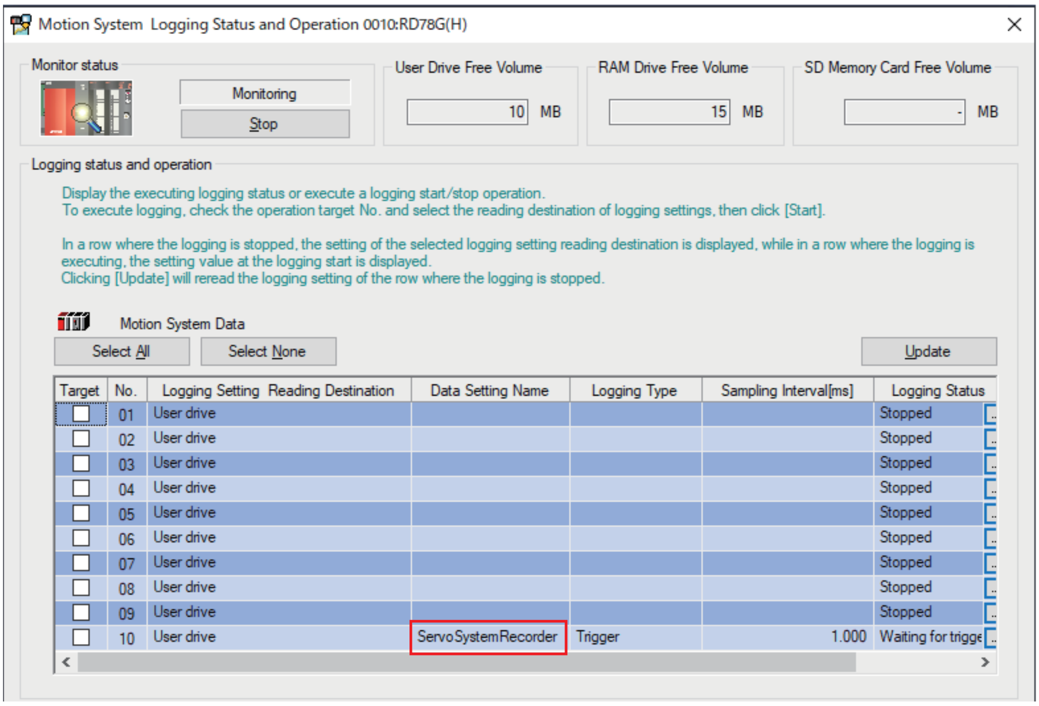
Variable/Structure name	Name	Details
RamUsage	RAM Usage	Stores current amount of system memory (RAM) used in the Add-on ServoSystemRecorder in k bytes.
RamMaxUsage	RAM Usage Maximum	Stores the maximum amount of system memory (RAM) used in the Add-on ServoSystemRecorder in k bytes.
BackupRamUsage	Backup 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.
Addon_ServoSystemRecorder	Version	Stores the version information of the Add-on ServoSystemRecorder.

Control details

Logging setting file auto generation

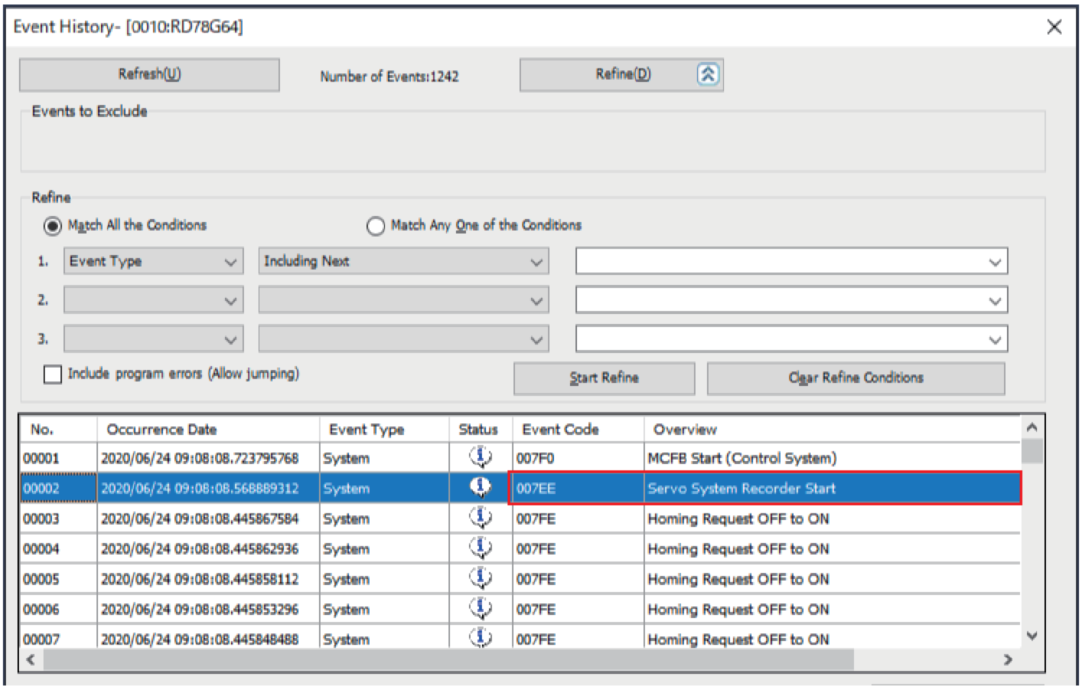
This add-on reads the axis setting in the project at the start of the motion system and automatically generates a logging setting file with collected data reflected with a number of axis setting required for error analysis.
The logging setting file is automatically generated every time the motion system 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 "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.

Motion Control Setting Function ⇨ [Tool] menu ⇨ [Logging function] ⇨ [Logging Setting] ⇨ "Motion System Logging Setting" window ⇨ [Online] ⇨ [Logging Status and Operation]



Also, the auto-generation completion can be confirmed if "007EE: Servo System Recorder Start" is shown in the "event history" window.

Motion Control Setting Function ⇨ [Online] menu ⇨ [Motion Monitor] ⇨ [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 704 How to delete the logging setting file

■ Storage location

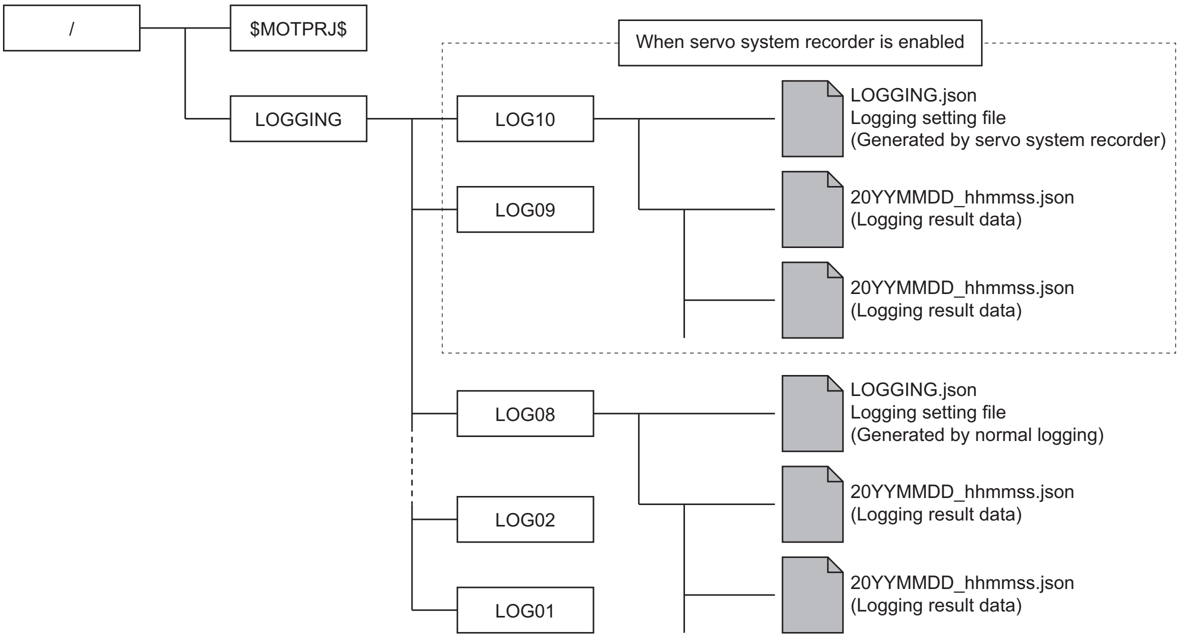
The logging setting file generated by this add-on is stored in an SD memory card or the user drive of the motion system. If the SD memory card is inserted, it is prioritized as a storage location of the logging setting file. (The storage location of the result file will be the same directory of the logging 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 device station.
- 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.

■ Editing procedure

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", "Servo System Recorder Setting File Creation Disabled Warning (warning code: 0F13H)" occurs and the servo system recorder does not start.
- Do not change the logging start condition of the logging setting file. The servo system recorder starts automatically even when the setting is changed to "Starts by user operation".

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 combination	
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.

- Axis monitor data (AxisName.Md.)

○: 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)
Object Data_TargetPos (Io_TargetPos)	Indicates a value of object data TargetPos (The position command to be sent to the device station)	○	○
Object Data_ PosActualValue (Io_PosActualValue)	Indicates a value of object data PosActualValue (The feedback of position to be received from the device station)	○	○
Object Data_TorqueActualValue (Io_TorqueActualValue)	Indicates a value of object data TorqueActualValue (The feedback of torque to be received from the device station)	○	○
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_VelActualValue (Io_VelActualValue)	Indicates a value of object data_ VelActualValue. (The feedback of velocity to be received from the device station) For the unit, refer to the manual of connected driver.	○	—*1
Object Data_TargetVelocity (Io_TargetVelocity)	Indicates a value of object data_ TargetVelocity (The velocity command to be sent to the device station) For the unit, refer to the manual of connected driver.	○	—*1

*1 To set Object Data_VelActualValue (Io_VelActualValue) and Object Data_TargetVelocity (Io_TargetVelocity) data even when there are 129 axes or more, add them by logging setting tool.

- System monitor data (System.Md.)

Item	Description
Processing time (OperationCycle[1].ProcessingTime)*1	Stores the processing time of the first operation cycle in a unit of μs.
Cycle over (OperationCycle[1].CycleOver)*1	Becomes TRUE when the cycle over of the first operation cycle is detected.
Latest motion area system error code (ErrorID)	Stores the latest error code.
Network area error code (NetWorkErrorID)	Stores the error code when a network error occurs.

*1 CYCLE_MONI structure

Precautions

When the axis operation process fast operation mode is enabled, the sampled value is always 0 because the following monitors are not updated.

- Object Data_TargetPos (AxisName.Md.Io_TargetPos)
- Object Data_PosActualValue (AxisName.Md.Io_PosActualValue)
- Object Data_TargetVelocity (AxisName.Md.Io_TargetVelocity)
- Object Data_VelActualValue (AxisName.Md.Io_VelActualValue)
- Object Data_TorqueActualValue (AxisName.Md.Io_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 696 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 Trigger Request (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 Communication Error	Minor error (Network error)
1C45H	SLMP Communication 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.

Co-recording function

The co-recording function synchronizes data-saving trigger generation between recording devices supporting the co-recording function.
Because each recording device can save data simultaneously by using this function, the triggers of the servo system recorder or recorders of programmable controllers on other stations can be generated at the same time as a trigger that detects a problem. In addition, by comparing and analyzing the data saved on different recording devices, a wide range of data anomalies can be found. (This function reduces the time to run an anomaly analysis of the entire system.)
The co-recording function in the Motion module applies to the logging data created by the servo system recorder function.
For details of the co-recording function, refer to the following.
MELSEC iQ-R System Recorder Co-recording Function Reference Manual

This section describes the operation of the Motion module for the co-recording function.

Control details

Enable co-recording function by setting Co-recording setting (System.PrConst.CoRecordingEnable) to "TRUE (Used)". (The initial value of Co-recording setting (System.PrConst.CoRecordingEnable) is "FALSE (Not used)".)
The co-recording operation status can be checked in Co-recording operation status (System.Md.CoRecordingStatus).

Co-recording Operation Status (System.Md.CoRecordingStatus)	Description
0: Stopped (CoRecordingStopping)	The co-recording is in the stop status.
1: In operating (CoRecordingEnable)	The co-recording is in the operating status.

Point

When the control CPU module does not support the co-recording function, or its co-recording setting is set to "Not used", "Co-recording Setting Error (error code: 1300H, 1301H)" will occur, and the co-recording function is not enabled.

■ Co-recording trigger sending/receiving

The servo system recorder detects triggers when a co-recording trigger detected in another module is received via a control CPU module. Additionally, when the servo system recorder detects a trigger, it notifies other modules via a control CPU module with a co-recording trigger.

Item	Description
Co-recording trigger receiving	When a recording trigger in another module is generated, and a co-recording trigger is received via a control CPU module, the servo system recorder sets the Trigger Request (System.LoggingRef[10(,9)].Cd.Trigger) of the data logging in operation (Logging Status (System.LoggingRef[10(,9)].Md. LoggingStatus) is "TRUE (Executing)") to "TRUE (Requested)" and detects a trigger.
Co-recording trigger sending	When a trigger is detected (Triggered (System.LoggingRef[10(,9)]).Md.Triggered) is TRUE (Triggered)) in the data logging operating, the servo system recorder sends a co-recording trigger to other modules via a control CPU module.

■ Co-recording trigger relay

Co-recording triggers are relayed within the programmable controller system that a module is installed, and to other stations that belong to the same network. For details, refer to "Co-recording function" in the following manual.
MELSEC iQ-R Motion Module User's Manual (Network)

Precautions

- For restrictions on the co-recording function, refer to the following.
  MELSEC iQ-R System Recorder Co-recording Function Reference Manual
- For each device, after generating a recording trigger, there is a delay in the time until the recording trigger is generated in the other recording devices. For delay times, refer to the following.
  MELSEC iQ-R System Recorder Co-recording Function Reference Manual
- When the logging setting file is edited and the "Trigger command" is excluded from the trigger conditions, co-recording trigger receiving will not operate even if the Co-recording setting (System.PrConst.CoRecordingEnable) is set to "TRUE (Used)". When using the co-recording function, do not exclude the trigger command from the trigger conditions.

Point

When the co-recording function does not operate, check the following items.

- Check that the firmware version of each module supports co-recording. For devices that support co-recording, refer to the following.
  MELSEC iQ-R System Recorder Co-recording Function Reference Manual
- Check that the co-recording setting is set to "Used" for each module.
  <Programmable controller setting>
  Set the co-recording setting to "Used".
  <Motion module setting>
  Set the Co-recording setting (System.PrConst.CoRecordingEnable) to "TRUE (Used)".
- Co-recording relay range
  For an add-on baseSystem version of "1.26" or later, the number of relay networks is fixed to 1.

Version combinations

This function can be used with ServoSystemRecorder add-on version "1.25" or later.
For the versions of other devices and engineering tools, refer to the following.
MELSEC iQ-R System Recorder Co-recording Function Reference Manual

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 pressing down 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 Servo System Recorder Enabled (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.
- Auto Logging Enabled (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 canceled.
- The error factors occurred in the logging are removed.

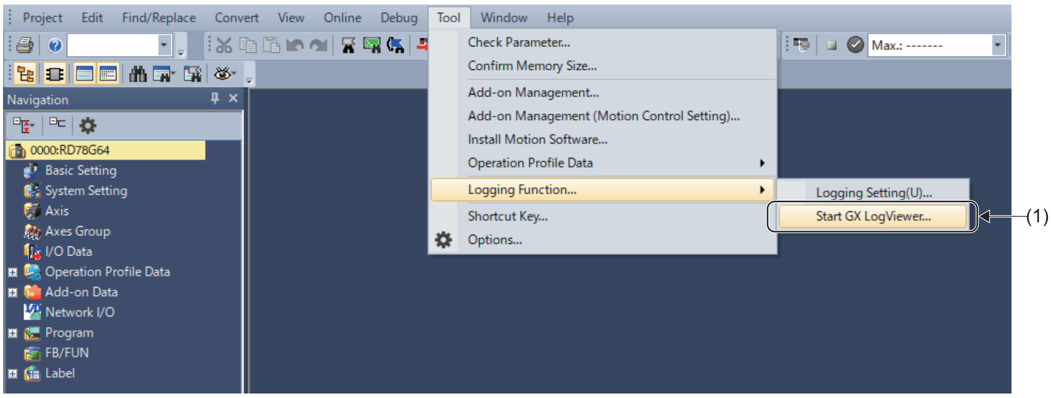
Precautions

- The logging cannot be restarted even if the Logging Request (System.LoggingRef[1..10].Cd.StartLog) of the system label is turned TRUE.
- "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.

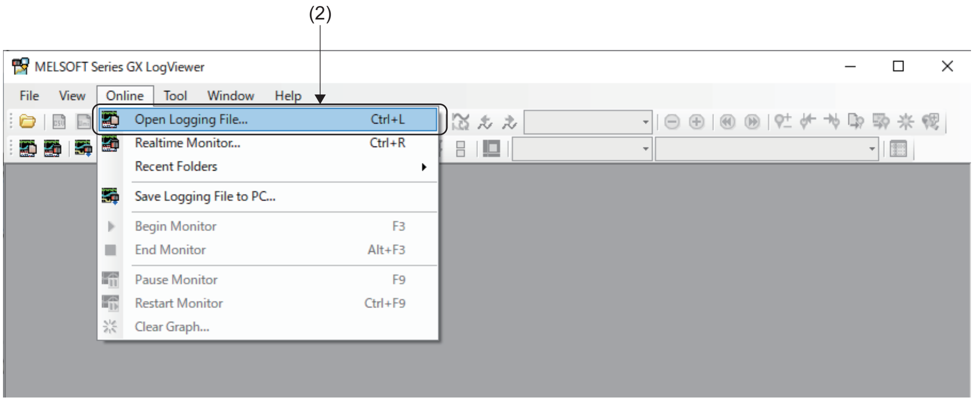
How to check the logging data

This section describes the procedures to check the logging data in the motion system.

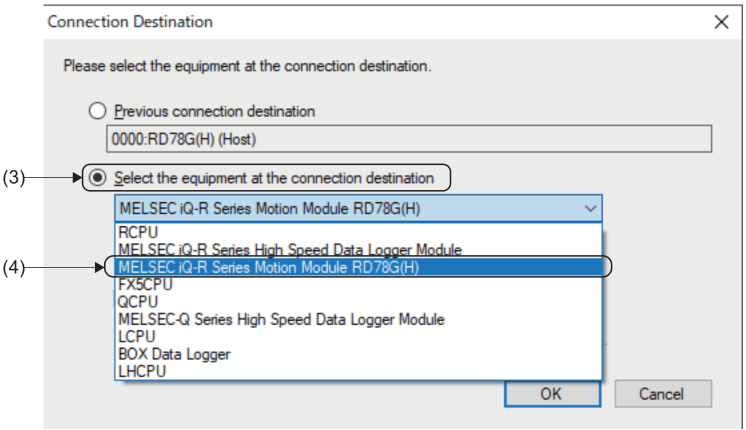
1. In "Motion Control Setting Function", select [Tool] menu ⇨ [Logging Function] ⇨ [Start GX LogViewer](1).



2. GX LogViewer starts. Select [Online] ⇨ [Open Logging File](2).

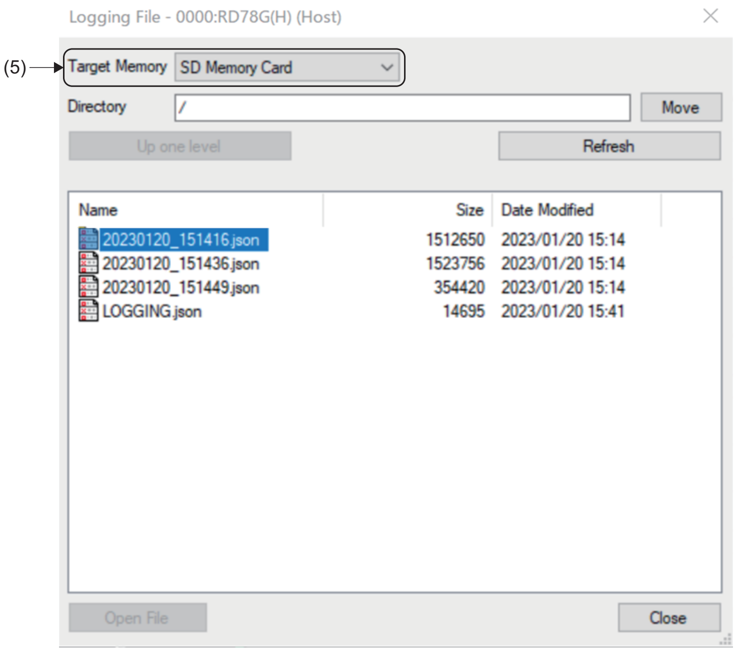


3. The "Connection Destination" window opens. Select "Select the equipment at the connection destination"(3), Select "MELSEC iQ-R Series Motion Module RD78G(H)"(4) from the pull-down list, and click the [OK] button.

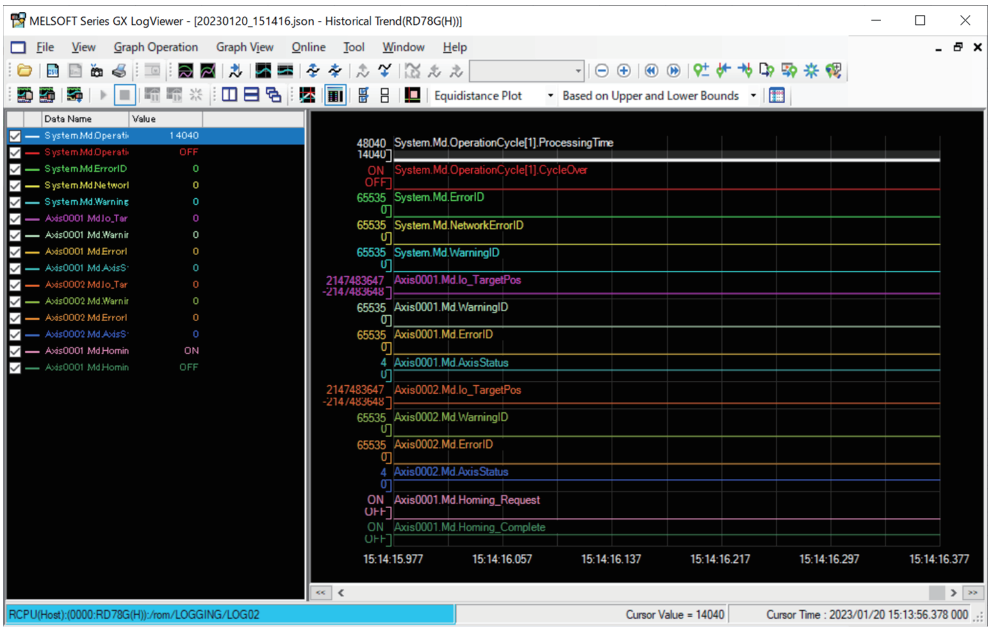


4. The "Logging File" window opens. Specify the "Target Memory"(5). (When the SD memory card is inserted, the storage location of the data logging file is the SD memory card.)

5. Select the logging data to be checked, and click the [Open File] button. (Multiple selection of logging data files is not possible.)

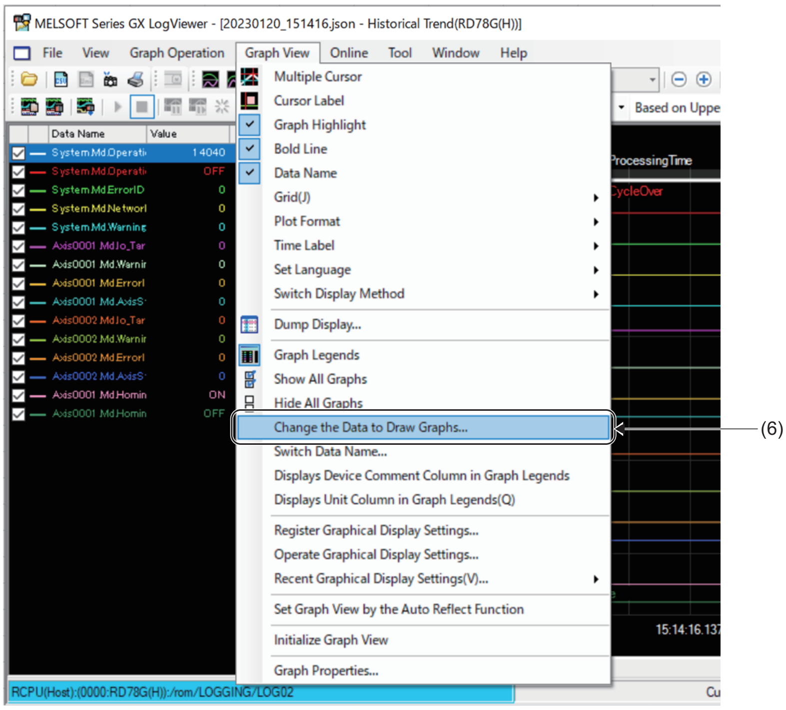


6. The "Historical Trend" window of the selected logging data opens. The contents of the logging data can be checked in the "Historical Trend" window.

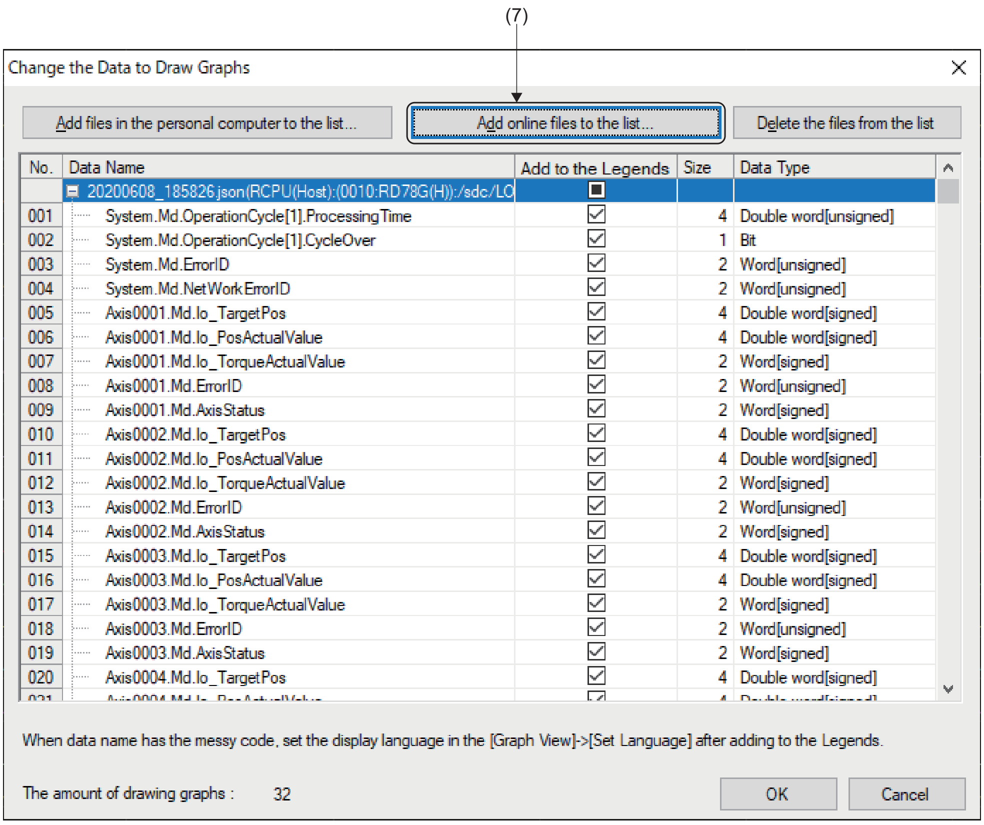


<The following procedures are required when there are 129 or more axes for logging.>

7. In GX LogViewer, select [Graph View] menu ⇨ [Change the Data to Draw Graphs](6).



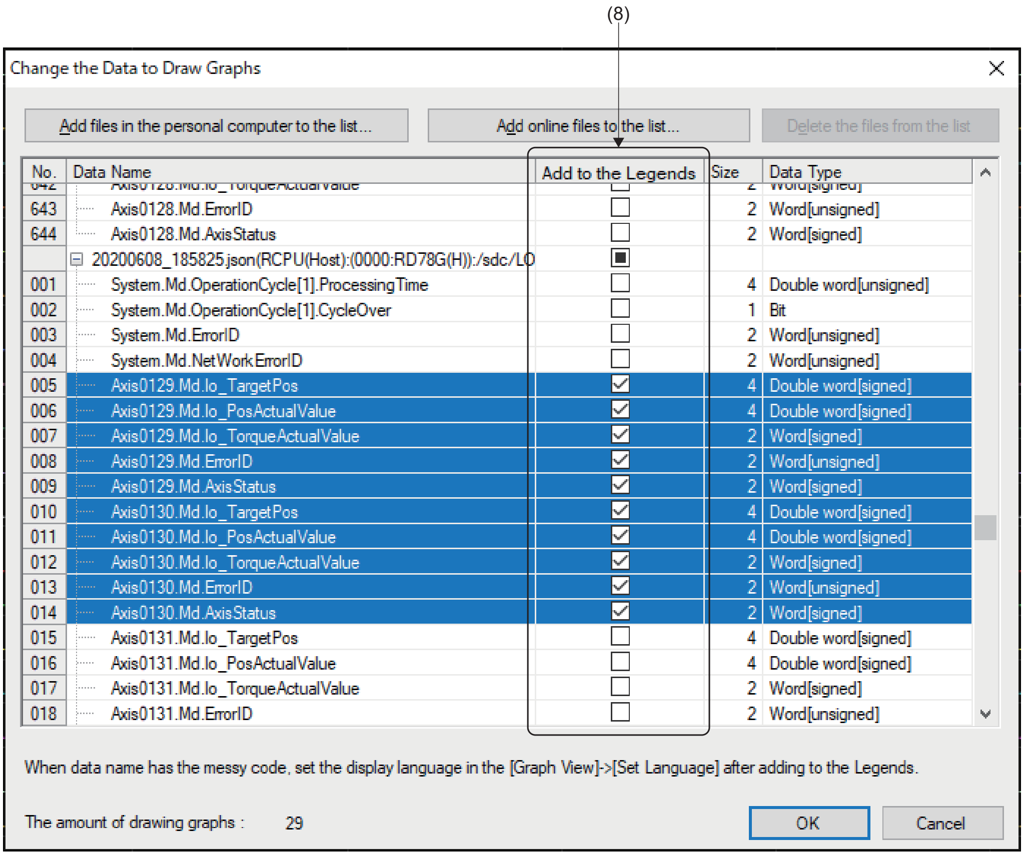
8. The "Change the Data to Draw Graphs" window opens, click the [Add online files to the list] button(7).



9. The "Connection Destination" window opens. Add data logging file by the same operation as step 3 to 4.

10. Check the "Add to the Legends"(8) column of the added data logging file.

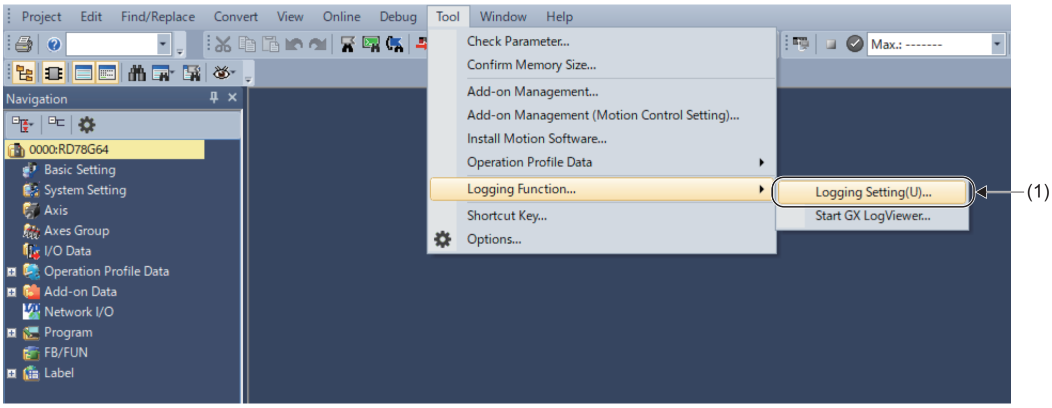
Up to 32 data are available to draw at the same time. Change the target data of drawing if necessary.



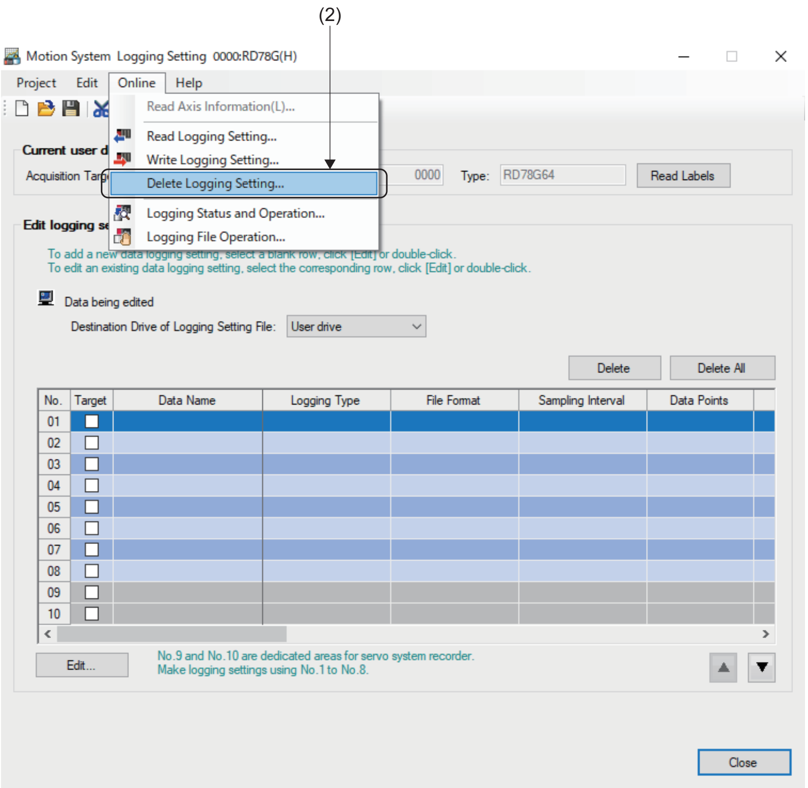
How to delete the logging setting file

This section describes the procedures to delete the logging setting file.

1. In "Motion Control Setting Function" , select [Tool] menu ⇨ [Logging function] ⇨ [Logging Setting](1).

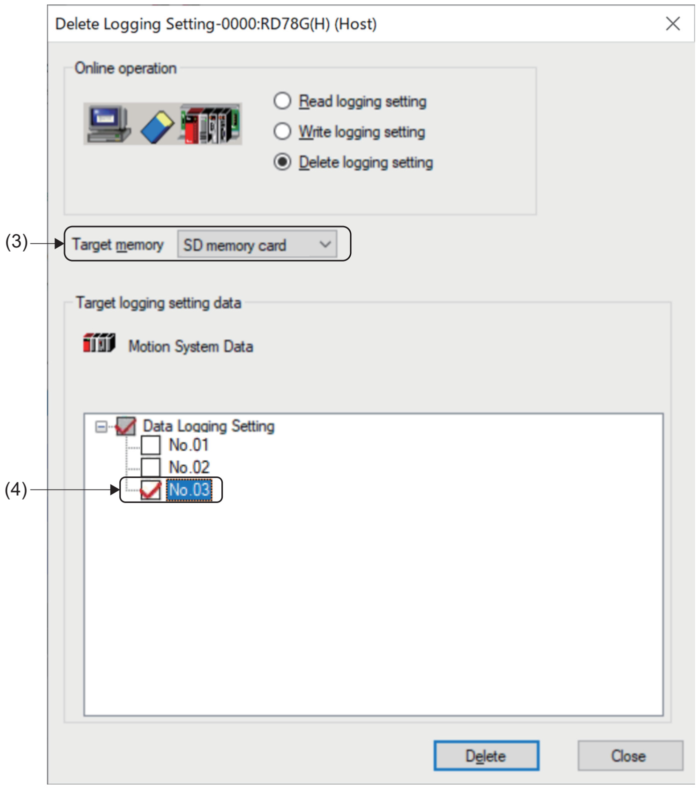


2. The "Motion System Logging Setting" window opens. Select [Online] ⇨ [Delete Logging Setting](2).



3. The "Delete Logging Setting" window opens. Select the memory which stores the logging setting file to be deleted from "Target memory"(3).

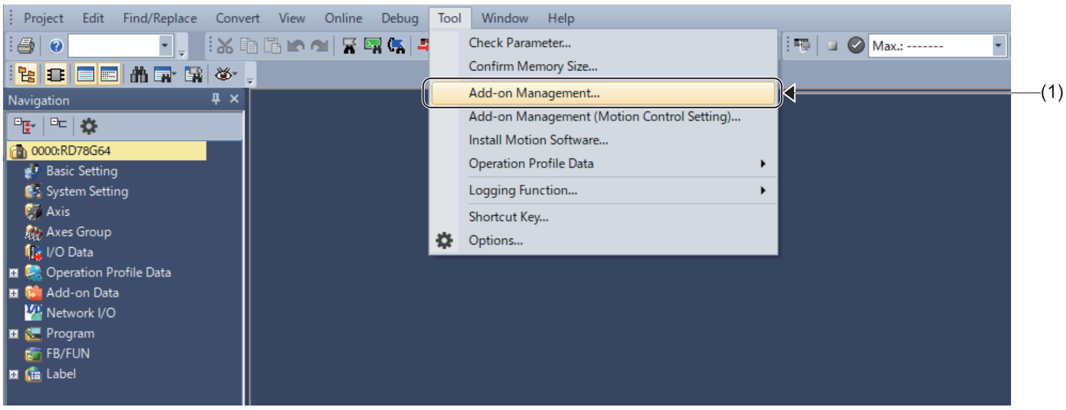
4. The stored logging setting file is displayed in "Target logging setting data". Insert a check(4) for the logging setting file to be deleted, and click the [Delete] button.



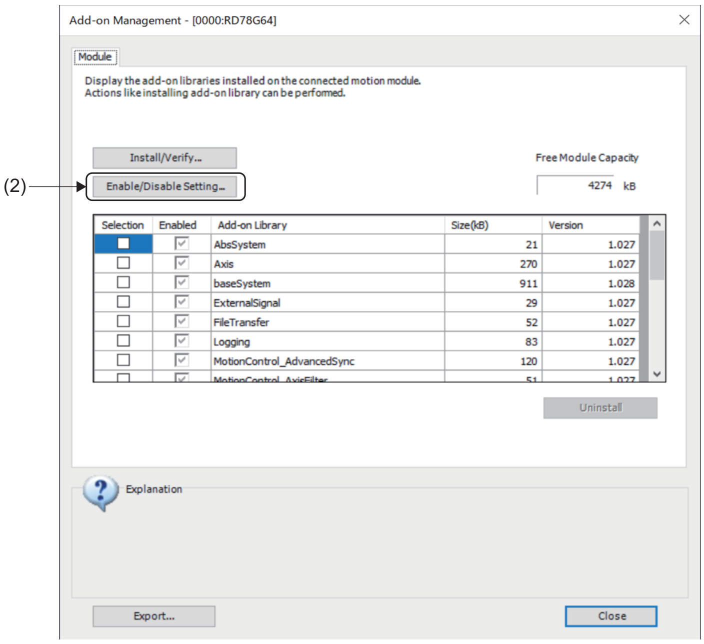
How to disable servo system recorder

When not using this function, disable the Add-on ServoSystemRecorder with the add-on library setting.
This section describes the procedures to set the add-on library to disable setting.

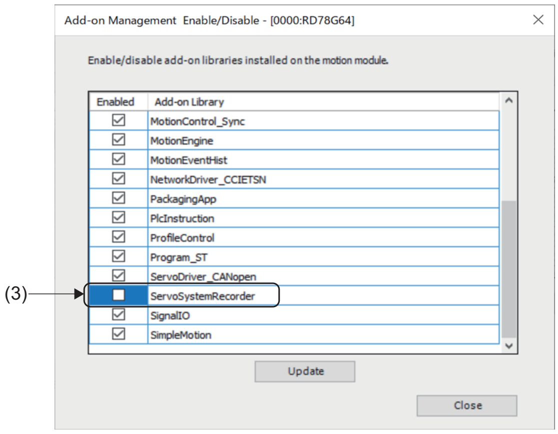
1. In "Motion Control Setting Function", select [Tool] menu ⇨ [Add-on Management](1).



2. The "Add-on Management" window opens. Click the [Enable/Disable Setting] button(2).

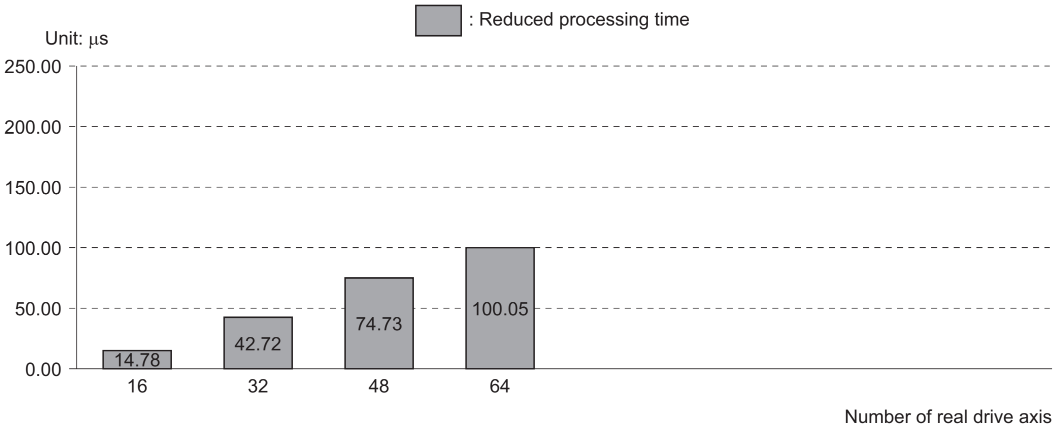


3. The "Add-on Management Enable/Disable" window opens. Uncheck(3) the Enable column of ServoSystemRecorder, and click the [Update] button.

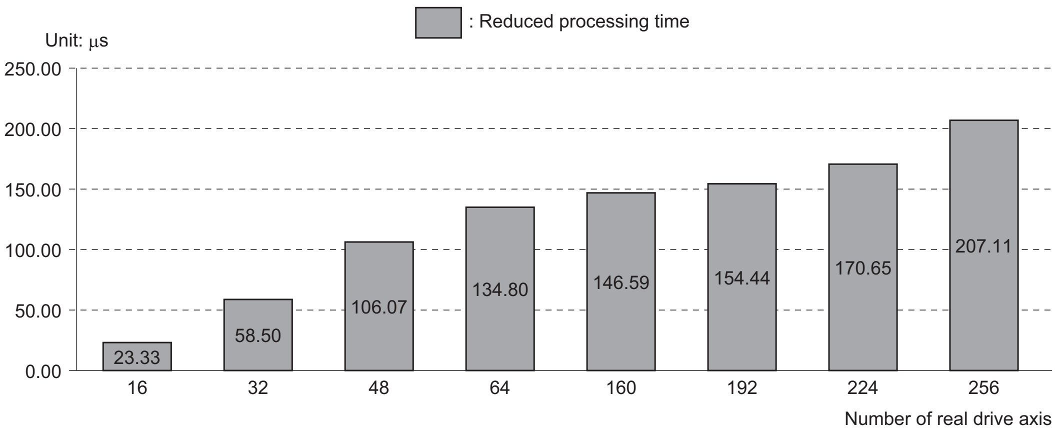


When not using this function the processing time of the operation cycle is reduced. An approximate guide of the reduced processing time is shown below.

- For RD78G



- For RD78GH



Precautions

Relevant add-ons and versions

The following add-ons are required besides this add-on.

- Logging version "1.7" or later
- Axis version "1.7" or later

When the above add-ons are being disabled while this add-on is enabled, "Servo System Recorder Inconsistency Incorrect Warning (warning code: 0F10H)" occurs.

System memory capacity

With the operation of this function, the system memory of Add-on Logging in addition to Add-on ServoSystemRecorder is used.

The system memory (RAM) usage of each add-on is shown below.

■ Add-on ServoSystemRecorder

- RamUsage

Motion module	RAM usage
RD78G	256K bytes
RD78GH	512K bytes

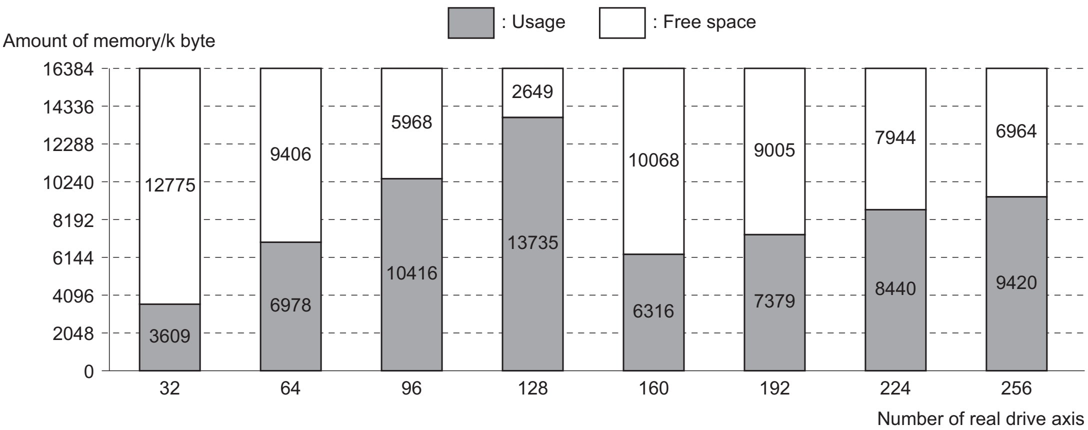
- BackupRamUsage

  Backup RAM is not used.

■ Add-on Logging

- RamUsage

  RAM is used when logging with the logging setting file output by the servo system recorder. An example is shown below.



- The default memory usage setting of the logging is 16384K bytes.
- 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.

Point

RAM usage may be insufficient when logging settings are changed. When RAM usage is insufficient, review the logging settings, or increase the RAM usage of Add-on Logging from system settings. For how to change RAM usage, refer to the following.

☞ Page 121 System Memory Settings

- BackupRamUsage

  Backup RAM is not used.

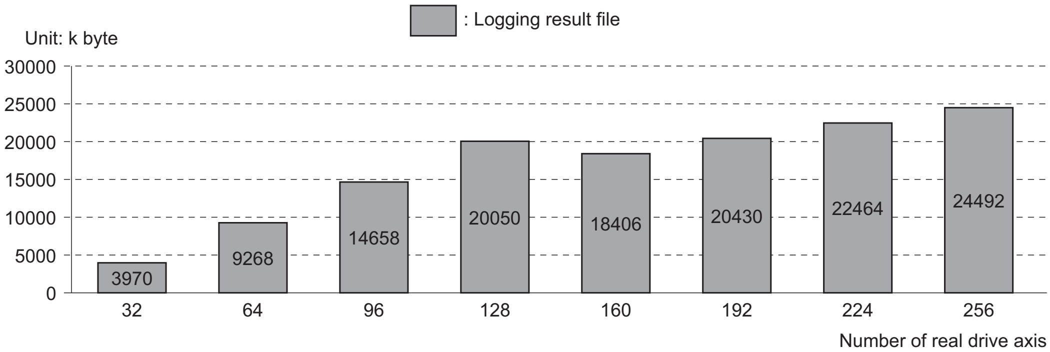
Logging setting file and logging result file sizes

The logging setting file and logging result file generated by this add-on are saved in a user drive or an SD memory card. For logging setting file and logging result file refer to the following.

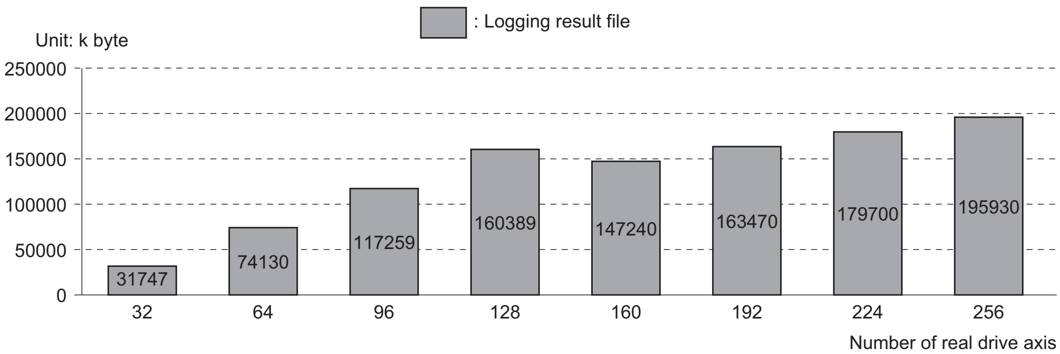
☞ Page 694 Details of logging setting

The default setting for the logging setting file size is up to 500K bytes. An approximate guide for the memory usage required for saving the logging result file is shown below.

- When saving to a user drive
  When using a user drive as the default setting, the number of saved files for the logging result file is "2".



- When saving to an SD memory card
  When using an SD memory card as the default setting, the number of saved files for the logging result file is "16".



Point

When editing the logging settings by adding collected data or number of records, the memory capacity may become insufficient. When memory capacity is insufficient, review the logging settings, or change the saving destination to SD memory card.

18 MOTION SERVICE PROCESSING

This section describes the motion service processing which is executed in the service task.

Operation of this function for each system state

○: Possible

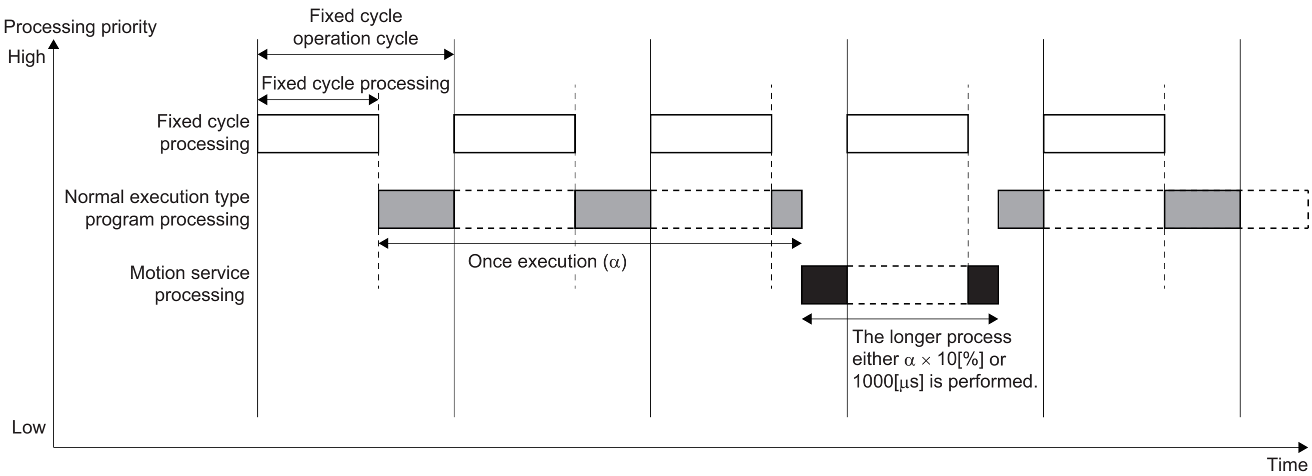
Status	Operation availability
STOP	○
RUN	○
Moderate error	○
Major error	○

18.1 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.



18.2 Motion Service 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.
Variable importing/reflection processing	Imports/reflects the variables whose import timing is "service".
Initialization processing for each function	Executes the initialization processing for each function. For details, refer to the specifications of each function.

19 HOW TO CONTROL FROM CPU MODULE

19.1 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)
For the setting method from the C Controller module, refer to the following.
MELSEC iQ-R C Controller Module User's Manual (Application)

Module label

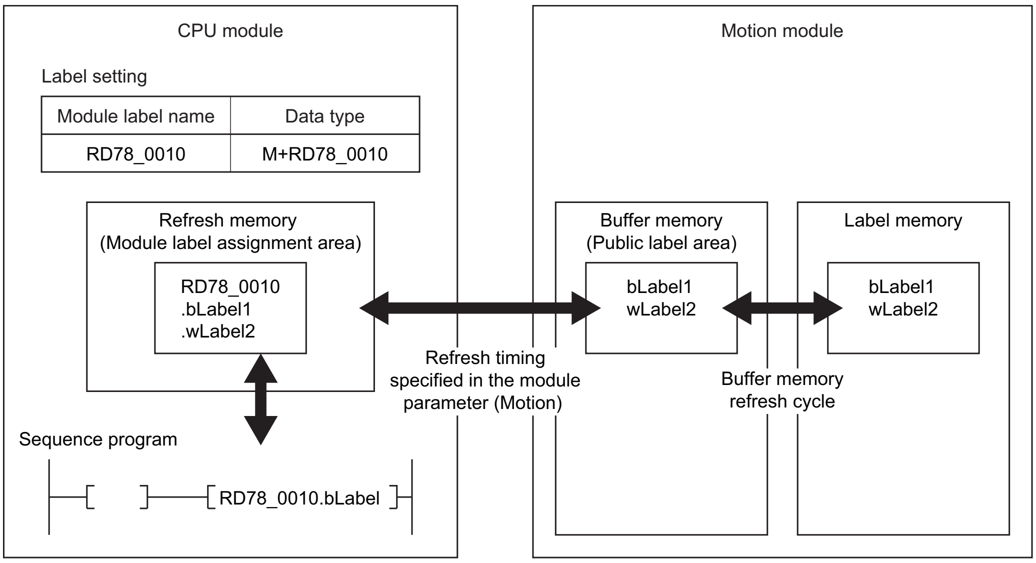
The global label in the Motion module is available as the module label from the CPU module.

■ 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 105 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 labels.
- 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_REAL 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

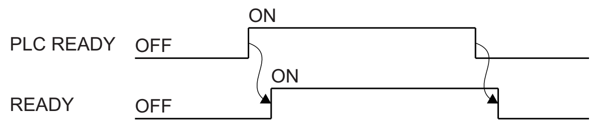
I/O signals

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 number 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 timing and conditions of the I/O signals are shown below.

■ Input signal (Device X)

Device No.	Signal name	Signal state	Description
X0	READY	OFF: Not READY ON: READY	• When the PLC READY turns from OFF to ON, the parameter setting range is checked. If no error is found, this signal turns ON. • When the PLC READY 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

*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 Synchronization flag is required.	The interlock by Synchronization flag is required.
Network control area	The interlock is not required.	

■ Output signal (Device Y)

Device No.	Signal name	Signal state	Description
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 sequence program. · When a stop error occurs in the PLC CPU, it is treated as power OFF. • When the data (parameter) is changed, the PLC READY is turned OFF depending on the parameter. • The following processes are carried out when the PLC READY turns from OFF to ON. · The parameter setting range is checked. · The READY signal turns ON. • The following processes are carried out when the PLC READY turns from ON to OFF. · The READY signal turns OFF. · The operating axis stops.

Buffer Memory

The buffer memory arrangement is shown in the following list.
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

Do not use the other areas than the above.

19.2 Using Motion control FB

This section describes how to use the Motion control FB in sequence programs of the CPU module.

Relevant variable

System monitor data (System.Md.)

Variable/Structure name	Name	Details
BuffermemoryFreeMcfbArea	Buffer Memory MCFB Communication Area Free Size	Displays the number of unused points of the area for Motion control FB.

Control details

The specifications of variables that are used for the Motion control FB of the CPU module side are shown below.

- Specifying AXIS_REF and AXES_GROUP_REF

  When using the Motion control FB 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 Motion control FB.

AXIS_REF and AXES_GROUP_REF to be specified	Setting
When 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 No. , axes group No., and module I/O Number of the control target) are set.
When AXIS_REF and AXES_GROUP_REF that are defined as labels in the CPU module are used.	Set the members (the axis No. , axes group No. and the module I/O Number to be controlled) of AXIS_REF and AXES_GROUP_REF by using the sequence program.

- Specifying the I/O number

  When using the Motion control FB of the CPU module side, the start I/O number of the target Motion module has to be specified for the I/O variables (I/O Number (AxisName.AxisRef.StartIO), etc.) used in the Motion control FB.

  Each Motion control FB definitely defines one input variable that requires the I/O Number (StartIO) setting. For the input variables that require the I/O Number (StartIO) setting, refer to "Motion Control FB Overview" in the following.

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

  For the Motion control FB that can input multiple variables whose member includes I/O Number (StartIO), the setting of I/O Number (StartIO) excluding the corresponding variables is not required. (The values are ignored.)

  For the start I/O number of the target module, refer to UNIT_LABEL.uIO of the module label or directly input the constant.

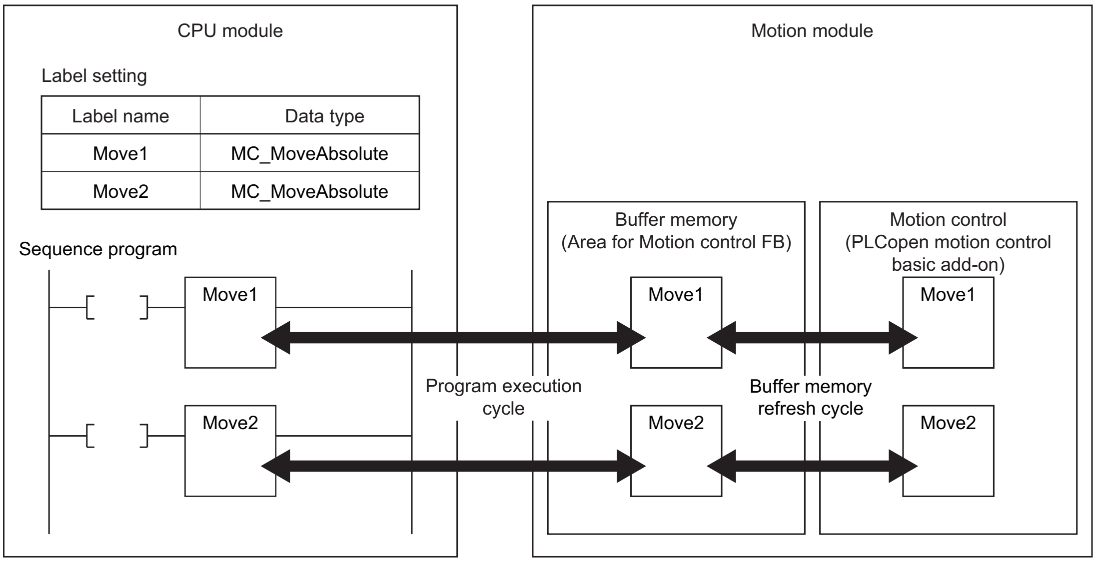
  Be sure to establish I/O Number (StartIO) of each Motion control FB before the FB is called for the first time.

- Specifying file names and variables (label, device)

  The files and the variables when using the FILE_LOCATION type structure (for specifying file names) or the TARGET_REF type structure (for specifying variables such as labels and devices) in a sequence program of the CPU module, they are the files and the variables used in the Motion module, not used in the CPU module.

Generating Motion control FB instance

Control corresponding to the Motion control FB can be carried out in the Motion module side by instantiating the Motion control FB as a global label or a local label on the CPU module and calling the Motion control FB on the sequence program.



When using the Motion control FB of the CPU module side, the following memory areas are used. The number of available Motion control FB instances must not exceed the size of each area.

Memory location	Name	How to check the available size
CPU module	Label memory	Check by using the engineering tool
Motion module	Buffer memory (Area for Motion control FB)	Buffer Memory MCFB Communication Area Free Size (System.Md.BuffermemoryFreeMcfbArea)
	Memory for PLCopen motion control basic add-on	• RAM Usage Maximum (System.PrConst.Addon_MotionControl_General.RamSizeMax) • RAM Usage (System.Md.Addon_MotionControl_General.RamUsage)

Point

- The buffer memory (area for the Motion control FB) and the memory for the motion control add-on are hold when the Motion control FB is called for the first time after the CPU module is set to RUN. If the free space is in short, "Insufficient memory capacity (error code: 349AH)" will occur when the Motion control FB is called for the first time, and the Motion control FB will not be executed.
- Even if the Motion control FB of the CPU module side is deleted, the memory on the Motion module side is not released. Therefore, if the Motion control FB is added or deleted repeatedly by writing during RUN, the memory area of the Motion module side will be in short and "Memory shortage (error code: 349AH)" may occur. In this case, reset the CPU module once.

Operating Motion control FB

■ Operation conditions

The conditions (available conditions) to operate the Motion control FB of the CPU module side are shown below.

<table>
<tr><th rowspan="2">CPU module</th><th colspan="2">Motion module</th></tr>
<tr><th>RUN</th><th>Other than RUN</th></tr>
<tr><td>RUN</td><td>Available</td><td rowspan="2">Unavailable
the Motion control FB in execution is stopped</td></tr>
<tr><td>Other than RUN</td><td>Available*1</td></tr>
</table>

*1 This state occurs when PLC Ready (System.Cd.SequenceReady) is being RUN by operated directly, and the stop error is not occurring 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 following.
☞ Page 128 Relevant variables

■ Processing order

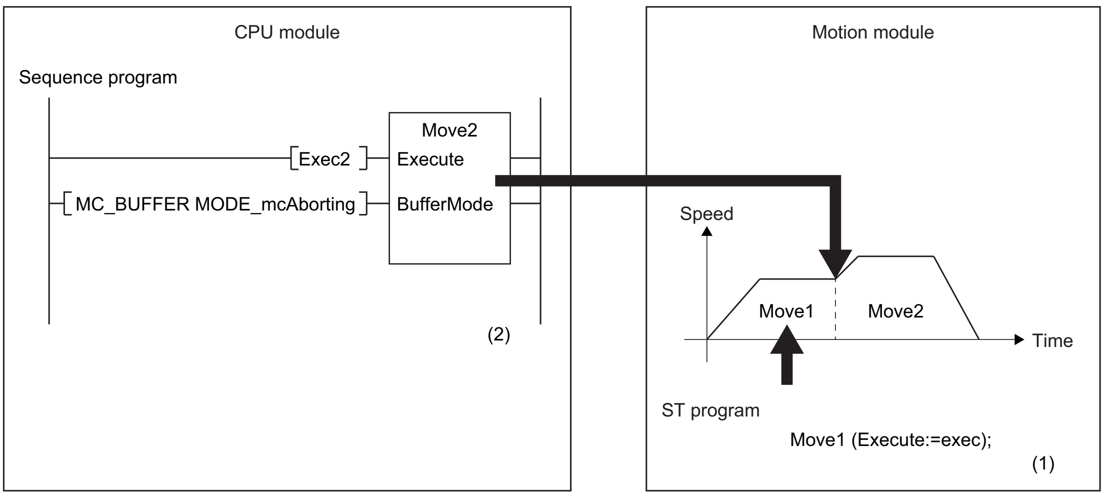
The processing order of the operation cycle in the Motion module corresponding to the Motion control FB instance of the CPU module side is the order when the Motion control FB is called for the first time.

■ Mix of the Motion control FB of CPU module side and the Motion control FB of the Motion module side

When the Motion control FB of the CPU module side and the Motion control FB 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 Motion control FB of the CPU module side and the Motion control FB of the Motion module side, these are treated as different control FBs in the Motion module.

Ex.

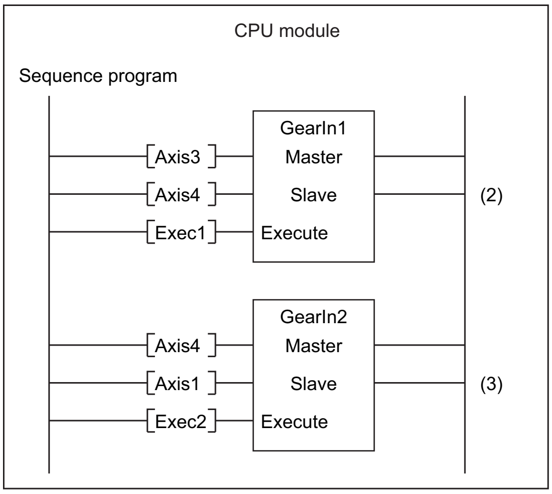
When executing Aborting to the on-going Motion control FB in the Motion module side by the Motion control FB of the CPU module side

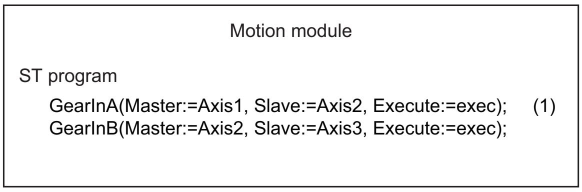


(1) Execute the Move1 of the Motion module side.
(2) When executing the Move2 of the CPU module side while the Move1 is in execution, the program is operated with the buffer mode specified by the Move2.

Ex.

When mixing the Motion control FBs of the single axis synchronous control in the Motion module side and the CPU module side





(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 and public variables

The refresh response time of I/O variables and public variables for the Motion control FB of the CPU module side is as shown below.

Motion control FB call cycle in the CPU module + Buffer memory refresh cycle × 2

■ Interlock of I/O variables when using Execute command (Execute) type FBs

For Execute command (Execute) type FBs, refer to "Basic operation of Execute command (Execute) type Motion control FBs" in the following manual.

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

Note that there is the difference in some of the specification as follows.

- Difference of the specification with the Motion control FBs of the Motion module side

  After the falling edge of Execute command (Execute), Abortion of execution (CommandAborted) is reset from 1to a few scans behind.

  When re-executing after the Motion control FBs have stopped, execute after confirming the reset of Executioncompletion (Done), Error (Error), and Abortion of execution (CommandAborted) have been performed.

■ Interlock of I/O variables when using Enabled (Enabled) type FBs

For Enable (Enable) type FBs, refer to "Basic operation of Enable (Enable) type Motion control FBs" in the following.

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

Precautions

For Motion control FBs, refer to precautions of the following.

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

The precautions only for the Motion control FB in a sequence program of the CPU module are shown below.

- When executing the Motion control FBs in the interrupt program of the CPU module, the Motion control FBs may not start. Use them with the following special relay ON.*1 Note that the completion signal of all dedicated instructions is maintained with ON at instruction completion by the following special relay ON so that the process which the completion signal is set to OFF should be created in a program when the program which is needed to set ON for only one cycle exists.

Number	Name
SM752	Dedicated instruction End bit control flag

*1 For details of the special relay, refer to the following.
MELSEC iQ-R CPU Module User's Manual (Application)

- When writing the Motion control FB in the control syntax such as the IF statement or the MC instruction, the Motion control FBs may not start.
  Do not write the Motion control FB in the control syntax.
- When using the Motion control FB in the CPU module side, do not execute the software reboot. If it is executed, turn on the power of the motion system again.
- The number of steps of FBs built into a program is different depending on the CPU module to use. the I/O definition, or Options setting of GX Works3. For Options setting of GX Works3, refer to the following.
  GX Works3 Operating Manual
- This FB uses the index register Z1. When using the interrupt program, do not use this index register.
  When the index register is used in the interrupt program, the FB may not operate correctly.
  When using the index register in the interrupt program, do not assign this index register as the local index register.
- Even if a warning of duplicated coil may occur in compiling, this is not particular problem in use.
- When using the subroutine type FBs in a ladder program, connect the I/O labels to both input and output sides.
- This FB uses the label initial value for each program. When specifying the program file using this FB by boot settings, specify the label initial value for each program by boot settings in the boot operation of the CPU module.
- If an Execute command (Execute) type Motion control FB is added by online change while Execute command (Execute) is TRUE, the added Motion control FB will not start. Change Execute command (Execute) from FALSE to TRUE after the online change is completed.

Point

- Errors (including warnings) of the CPU module side during Motion control FB execution are output as errors of the Motion module. For the error code output to Error code (ErrorID) of the Motion control FB, refer to the following.
  MELSEC iQ-R Programming Manual (Motion Control Function Blocks)
  However, when the I/O number (Page 715 Relevant variable) of the Motion module specified with the Motion control FB 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.
- The Motion control FBs do not include an error recovery process. Create the error recovery process separately according to the user setting system or required operation.
- The Motion control FBs do not detect the alarms and the warnings which occurred in the servo amplifier. Create the monitoring process of the alarms and the warnings which occurred in the servo amplifier separately. For the alarms and the warnings which occurred in the servo amplifier, refer to the manuals of the servo amplifier in use.

20 FILE CONTROL

20.1 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

System monitor data (System.Md.)

Variable/Structure name	Name	Details
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.

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	■When using RD78G 16M bytes ■When using RD78GH 64M bytes	■When using RD78G 8M bytes ■When using RD78GH 24M bytes	■When using RD78G 16M bytes ■When using RD78GH 32M bytes	■When using RD78G 240k bytes ■When using 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 Flash ROM Write Count Index Value (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 721 Drive
☞ Page 725 Storage files

Ex.

/sdc/$MOTPRJ$/calc_profile/cam0001.csv

- Relative path

  Paths that do not start with "/" are relative paths. 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 734 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 723 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	○	×
PRG.PRM	Program setting	○	×
/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	×
SimpleMotionRom.bin			For the Simple motion mode (Data saving of flash)	○

■ 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
AdvSyncInputList.json		Advanced synchronous input axis instance list
AdvSyncOutputList.json		Advanced synchronous output axis instance list

■ lch

File name	Usage
abs_axis.bin	Axis absolute position data
EVENT.LOG	Motion event history
SimpleMotionFram.bin	For the Simple motion mode (Data saving of FRAM)
abs_advsync.bin	Advanced synchronous control backup data

■ sdc

File name	Usage
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 725 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 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 737 Filing

2. If function such as the absolute position system which uses latch drive is being used, use Backup/Restore Request (System.Cd.BackupRestore) to stop the data update of the latch drive.
For details, refer to the following.
☞ Page 99 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 736 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 Backup/Restore Request (System.Cd.BackupRestore). For details, refer to the following.
☞ Page 99 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 Backup/Restore Request (System.Cd.BackupRestore) to stop the data update of the latch drive. For details, refer to the following.

☞ Page 99 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 736 Data copy / Data move

3. If the files in the latch drive are changed following procedure 2, execute restored data saving request by using Backup/Restore Request (System.Cd.BackupRestore). For details, refer to the following.

☞ Page 99 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.

20.2 Parameter Read/Write Function

This function reads/writes parameters.
It can read and write objects of the device stations.

Relevant FBs

For details on Motion control FBs, refer to the following.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

Motion control FB	Name	Description
MC_ReadParameter	Parameter read	Read objects of the device stations.
MC_WriteParameter	Parameter write	Write objects of the device stations.

Control details

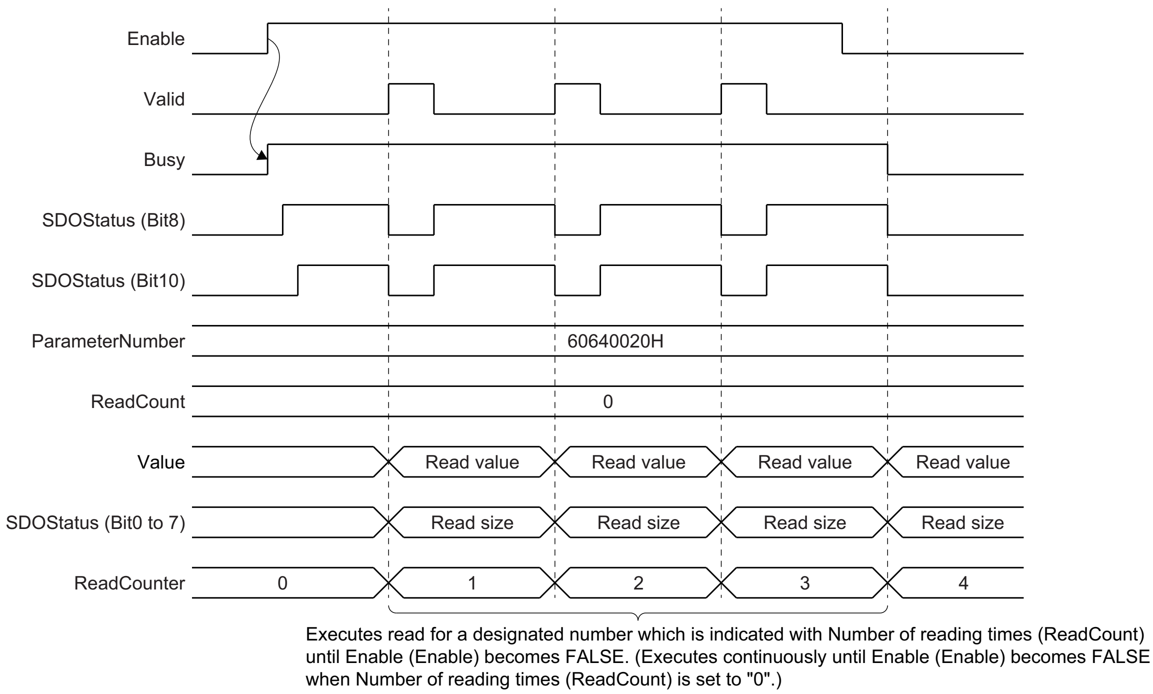
Slave object (Transient)

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 (response code (SDO Abort Code)) for transient request is received from the device station, Error (Error) becomes TRUE and then the error "SDO Communication Error (error code: 1800H)" is stored to Error code (ErrorID) and the response code (SDO Abort Code) from device stations is stored to Transient error code (SDOErrorID).

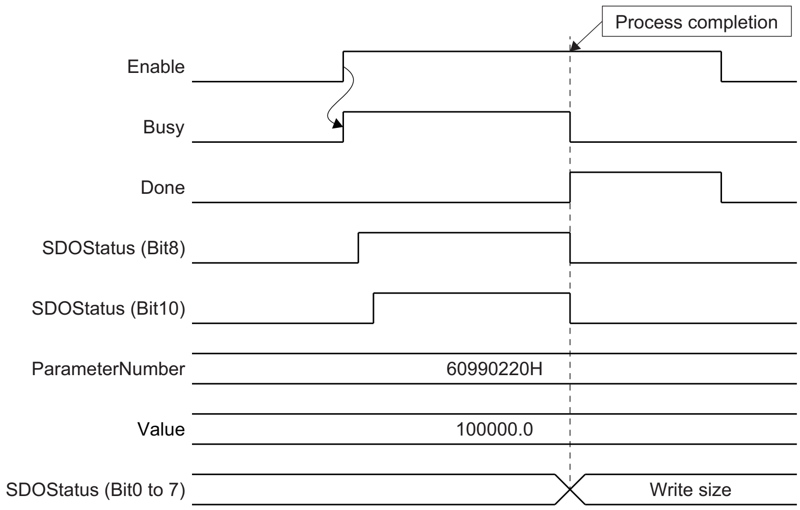
■ The timing of send/receive

- At successful completion

■For executing parameter read

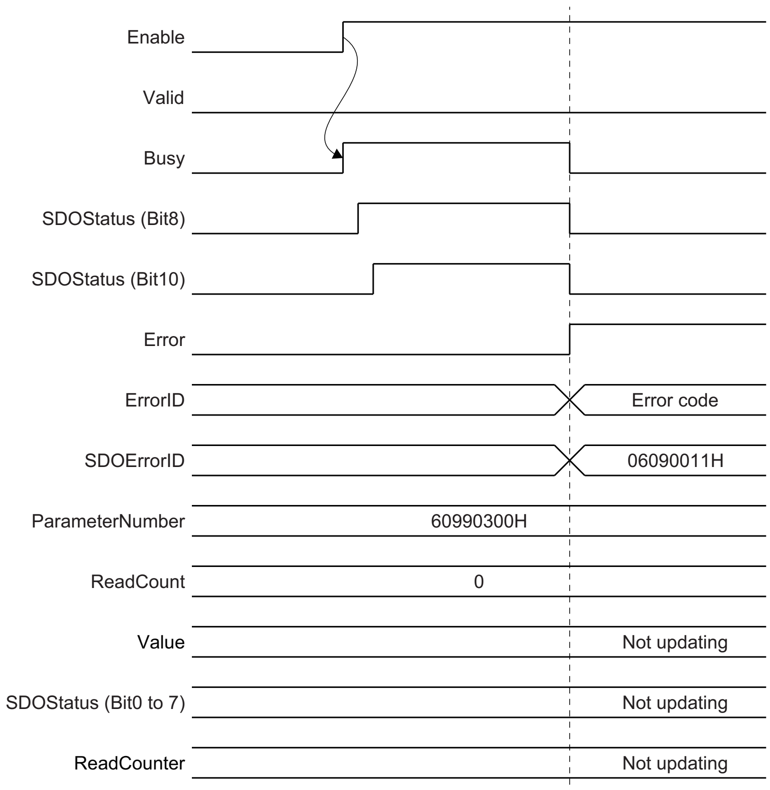


■For executing parameter write



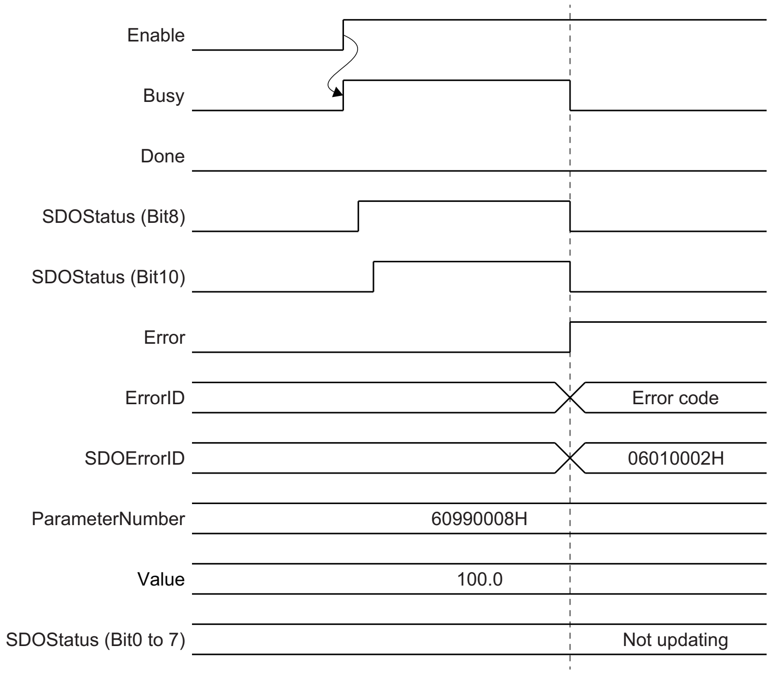
- At unsuccessful completion (SDO communication error has occurred.)

■For executing parameter read



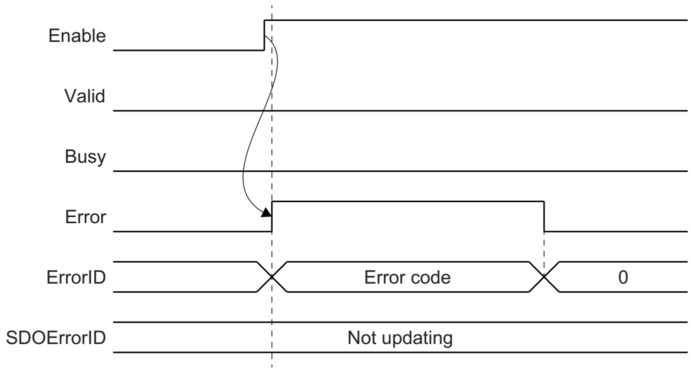
20

■For executing parameter write

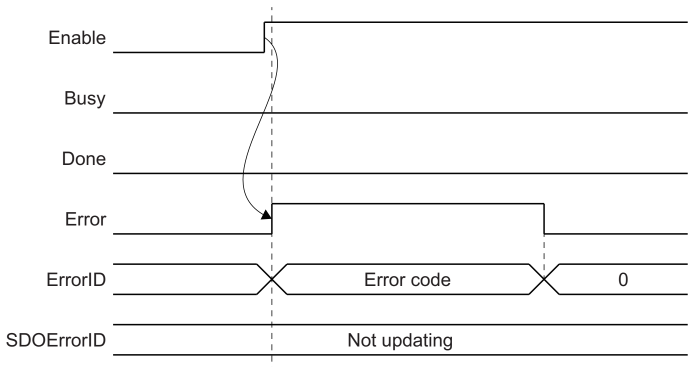


- At unsuccessful completion (the I/O variables and the Input variables are abnormal

■For executing parameter write



■For executing parameter write



The response code (SDO abort code)

For the response code by using the transient transmission function when the object data send/receive is unsuccessful, refer to "Response Code (SDO Abort Code)" in the following manual.

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

Precautions

- The transient transmission can be executed up to four toward the same axis. If MC_ReadParameter (Parameter Read)/ MC_WriteParameter (Parameter Write) for the slave object is executed toward the axis which is already executing four transient transmissions, "Parameter Reading And Writing FB Execution Disabled Error (error code: 34A1H)" occurs and the next FB is not executed. (The FB 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 (Homing) is carried out, "ABS Reference Point Read Error (error code: 1AAAH)" might occur in MC_Home (Homing).
- When reading/writing the integer type/BOOL type parameters or slave objects, the setting value of MC_WriteParameter (Parameter Write) is converted into integer type or BOOL type and taken. Also, the written value in MC_ReadParameter (Parameter Read) is respectively converted into LREA type or BOOL type and output.

Ex.

When writing parameters by specifying two bytes size with MC_WriteParameter (Parameter Write).

Setting value (Value) (Data type: LREAL)	Write value (2 bytes data)	
	Specifies unsigned (Options (Options): Bit 16) "0: Signed"	Specifies unsigned (Options (Options): Bit 16) "1: Unsigned"
10000.5	10000.0	10000.0
-10000.4	-10000.0	55536.0
65535.0	-1.0	65535.0
-65535.0	1.0	1.0
65536.0	0.0	0.0
131071.0	-1.0	65535.0

When reading parameters by specifying two bytes size with MC_ReadParameter (Parameter Read).

Parameter value	Read value (Value) (Data type: LREAL)	
	Specifies unsigned (Options (Options): Bit 16) "0: Signed"	Specifies unsigned (Options(Options): Bit 16) "1: Unsigned"
10000	10000.0	10000.0
-20000	-20000.0	45536.0
65535	-1.0	65535.0
65536	0.0	0.0
131071	-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 how to save parameters of device objects , refer to the manual for the device stations.
- When device stations in transient transmission are disconnected, all the transient requests for disconnected device stations are fail, and "SDO Communication Error (error code: 1800H)" occurs.

20.3 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 File Transfer Command (System.Cd.FileTransfer_Command) by setting TRUE in File Transfer Execution Request (System.Cd.FileTransfer_Execute).
While the file transfer is executed, File Transfer Executing State (System.Md.FileTransfer_State) is set to Executing (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

System parameter constant (System.PrConst.)

Variable/Structure name	Name	Details
FileTransfer_LogCapacity	File Transfer Log Capacity	Specifies the log file capacity in a unit of k byte. • -1 to 2048 [k bytes]
FileTransfer_AcFile_lch	File Transfer Access Control (Files in Latch Drive)	Specifies the access control to files in the latch drive. For details, refer to the following. ☞ Page 735 Access control setting
FileTransfer_AcFile_ram	File Transfer Access Control (Files in RAM Drive)	Specifies the access control to files in the RAM drive. For details, refer to the following. ☞ Page 735 Access control setting
FileTransfer_AcFile_rom	File Transfer Access Control (Files in User Drive)	Specifies the access control to files in the user drive. For details, refer to the following. ☞ Page 735 Access control setting
FileTransfer_AcFile_sdc	File Transfer Access Control (Files in SD Memory Card)	Specifies the access control to files in the SD memory card. For details, refer to the following. ☞ Page 735 Access control setting
FileTransfer_AcFile_sys	File Transfer Access Control (Files in System Drive)	Specifies the access control to files in the system drive. For details, refer to the following. ☞ Page 735 Access control setting
FileTransfer_AcLabel	File Transfer Access Control (Label)	Specifies the access control to labels. For details, refer to the following. ☞ Page 735 Access control setting

System monitor data (System.Md.)

Variable/Structure name	Name	Details
FileTransfer_State	File Transfer Executing State	Displays the execution state of file transfer. • 0: Waiting for Execution Request (Ready) • 1: Executing (Executing) • -1: Error Occurrence (Error_)

System control data (System.Cd.)

Variable/Structure name	Name	Details
FileTransfer_Execute	File Transfer Execution Request	Requests the execution of file transfer. When the file transfer is completed, FALSE is set automatically. • FALSE: Not executed • TRUE: Executes the file transfer
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 736 Command format

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", "Out of Options Range (error code: 1A4EH)" occurs, and the function will 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 specified file, a file in a specified 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 specified file, a file in a specified 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 specified 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.

Format
[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 723 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, File Transfer Executing State (System.Md.FileTransfer_State) turns "-1: Error Occurrence (Error_)" and the execution ends.
- When the command format is abnormal, File Transfer Executing State (System.Md.FileTransfer_State) turns "-1: Error Occurrence (Error_)" and the execution ends.
- For the operation to the file set with the read only attribute, refer to the following.
  ☞ Page 725 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
```

20

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 724 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. Target data: Label Argument: label • For the global label of the filing target, refer to "List of Variables" in the following. (The label whose attribute is LIST_WRITE_BACK is the filing target.) 📖MELSEC iQ-R Programming Manual (Motion Control Function Blocks) • Read permission to the target data is required.
Argument 2	• Specify the output destination of the filing data. • When Label is specified to argument 1, specify the path of the label default value file to be rewritten. (The file name cannot be omitted). • Outputting the file with the specified name is allowed as specifying the arbitrarily named file. • 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 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.
- When filing (backing up to the label default value file) the label value, the value set to the first element becomes the default value of all elements for array labels at reboot.

■ 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 which includes the file name after compression. • Specifies the following for the compressed file extension depending on the intended use. Compression format: Zip file format Extension: 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

- Compresses 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, File Transfer Executing State (System.Md.FileTransfer_State) turns "-1: Error Occurrence (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.

20

■ 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 File Transfer Log Capacity (System.PrConst.FileTransfer_LogCapacity) and the operation depends on the setting value as follows.

File Transfer Log Capacity (FileTransfer_LogCapacity)	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 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 File Transfer Execution Request (System.Cd.FileTransfer_Execute) and File Transfer Command (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%).

Relevant add-ons

The following add-on is required to use this function.
- FileTransfer

20.4 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	○

20

Relevant variables

System monitor data (System.Md.)

Variable/Structure name	Name	Details
Storage_sdc	SD Memory Card Information	Displays the SD memory card information. SD Memory Card Information (Storage_sdc) is a STORAGE_INFORMATION structure
Storage_sdcProtected	SD Memory Card Protected	Displays whether the write protect switch of the SD memory card is ON/OFF. • FALSE: Write protect switch OFF • TRUE: Write protect switch ON
Storage_sdcInserted	SD Memory Card Inserted	Displays whether the SD memory card is inserted. • FALSE: Not inserted • TRUE: Inserted
Storage_sdcForcedDisabled	SD Memory Card Force Disabled	Displays whether the usage of SD memory card is being forced stopped. • FALSE: Not forced stopped • TRUE: Force stopped

System control data (System.Cd.)

Variable/Structure name	Name	Details
Storage_sdcRemovalProhibit	SD Memory Card Removal Prohibition	Specifies whether to prohibit removal of the SD memory card. • FALSE: Removal allowed • TRUE: Removal prohibited
Storage_sdcForcedDisable	SD Memory Card Force Disabled Command	Specifies whether to restrict usage of the forced stop for the SD memory card. • FALSE: Forced stop release • TRUE: Forced stop

STORAGE_INFORMATION

Variable/Structure name	Name	Details
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.
- The SD memory card is handled as the drive in the internal of the motion system and the engineering tool. For details on the SD memory card, refer to the following.
  ☞ Page 720 Memory and File

Installation and removal of SD memory card

For details, refer to "Installation and Removal of SD Memory Card" of Part 1 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.

21 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
Security key authentication	Prevents unauthorized viewing of a program in the engineering tool. Also prevents unauthorized program execution in a motion system. (By using a security key.) A program locked by a security key can only be executed with a motion system that has the same security key set to it.
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 details, refer to "Security" in the following manual. 📖MELSEC iQ-R Motion Module User's Manual (Network)
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 details , refer to "Security" in the following manual. 📖MELSEC iQ-R Motion Module User's Manual (Network)

Operation of this function for each system status

○: Possible

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

21.1 Security Key Authentication Function

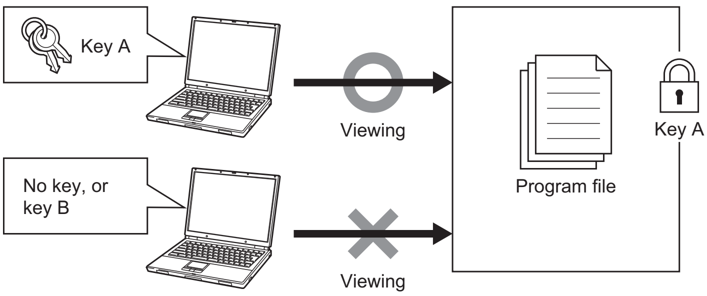
The security key authentication function is used to protect the user data by setting a common security key to the project and motion system to restrict the personal computer which operates the project and motion system which runs the project.
The security key is created in an engineering tool and registered to the personal computer, or motion system.
Security keys are set for the following.

Security key target		Application
Program file	Security keys can be registered for each program file, however, the same security key is used for all program files in a project.	• Restricts the viewing of program files inside the project. • Restricts the execution of programs with motion systems.
Personal computer	Multiple security keys can be registered to a single personal computer.	Protected program files can be viewed with a security key.
Motion system	Only one security key can be written per motion system.	Restricted programs can be executed by security keys.

Operations that can be restricted with security key authentication function

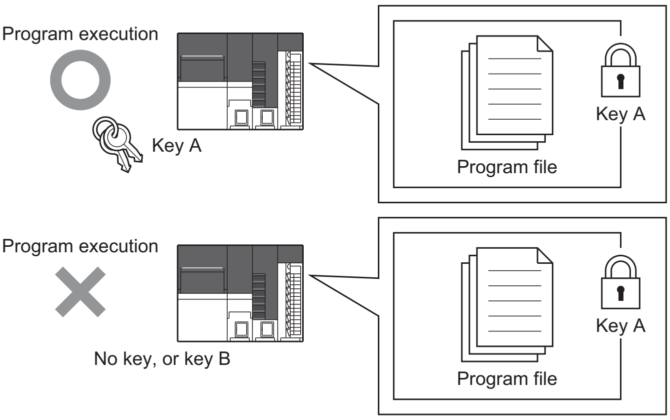
■ Program viewing

Viewing a program on an engineering tool is controlled based on whether the security keys for program files and the personal computer match.



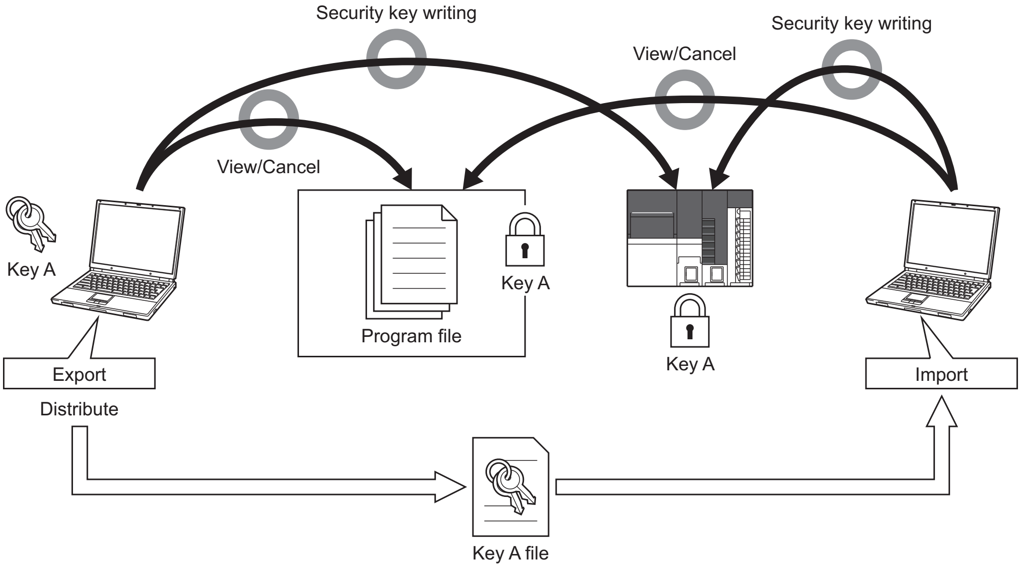
■ Program execution inside motion system

Program execution in the motion system is controlled based on whether the security keys of program files written to the motion system and the motion system security key match.



Copying security keys

Security keys registered to a personal computer can be imported or exported using an engineering tool for use at another personal computer. Expiry dates can be set for copied security keys.



Point

- When the project data or the security key set to a motion system cannot be imported in an engineering tool, the security key cannot be canceled with repairs or maintenance. Thus the project data can never be viewed or edited. Mitsubishi Electric accepts no responsibility for losses to customers, other individuals, or organizations as a result of not being able to view or edit project data.
- To prevent losses occurring when a personal computer at which security keys are registered malfunctions, create backups by importing registered security keys to another personal computer, or storing exported files containing registered security keys in a safe location.

Control details

The operations of the security key authentication function and the control details are shown below.

Operation target	Item		Details
Security key	Security key creating/deleting	Security key creating	Generates a security key with an engineering tool.
		Security key deleting	Deletes the security key from an engineering tool.
	Security key copying	Security key exporting	Writes a security key that can be used in the engineering tool of another personal computer to a file.
		Security key importing	Registers an exported security key to an engineering tool.
Project	Registration of security key to program file	Registration of security key to program file	Locks specified project data with a security key.
		Deleting security key of program file	Cancels the security key lock of specified project data.
Motion system	Writing/deleting of security key to motion system	Writing of security key to motion system	Writes a security key to a motion system.
	Deleting of security key	Deleting security key of motion system	Deletes a security key of a motion system.
	Checking security key information		Checks the information (key name/date of generation) of the security key registered to the motion system.
	Preventing unauthorized program execution		Prevents unauthorized program execution by verifying the security key written to the motion system, and the security key of the program file.

Creating/deleting the security key

The creating/deleting of a security key is executed on the security key management screen of an engineering tool.

For details of the operation procedures, refer to help of an engineering tool.

■ Creating the security key

Creates a new security key in the security key management screen of an engineering tool and registers it to a personal computer.

The information that the security key contains is shown below.

Item	Details
Name	The name of the security key (32 characters).
Date of generation	The date the security key was generated in an engineering tool.
Authentication code	The information for verifying the validity of the security key during security key authentication.
Password for importing*1	The password to be input when importing the security key.
Operation restrictions*1	Operation restrictions • Availability to re-export the security key • Availability to change the lock status of project data • Availability to write the security key to a motion system
Expiry date*1	Time period that the security can be used.

*1 Can be added when exporting. Newly created keys do not have operation restrictions and expiry dates. For operation restrictions and expiry dates, refer to the following.

☞ Page 747 Copying the security key

■ Deleting the security key

Deletes the security key registered to the personal computer.

Point

When a security key registered to a personal computer is deleted without being exported, the project data that is locked by that security key can no longer be viewed or edited. Before deleting a security key on a personal computer, cancel the lock on the project data.

Copying the security key

Security keys can be copied (exported/imported) and used on a personal computer other than the personal computer on which the security key was generated.

Security keys are copied (imported/exported) on the security key management screen of an engineering tool.

For details of the operation procedures, refer to help of an engineering tool.

■ Exporting the security key

Exports the security key registered to a personal computer as a file that can be imported.

An expiry date and operation restrictions can be added to the security key to be exported.

- Expiry date

  When the expiry date elapses on an engineering tool, the security key becomes invalid. Programs can no longer be viewed and the security key cannot be written to a motion system. Export/import the security key again. When the expiry date of a security key in a motion system elapses, the unauthorized execution of a program is still prevented.
- Operation restriction

  Restricts the operations available on the engineering tool of the personal computer that the security key is imported. For details of the operation procedures, refer to help of an engineering tool.

21

■ Importing the security key

Imports the exported security key file to a personal computer and registers the security key.

Registering the security key to the program file

Security keys are registered to/deleted from a program file on the security key management screen of an engineering tool.

For details of the operation procedures, refer to help of an engineering tool.

The name and date of generation of the security key registered to a program file in a motion system are displayed in the security key management screen of an engineering tool.

■ Registering the security key to the program file

Registers the security key to a program file.

The user can select whether to register a security key to each program file or not. However, all program files in a project will have the same security key.

The files to which a security key can be set, and the details of protection are shown below.

For the other files used for control in a motion system, refer to the following.

☞ Page 725 Storage files

Data type	File extension	Details of protection
Motion program file	*.PRG	• Restricts viewing of program files in the project. • Restricts execution of programs in the motion system.
FB/FUN file	*.PFB	

■ Deleting the security key from the program file

Deletes the security key registered to the program file.

Writing/deleting the security key to the motion system

A program can be executed by writing the same security key as the program file to the motion system.

The security key is written to/deleted from the motion system with the "Security Key Setting" screen of the engineering tool.

For details of the operation procedures, refer to help of an engineering tool.

A security key that is written to a motion system can be deleted from a personal computer that does not have the security key registered to it.

Point

- When a motion system security key is deleted, the data written to the motion system is still locked and the program cannot be executed. Project data with the lock canceled must be written to the motion system again.
- The writing/deleting of the security key of the motion system is registered in the event history.
- The name and date of generation of the security key in the motion system can be checked in the "Security Key Setting" screen of the engineering tool.

Preventing the unauthorized execution of programs

When there is a program with a security key registered in the motion system, the security key of the motion system and the security key of the program file are verified when PLC READY turns OFF to ON.

When the security keys of all program files match the security key of the motion system, the program can be executed.

When the security key of the program file does not match the security key of the motion system, or a security key is registered to the program file but there is no security key in the motion system, "Security key authentication error" (error code: 2300 to 2302H) occurs, and the motion system does not start.

The motion system operations depending on whether a security key exists or not are shown below.

Security key in the motion system	Security key in the program file	Security key match/mismatch	Ability to execute the program
Not written	Not registered	—	Program can be executed
	Registered	—	Security key authentication error occurs and program cannot be executed
Written	Not registered	—	Program can be executed
	Registered	Match	
		Mismatch	Security key authentication error occurs and program cannot be executed

Precautions

Program asset outflow cannot be prevented if a personal computer at which a security key is set is misused by a third party, and therefore customers must take the following countermeasures to protect against this.

- Personal computer theft prevention measures (use of wire locks, etc.)
- Personal computer user management (deletion of redundant user accounts, strict control of login information, adoption of fingerprint authentication, etc.)

Furthermore, if a security key is lost, locked project data cannot be viewed or edited. Mitsubishi Electric accepts no responsibility for losses to customers, other individuals, or organizations as a result of this. Customers must therefore take the following countermeasures to protect against this.

- Import registered security keys to another personal computer.
- Store exported files containing registered security keys in a safe location.

Version combinations

The security key authentication function supported versions are shown below.

This function can be used when the following versions of the engineering tool and motion system software are combined.

Engineering tool version		Motion system software version
GX Works3	Motion control setting function	
"1.087R" or later	"1.035M" or later	"24" or later

The operations for version combinations of the engineering tool and the motion system are shown below.

Writing/deleting the security key to the motion system

The availability of writing/deleting the security key of the motion system using the engineering tool is shown below.

Engineering tool version		Motion system software version	Details
GX Works3	Motion control setting function		
"1.087R" or later	"1.035M" or later	"24" or later	A security key can be written/deleted.
		Earlier than "24"	A security key cannot be written as the motion system is unsupported.
	Earlier than "1.035M"	"24" or later	A security key cannot be written as the engineering tool is unsupported.
		Earlier than "24"	A security key cannot be written as the engineering tool is unsupported.

Writing the project data to the motion system

The availability of writing/deleting the project data to the motion system when there is a program with a security key registered by the engineering tool is shown below.

Engineering tool version		Motion system software version	Details
GX Works3	Motion control setting function		
"1.087R" or later	"1.035M" or later	"24" or later	Project data can be written.
		Earlier than "24"	Project data cannot be written by an engineering tool.
	Earlier than "1.035M"	"24" or later	Project data cannot be written as the project data cannot be opened by an engineering tool.
		Earlier than "24"	

Reading the project data from the motion system

The availability of reading the project data with an engineering tool when there is a program with a security key registered in the motion system is shown below.

Engineering tool version		Motion system software version	Details
GX Works3	Motion control setting function		
"1.087R" or later	"1.035M" or later	"24" or later	Project data can be read.
		Earlier than "24"*2	
	Earlier than "1.035M"*1	"24" or later	Project data cannot be read as the engineering tool is unsupported.
		Earlier than "24"*2	

*1 Applies to when an engineering tool is downgraded to an unsupported version after writing a program file with a security key registered by a supported version of an engineering tool.

*2 Applies to when a motion system is downgraded to an unsupported version after writing a project file with a security key set by a supported version of a motion system.

Preventing the unauthorized execution of programs

The operations for when a security key is set to a program file and not, and the version of the motion system are shown below.

Program file	Software version of the motion system	Details
Security key set	"24" or later	Program can be executed.
	Earlier than "24"	Program cannot be executed.
No security key set	"24" or later	Program can be executed.
	Earlier than "24"	

22 MOTION MODULE SOFTWARE INSTALLATION

This chapter describes how to install software of the motion system.
Various software can be updated at once by installing the motionsystem software.
If needed, boot software and software can also be installed individually.
However, when installing the software which version is "07 or earlier", install it after installing boot software.

22.1 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 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, "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.
  If the combination of the software version does not match, "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.
  ☞ Page 879 List of Add-on Library
  ☞ Page 884 List of Boot software

Operation of this function for each system status

○: Possible, △: Possible (Restricted)

Status	Operation availability
STOP	○
RUN	○
Moderate error	○
Major error	△ (Operation may not be possible according to the error status)

Relevant variables

System monitor data (System.Md.)

Variable/Structure name	Name	Details
Version_BootSw	Boot Software Version	Displays the version of the boot software
Version_NetworkBootSw	Network Boot Software Version	Displays the version of the network boot software
Version_BaseSystemSw	Basic System Software Version	Displays the version of the basic system software

Software file configuration of motion system

The configuration of motion system software downloaded from MITSUBISHI ELECTRIC FA Global Website is shown below.

The following files are packed in the motion system software (sw12dnn-rmtufw_□.zip).

File		
Motion system software for RD78G(H) (sw12dnn-rmtufw_□.zip)*1		
	Motion system software package (RD78GSW.smpk)	
	instnew	
		Basic system software (baseSystem.smpk)
		Each add-on library (.mpk)
	instboot	
		Boot software(.img)

*1 "□" is the version of the Basic system software.

The following files are packed in the motion software package (RD78GSW.smpk).

File	
Motion system software for RD78G(H) (RD78GSW.smpk)	
	Basic system software
	Each add-on

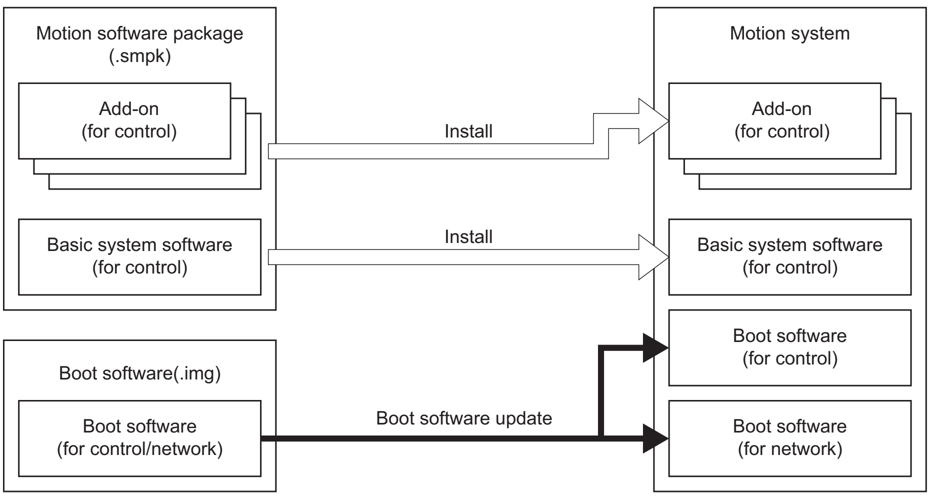
At installing, expanding is performed as follows on the motion system.

File	
/sys drive	
	Basic system software
	Each add-on

Boot software will not be recognized as a drive.

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



How to batch install the motion system software

This section describes how to batch install motion system software.

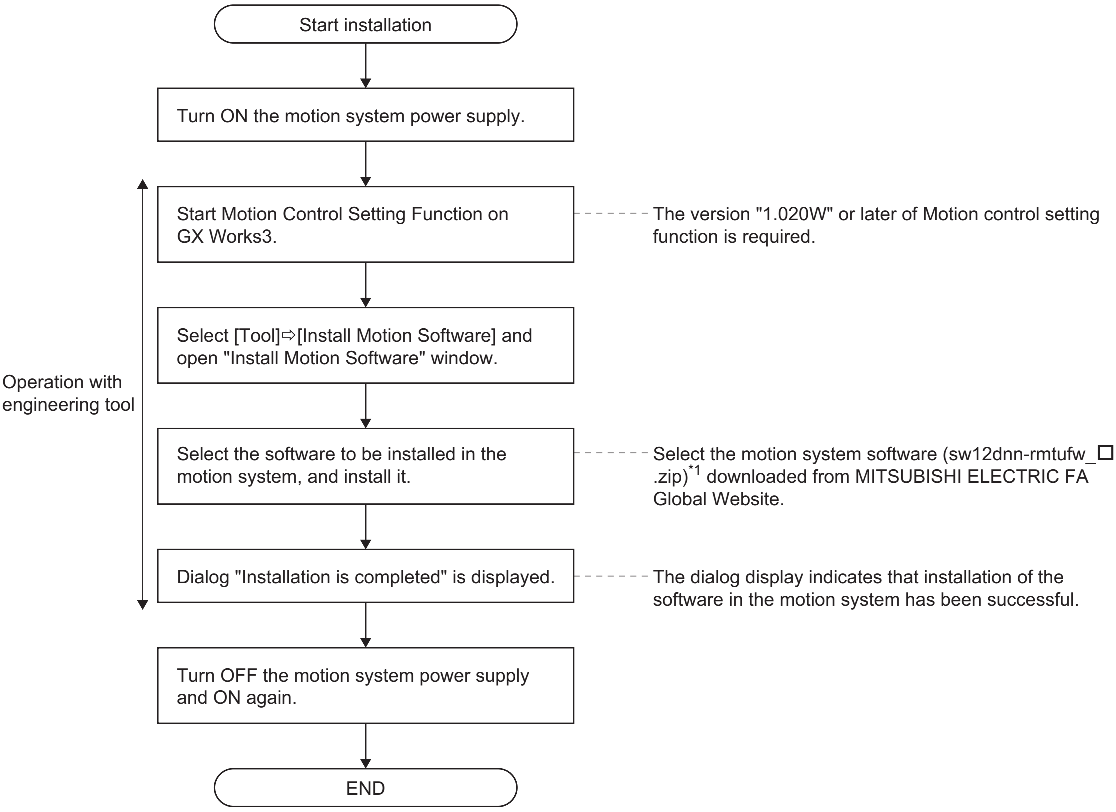
- When installing the motion system software from the motion system which version is "16" or later, the methods can be selected either installing from the engineering tool or installing from the SD memory card.
- When installing the motion system software from the motion system which version is "14" or earlier, install from the SD memory card.

Installation procedure using engineering tool

The batch installation procedure of the motion system software using engineering tool is shown below.

Point

When installing from the engineering tool, the version "1.020W" or later of Motion control setting function is required.



*1 "□" is the version of the Basic system software.

Install the software according to the instructions shown on "Motion Control Setting Function" window on GX Works3. For details, refer to Help of Motion Control Setting Function on GX Works3.

Installation procedure using SD memory card

The batch installation procedure of the motion system software using SD memory card is shown below.

- To install the version of motion system software, "08" or later, install the software and the boot software simultaneously by batch installation. When the batch installation is executed, the separate installtation 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 separately because the batch installation is not supported. For the installation, refer to the following.

☞ Page 758 Basic System Software Update Function
☞ Page 761 How to update the boot software

Point

To install the software separately, update the boot software first, and then install the software.

Start installation

1. Using a personal computer, store the motion system software file in the SD memory card. — Open the motion system software (.zip) downloaded from MITSUBISHI ELECTRIC FA Global Website, and copy [instnew] and [instboot] folders in the SD memory card root.
2. Insert the SD memory card in the Motion module SD memory card slot.
3. Turn ON the motion system power supply while pressing the SD memory card access control switch. — System detects that the switch is pressed down when the motion system power supply turns ON, and automatically transits to installation mode.
4. Release the SD memory card access control switch about 3[s] later after power turns ON. — Once the installation mode starts, it is continued even if a hand is released from the switch. Both the RUN LED and the PROGRAM RUN LED flash (every 500 [ms]) during installation.
5. The RUN LED turns off, and the PROGRAM RUN LED and the CARD READY LED turn on. — The RUN LED turns off, and the PROGRAM RUN LED and the CARD READY LED turn on after completing the installation.
6. Turn OFF the motion system power supply.

END

Point

- Use an SD memory card formatted by the CPU module. When an SD memory card formatted by the Windows format function is used, the motion system software installation may fail.
- When formatting an SD memory card using the CPU module, use the CPU memory operation function in GX Works3. For details on operation, refer to the following.
  GX Works3 Operating Manual
- 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 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.
- 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 and the synchronization flag do not turn ON during installation execution from the SD memory card.

22

■ LED displays during motion system software installation

The following shows the LED displays during motion system software installation.

Motion system status	LED display			Description	Action
	RUN	PROGRAM RUN	CARD READY		
Normal	Flashing (every 500 ms)	Flashing (every 500 ms)	OFF	Executing the installation.	Wait until the installation is completed.
	OFF	ON	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.
Error	OFF	Flashing (every 200 ms)	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 Basic System Software Version (System.Md.Version_BaseSystemSw) or the engineering tool.

■ Checking with the engineering tool

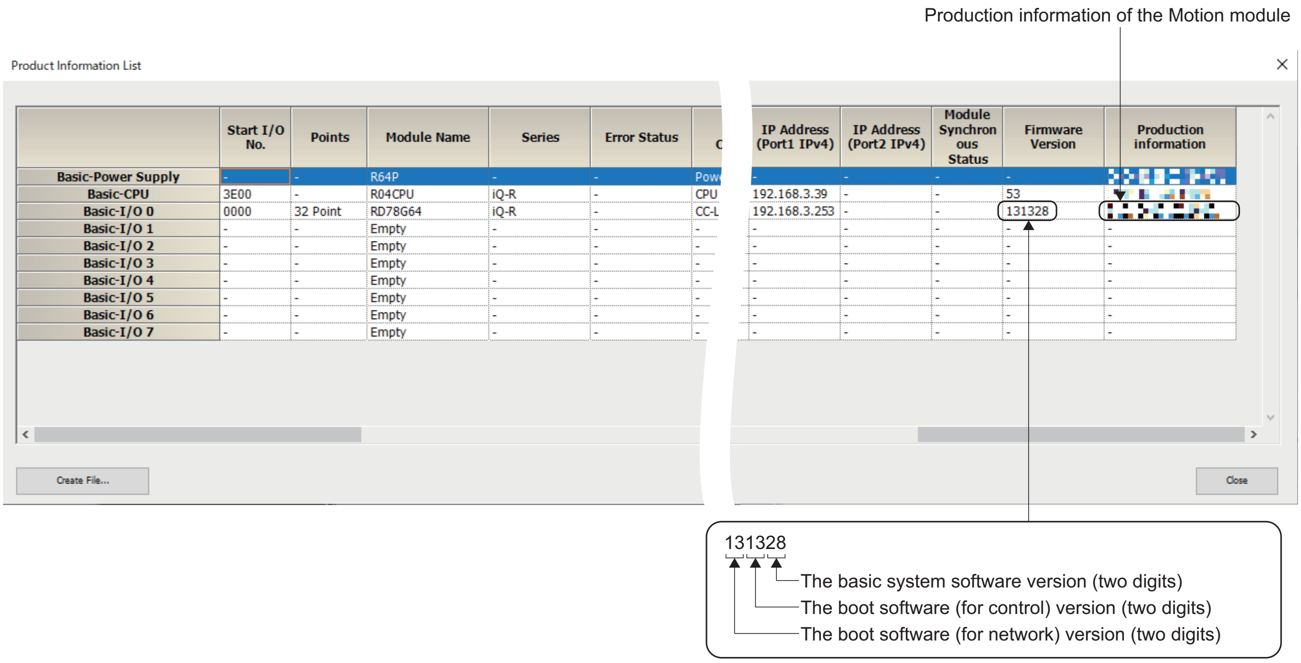
Check on the product information list screen.

[Diagnostics] menu ⇨ [System monitor] ⇨ [Product Information List] button

The basic system software and boot software (for control / for network) versions are displayed.

Point

When the version of motion system software is "07" 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

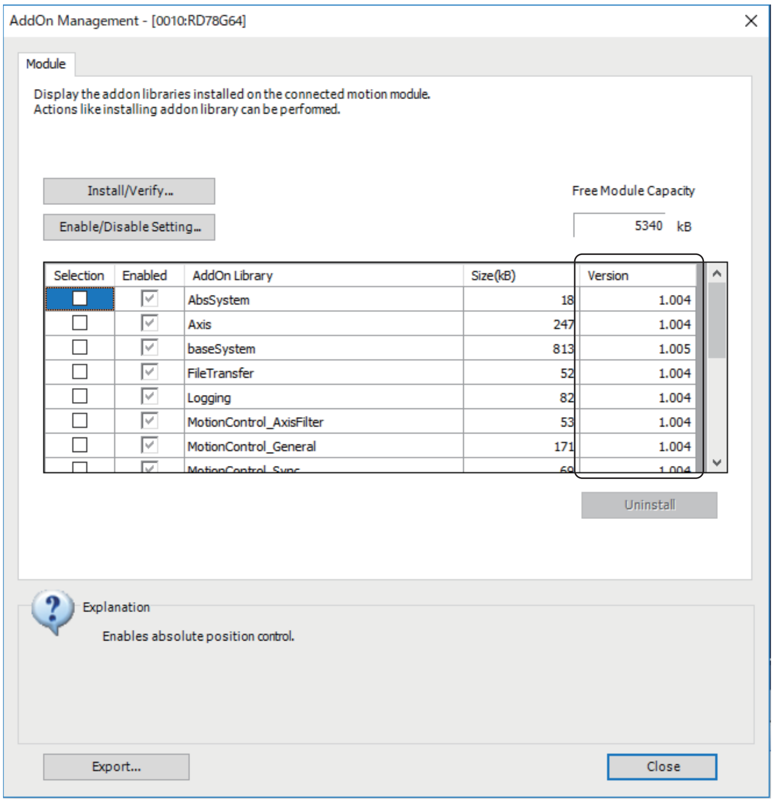
Check the add-on version on the engineering tool.

■ Checking with the engineering tool

Check on the add-on management screen.

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ [Target module] ⇨ [Module Extended Parameter] ⇨ [Motion Control Setting Function] ⇨ [Tool] menu ⇨ [Add-on Management]

The add-on version is displayed.



Checking the boot software version

The boot software version can be checked with Boot Software Version (System.Md.Version_BootSw).

Precautions

Relevant add-ons

This function can be performed without add-ons.

System memory capacity

System memory is not required for the software installation.

22

22.2 Basic System Software Update Function

How to install the basic software

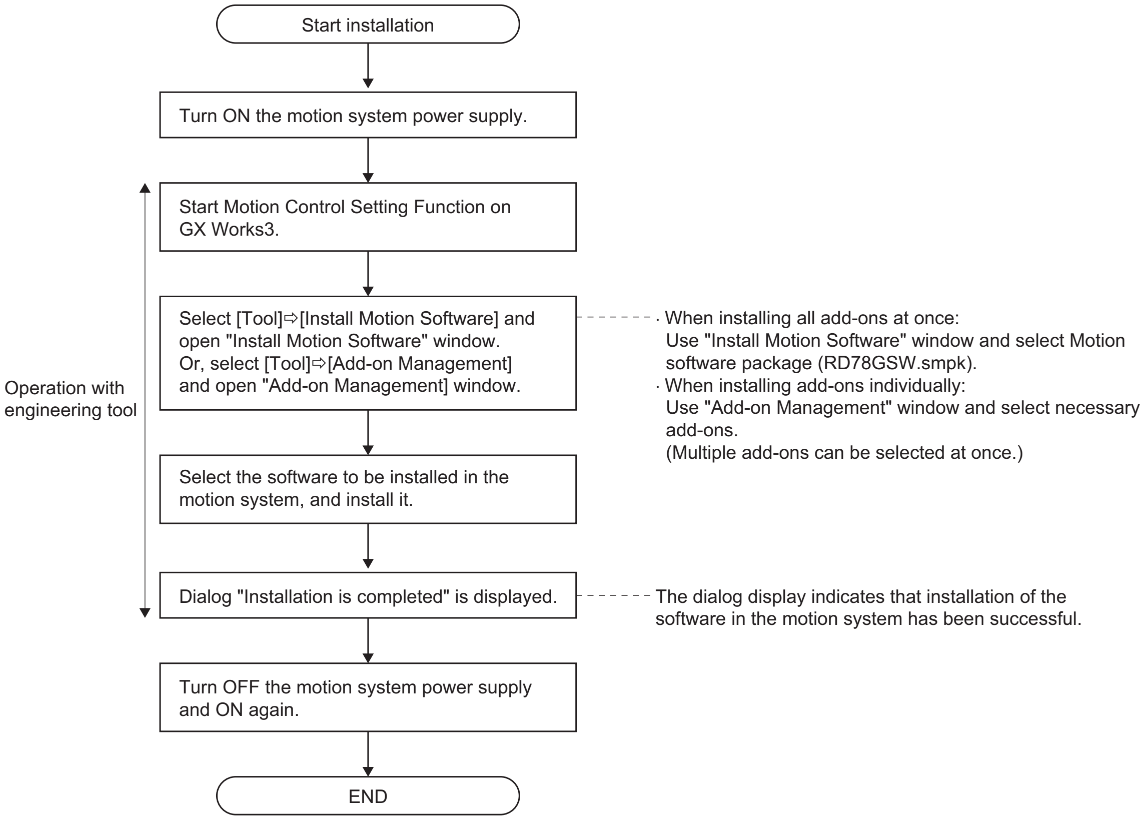
The procedure for installing the software is shown below.

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 743 SECURITY

Installation procedure using the engineering tool

The following shows a procedure to install the motion software by using the engineering tool.



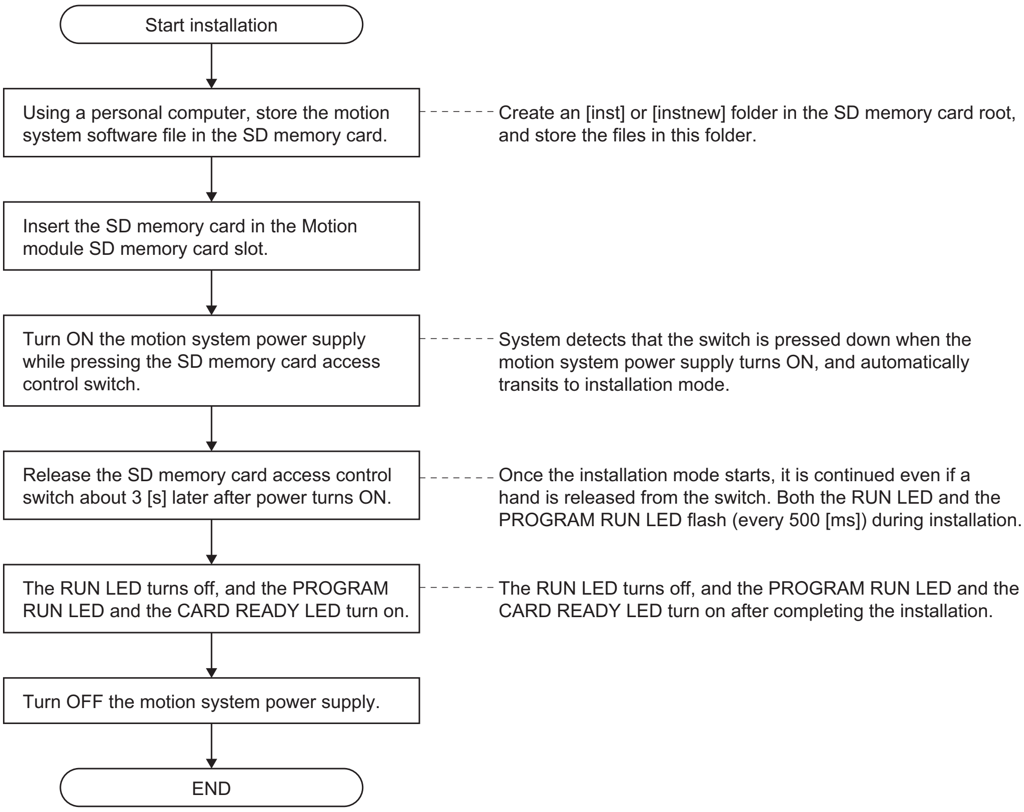
Install the software according to the instructions shown on "Motion Control Setting Function" window on GX Works3. For details, refer to Help of Motion Control Setting Function on GX Works3.

Point

- The installed files can be verified with the files on the personal computer by using Motion Control Setting Function on GX Works3.
- 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.



Point

- The motion software creates the [inst] folder or [instnew] folder in the root of the SD memory card. Store the file in the folder.
  (1) [instnew] folder: When newly installing the file (The file is transmitted after the system drive is formatted.)
  (2) [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.
  (1) When reading the SD memory card is abnormal, check the SD memory card.
  (2) When the installation file is abnormal or when it cannot be recognized, check the installation file.
  (3) 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 and the synchronization flag do not turn ON during installation execution from the SD memory card.

■ LED displays during basic system software installation

The following shows the LED displays during basic system software installation.

Motion system status	LED display			Description	Action
	RUN	PROGRAM RUN	CARD READY		
Normal	Flashing (every 500 ms)	Flashing (every 500 ms)	OFF	Executing the installation	Wait until software installation is completed.
	OFF	ON	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.
Error	OFF	Flashing (every 200 ms)	Flashing (every 200 ms)	The installation is completed with an error.	Turn the power of motion system OFF to ON, or reset the system, then execute software installation again.

22.3 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 checked with Boot Software Version (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.

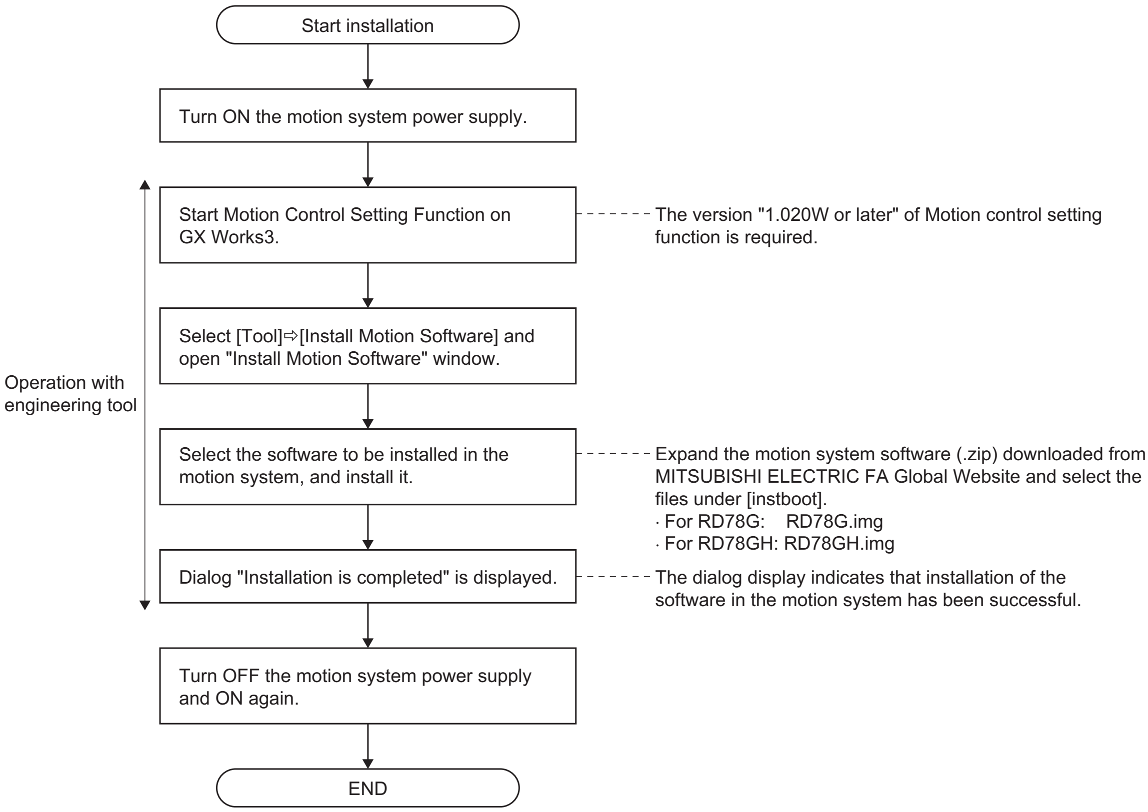
How to update the boot software

The method to update boot software is shown below.

- When installing the motion system software from the motion system which boot software version is "07" or later, the methods can be selected either installing from the engineering tool or installing from the SD memory card.
- When installing the motion system software from the motion system which boot software version is "06" or earlier, install from the SD memory card.

Installation procedures using the engineering tool

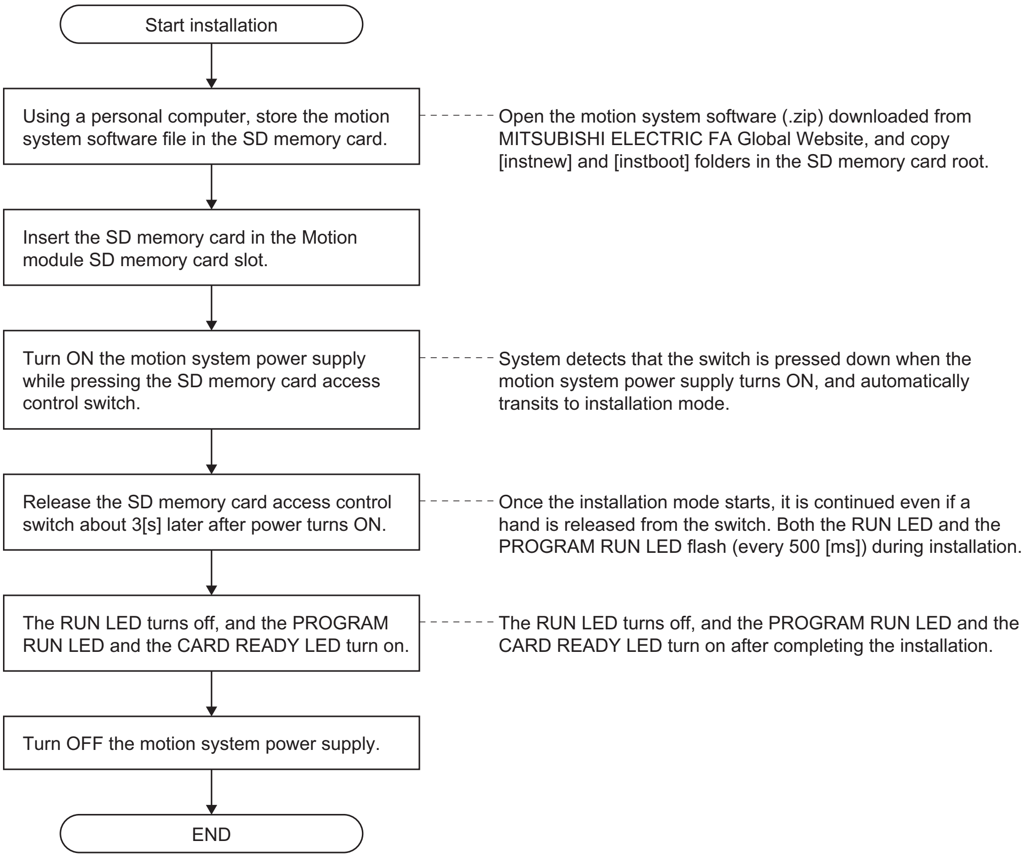
This section describes the batch installation procedure of the boot software using engineering tool.



Install the software according to the instructions shown on "Motion Control Setting Function" window on GX Works3. For details, refer to Help of Motion Control Setting Function on GX Works3.

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.
  (1) When reading the SD memory card is abnormal, check the SD memory card.
  (2) 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 might 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 restore the boot software, follow the procedure above after temporarily changing the rising edge time in the entire system to 15 s or less in the system configuration.
- 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 and the synchronization flag do not turn ON during installation execution from the SD memory card.

■ LED displays during boot software update

The following shows the LED displays during boot software update.

Motion system status	LED display			Description	Action
	RUN	PROGRAM RUN	CARD READY		
Normal	Flashing (every 500 ms)	Flashing (every 500 ms)	OFF	Updating the boot software	Wait until boot software update is completed.
	OFF	ON	ON	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.
Error	OFF	Flashing (every 200 ms)	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 version

For the boot software versions, refer to the following.

☞ Page 884 List of Boot software

23 TROUBLESHOOTING

23.1 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.

☞ 22 MOTION 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 normall • 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[*2] • OFF: Not communicating data[*2]
L ER LED	Displays the port status. ■When using RD78G • ON: Abnormal data received • OFF: Normal data received ■When using RD78GH <When using add-on baseSystem version "1.28" or earlier> • ON: Abnormal data received • OFF: Normal data received <When using add-on baseSystem version "1.30" or later> • ON: Abnormal data received or loopback being performed[*3] • OFF: Normal data received and loopback not performed
LINK LED	Displays the link status. • ON: Link-up • OFF: Link-down

*1 Includes the cyclic transmission data and the transient transmission data of CC-Link IE TSN.
When multiple errors occur, the error status is displayed in the order of major, moderate, and minor.

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.

*2 Error status can be determined by status of the RUN LED and the ERR LED.

*3 If a station for which loopback is being performed is a station whose station number has not been set, the L ER LED of that station does not turn on.
If a station whose station number has not been set enters the ring structure, the L ER LED of the loopback station does not turn on.

23.2 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 683 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.

Operation of this function for each system status

○: Possible

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

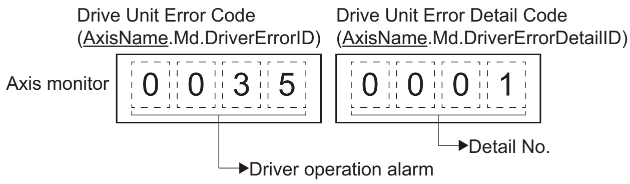
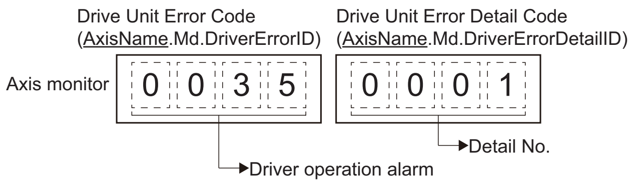
Overall block diagram

Refer to the following.

☞ Page 683 History Data

Relevant variables

Axis monitor data (AxisName.Md.)

Variable/Structure name	Name	Details
Warning	Axis Warning Detection	Becomes TRUE at axis warning occurrence.
Error	Axis Error Detection	Becomes TRUE at axis error occurrence.
DriverError	Drive Unit 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 Unit Error Code	Stores upper 16 bits of "Current alarm (2A41H)" of the slave object at drive unit error occurrence. <Example> For MR-J5(W)-G  • Stores "0035H" when servo alarm [AL.035. 1_Command frequency error] occurs at drive unit.
DriverErrorDetailID	Drive Unit Error Detail Code	Stores lower 16 bits of "Current alarm (2A41H)" of the slave object at drive unit error occurrence. <Example> For MR-J5(W)-G  • Stores "0001H" when servo alarm [AL.035. 1_Command frequency error] occurs at drive unit.

Axes group monitor data (AxesGroupName.Md.)

Variable/Structure name	Name	Details
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.

System parameter constant (System.PrConst.)

Variable/Structure name	Name	Details
ExcludeWarning	Excluded Warning	Sets the warning which is not to be detected. <Example> • "": Not set • Not detect the warnings of "0x1000, 0x1001": 0x1000, 0x1001. • Not detect the warnings from "0x1000-0x1010": 0x1000 to 0x1010.

System monitor data (System.Md.)

Variable/Structure name	Name	Details
Warning	Motion Area System Warning Detection	Becomes TRUE at warning occurrence.
Error	Motion Area System Error Detection	Becomes TRUE at error occurrence.
NetworkError	Network Area Error Detection	Becomes TRUE at network error occurrence.
WarningID	Latest Motion Area System Warning Code	Stores the latest warning code.
ErrorID	Latest Motion Area System Error Code	Stores the latest error code.
NetworkErrorID	Network Area Error Code	Stores the error code at network error occurrence.

Control details

The latest error code occurred in motion is output in Latest Motion Area System Error Code (System.Md.ErrorID).
When multiple errors occur, follow the conditions below whether the error code output to Latest Motion Area System Error Code (System.Md.ErrorID is updated).

Occurring error	Occurred error					
Latest Motion Area System Error Code (System.Md.ErrorID)	Axis Error Code (AxisName.Md.ErrorID)/Axes Group Error Code (AxesGroupName.Md.ErrorID)/Latest Motion Area System Error Code (System.Md.ErrorID)			Network Area Error Code (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

Excluded warning setting

- The specified warning can be ignored by setting Excluded Warning (System.PrConst.ExcludeWarning).
- When the warning set to Excluded Warning (System.PrConst.ExcludeWarning) has occurred, the relevant variables and each histories are not updated.
- If set Excluded Warning (System.PrConst.ExcludeWarning), each checking process to detect warnings is not omitted, so that the processing time is not reduced.
- If set the error code to Excluded Warning (System.PrConst.ExcludeWarning), the error is not excluded.

Precautions

Necessary slave object

■ PDO

- CurrentAlarm (monitor of drive unit errors)

■ SDO

None

Relevant add-ons

The following add-on is required to use this function.

- MotionControl_General

23.3 Error and Warning Reset

Errors and warnings occurred in the motion system can be reset by the following methods.

Operation of this function for each system status

○: Possible, △: Possible (restricted)

Status	Operation availability
STOP	○
RUN	○
Moderate error	△ (only for the axis and the axes group error reset)
Major error	△ (only for the axis and the axes group error reset)

Data to be reset

■ System error reset

- All errors/warnings

Error type	Description	Variable name
System	Motion area system error detection	System.Md.Error
	Latest motion area system error code	System.Md.ErrorID
	Motion area system warning detection	System.Md.Warning
	Latest motion area system warning code	System.Md.WarningID
	Network area error detection	System.Md.NetworkError
	Network area 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 unit error detection	AxisName.Md.DriverError
	Drive unit error code	AxisName.Md.DriverErrorID
	Drive unit 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 unit error detection	AxisName.Md.DriverError
	Drive unit error code	AxisName.Md.DriveErrorID
	Drive unit 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 unit error detection*1	AxisName.Md.DriverError
	Drive unit error code*1	AxisName.Md.DriverErrorID
	Drive unit error detail code*1	AxisName.Md.DriverErrorDetailID

*1 Only axis which belongs to axes group

Relevant variables

Axis control data (AxisName.Cd.)

Variable/Structure name	Name	Details
ErrorReset	Axes Group Error Reset	Resets axes group errors/warnings.

Axes group control data (AxesGroupName.Cd.)

Variable/Structure name	Name	Details
ErrorReset	Axes Group Error Reset	Resets axes group errors/warnings.

System control data (System.Cd.)

Variable/Structure name	Name	Details
ErrorReset	System Error Reset	Resets all errors and warnings.

Relevant FBs

For details on Motion control FBs, refer to the following.

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

Motion control FB	Name	Description
MCv_MotionErrorReset	Motion Error Reset	Resets all errors and warnings of the Motion system.
MC_Reset	Axis Error Reset	Resets errors and warnings of the axis.
MC_GroupReset	Axes Group Error Reset	Resets errors and warnings of the axes group and each axis in the axes group.

System error reset

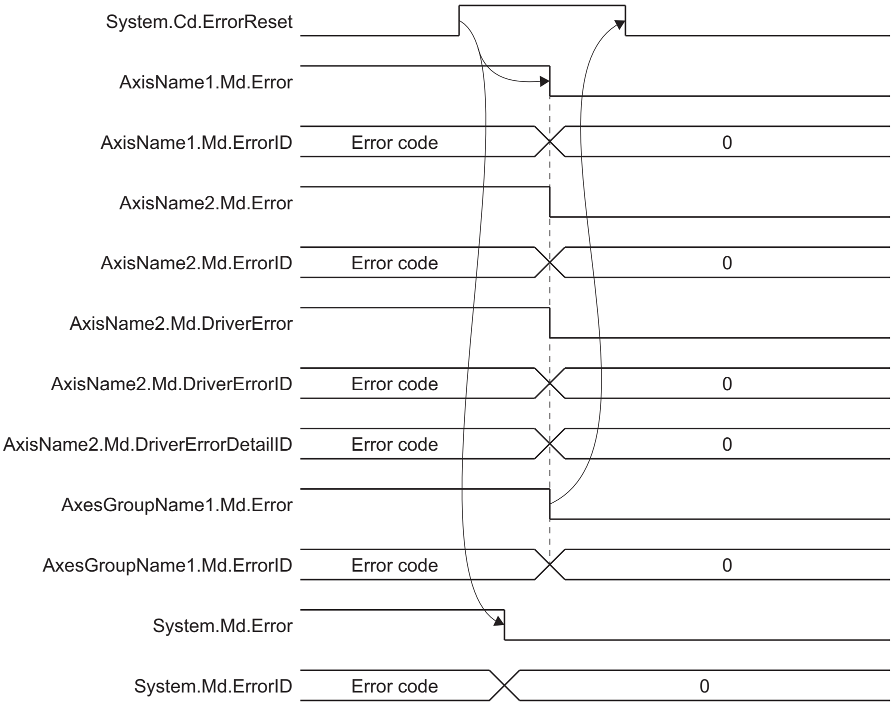
Reset all errors/warnings of the motion system. There are System Error Reset (System.Cd.ErrorReset), MCv_MotionErrorReset (Motion Error Reset) and clear error button on the module diagnostics window of GX Works3 as a reset method. (System Error Reset (System.Cd.ErrorReset) does not become TRUE as the reset is performed by clear error button.)

Control details

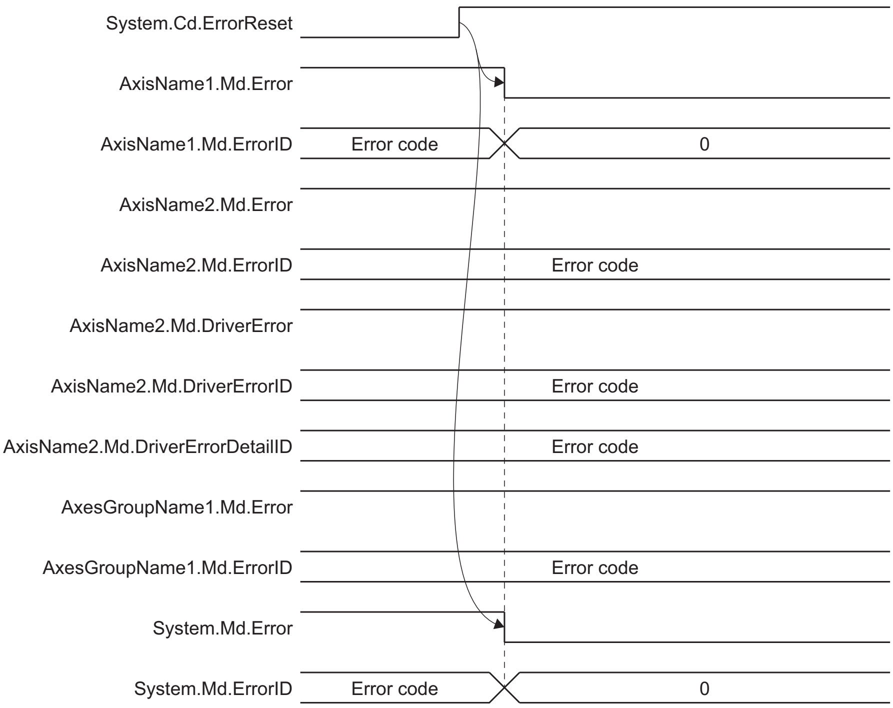
■ System Error Reset (System.Cd.ErrorReset)

- Reset is executed at the rising edge detection of System Error Reset (System.Cd.ErrorReset).
- When the reset is completed, System Error Reset (System.Cd.ErrorReset) becomes FALSE.
- If the error reset is not succeeded, System Error Reset (System.Cd.ErrorReset) remains TRUE.

■When the FB is normally completed (When the error reset is succeeded)



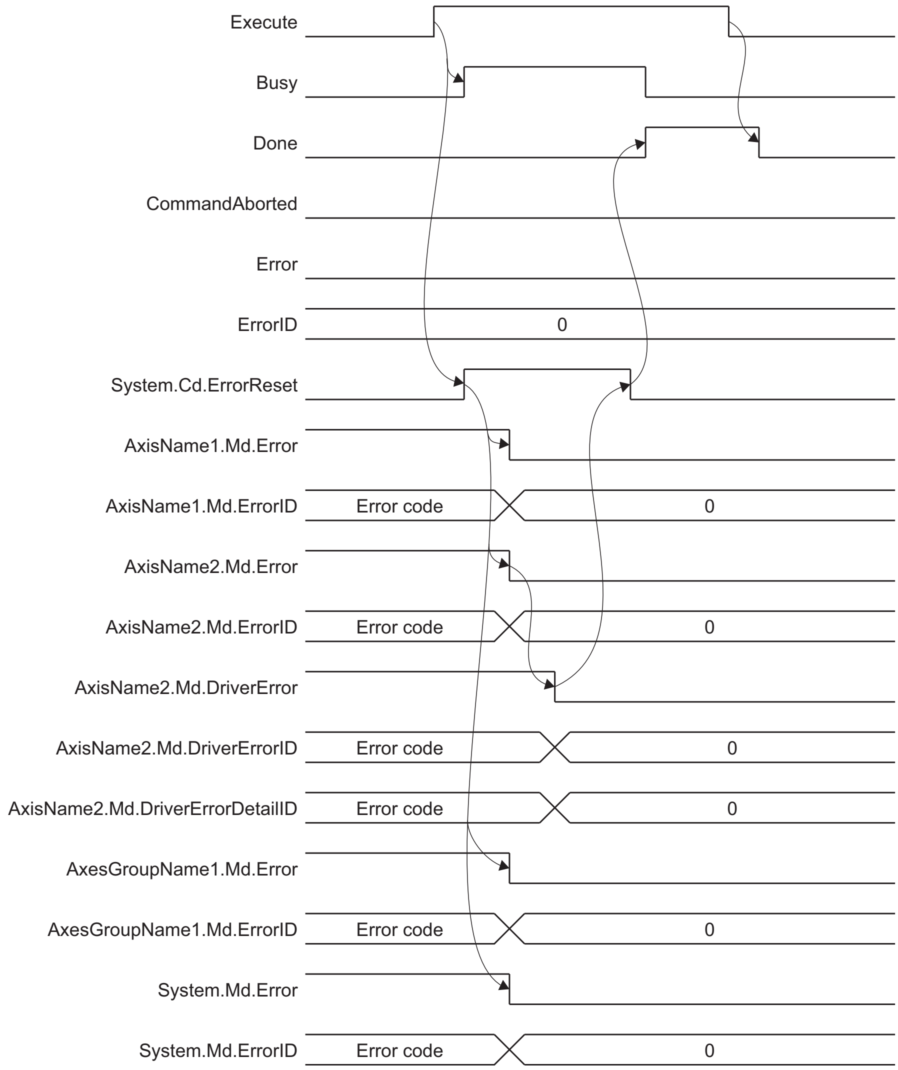
■When an error occurs (When the error reset is failed)



■ MCv_MotionErrorReset (Motion Error Reset)

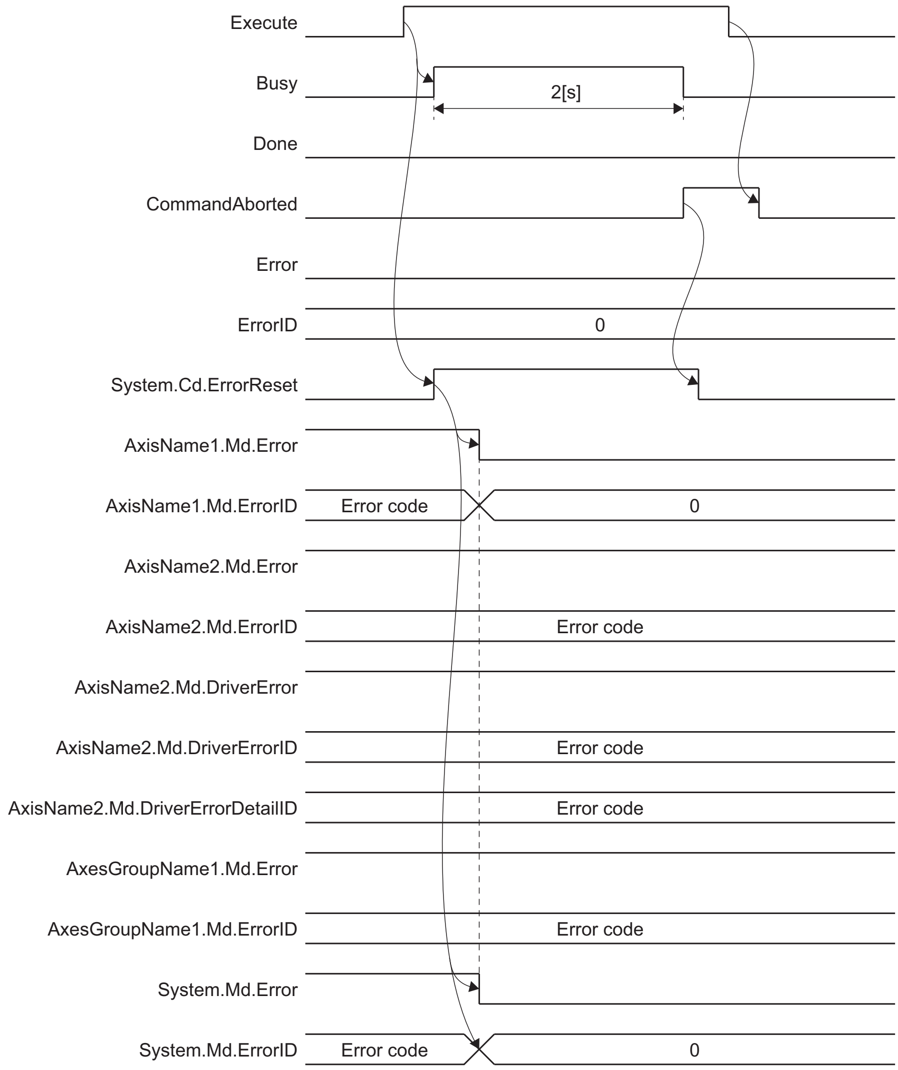
- The FB is executed with Execute command (Execute) = TRUE and Executing (Busy) becomes TRUE as the process starts, and then System Error Reset (System.Cd.ErrorReset) is set to TRUE.
- Executing (Busy) becomes FALSE and Execution completion (Done) becomes TRUE as clearing the errors/warnings of the motion system is completed.
- The errors/warnings are not cleared if Execute command (Execute) becomes TRUE while the error/warning causes still remain. In this case, Abortion of execution (CommandAborted) becomes TRUE and System Error Reset (System.Cd.ErrorReset) is set to FALSE if the errors are not cleared within two [s] after command executed. Temporarily set Execute command (Execute) to FALSE, remove the causes of the errors/warnings, then set Execute command (Execute) to TRUE again.

■When the FB is normally completed



23

■At timeout



Precautions

- Do not operate System Error Reset (System.Cd.ErrorReset) directly while System Error Reset (SYSTEM.Cd.ErrorReset) is executed.
- Use MCv_MotionErrorReset (Motion Error Reset) without operating System Error Reset (System.Cd.ErrorReset) to perform the error reset from the CPU module.

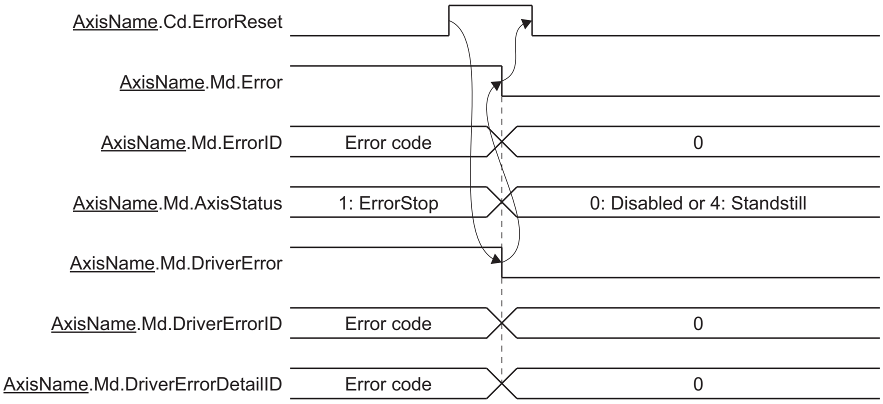
Axis error reset

Reset the error/warning of a single axis and drive unit error by Axis Error Reset (AxisName.Cd.ErrorReset).

Control details

■ Axis Error Reset (AxisName.Cd.ErrorReset)

- Reset is executed at the rising edge detection of Axis Error Reset (AxisName.Cd.ErrorReset).
- When the reset is completed, Axis Error Reset (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 Axis Error Reset (AxisName.Cd.ErrorReset) is TRUE.



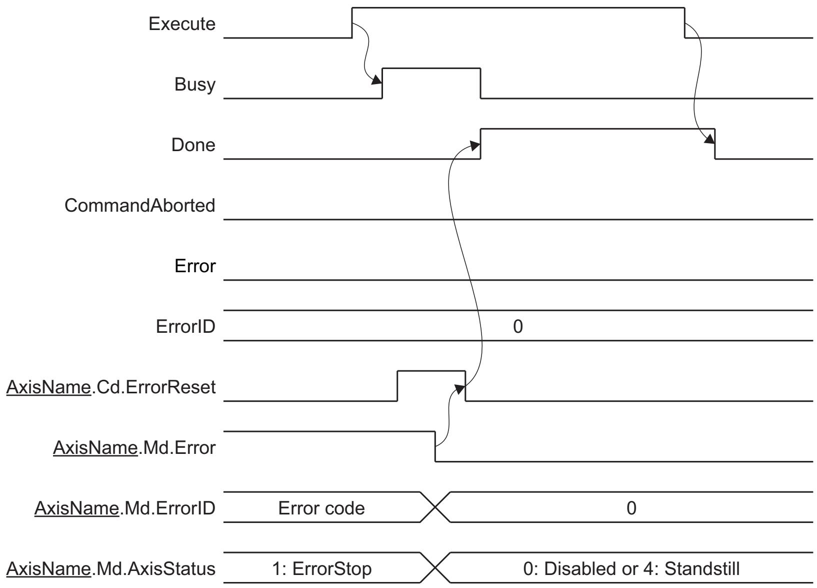
23

■ MC_Reset (Axis Error Reset)

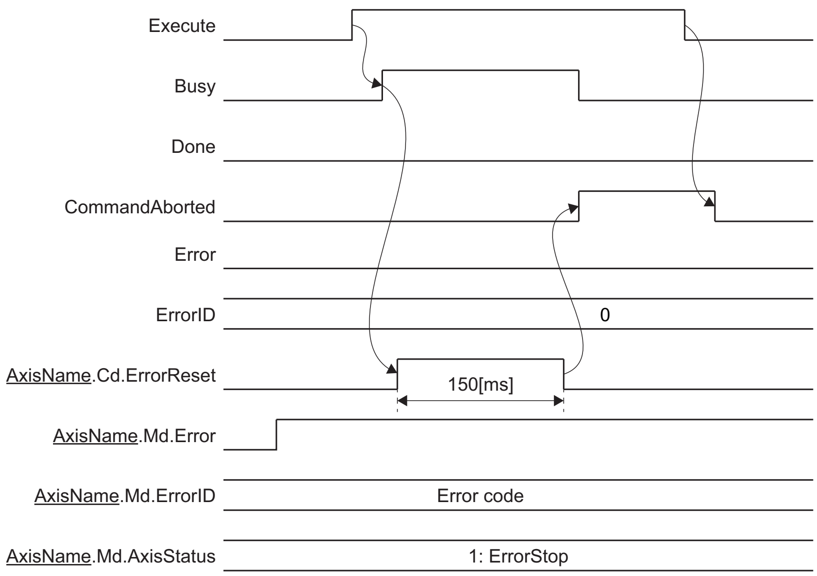
- The FB is executed with Execute command (Execute) = TRUE and Executing (Busy) becomes TRUE as the process starts, and then Axis Error Reset (AxisName.Cd.ErrorReset) of the target axis is set to TRUE.
- Executing (Busy) becomes FALSE and Execution completion (Done) becomes TRUE as clearing the errors/warnings of the axis is completed.
- Executing (Busy) is set to FALSE and Execution completion (Done) becomes TRUE after the warnings are cleared if execution is carried out as the axis status (AxisName.Md.AxisStatus) is "1: Stopping on error (ErrorStop)".
- The errors/warnings are not cleared if Execute command (Execute) is set to TRUE while the error/warning causes of the axis still remain. If the errors are not cleared within 1 [s] after command executed, Abortion of execution (CommandAborted) becomes TRUE. Set Execute command (Execute) to FALSE temporarily to clear the error/warning causes and set Execute command (Execute) to TRUE again.
- If the error occurs in the FB, Error (Error) is set to TRUE and the error code is stored in Error code (ErrorID). For details on error codes, refer to the following.

☞ Page 764 TROUBLESHOOTING

■When the FB is normally completed



■At timeout



Precautions

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

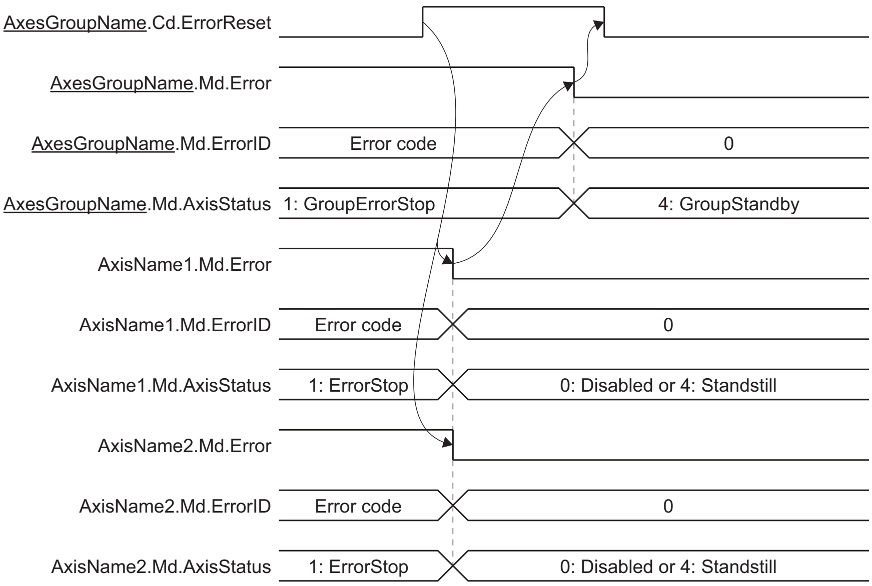
Axes group error reset

Reset the error/warning of the axes group by Axes Group Error Reset (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

■ Axes Group Error Reset (AxesGroupName.Cd.ErrorReset)

- Reset is executed at the rising edge detection of Axes Group Error Reset (AxesGroupName.Cd.ErrorReset).
- When the reset is completed, Axes Group Error Reset (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, Axes Group Error Reset (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 group is stopped completely and Axes Group Error Reset (AxesGroupName.Cd.ErrorReset) is TRUE.
- The error reset can be canceled by changing Axes Group Error Reset (AxesGroupName.Cd.ErrorReset) to FALSE.

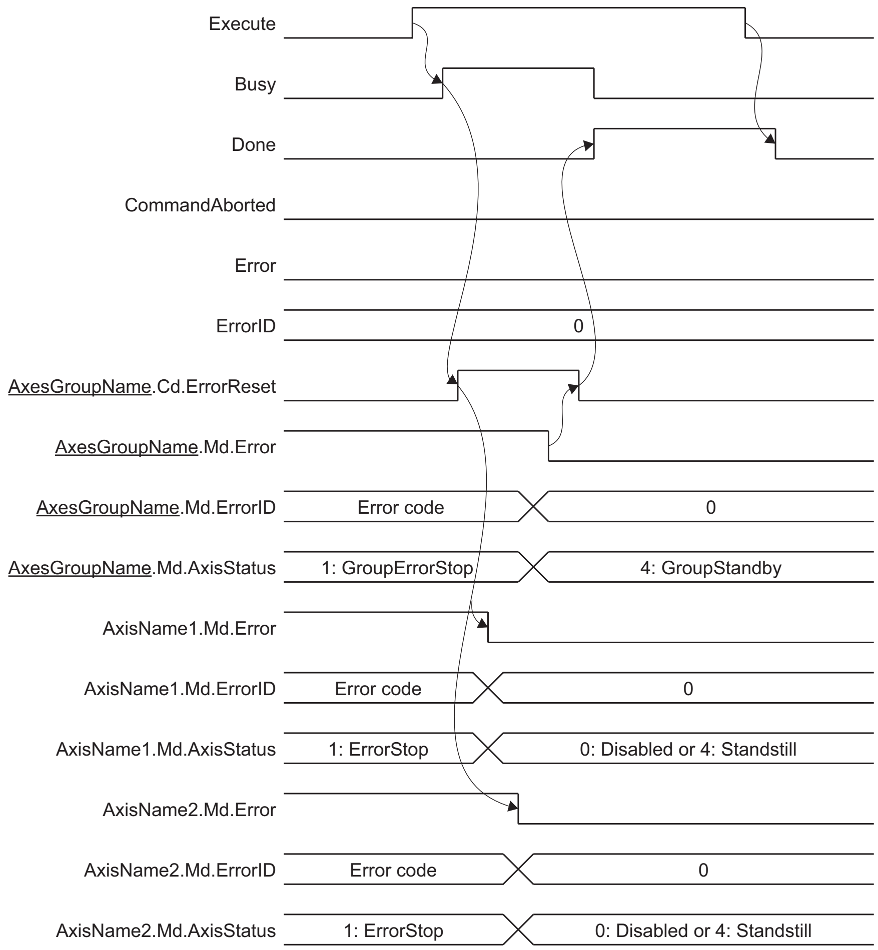


■ MC_GroupReset (Axes Group Error Reset)

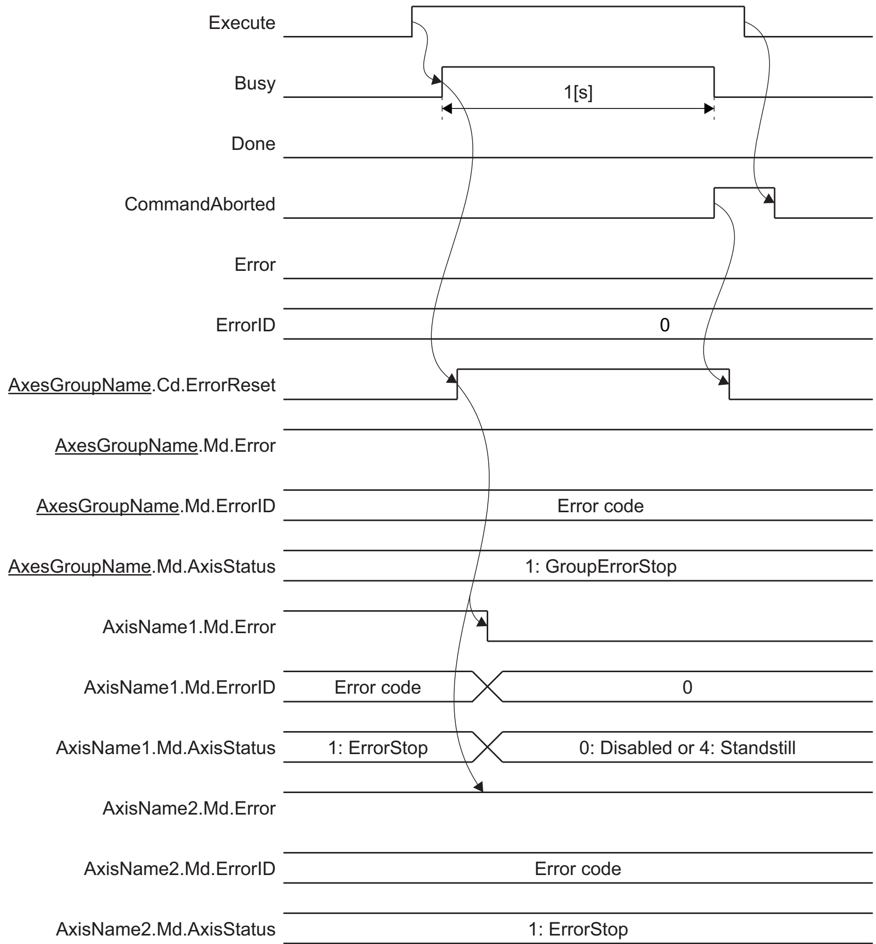
- As the process start by executing this FB at detecting rising edge of Execute command (Execute), Executing (Busy) becomes TRUE, and then Axes Group Error Reset (AxesGroupName.Cd.ErrorReset) of the target axes group is set to TRUE.
- Executing (Busy) becomes FALSE and Execution completion (Done) becomes TRUE as clearing the errors/warnings of the axis and the axes group is completed.
- Executing (Busy) becomes FALSE and Execution completion (Done) becomes TRUE after the warnings are cleared if execution is carried out as the axes group status (AxesGroupName.Md.GroupStatus) is not "1: Stopping on error (GroupErrorStop)".
- The errors/warnings are not cleared if Execute command (Execute) is set to TRUE while the error/warning causes still remain. In this case, Abortion of execution (CommandAborted) becomes TRUE and becomes FALSE Axes Group Error Reset (AxesGroupName.Cd.ErrorReset of the target axes group if the errors are not cleared within 1 [s] after command executed. Set Execute command (Execute) to FALSE temporarily to clear the error/warning causes and set Execute command (Execute) to TRUE again.
- If the error occurs in the FB, Error (Error) is set to TRUE and the error code is stored in Error code (ErrorID). For details of error codes, refer to the following.

☞ Page 764 TROUBLESHOOTING

■When the FB is normally completed



■At timeout



Precautions

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

Precautions

Necessary slave object

■ PDO

None

■ SDO

- Reset alarm (2A46H)*1

*1 For using the real drive axis, the real encoder axis

Relevant add-ons

The following add-on is required to use this function.

- AXIS
- MotionControl_General*1
- ServoDriver_CANopen*2

*1 For using the FBs in Relevant FBs
 · MCv_MotionErrorReset (Motion Error Reset)
 · MC_Reset (Axis Error Reset)
 · MC_GroupReset (Axes Group Error Reset)
*2 For using the real drive axis, the real encoder axis

23.4 Troubleshooting by Symptom

This section describes troubleshooting for the Motion module.

Servo ON failed

When Servo ON failed, check the following items.

<table>
<tr><th colspan="2">Check item</th><th>Action</th></tr>
<tr><td colspan="2">Are both MC_Power (Operation Available) and MCv_AllPower (All Axes Operation Available) being used?</td><td>Check whether both FBs perform Servo ON or not.</td></tr>
<tr><td>MCv_AllPower (All Axes Operation Available) is executed from the PLC CPU</td><td>Is I/O Number (StartIO) for the target Motion module set?</td><td>Set I/O Number (StartIO) for the target Motion module.</td></tr>
<tr><td>Device label of MR-J5(W)-G or MR-JET-G is set</td><td>In network I/O setting, is device label (Controlword of data type RWw) of MR-J5(W)-G or MR-JET-G targeted labeling?</td><td>Clear the device label (Controlword of data type RWw) of MR-J5(W)-G or MR-JET-G from labeling target.</td></tr>
</table>

A motor does not rotate

When a motor does not rotate, check the following items.

<table>
<tr><th colspan="2">Check item</th><th>Action</th></tr>
<tr><td>Servo ON is TRUE</td><td>Is Driver unit conversion numerator/Driver unit conversion denominator being set?</td><td>Convert machine feed value to the set value of the driver unit by using driver unit conversion function.</td></tr>
<tr><td>Acceleration/ Deceleration method is set to Acceleration/ Deceleration specification method (default)</td><td>Is Acceleration or Deceleration set as Acceleration (Acceleration)/Deceleration (Deceleration), not acceleration/deceleration time?</td><td>Correct the setting value of acceleration/deceleration.</td></tr>
</table>

23.5 List of Warning Codes

Warning code	Error	Warning name	Error detail and cause	Detection timing	Operation at warning occurrence	Action
0D00H	Axis	Absolute Position Data Incorrect Warning	• The absolute position data was incorrect at current position restoration for the following causes. • The axis type was changed. • The absolute position data was erased. • The homing request was 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' on driver side was detected. • Slave object 'Polarity(607EH)' b7: position polarity was changed. • Slave object 'HomeOffset (607CH)' was changed. • The connected driver device was changed. • The encoder resolution of the connected driver device was changed. ■Real encoder axis (via drive unit) only • 'Absolute position erased' on the scale measurement encoder was detected. • The encoder resolution of the scale measurement encoder was changed.	During control	The current position is restored with the same content as the incremental system, and the homing request becomes TRUE.	Please execute homing.
0D01H	Axis	Start during Operation Warning	An instruction that cannot start during operation was executed.	During control	Do 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 Do not start after the execution of FB is interrupted. ■During control Decelerates and continues stop/wait until the other JOG command becomes FALSE.	Adjust settings so that JogForward and JogBackward do not become TRUE at the same time.
0D03H	Axis/Axes group	Velocity Limit Value Over Warning	The target velocity exceeded the following when changing velocity/ velocity override factor, or velocity exceeded the following during control. • Positive Direction Velocity Limit Value (AxisName.Pr.VelocityLimit_Positive) (Detail Code: 0001H) • Negative Direction Velocity Limit Value (AxisName.Pr.VelocityLimit_Negative) (Detail Code: 0002H) • Velocity Limit Value (AxesGroupName.Pr.VelocityLimit) (Detail Code: 0003H)	At control change During control	■At control change Continues the operation at the previous speed. ■During control Continues the operation over the speed limit.	Set the velocity within the following. • Positive Direction Velocity Limit Value (AxisName.Pr.VelocityLimit_Positive) • Negative Direction Velocity Limit Value (AxisName.Pr.VelocityLimit_Negative) • Velocity Limit Value (AxesGroupName.Pr.VelocityLimit)

Warning code	Error	Warning name	Error detail and cause	Detection timing	Operation at warning occurrence	Action
0D04H	Axis/Axes group	Acceleration Time Limit Over Warning	Acceleration time exceeded 8400 [s] when changing velocity/ acceleration (acceleration/ deceleration time)/velocity override factor/acceleration override factor.	During control	Continues the operation at the previous speed/ acceleration (acceleration/ deceleration time).	Set velocity/acceleration (acceleration/deceleration time)/velocity override factor/ acceleration override factor so that acceleration time does not exceed 8400 [s].
0D05H	Axis/Axes group	Deceleration Time Limit Over Warning	The following occurred when changing velocity/deceleration (acceleration/deceleration time)/ velocity override factor/ acceleration override factor. • Deceleration time exceeded 8400 [s]. • The deceleration time for automatically decelerating from the target velocity exceeded 8400 [s].	During control	Continues the operation at the previous speed/ deceleration (acceleration/ deceleration time).	Set velocity/deceleration (acceleration/deceleration time)/velocity override factor/ acceleration override factor as follows. • Adjust settings so that the deceleration time does not exceed 8400 [s]. • Adjust settings so that the deceleration time for automatically decelerating from the target velocity does not exceed 8400 [s].
0D06H	Axis/Axes group	Acceleration Limit Value Over Warning	Target acceleration exceeded AxisName(AxesGroupName).Pr.AccelerationLimit when changing acceleration/acceleration override factor.	During control	Continues the operation at the previous acceleration.	Set target acceleration so that it does not exceed AxisName(AxesGroupName). Pr.AccelerationLimit.
0D07H	Axis/Axes group	Deceleration Limit Value Over Warning	Target deceleration exceeded AxisName(AxesGroupName).Pr. DecelerationLimit when changing deceleration/acceleration override factor or when a stop cause occurred.	During control	■At control change Continues the operation at the previous deceleration. ■At stop cause occurrence Operates at the deceleration limit value.	Set target deceleration so that it does not exceed AxisName(AxesGroupName). Pr.DecelerationLimit.
0D08H	Axis/Axes group	Position Command Unit Mismatch Warning	• AxisName(AxesGroupName).Pr. Unit_Position has been set to different values between an axes group and the interpolation axes. • AxisName.Pr.Unit_Position has been set to different values between the master axis and slave axis.	At start / At multiple start	—	• Set the same value to AxisName(AxesGroupName).Pr.Unit_Position for the axes group and interpolation axes. • Set the same value to AxisName.Pr.Unit_Position for the master axis and slave 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 Velocity Override Factor 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 Factor 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.

Warning code	Error	Warning name	Error detail and cause	Detection timing	Operation at warning occurrence	Action
0D0FH	Axis/Axes group	Out of Velocity Range Warning	The value is out of range for Velocity.	During control	Continues the operation with the previous speed.	Set Velocity within the range.
0D10H	Axis/Axes group	Overrun Warning	Target position turned overrun because the deceleration distance for command output velocity 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 Velocity Mode Specified Mismatch Warning	The VelocityMode of the under-control FB is different from that of the multiply-started FB.	At multiple start	Operates by "VelocityMode" of each FB.	Set the same VelocityMode for the under-control FB and the multiply-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 velocity.
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 settings for MC_SetPosition.
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 Velocity 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.

23

Warning code	Error	Warning name	Error detail and cause	Detection timing	Operation at warning occurrence	Action
0D1EH	Axes group	Velocity Command Unit Mismatch Warning	The velocity command unit is different from that of the under-conrol FB.	At multiple start	—	Set the same value to AxisName(AxesGroupName). Pr.Unit_Velocity for the axes group and interpolation axes.
0D1FH	Axis	Driver Warning	Warning occurred in the driver.	During operation	Executes the operation.	Please check the error details and take corrective actions according to AxisName.Md.DriverErrorID. (Please refer to the manual of each driver for details on AxisName.Md.DriverErrorID.)
0D20H	Axis	Velocity Limit Value Over Warning on Direction Change	Target velocity exceeded the velocity limit value on that direction when accelerating again from velocity 0 on direction change.	During control	Continues the operation with speed 0.	Adjust settings so that the target velocity does not exceed velocity limit value after direction change.
0D21H	Axis	Out of End Velocity Range during Operation Warning	The value is out of range for EndVelocity when changing to the end target velocity.	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.
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 Velocity 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 Operation Cycle Converted Velocity Range Warning	The velocity converted from Velocity by operation cycle is out of the range.	During control	Continues the operation with the previous speed.	Set Velocity so that the velocity converted by operation 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.
0D32H	Axis	Acceleration Time Exceeded during Direction Change Warning	Acceleration time exceeded 8400 [s] when accelerating again from velocity 0 on direction change.	During control	Continues the operation at speed 0.	Adjust settings so that the acceleration time after direction change does not exceed 8400 [s].

Warning code	Error	Warning name	Error detail and cause	Detection timing	Operation at warning occurrence	Action
0D33H	Axis/Axes group	Deceleration Time Over Warning on FB Switching	Deceleration time exceeded 8400 [s] on FB switching.	During control	Continues the operation with the velocity and deceleration before the FB switch.	Set velocity, deceleration, and jerk so that deceleration time does not exceed 8400 [s].
0D34H	Axis/Axes group	Out of Velocity Range Clamping Warning	The velocity-overridden velocity was below the velocity lower limit value.	At start At control change	Operates at velocity 0.	Adjust settings so that the velocity after velocity override is within the range.
0D35H	Axis/Axes group	Out of Acceleration Range Clamping Warning	■Acceleration/deceleration specification method The acceleration after acceleration override was out of range. ■Time-fixed acceleration/ deceleration method. The acceleration or deceleration calculated from Acceleration was out of range.	At start At control change	■Acceleration/ deceleration specification method Operates at upper limit value of acceleration or 0. ■Time-fixed acceleration/ deceleration method Operates at upper limit value of acceleration (deceleration) or 0.	■Acceleration/deceleration specification method Adjust settings so that the acceleration after acceleration override is within the range. ■Time-fixed acceleration/ deceleration method Adjust settings so that the acceleration or deceleration calculated from Acceleration is within the range.
0D36H	Axis/Axes group	Out of Deceleration Range Clamping Warning	The deceleration after acceleration override was out of range.	At start At control change	Operates at upper limit value of deceleration or 0.	Adjust settings so that the deceleration after acceleration override is within the range.
0D37H	Axis/Axes group	Out of Acceleration/ Deceleration Time Range Clamping Warning	The acceleration/deceleration time after acceleration override was out of range.	At start At control change	Operates at upper limit value of acceleration/ deceleration time or 0.	Adjust settings so that the acceleration/deceleration time after acceleration override is within the range.
0D38H	Axis/Axes group	Out of Operation Cycle Converted Acceleration Range Warning	Because the acceleration converted from Acceleration by operation cycle is low, there is a possibility that the error of set velocity is large.	At start At control change	Continues the control.	• Set Acceleration so that the acceleration converted by operation cycle becomes larger. • Increase the operation cycle.
0D39H	Axis/Axes group	Out of Operation Cycle Converted Deceleration Range Warning	Because the deceleration converted from Deceleration by operation cycle is low, there is a possibility that the error of set velocity is large.	At start At control change	Continues the control.	• Set Deceleration so that the deceleration converted by operation cycle becomes larger. • Increase the operation cycle.
0D3AH	Axis/Axes group	Out of Operation Cycle Converted Jerk Range Warning	Because the jerk converted from Jerk by operation cycle is low, there is a possibility that the error of set velocity or set acceleration is large.	At start At control change	Continues the control.	• Set Jerk so that the jerk converted by operation cycle becomes larger. • Increase the operation cycle.
0D3BH	Axis/Axes group	Jerk Acceleration/ Deceleration Disabled Warning	Because the jerk converted from Jerk by operation cycle is 0, it does not operate as a jerk acceleration/deceleration.	At start	Operates as trapezoidal acceleration/ deceleration.	• Set Jerk so that the jerk converted by operation cycle becomes larger. • Increase the operation cycle.

Warning code	Error	Warning name	Error detail and cause	Detection timing	Operation at warning occurrence	Action
0D3EH	Axis	Advanced Synchronous Control Operation Overflow Warning	Overflow occurred during the operation of each module of advanced synchronous control. ■Input Axis • The movement amount of the data source overflowed. (Detail code: 0001H) • Movement amount overflowed during smoothing compensation. (Detail code: 0002H) • Movement amount overflowed during phase compensation. (Detail code:0003H) • Phase compensation amount overflowed. (Detail code: 0004H) • The cumulated value of the moving direction restriction amount overflowed. (Detail code: 0005H) ■Output Axis • The movement amount of the output axis overflowed during phase compensation. (Detail code: 0100H)	During operation	Continues the control. It is controlled with the minimum or maximum value.	• Reduce the input axis velocity. • Review the setting value of the time constant to reduce the smoothing level, or increase the operation cycle. • Reduce the advance time for phase compensation. • Check the enabled direction of the moving direction restriction setting. (The setting may be reversed.) • Check if the input axis moves to the reverse direction of the enabled direction.
0D3FH	Axis	Out of Advanced Synchronous Control Variable Range Warning	The parameters or control data were out of range during advanced synchronous control.	During operation	Continues the operation with the setting before the change.	Set the value of the target variable within the range.
0D40H	Axis	Advanced Synchronous Control Restoration Incorrect Warning	The absolute position data was incorrect when restoring current position due to the following causes. • Target axes of the input axis and output axis of advanced synchronous control are different before and after the axis connection. (Detail code: 0001H) • File versions of the backup data are different. (Detail code: 0002H)	When connecting driver	Does not execute restoration taking into account the previous value.	The target axes and axis label IDs of the input axis and output axis must be the same before and after the axis connection.
0D44H	Axis	No Cam Table Warning	The cam table specified in CamTableID does not exist.	During operation	Continues the operation with the setting before the change.	Open cam data to the open area with MC_CamTableSelect.
0D45H	Axis	Unable to Change Cam Length per Cycle Under Advanced Synchronous Control Warning	Under advanced synchronous control, when using operation profile data consisting of stroke ratio cam data where the cam data starting point was not 0, "Length per Cycle" was changed.	During operation	Continues operation with the length per cycle before the change.	Set the cam starting point (AdvOutputName.Pr.Cam.StartingPoint) to 0.

Warning code	Error	Warning name	Error detail and cause	Detection timing	Operation at warning occurrence	Action
0D46H	Axis	Operation Profile Data Settings Incorrect Under Advanced Synchronous Control Warning	Any of the following operation profile data settings have been set to values that cannot be used under advanced synchronous control. • Profile data type incorrect (Detail code: 0001H) • Operation profile data format incorrect (Detail code: 0002H) • The operation is not repetitive. (Detail code: 0003H) • The master axis (input) absolute coordinate is not set to relative coordinate. (Detail code: 0004H) • The slave axis (output) absolute coordinate is not set to relative coordinate. (Detail code: 0005H) • The starting point is not 0. (Detail code: 0006H) • The initial stroke is not 0. (Detail code: 0007H) • The cam starting point (AdvOutputName.Pr.Cam.StartingPoint) exceeded the resolution. (Detail code: 0008H) • The target operation profile data is being operated by another FB. (Detail code: 0009H)	During operation	Continues the operation with the setting before the change.	• Set the profile data type to "1: Cam Data". (Detail code: 0001H) • Set the operation profile data format to "0: Linear Interpolation", "1: Section Interpolation", or "2: Spline Interpolation". (Detail code: 0002H) • Set repetitive operation to "1: Enable". (Detail code: 0003H) • Set master axis (input) absolute coordinate to "0: Relative Coordinate". (Detail code: 0004H) • Set slave axis (output) absolute coordinate to "0: Relative Coordinate". (Detail code: 0005H) • Set the starting point to "0". (Detail code: 0006H) • Set the initial stroke to "0". (Detail code: 0007H) • Set the cam starting point (AdvOutputName.Pr.Cam.StartingPoint) to a value smaller than the resolution. (Detail code: 0008H) • End the FB that is operating the target operation profile data. (Detail code: 0009H)
0D47H	Axis	Add-on Acquisition Failure Warning	Failed to acquire the necessary add-on for operation.	During operation	Continues the control.	Check the detailed information, then install the displayed add-on (*.mpk) and enable it.
0D48H	Axes group	Operation Profile Data Control Profile ID Setting Incorrect Warning	During the control of a multiple axes positioning data operation, the positioning data was switched to the next one in the M code outputting state, from one whose operation pattern was continuous path.	During control	Continues the control.	Do not set the M code for positioning data whose operation pattern is continuous path, or reset the M code before the data switches.
0D49H	Axis	Synchronous Control Variable Out-of-Range Warning	Out-of-range value set for parameter or control data during synchronous control. • Out of spindle offset range (Detail code: 0001H) • Out of the follow shaft offset range (Detail code: 0002H) • Out of spindle coefficient range (Detail code: 0003H) • Out of the dependent coefficient range (Detail code: 0004H) • Out of cam table ID range (Detail code: 0005H)	During operation	Continues the operation with the setting before the change.	Set the target variable to a value within a range.
0F00H	System	Software Reboot Disabled Warning	Software 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.

Warning code	Error	Warning name	Error detail and cause	Detection timing	Operation at warning occurrence	Action
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 review 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 acquire the 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 Setting 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 access 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.
0F08H	System	Cycle Over Warning	Operation 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 communication cycle interval setting or buffer memory refresh cycle to a larger value. • When using by a operation cycle of 31.25 μs, please begin with a minimum configuration of add-ons, and then add more add-ons as required according to the load of the system. • When the inter-module synchronization function is enabled, please be sure to execute the EI instruction with the PLC to enable I44 interrupt programs.
0F09H	System	File Transfer Execution Request Acquisition Disabled Warning	Failed to acquire the 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 acquire the address of System.Md.FileTransfer_State.	At system start	Cannot execute the file transfer of the command label.	Start the system again.

Warning code	Error	Warning name	Error detail and cause	Detection timing	Operation at warning occurrence	Action
0F0BH	Axis	Maximum Setting Axis Number Over Warning	Axis setting that 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 the user root path. • If the label incorrect error occurred, please refer to the corrective action for label incorrect error.
0F0EH	System	Module Extended Parameter Storage Location Warning	The project content of the storage location different from module extended parameter storage location setting was changed.	At system start	The written module extended parameter is not enabled.	Please check the module extended parameter storage location 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	• The add-on on which it is dependent has not been installed. (Detail Code: 0001H) • There is an inconsistency in the versions with other add-ons. (Detail Code: 0001H) • There is an inconsistency in the versions with engineering tool. (Detail Code: 0002H)	At power ON	The servo system recorder does not start.	• Please install the software again. • Please install the add-on which it is dependent. • Please install the correct version of the add-on. • Please execute project writing using the engineering tool that supports servo system recorder.
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.
0F13H	System	Servo System Recorder Setting File Creation Disabled Warning	• There is a user-created logging setting file in save destination. (Detail Code: 0001H) • Memory capacity in save destination is insufficient. (Detail Code: 0002H) • Cannot output logging setting file because of system error. (Detail Code: 0003H)	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 the save destination. • Please check the free capacity in the save destination. • Please check if any error has occurred to other add-ons.
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 At servo system recorder stop	The logging does not starts.	• Please stop logging setting 10 or 9 if they are being executed. • Please check the logging error code.
0F15H	System	SD Memory Card Removed Warning	The SD memory card was removed without being disabled.	At removing the SD memory card.	Continues the operation	Remove SD memory card after disabling it.

23

Warning code	Error	Warning name	Error detail and cause	Detection timing	Operation at warning occurrence	Action
0F18H	System	Advanced Synchronous Control Backup Memory Insufficient Warning	The necessary capacity for the input axis and output axis to be backed up exceeds MotionControl_AdvancedSync_System.PrConst.AddonMemory.	At system start	Does not save the backup data of the input axis and output axis for the capacity exceeding the setting of MotionControl_AdvancedSync_System.PrConst.AddonMemory.	Review the setting of MotionControl_AdvancedSync_System.PrConst.AddonMemory.
0F2FH	System	Add-on Warning	An add-on warning occurred.	Depends on the add-on.	Depends on the add-on	Check the manual of the add-on for details about the detail code.

23.6 List of Error Codes

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
1080H	System	Writing to Flash ROM Count Error	The number of writes to the flash ROM has exceeded 100000.	At module start	Does not execute the writes to the flash ROM.	Replace the module.
1300H	System	Co-recording Setting Error	Because the firmware version of the control CPU module does support the co-recording function, co-recording does not operate.	At module start	Co-recording does not operate.	Please check the version of the control CPU module and use a product that supports the function.
1301H	System	Co-recording Setting Error	Because the co-recording setting of the control CPU module is not set to "Use", co-recording does not operate.	At trigger detection	Co-recording does not operate.	Please set the co-recording setting of the control CPU module to "Use".
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 axis emulation enabled axis.	During FB execution	The Error of FB turns ON, and the error code of SDO is output to SDOErrorID.	• Remove the error cause according to the SDO error code. • Reduce the number of SLMPs to be issued, or modify the program to issue the next request after the response is received. • Modify program to not issue SLMP requests for axes with axis emulation enabled.
1A00H	Axis	Software Stroke Limit Over (Target Position)	A value out of the software stroke limit range was set for the target position.	At start	Does not start.	Set the target position within the software stroke limit range.
1A01H	Axis	Software Stroke Limit Over (Start Position)	• Started at a position out of the software stroke limit range. • Started at a position out of the software stroke limit range towards a direction other than the returning enable direction.	At start	Does not start.	Set ONLY_INSIDE or DISABLE to AxisName.Cd.SwStrokeLimit_Override and carry out a move to the movable range.
1A02H	Axis	Start when Software Stroke Limit is Valid	It was started except when AxisName.Pr.SwStrokeLimit_Target is invalid.	At start	Does 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.	Carry out a move to the 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.	Carry out a move to the software stroke limit range.
1A05H	Axis/Axes group	Out of Target Position Range	• The value is out of range for Position/Distance. • In the case of target position designation beyond the ring counter, Direction was set to the shortest path and Position was set out of range.	At start	Does not start.	• Set Position/Distance within the range. • Set the Direction to something other than the shortest path.
1A06H	Axis	Out of Limit Velocity Range	The value is out of range for LimitVelocity.	At start	Does not start.	Set LimitVelocity within the range.

23

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
1A07H	Axis/Axes group	Velocity Limit Value Over	The target velocity exceeded the following. • AxisName.Pr.VelocityLimit_Positive (Detail Code: 0001H) • AxisName.Pr.VelocityLimit_Negative (Detail Code: 0002H) • AxesGroupName.Pr.VelocityLimit (Detail Code: 0003H)	At start At multiple start	■At start Does 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
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	Does not start.	Adjust 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	Does not start.	Set Acceleration within the range.
1A0EH	Axis/Axes group	Acceleration/ Deceleration Method Mismatched	The acceleration/deceleration method of the under-control FB is different from that of the multiply-started FB.	At multiple start	Stops the operation.	Set the same acceleration/ deceleration method for the under-control FB and the multiply-started FB.
1A0FH	Axis	Acceleration Time Over at Start	Acceleration time exceeded 8400 [s].	At start	Does not start.	Set target velocity/target acceleration/target deceleration/ target jerk so that acceleration time does not exceed 8400 [s].
1A10H	Axis	Deceleration Time Over at Start	• Deceleration time exceeded 8400 [s]. • The deceleration time for automatically decelerating from the target velocity exceeded 8400 [s].	At start	Does not start.	Set target velocity/target acceleration/target deceleration/ target jerk as follows. • Adjust settings so that the deceleration time does not exceed 8400 [s]. • Adjust settings so that the deceleration time for automatically decelerating from the target velocity does not exceed 8400 [s].
1A11H	Axis	Out of Acceleration Range	The value is out of range for Acceleration.	At start	Does not start.	Set Acceleration within the range.
1A12H	Axis	Out of Deceleration Range	The value is out of range for Deceleration.	At start	Does 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	Does 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	Does not start.	Set the velocity of converted driver velocity unit within the setting range.
1A18H	Axis	Out of Velocity Initial Value Selection Range	The value is out of range for velocity initial value selection of Options.	At start	Does not start.	Set velocity 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	Does 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	Does not start.	Execute the instruction in the driver supporting csv.
1A1CH	Axis	cst Unsupported Driver	The driver does not support cst.	At start	Does not start.	Execute the instruction in the driver supporting cst.

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
1A1DH	Axis	Control Mode Switching Error	Driver control mode switching did not complete within 1 [s] 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 an error occurred in the driver or whether the setting is incorrect, and take necessary actions. • When the MR-J5(W)-G or MR-JET-G is connected: When switching the driver control mode while the motor is operating, please set ZSP disabled selection at control switching (PC76.1) to "1: Disabled".
1A1EH	Axis	Out of Target Torque Range	The value is out of range for Torque.	At start	Does 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	Does not start.	Set TorquePositiveRamp within the range.
1A20H	Axis	Out of Torque Negative Direction Ramp Range	The value is out of range for TorqueNegativeRamp.	At start	Does not start.	Set TorqueNegativeRamp within the range.
1A22H	Axis	Start at Homing Incomplete	The positioning was started when homing request flag is ON.	At start	Does 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	Does 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.AxesGroup Status was other than GroupStandby.	At start	Does not start.	Start the axes group operation when AxesGroupName.Md.AxesGrou pStatus is GroupStandby.
1A2DH	Axis	FLS Signal Detection (at Start)	The signal input from FLS signal at start was detected.	At start	Does not start.	Set ONLY_INSIDE or DISABLE to AxisName.Cd.HwStrokeLimit_O verride and carry out a move to the controllable range.
1A2EH	Axis	RLS Signal Detection (at Start)	The signal input from RLS signal at start was detected.	At start	Does not start.	Set ONLY_INSIDE or DISABLE to AxisName.Cd.HwStrokeLimit_O verride and carry out a 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_O verride and carry out a 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 carry out a move to the controllable range.
1A34H	Axis/Axes group	Out of Gear Denominator Range	The value is out of range for RatioDenominator.	At start	Does not start.	Set the value of RatioDenominator within the range.
1A36H	Axis/Axes group	Velocity Limit Value Over during Controlling	The set velocity exceeded the following when AxisName.Pr.VelocityLimit_Over Operation is 3:ImmediateStop. • AxisName.Pr.VelocityLimit_Posit ive (Detail Code: 0001H) • AxisName.Pr.VelocityLimit_Neg ative (Detail Code: 0002H)	During control	Operates exeeding "AxisName.Pr.Velocit yLimit_Positive" or "AxisName.Pr.Velocit yLimit_Negative".	Set the velocity within the following. • AxisName.Pr.VelocityLimit_Po sitive • AxisName.Pr.VelocityLimit_Ne gative
1A37H	Axis	Out of Direction Selection Range	The value is out of range for Direction.	At start At multiple start	■At start Does not start. ■At multiple start Stops the operation.	Set Direction within the range.

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
1A38H	Axis	Acceleration Limit Value Over	Target acceleration exceeded AxisName(AxesGroupName).Pr.AccelerationLimit.	At start At multiple start	■At start Does 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 exceeded AxisName(AxesGroupName).Pr.DecelerationLimit.	At start At multiple start	■At start Does 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 Does 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	[5: Servo OFF After Deceleration to Stop (ServoOffAfterDecelStop)] was selected for AxisName.Pr.StopMode_ServoOff, and ServoOn of MC_Power MCv_AllPower) was set to FALSE during operation.	During control	Deceleration stops.	Set TRUE to ServoOn in MC_Power (MCv_AllPower).
1A3CH	Axis	Immediate Stop by Servo OFF During Operation	• [4: Servo OFF After Immediate Stop (ServoOffAfterImmediateStop)] was selected for AxisName.Pr.StopMode_ServoOff, and ServoOn of MC_Power(MCv_AllPower) was set to FALSE during operation. • Enable of MC_Power(MCv_AllPower) was set to FALSE 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 Does 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	Does 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	Does 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	Does 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	Does not execute FB.	Set the 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	Does not execute FB.	Set the value within the range for SlaveOffset.
1A43H	Axis	Out of Master Axis Coefficient Range	• The value is out of range for MasterScaling. • The result of multiplying MasterScaling and the length per cycle is out of the positioning range.	At start When executing FB	Does not execute FB.	• Set a value within the range for MasterScaling. • Set values so that the result of multiplying MasterScaling and the length per cycle is within the positioning range.

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
1A44H	Axis	Out of Slave Axis Coefficient Range	• The value is out of range for SlaveScaling. • The result of multiplying SlaveScaling and the stroke amount is out of the positioning range.	At start When executing FB	Does not execute FB.	• Set the value of SlaveScaling within the range. • Set values so that the result of multiplying SlaveScaling and the stroke amount is within the positioning range.
1A45H	Axis	Out of Master Axis Follow-up Distance Range	The value is out of range for MasterStartDistance.	At start When executing FB	Does not execute FB.	Set the value of MasterStartDistance within the range.
1A46H	Axis	Out of Master Axis Synchronization Start Position Range	The value is out of range for MasterSyncPosition.	At start When executing FB	Does not execute FB.	Set the value of MasterSyncPosition within the range.
1A47H	Axis	Out of Start Mode Range	The value is out of range for StartMode.	At start When executing FB	Does not execute FB.	Set the value of StartMode within the range.
1A48H	Axis/Axes group	Out of Master Axis Data Source Range	The value is out of range for MasterValueSource.	At start When executing FB	Does not execute FB.	Set the value of MasterValueSource within the range.
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	Does not execute FB.	Check axes specified by Master.
1A4AH	Axis	Out of BufferMode Range	The value is out of range for BufferMode.	At start	Does 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	Does not execute FB.	Open cam data to the open area with MC_CamTableSelect.
1A4DH	Axis/Axes group	Out of Velocity Range	The value is out of range for Velocity.	At start	Does not start.	Set Velocity within the range.
1A4EH	Axis	Out of Options Range	The value is out of range for Options.	At start	Does 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	Does 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	Does 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	Does not start.	Set the value of GearRatioDenominatorM1 within the range.
1A52H	Axis	Out of Master Axis 2 Gear Denominator Range	The value is out of range for GearRatioDenominatorM2.	At start	Does not start.	Set the value of GearRatioDenominatorM2 within the range.
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.

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
1A59H	Axes group	Axes Group Configuration Axis Operation Cycle Incorrect	Operation cycles of configuration axes in axes group are 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 operation cycle in all configuration axes.
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.PrConst.Addon_Axis.RamSizeMax, System.PrConst.Addon_MotionEngine.RamSizeMax. • 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. • The station-specific mode was set to Motion mode when AxisName.PrConst.FastOperationMode was 5FE2H (fast operation mode). • The mapping for fast operation mode was changed when AxisName.PrConst.FastOperationMode was 5FE2H (fast operation 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 Velocity Mode Range	The value is out of range for velocity mode.	At start At multiple start	■At start Does not start. ■At multiple start Stops.	Correct the velocity mode so that it is within the range.
1A62H	Axes group	Circular Interpolation Axes 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 Does not start. ■At multiple start Stops the operation.	• Set the configuration axis in CircAxes with an axis that has been set up. • 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 Does 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 Does 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 Does not start. ■At multiple start Stops the operation.	Set center point (AuxPoint) so that start point is not equal to center point.

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
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 Does 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 Does 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 Does 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 Does 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 Does 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 Does 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 Does 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 Does 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 Does 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 Does 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 Does not start. ■At multiple start Stops the operation.	Set CircularErrorTolerance within the range.
1A71H	Axes group	Circular Interpolation Error Allowable Value Over	The difference 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 Does 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 Does 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 Does not start. ■At multiple start Stops the operation.	Set Pitch within the range.

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
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 Does 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	AxisName.PrConst.TorqueLimit_PositiveInitial was set to a value larger than AxisName.PrConst.TorqueLimit_Max.	At power ON	Generating axis fails.	Set AxisName.PrConst.TorqueLimit_PositiveInitial within the range of 0 to AxisName.PrConst.TorqueLimit_Max.
1A7AH	Axis	Out of Reverse Direction Torque Limit Initial Value Specification Range	AxisName.PrConst.TorqueLimit_NegativeInitial was set to a value larger than AxisName.PrConst.TorqueLimit_Max.	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 velocity designation.	During control	Stops the operation.	Set target position/movement distance so that movement amount is not insufficient.
1A7CH	Axes group	Motion 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 velocity 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	Does 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	Does 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	Does 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	Does not start.	Set AxisName(AxesGroupName).Pr.OverrunOperation within the range.

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
1A90H	Axis	Synchronous Encoder Via Drive Unit Setting Incorrect	• The real drive axis with the same station address as the real encoder axis via drive unit 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 unit 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 unit is not mapped to the real drive axis with the same station address. (Detail Code: 0005H) • Absolute position control 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 unit. • Enable scale measurement function of the connected driver. • Check if the slave object required to be set for the real encoder axis via drive unit exists. • Map the slave object required to be set for the real encoder axis via drive unit 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 control 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 multidrop number 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 multidrop number 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 axis emulation function. • A moderate error occurred in the motion module.	At power ON	Generating axis fails.	• Specify a station address where a device is connected, when the axis emulation function is not to be used. • Refer to the remedy for the moderate error that occurred in the Motion module.
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 the initial value when AxisName.PrConst.FastOperationMode was 5FE2H (fast operation mode).	At power ON	Generating axis fails.	• Please specify data that can be set as slave object. • Please set the signal for the slave object data that needs to be set for each axis type. • Please check whether it was changed from the initial value when setting fast operation mode.
1A97H	Axes group	Axes Group Status Incorrect (When Axes Group is Disabled)	MC_GroupDisable was executed when AxesGroupName.Md.GroupStatus is other than GroupStandby or GroupErrorStop.	Axes Group Disable	Stops the operation.	Execute MC_GroupDisable when AxesGroupName.Md.GroupStatus is GroupStandby or GroupErrorStop.
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.

23

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
1A99H	Axes group	No axis	A 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 an existing axis in configuration axis.
1A9AH	Axis	Absolute Position Control 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 control setting of connection device.
1AA3H	Axis	Out of Ring Counter Range	• The settings of ring counter upper limit value and lower limit value exceeded the positioning range. • Ring counter lower limit value > Ring counter upper limit value. • 0 < |Ring counter upper limit value - Ring counter lower limit value| < 2.0 • The setting values 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 Conversion 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 Conversion 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 add-on SignalIO when set the 'Counter Disabling Signal'.	At power ON	Generating axis fails.	Install the add-on SignalIO and enable it.
1AA8H	Axis/Axes group	Necessary Slave Object Unset	Necessary slave object for control command execution is not set to axis (configuration axis in the case of an axes group).	At start At multiple start	■At start Does not start. ■At multiple start Stops the operation.	• Please set necessary object in control command. • Please use single axis synchronization for virtual linked axis.
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 Does 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 unit normally for homing.	During operation	Homing is not completed normally.	• Execute homing again. • When reading or writing parameters of a device station 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 was 0 during a linear interpolation with reference axis designated.	At start At multiple start	■At start Does 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 multiple start	■At start Does not start. ■At multiple start Stops the operation.	Set torque lamp function selection of Options within the range.

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
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) • Hardware stroke limit override incorrect at homing start (Detail Code: 0005H) • Axis/axes group status incorrect (Detail Code: 0100H) • Axes group operating (Detail Code: 0101H) • Configuration axis status incorrect (Detail Code: 0102H)	At start	Does not start.	• Adjust settings so that it will not start when PLC READY is OFF. • Adjust settings so that it will not start when READY is OFF. • Check the sequence program which turns PLC READY ON/OFF. • Do not set ONLY_INSIDE for AxisName.Cd.HwStrokeLimit_Override at homing start. • Check the status of the drive unit power supply, the wiring with the drive unit, and connector status. • Start it after AxesGroupName.Md.GroupStatus changes into GroupStandby. • Start it after AxisName.Md.AxisStatus changes into Standstill. • Disable the 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 take corrective actions according to AxisName.Md.DriverErrorID. (Please refer to the manual of each driver for details on 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.
1AB0H	Axis	PLC READY OFF During Operation	PLC READY was turned OFF during operation.	During operation	Stops the operation.	Check the sequence program which turns PLC READY 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	Does not start.	Enable the AxisName.PrConst.RingCount_Enable.

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
1AB6H	Axis	External Signal String Incorrect	• Incorrect string was set in the string of external signal. • A signal that does not exist or is incorrect was set. • An unconnected or non-existent station was set when [OBJ] was specified. • A non-existent object was set for TPDO/RPDO when [OBJ] was specified.	At start	Does not start.	• Set to a valid string for the external signal. • Set a signal that can be used. • Connect the target station if [OBJ] has been specified. • Set the target object for TPDO/RPDO if [OBJ] has been specified.
1AB7H	Axes group	Path Choice Setting Incorrect	Incorrect value was set in PathChoice.	At start At multiple start	■At start Does not start. ■At multiple start Stops the operation.	Set the value of PathChoice to one that corresponds to CircMode.
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 Does 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 Axes 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 Does not start. ■At multiple start Stops the operation.	• Set the configuration axis in LinearAxes with an axis that has been set up. • 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	Add-on SignalIO Acquisition Failure	Failed to acquire the add-on SignalIO.	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	Does not execute FB.	• Set the value of Master.AxisNo within the range. • Set to an axis No. for which an instance has been created.
1AD9H	Axis	Out of Slave Axis No. Range	The value is out of range for Slave.AxisNo.	At start When executing FB	Does not execute FB.	• Set the value of Slave.AxisNo within the range. • Set to an axis No. for which an instance has been created.
1ADAH	Axis	Out of BufferMode Range	The value is out of range for BufferMode.	At start When executing FB	Does not execute FB.	Set the value of BufferMode within the range.
1ADBH	System	Command Filter Add-on System Memory Shortage	System memory for the add-on is insufficient.	At start At multiple start	Does not start.	Please check the System.PrConst.Addon_MotionControl_AxisFilter.RamSizeMax.
1ADCH	System	Add-on Memory Shortage	System memory for add-ons related to operation is insufficient.	At start At multiple start	Does not start.	Please check the System.PrConst.Addon_*.RamSizeMax for related add-ons.
1ADEH	Axes group	Out of Linear Movement Amount Range	An interpolation axis for which the movement amount exceeds "4294967296.0 (= 2^{32})" exists during a linear interpolation with vector speed designated.	At start At multiple start	■At start Does not start. ■At multiple start Stops the operation.	Set the value of Position or Distance so that the movement amount of each interpolation axis does not exceed "4294967296.0 (= 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 software reboot after checking the System.PrConst.Addon_AbsSystem.BackupRamSizeMax.

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
1AE0H	Axis	Driver Restoration Data Incorrect	The parameter of connected driver device was incorrect when restoring current position. • Parameter of connected driver device was incorrect (Detail code: 0001H) • Restoration data of connected driver device was incorrect (Detail code: 0002H)	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	Does 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 Operation Cycle Converted Velocity Range	The velocity converted from Velocity by operation cycle is out of the range.	At start	Does not start.	Set Velocity so that the velocity converted by operation 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.	Please check the motor and encoder cable.
1AE6H	Axis	Driver Command Discard Detection	The command discard (Statusword bit12 OFF) of the connected 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 manual of each driver for details about 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	Velocity Range Over during Controlling	Set velocity exceeded the velocity command range.	During operation	Stops the operation.	Adjust settings so that the set velocity does not exceed the velocity 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	Does 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.

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
1AF0H	Axis	Advanced Synchronization Control Axis Unset	An invalid axis or an axis that has not been set in the advanced input/output axis setting was designated to the main shaft, auxiliary shaft, sub input axis and slave axis of advanced synchronous control. • Main Shaft (Detail code: 0001H) • Main Shaft Sub (Detail code: 0002H) • Auxiliary Shaft (Detail code: 0003H) • Slave Axis (Detail code: 0004H)	When executing FB	■At start Does not start.	Review the following settings of the axes corresponding to the detailed code. • Designate an axis that has been set in the advanced synchronous control input/ output axis setting to the main shaft, auxiliary shaft, sub input axis and slave axis of advanced synchronous control. • Correct the setting values if the advanced synchronous control input/output axis is outside the parameter range.
1AF1H	Axis	Advanced Synchronization Control Operation Overflow	Overflow occurred in any of the modules of the output axis of synchronous control. • Composite Main Shaft Gear (Detail code: 0001H) • Main Shaft Gear (Detail code: 0002H) • Main Shaft Clutch (Detail code: 0003H) • Main Shaft Module Speed Change Gear (Detail code: 0004H) • Auxiliary Shaft (Detail code: 0005H) • Auxiliary Shaft Gear (Detail code: 0006H) • Auxiliary Shaft Clutch (Detail code: 0007H) • Auxiliary Shaft Module Speed Change Gear (Detail code: 0008H) • Composite Auxiliary Shaft Gear (Detail code: 0009H) • Output Axis Speed Change Gear (Detail code: 000AH) • Output Axis Phase Compensation (Detail code: 000BH) • Output Axis Smoothing (Detail code: 000CH) • Output Axis Synchronous Control Change (Detail code: 000DH) • Output Axis Cam Operation (Detail code: 000EH)	During operation	Stops the operation.	Check the following settings of the modules corresponding to the detail codes. ■Common • Reduce the input axis velocity of the corresponding module. ■Main Shaft Gear/Auxiliary Shaft Gear • Reduce the gear ratio (gear numerator/gear denominator) of the corresponding module. ■Output Axis Smoothing • Review the setting value of the time constant to reduce the smoothing level, or increase the operation cycle. ■Composite Main Shaft Gear/ Composite Auxiliary Shaft Gear • Reduce the input values of the main and sub input axes of the main shaft. • Reduce the input values of the main shaft and auxiliary shaft. ■Output Axis Phase Compensation • Reduce the advance time or increase the time constant for phase compensation. ■Output Axis Synchronous Control Change • Reduce the synchronous control change value, or increase the synchronous control change reflection time. ■Output Axis Cam Operation • Review the cam waveform so that the stroke does not change rapidly.

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
1AF2H	Axis	Advanced Synchronous Control Restoration Disabled	■For current position per cycle restoration When the Cam Axis Position Restoration Object was "0: Current Position per Cycle Restoration", the current position per cycle corresponding to the value output at the start of the advanced synchronous control could not be restored. (Occurs in reciprocating cam patterns.)	At start	■At start Does not start.	■For current position per cycle restoration • Move the slave axis to make sure that the current value per cycle is within the stroke of the reciprocating cam pattern before starting the advanced synchronous control. • Set the cam reference position to make sure that the current value per cycle is within the stroke of the reciprocating cam pattern.
			■For cam set position restoration When the Cam Axis Position Restoration Object was "2: Cam Set Position Restoration, the difference between the restored output value and the set value of the slave axis at the start of the advanced synchronous control exceeded the value of the Cam Set Position Restoration: Allowable Movement Amount Setting, and the set position could not be restored.			■For cam set position restoration • Calculate the restored output value using the cam position calculation function and move the slave axis accordingly before starting advanced synchronous control. • Increase the Cam Set Position Restoration: Allowable Movement Amount Setting if it is extremely small, for example, 0.
1AF3H	Axis	Out of Parameter Range (Advanced Synchronous Control Output Axis)	Parameter is out of the range.	At start At multiple start	■At start Does not start. ■At multiple start Stops the operation.	Set the value within the range.
1AF4H	Axis	Out of Advanced Synchronous Control Output Axis Ring Counter range	If ring counter has been enabled for the output axis, the setting value of the ring counter is invalid.	At power ON	■At power ON Generation of the advanced synchronous control output axis fails.	Set the ring counter upper and lower limit values as either of the following. • Ring counter upper limit value = 10000000000.0, Ring counter lower limit value = -10000000000.0 • 0.0 < Ring counter upper limit value ≤ 2147483648.0, Ring counter lower limit value = 0.0
1AF5H	Axis/Axes group	FB Input Variable Error	When starting an FB, an error occurred due to any of the following causes. • PositioningData.ProfileID.Number is out of range. (Detail Code: 0003H) • StartDataNo is out of range. (Detail Code: 0004H) • StepMode is out of range. (Detail Code: 0005H)	At start At multiple start During control	■At start Does not start. ■At multiple start Stops the operation. ■During control Stops the operation.	• Set PositioningData.ProfileID.Number within the range. (Detail Code: 0003H) • Set StartDataNo within the range. (Detail Code: 0004H) • Set StepMode within the range. (Detail Code: 0005H)
1AF6H	Axis/Axes group	Add-on Acquisition Failure	Failed to acquire the necessary add-on for Axis Generation, operation.	At power ON At ready ON At start At multiple start	■At power ON/At ready ON Axis is not generated. ■At start Does not start. ■At multiple start Stops the operation.	Check the detailed information, then install the displayed add-on (*.mpk) and enable it.

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
1AF8H	Axis	Operation Profile Data Settings Incorrect Under Advanced Synchronous Control	Any of the following operation profile data settings have been set to values that cannot be used under advanced synchronous control. • Profile data type incorrect (Detail code: 0001H) • Operation profile data format incorrect (Detail code: 0002H) • The operation is not repetitive. (Detail code: 0003H) • The master axis (input) absolute coordinate is not set to relative coordinate. (Detail code: 0004H) • The slave axis (output) absolute coordinate is not set to relative coordinate. (Detail code: 0005H) • The starting point is not 0. (Detail code: 0006H) • The initial stroke is not 0. (Detail code: 0007H) • The cam starting point (AdvOutputName.Pr.Cam.StartingPoint) exceeded the resolution. (Detail code: 0008H) • The target operation profile data is being operated by another FB. (Detail code: 0009H) • The output coordinate is set to a value exceeding 32 bits. (Detail code: 000AH) • An output value calculated by extending the line segment between two coordinates exceeded 32 bits. (Detail code: 000BH)	At start At multiple start During control	■At start Does not start. ■At multiple start Stops the operation. ■During control Stops the operation.	• Set the profile data type to "1: Cam Data". (Detail code: 0001H) • Set the operation profile data format to "0: Linear Interpolation", "1: Section Interpolation", or "2: Spline Interpolation". (Detail code: 0002H) • Set repetitive operation to "1: Enable". (Detail code: 0003H) • Set master axis (input) absolute coordinate to "0: Relative Coordinate". (Detail code: 0004H) • Set slave axis (output) absolute coordinate to "0: Relative Coordinate". (Detail code: 0005H) • Set the starting point to "0". (Detail code: 0006H) • Set the initial stroke to "0". (Detail code: 0007H) • Set the cam starting point (AdvOutputName.Pr.Cam.StartingPoint) to a value smaller than the resolution. (Detail code: 0008H) • End the FB that is operating the target operation profile data. (Detail code: 0009H) • If the operation profile data format is "0: Linear Interpolation", correct the cam data so that no output values will exceed 32 bits. (Detail code: 000AH) • Adjust the slope between the two coordinates so that the output value calculated by extending the line segment between them does not exceed 32 bits within the length per cycle. (Detail code: 000BH)
1AF9H	Axis	Cycle Assignment Incorrect (Axis)	"Communication Period Setting" was set a period that can not be used in the control cycle of an axis.	At power ON At ready ON	Generating axis fails.	If "Set" is set for "Setting in Units of 1 µs" in "Basic Period Setting" of "Module Parameter (Network)", set one of the following in "Communication Period Interval Setting (Set it in Units of 1µs)". ■Earlier than version "32" 125.00 µs, 250.00 µs, 500.00 µs, 1000.00 µs, 2000.00 µs, 4000.00 µs, 8000.00 µs ■Version "32" or later 125.00 µs, 250.00 µs, 500.00 µs, 1000.00 µs, 1500.00 µs, 2000.00 µs, 2500.00 µs, 3000.00 µs, 3500.00 µs, 4000.00 µs, 4500.00 µs, 5000.00 µs, 5500.00 µs, 6000.00 µs, 6500.00 µs, 7000.00 µs, 7500.00 µs, 8000.00 µs

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
1AFAH	Axes group	Operation Profile Data Control Profile ID Setting Incorrect	The operation profile data specified in the PROFILE_ID type data is invalid due to any of the following causes. • It was not opened to the open area specified in the PROFILE_ID type data. (Detail Code: 0001H) • The type of the open area specified in the PROFILE_ID type data is invalid. (Detail Code: 0002H) • The target operation profile data is being operated by another FB. (Detail Code: 0003H) • The value is out of range for OperationPattern. (Detail Code: 1001H) • The value is out of range for ControlMethod. (Detail Code: 1002H) • The value is out of range for DwellTime. (Detail Code: 1003H) • The value is out of range for McodeOutput_Override. (Detail Code: 1004H) • The value is out of range for JumpDestinationDataNo. (Detail Code: 1005H) • The value is out of range for LoopCount. (Detail Code: 1006H) • The value is out of range for ConditionSignalNo or the signal has not been configured for the No. specified with ConditionSignalNo. (Detail Code: 1007H) • For some positioning data whose control method is JUMP, JumpDestinationDataNo has been set to a positioning data No. whose control method is also JUMP. (Detail Code: 1101H) • Positioning data of another control method was continuously executed exceeding the limit. (Detail Code: 1102H) • An invalid string was specified for ConditionSignalN (N: 1H to AH).Source.Target. (Detail Code: 1210H+N) • The value is out of range for ConditionSignalN (N: 1H to AH).Detection. (Detail Code: 1220H+N) • The value is out of range for ConditionSignalN (N: 1H to AH).CompensationTime. (Detail Code: 1230H+N) • The value is out of range for ConditionSignalN (N: 1H to AH).FilterTime. (Detail Code: 1240H+N)	At start At multiple start During control	■At start Does not start. ■At multiple start Stops the operation. ■During control Stops the operation.	• Open the operation profile data to the open area specified in the PROFILE_ID type data. (Detail Code: 0001H) • Specify an open area that is available for operation profile data control. (Detail Code: 0002H) ■MCv_MovePositioningData MC_POSITIONING_DATA_ID type • End the FB that is operating the target operation profile data. (Detail Code: 0003H) • Set OperationPattern within the range. (Detail Code: 1001H) • Set ControlMethod within the range. (Detail Code: 1002H) • Set DwellTime within the range. (Detail Code: 1003H) • Set McodeOutput_Override within the range. (Detail Code: 1004H) • Set JumpDestinationDataNo within the range. (Detail Code: 1005H) • Set LoopCount within the range. (Detail Code: 1006H) • Set ConditionSignalNo within the range or set the condition signal for the No. specified with ConditionSignalNo. (Detail Code: 1007H) • Set JumpDestinationDataNo to a positioning data No. whose control method is not JUMP. (Detail Code: 1101H) • Set the positioning data, which has another control method, so that it won't be continuously executed exceeding the limit. (Detail Code: 1102H) • Set ConditionSignalN (N: 1H to AH).Source.Target to a valid string. (Detail Code: 1210H+N) • Set ConditionSignalN (N: 1H to AH).Detection within the range. (Detail Code: 1220H+N) • Set ConditionSignalN (N: 1H to AH).CompensationTime within the range. (Detail Code: 1230H+N) • Set ConditionSignalN (N: 1H to AH).FilterTime within the range. (Detail Code: 1240H+N)

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
1C40H	System	Network Parameter Error (Total No. of Stations < Station No.)	■Add-on baseSystem version "1.15" or earlier The station number of the device station to connect to the network in "Network Configuration Settings" exceeds the total number of devicestations configured.	At power ON	The station can not be connected to the drive unit.	Set the station number of the device station to connect to the network to be less than or equal to the total number of device stations.
1C41H	Axis	Watchdog Counter Error	A watchdog counter error of driver was detected.	During communication	Stops immediately.	• When a stop error of the CPU module is detected, please check the error of the CPU module and take corrective action. • Please review the communication period. • Please check for errors in surrounding environment, e.g. noise, and remove the error factor. • After the above procedure, please switch the power off and then on, or reset the CPU module.
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 Communication 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 manual of the driver .
1C44H	System	Network Parameter Error (Station No. Order Error)	■Add-on baseSystem version "1.15" or earlier The station numbers of the device stations to connect to the network are not listed in ascending order in "Network Configuration Settings".	At power ON	The station can not be connected to the drive unit.	Please set the station numbers of the device stations to connect to the network in ascending order.
1C45H	System	SLMP Communication 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 set in the network configuration settings only.
1C47H	System	Communication Cycle Unsupported Driver Connection	An MR-J5(W)-G or MR-JET-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(W)-G or MR-JET-G amplifier and apply the latest version.
1C48H	System	PDO Mapping Setting Error	Abnormal response for PDO mapping setting was received from a device 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 (SDO Abort Code) and review the PDO mapping setting.
1C49H	System	Configuration Mismatch on Reconnection	A model name mismatch was detected on reconnection of device station.	At start communication	The station can not be connected to the drive unit.	Please connect the device station with the same model name as the disconnected device station.

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
1C4CH	System	Power Not Turned ON After Parameter Distribution	After master-slave operation was enabled, the device station parameter automatic setting was performed.	At power ON	The station can not be connected to the drive unit.	Start the system again.
1C4DH	System	Number of Master Axes Error	The number of master axes, set with the servo parameter "Driver communication setting (PD15.0)" in the device station parameter automatic setting, exceeds the number of settable axes.	At power ON	The station can not be connected to the drive unit.	Keep the number of master axes, set with the servo parameter "Driver communication setting (PD15.0)" in the device station parameter automatic setting, at or below the number of settable axes.
1C4EH	System	Master Axis No. Error	The station No. of an own axis is specified for the servo parameter "Driver communication setting - Slave - Master axis 1 - Station No. setting (PD22)" in the device station parameter automatic setting.	At power ON	The station can not be connected to the drive unit.	Review the master axis No. specified for the servo parameter "Driver communication setting - Slave - Master axis 1 - Station No. setting (PD22)" in the device station parameter automatic setting.
1C4FH	System	Master Axis Setting Error	The station No. of the master axis has not been specified for the servo parameter "Driver communication setting - Slave - Master axis 1 - Station No. setting (PD22)" in the device station parameter automatic setting.	At power ON	The station can not be connected to the drive unit.	Review the master axis No. specified for the servo parameter "Driver communication setting - Slave - Master axis 1 - Station No. setting (PD22)" in the device station parameter automatic setting.
1C50H	System	Driver Communication Setting Error	Driver communication settings of the Motion module and servo amplifier are different.	At power ON	The station can not be connected to the drive unit.	• Check the driver communication settings and the actually connected servo amplifier. • After setting the module type and configuring the servo parameters "Driver communication setting (PD15.0), (PD15.1)", "Driver communication setting - Slave - Master axis 1 - Station No. setting (PD22)", and "Driver communication setting - Slave - Master axis 1 - Control slave axis No. setting (PD23.1)" in the device station parameter automatic setting, write the settings you made to the project, then switch the power off and on again or reset the PLC. After that, power the servo amplifier off and then on, and power cycle or reset the PLC again.
1C51H	Axis	Slave Axis Specification Error	The station address of the slave axis was set in the station address setting.	When connecting driver	Unable to connect to the driver.	Set a station address other than the slave axis in the station address setting.

23

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
1C80H	System/ Axis	Cycle Over	Operation 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 communication cycle interval setting or buffer memory refresh cycle to a larger value. • When using by a operation cycle of 31.25 μs, please begin with a minimum configuration of add-ons, and then add more add-ons as required according to the load of the system. • When the inter-module synchronization function is enabled, please be sure to execute the EI instruction with the PLC to enable I44 interrupt programs.
1C81H	System	Cycle Assignment Incorrect	The value is out of range for system basic cycle, the first operation 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.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.
1C85H	System	Event History File Incorrect	• An error was detected in the event history file. • The memory capacity of add-on MotionEventHist is insufficient. • The drive specified for System.PrConst.EventHistoryMotion_Path is not mounted.	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 a dd-on MotionEventHist. • Please check the status of the target drive (System.Md.Storage_*).
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.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 settings. • If the device label import error occurred, please refer to the corrective action for device label import error.
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 described in the detailed information is invalid function.	Please re-install the baseSystem.smpk
1C8FH	System	Add-on SignalIO Acquisition Failure	Failed to acquire the add-on SignalIO.	At power ON At ready ON	■At power ON Generating axis fails. ■At ready ON The READY does not turn ON.	Install SignalIO.mpk and enable it.

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
1C91H	System	Insufficient Memory Capacity of Add-on SignalIO	Cannot create a signal, because the memory capacity of the add-on SignalIO is insufficient.	At power ON At ready ON	■At power ON Generating axis fails. ■At ready ON] The READY does not turn ON.	• Increase the memory capacity of add-on SignalIO. • Increase the memory capacity of add-on ExternalSignal.
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.
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. • A moderate error occurred in the motion module.	At power ON	Interrupts the process to import the labels.	• Please check the settings of network configuration and then create device labels again. • Refer to the remedy for the moderate error occurred in the motion module.
1C99H	System	Backup Data Saving Failure	• Failed to save the backup data because of insufficient capacity in the backup data write area. • The backup data is corrupted.	During operation	Backup data are not saved.	• Please delete unnecessary cam data and positioning data. • Please execute parameter initialization and flash ROM write.
1C9BH	System	Advance Synchronous Control Add-on Memory Shortage	System memory for add-on AdvancedSync is insufficient. • Failed to create the input/output data. (Detail code: 0001H) • Failed to create the backup data. (Detail code: 0002H)	At power ON	The add-on is invalid function.	Increase the memory capacity of add-on AdvancedSync. Delete unnecessary labels and reduce memory usage.

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
1C9CH	System	Advanced Synchronous Control Smoothing Memory Shortage	The smoothing memory could not be secured because the system memory of add-on AdvancedSync is insufficient. The memory could not be secured for the smoothing time constants corresponding to the detail codes shown below. • PrConst.SmoothingTimeConstant (Detail code: 0001H) • PrConst.MasterSpeedChangeGearSmoothingTimeConstant (Detail code: 0002H) • PrConst.MasterClutchSmoothingTimeConstant (Detail code: 0003H) • PrConst.AuxSpeedChangeGearSmoothingTimeConstant (Detail code: 0004H) • PrConst.AuxClutchSmoothingTimeConstant (Detail code: 0005H) • PrConst.OutSpeedChangeGearSmoothingTimeConstant (Detail code: 0006H)	At power ON At ready ON	The corresponding input axis/output axis becomes invalid.	• Increase the memory capacity of add-on AdvancedSync. • Review the setting value of the time constant to reduce the smoothing level, or increase the operation cycle.
1C9DH	System	Advanced Synchronous Control Input Axis Incorrect	Invalid values have been set in axis parameters of the input axes of advanced synchronous control. (Refer to the detail code in event history for the cause.) • The label ID of the input axis of advanced synchronous control is duplicated. (Detail code: 0001H) • The axis No. of the input axis of advanced synchronous control is duplicated. (Detail code: 0002H)	At power ON	■At power ON Generation of the advanced synchronous control input axis fails.	• Review the settings so that the label IDs of the input axis for advanced synchronization control do not overlap. • Review the settings so that the axis Nos. of the input axis for advanced synchronization control do not overlap.
1C9EH	System	Advanced Synchronous Control Output Axis Incorrect	An invalid value is set in axis parameters of the output axis of advanced synchronous control. (Refer to the detail code in event history for the cause.) • The label ID of the output axis of advanced synchronous control is duplicated. (Detail code: 0001H) • The axis No. of the output axis of advanced synchronous control is duplicated. (Detail code: 0002H) • An invalid axis type (real encoder axis/virtual encoder axis) is set for the output axis of advanced synchronous control. (Detail code: 0003H)	At power ON	■At power ON Generation of the advanced synchronous control output axis fails.	• Set the label ID of the output axis of advanced synchronous control avoiding duplication. • Set the axis No. of the output axis of advanced synchronous control avoiding duplication. • Set a valid axis type for the output axis of advanced synchronous control.
1CAFH	System	Add-on Error (Minor)	An add-on error occurred.	Depends on the add-on	Depends on the add-on	Check the manual of the add-on for details about the detail code.
1D80H	System	Out of Parameter Range (Axis)	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 Generating axis fails. ■At ready ON The READY does not turn ON.	Change the specified value of the parameter in detailed information within the range.

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
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 does not turn ON.	Change the specified value of the parameter in detailed information within the range.
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 does not turn ON.	Change the specified value of the parameter in detailed information within the range.
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.
1D85H	System	Out of Parameter Range (Advanced Synchronous Control Input Axis)	Parameter is out of the range.	At power ON	■At power ON Generation of the advanced synchronous control input axis fails.	Set the value within the range.
1D86H	System	Out of Parameter Range (Advanced Synchronous Control Output Axis)	Parameter is out of the range.	At power ON	■At power ON Generation of the advanced synchronous control output axis fails.	Set the value within the range.
2300H	System	Security Key Authentication Error	The security key which locks the program does not match the key written to the motion system.	At ready ON	The program stops.	Please review the security key.
2301H	System	Security Key Authentication Error	Program is locked by the security key, but there is no security key written to the motion system.	At ready ON	The program stops.	Please review the security key.
2302H	System	Security Key Authentication Error	• The security key set in the file is corrupted and does not match the key written to the motion system. • The security key set in the motion system is corrupted and does not match the key set in the file.	At ready ON	The program stops.	Please write files to the motion again. If the same error occurs again even after taking the above, the possible cause is a hardware failure of the motion system. Please consult your local Mitsubishi representative.
3200H	System	File Access Error	The following operations were performed via engineering tool or SLMP. • An access (read/write) from another request source to the file which was being written was executed. • Writing from another request source to the file which was being accessed (read/write) was executed.	When accessing file	Fails to access the files.	Please execute the next access after the processing of one request source is completed.

23

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
3203H	System	WDT Error Detected	The following errors were detected in the software or hardware. <Software> ■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 Operation processing or other processing was in abnormal conditions. Processing stopped for more than 1 [s]. <Hardware> An error was detected in the hardware.	During operation	• The axis in operation stops immediately or decelerate stops. • Stops the system.	Please take the following actions. <Software> ■Programming Reduce the number of command execution of initial execution type and normal execution type by program. ■Execution time monitor • Change the communication cycle interval setting or buffer memory refresh cycle to a larger value. • Change the fixed scan interval setting of fixed scan execution type program to a larger value. <Hardware> • Please take measures to reduce noise. • Please reset the Motion module. If the same error occurs again even after taking the above, the possible cause is a hardware failure of the Motion module. Please consult your local Mitsubishi representative.
3205H	System	Add-on Library Load Error	Add-on • The add-on file is broken. • The add-on on which it is dependent has not been installed. • An add-on that cannot be combined with this one has been installed. • There is an inconsistency in the versions.	At power ON	Add-on The motion system does not start.	Add-on • Please install the software again. • Please install the add-on on which it is dependent. • Please delete the add-on that cannot be combined. • Please install the correct version of the add-on.
3207H	System	Drive Error	• Failed to mount the drive. • Check disk ended with error. • A 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 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	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
320CH	System	Cycle Over	Operation processing or other fixed scan processing is not completed within the set cycle.	During operation	The axis in operation stops immediately or decelerate stops.	• Change the fixed scan interval setting of fixed scan execution type program to a larger value. • Change the communication cycle interval setting or buffer memory refresh cycle to a larger value. • When using by a operation cycle of 31.25 μs, please begin with a minimum configuration of add-ons, and then add more add-ons as required according to the load of the system. • When the inter-module synchronization function is enabled, please be sure to execute the EI instruction with the PLC to enable I44 interrupt programs.
320DH	System	Base System Error	• Failed to configure labels. • Failed in initial process.	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 in communication between motion area and network area.	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.
3217H	System	Add-on SimpleMotion Acquisition Failure	Failed to acquire the add-on SimpleMotion.	At power ON	The motion system does not start.	Install SimpleMotion.mpk and enable it.
3218H	System	Add-on SimpleMotion Internal Error	Failed in initial process of the add-on SimpleMotion.	At power ON	The motion system does not start.	Please install the software again.
3219H	System	Add-on SimpleMotion Memory Shortage	System memory for the add-on SimpleMotion is insufficient.	At power ON	The motion system does not start.	Please install the software again.
3228H	System	Standard Station Communication Cycle Combination Error	After setting the device station as a standard station in "Network Configuration Settings", a value less than 125 μs was set for "Communication Period Interval Setting".	At power ON	• Data link does not start. • The READY does not turn ON.	In "Network Configuration Settings", set the device station that is currently a standard station as a motion control station, or set "Communication Period Interval Setting" to a value equal to or more than 125 μs.
3229H	System	Motion Synchronization Station Send/Receive Data Refresh Method Setting Combination Error	System.PrConst.Link_MotionStationRefreshType (motion synchronization station send/ receive data refresh method) is set to 0 (respond preferred method) while Communication Period Setting is set to 31.25 μs.	At power ON	• Unable to connect to the driver. • Refresh of the slave label is not executed.	Set "Communication Period Setting" to 62.5 μs or more, or change the System.PrConst.Link_MotionStationRefreshType (motion synchronization station send/ receive data refresh method) to 1 (operation cycle preferred method).

23

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
322AH	System	Communication Period Interval Setting Combination Error	The following settings were made after setting a value less than 125 μs for "Communication Period Interval Setting". • "Multicast" was set in "Communication Mode". • "Ring" was set in "Network Topology".Set a value equal to or more than 125 μs for	At power ON	• Data link does not start. • The READY does not turn ON.	Set "Communication Period Interval Setting" to a value equal to or more than 125 μs, or perform the following settings. • Set "Unicast" to "Communication mode". • Set "Line or Star or mixture of Line/Star" to "Network Topology".
322BH	System	Motion Control Station Setting Error	Any of the items in "Network Configuration Settings" under "Basic Settings" are set as follows. • A local station was set as the motion control station. • The communication period setting of the remote station designated as the motion control station was set to other than "Basic Period".	At power ON	• Data link does not start. • The READY does not turn ON.	Please correct the parameters shown in the cause.
322CH	System	Link Device Setting Error	The following settings were made while the start/end of the link device for the device station which was set as the motion control station in "Network Configuration Settings" had not been set. • A local station was set in "Network Configuration Settings". • "Communication Mode" was set to "Multicast".	At power ON	• Data link does not start. • The READY does not turn ON.	Through GX Works3 version 1.085P or later, set the start/end of the link device for the station which is set as the motion control station in network configuration settings, or make the following settings. • Change the device station set as the motion control station in "Network Configuration Settings" to be a standard station. • Remove local stations form "Network Configuration Settings". • Set "Unicast" to "Communication Mode".
322FH	System	Add-on Error (Moderate)	An add-on error occurred	Depends on the add-on	Depends on the add-on	Check the manual of the add-on for details about the detail code.

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
3230H	System	Driver Communication Setting Error 1	• There are abnormal values in the driver communication settings (servo parameters in the device station parameter automatic setting). • There are errors in the network configuration settings.	At power ON	Data link does not start.	• Please check the driver communication settings (servo parameters in the device station parameter automatic setting). • Please set the number of slave axes drove by the master axis within the range of 1 to 8. • Please do not enable driver communication for multi-axis servo amplifiers. • Please set the servo parameter "Driver communication setting - Slave - Master axis 1 - Control slave axis No. setting (PD23.1)" of each slave axis belonging to the same master axis avoiding duplication. • Please do not enable driver communication for stations for which the station-specific mode setting in the network configuration settings has been set to "Motion Mode (Safety)". • Please set the master axis to a real drive axis, and set slave axes to motion control stations. • Please specify an authentication Class B device for a master-slave operation enabled axis. • Please review the Master-slave Operation - Slave Axis Station Address Setting. • Please write the project where master-slave operation has been enabled.
3231H	System	Driver Communication Setting Error 2	The communication cycle or communication speed setting is not supported by driver communication.	At power ON	Data link does not start.	• Set the factor of "Low-Speed" in "Multiple Period Setting" to be 128 times or less after multiplying by the driver communication factor (basic). • Please set the communication speed to 1 Gbps. • Please invalidate the inter-module synchronization. • To connect a CC-Link IE TSN Class A device station, please set the "TSN HUB Setting" to "Use TSN HUB".
3233H	System	File Error	Parameter settings are corrupted.	When importing file	The file is not imported. The motion system does not start.	Please write the program again.
3234H	System	Ring Topology Unsupported Station Detection	A station that did not support ring topology was detected.	At start communication	Does not data link.	Update stations that do not support ring topology to a version that supports ring topology. Alternatively, set "Line", "Star", or "Line/Star" in "Network Topology" under "Basic Settings" of the master station.

23

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
3235H	System	Driver Communication Cyclic Transmission Time Setting Error	The cyclic transmission time setting does not support driver communication function.	At power ON	Data link does not start.	Set the value of "Cyclic Transmission Time" in "Communication Period Setting" in "Basic Settings" of the master station to not less than Cyclic transmission time of detailed information and not more than 216 μs. If the cyclic transmission time of detailed information exceeds 216 μs, reduce the number of stations connected to the master station and the number of link devices allocated to each device in "Network Configuration Settings".
3236H	System	Driver Communication Period Interval Setting Error	"Communication Period Interval Setting" does not support Driver communication function.	At power ON	Data link does not start.	In "Communication Cycle Setting" of "Basic Settings" of the master station, set the value of "Communication Cycle Period Interval" shall be set so that it is greater than or equal to the communication cycle interval of detailed information and less than or equal to 4000 μs, and one of the following shall apply. • 125 μs • 250 μs • 500 μs • 1000 μs • 2000 μs • 4000 μs If the communication cycle interval for detailed information exceeds 4000 μs, do one of the following. • Decrease the "Cyclic Transmission Time" setting. • Reduce the number of devices connected to the master station and the number of link devices allocated to each unit in "Network Configuration Settings".
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	Does 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	Does 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	Does 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	Does 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	Does not execute FB.	• Set the value within the range of 1 to 10000. • Set the axis No. in which instance is created.

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
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	Does not execute FB.	• Set the value within the range of 1 to 10000. • Set the axis No. in which instance is created.
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	Does 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 RecordCount.	When executing FB	Does not execute FB.	Set the value within the range.
3408H	System	Latch Count Incorrect	The value has been out of range for RecordCount when RecodeMode=RecordCount, RingBuffer.	When executing FB	Does not execute FB.	Set the value of RecordCount within the range.
3409H	System	Out of Touch Probe ID Range	The value is out of range for TouchProbeID.	When executing FB	Does 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	Does 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 RecordMode.	When executing FB	Does not execute FB.	Set the value of MC_RECORD_MODE within the range.
340FH	System	No Operation Profile Data	• PROFILE_DATA.Location is not set. • The operation profile data does not exist in PROFILE_DATA.Location.	When executing FB During operation	• Does not execute FB. • The execution of FB is interrupted.	Set the operation profile data storage location for PROFILE_DATA.Location.
3410H	System	Operation Profile Data Incorrect	• The value is out of range for the data. • Add-on that supports the operation profile data 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 operation profile data format.
3411H	System	Operation Profile Data Being Operated	Operation was executed while the operation profile data was being operated by another FB.	During operation	The execution of FB is interrupted.	Operate it again after another FB is completed (Busy of FB is FALSE.).
3412H	System	Operation Profile Data Error	The operation profile data is broken, or the format is incorrect.	During operation	The execution of FB is interrupted.	Set the operation profile data in a correct format.
3413H	System	Out of Offset Range	Offset exceeds the number of elements of the operation profile data (in a file or open area).	During operation	The execution of FB is interrupted.	Set Offset to a value that does not exceed the number of elements of the operation profile data (in a file or open area).
3414H	System	Out of Operation Profile Data ID Range	Set the value of ProfileData.ID.Number or ProfileID.Number within the range.	At start During operation	■At start Does not execute FB. ■During operation Operates with previous setting value.	Set the value of ProfileData.ID.Number or ProfileID.Number within the range.
341DH	System	Out of MasterAbsolute Range	The value is out of range for MasterAbsolute.	When executing FB	Does not execute FB.	Set the value within the range.
341EH	System	Out of SlaveAbsolute Range	The value is out of range for SlaveAbsolute.	When executing FB	Does not execute FB.	Set the value within the range.
342FH	System	SD Memory Card Detached	The SD memory card was removed without being disabled	When removing SD memory card	—	Remove the SD memory card after disabling it.
343FH	System	Out of Backlash Amount Range	The value is out of range for BacklashAmount.	When executing FB	Does not execute FB.	Set the value within the range.

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
344EH	System	MC_SetPosition Instruction Error	During the execution of MC_SetPosition, an error occurred due to any of 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) • Another MC_SetPosition was executed for the specified axis (Detail Code: 0008H) • Options is out of range (Detail Code: 0009H)	At start During operation	• Does 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.
3450H	System	MCv_SetTorqueLimit Instruction Error	During the execution of MCv_SetTorqueLimit, an error occurred due to any of the 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) • Another MCv_SetTorqueLimit was executed for the specified axis. (Detail Code: 0006H) • Options is out of range. (Detail Code: 0007H)	At start During operation	Does 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	Does not execute FB.	Set the value of ExecutionMode within the range.
3452H	System	Insufficient Operation Profile Data 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	Operation Profile Data Control Unsupported	Control FB was executed for the unsupported operation profile data.	At start	Does not execute FB.	Open operation profile data which supports the operation profile data control FB in the open area.
345EH	System	Operation Profile Data Read/Write Unsupported	Operation profile data which does not support operation profile data read/write is set.	During operation	The execution of FB is interrupted.	Set operation profile data 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	• An incorrect type is set in Data1 or Data2. • Add-on that supports the operation profile data 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 operation profile data 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.

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
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	• Values other than Offset = 0 and Points = 0 were set when read target/write target was a file. • Points were set to 0 when the writing destination was the open area and Offset wasn't set to 0. • The source DataNo. isn't 0 or different from the writing destination DataNo. when writing multiple axes positioning data to the open area.	During operation	The execution of FB is interrupted.	• Set Offset = 0 and Points = 0 when the read target/write target was a file. • Set Points to 0 when the writing destination was the open area and Offset wasn't set to 0. • Set the writing source DataNo. to 0 or set the same value as the writing destination DataNo. when writing multiple axes positioning data to the open area.
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 the value of Cycle within the range.
3467H	System	Current Value per Cycle Change Unsupported	FB which does not support the current value per cycle change is set.	At start	Does 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	Does not execute FB.	Set an existing instance ID.
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.
3492H	System	Operation Profile Data 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	Operation Profile Data Add-on Internal Error	An error occurred in internal of operation profile data 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 Enabled)	MC_GroupEnable was executed when AxisName.Md.AxisStatus of a configuration axis is other than Standstill or Disabled.	When Axes Group is Valid	The axes group status does not turn to "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 Enabled)	MC_GroupEnable was executed when AxesGroupName.Md.GroupStatus is other than GroupStandby or GroupDisabled.	When Axes Group is Valid	Does 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.

23

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
3497H	System	MCv_ChangeCycle Instruction Error	During the execution of MCv_ChangeCycle, an error occurred due to any of the following causes. • Cycle is out of range. (Detail Code: 0001H) • ExecutionMode is out of range. (Detail Code: 0002H) • MCv_ChangeCycle was executed when the axis status was SyncronizedMotion. (Detail Code: 0004H) • Another MCv_ChangeCycle was executed with the specified instanceID. (Detail Code: 0008H) • Options is out of range. (Detail Code: 0009H) • MCv_ChangeCycle was executed when the axis status was Standstill. (Detail Code: 000AH)	At start During operation	• Does not execute FB. • The execution of FB is interrupted.	• Set Cycle within the range. (Detail Code: 0001H) • Set ExecutionMode within the range. (Detail Code: 0002H) • Execute it when the axis status is Standstill. (Detail Code: 0004H) • Execute another MCv_ChangeCycle after the current one is completed. (Detail Code: 0008H) • Set Options within the range. (Detail Code: 0009H) • Execute it when the axis status is SyncronizedMotion (Detail Code: 000AH)
3499H	System	Add-on Memory Shortage	System memory for the related add-ons is insufficient.	At start up	Does not execute FB.	Please increase System.PrConst.Addon_*.Ram SizeMax 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 Factor (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 Factor (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 Factor (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	Does not execute FB.	• Set the number of elements specified for OutputBuffer to be more than RecordCount. • Adjust 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	Does 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	Does 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	Does 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	Does not execute FB	Set the value within the range.
34A4H	System	Touch Probe Disabling Unsupported	FB which does not support the touch probe disabling is set.	When executing FB	Does not execute FB.	Set FB which supports the touch probe disabling.

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
34A5H	System	Signal Target Incorrect	• An incorrect string was set for the Target of the signal. • An unconnected or non-existent station was set when [OBJ] was specified. • A non-existent object was set for TPDO/RPDO when [OBJ] was specified.	When executing FB	Does not execute FB.	• Set to a valid string for the Target of the signal. • Connect the target station if [OBJ] has been specified. • Set the target object for TPDO/RPDO if [OBJ] has been specified.
34A6H	System	Out of Detection Setting Range of Signal	The value is out of range for Detection.	When executing FB	Does not execute FB.	Set the value of Detection within the range.
34A7H	System	Out of CompensationTime Setting Range of Signal	The value is out of range for CompensationTime.	When executing FB	Does not execute FB.	Set the value of CompensationTime within the range.
34A8H	System	Out of FilterTime Setting Range of Signal	The value is out of range for FilterTime.	When executing FB	Does not execute FB.	Set FilterTime within the range.
34ABH	System	Add-on Acquisition Failure	Failed to acquire the necessary add-on for FB execution.	When executing FB	Does not execute FB.	Check the detailed information, then install the displayed add-on (*.mpk) and enable it.
34ACH	System	MCv_AdvCamPositionPerCycleCalc Instruction Error	During the execution of MCv_AdvCamPositionPerCycleCalc, an error occurred due to any of the following causes. • RingCountUpperValue is out of range. (Detail Code: 0001H) • LengthPerCycle is out of range. (Detail Code: 0002H) • StrokeAmount is out of range. (Detail Code: 0003H) • PositionPerCycle is out of range. (Detail Code: 0004H) • ReferenceSetPosition is out of range. (Detail Code: 0005H) • SetPosition is out of range. (Detail Code: 0006H) • StartingPoint is out of range. (Detail Code: 0007H)	During operation	The execution of FB is interrupted.	• Set RingCountUpperValue within the range. (Detail Code: 0001H) • Set LengthPerCycle within the range. (Detail Code: 0002H) • Set StrokeAmount within the range. (Detail Code: 0003H) • Set the values so that PositionPerCycle < LengthPerCycle. (Detail Code: 0004H) • Set ReferenceSetPosition within the range. (Detail Code: 0005H) • Set SetPosition within the range. (Detail Code: 0006H) • Set StartingPoint within the range. (Detail Code: 0007H)
34ADH	System	MCv_AdvCamSetPositionCalc Instruction Error	During the execution of MCv_AdvCamSetPositionCalc, an error occurred due to any of the following causes. • RingCountUpperValue is out of range. (Detail Code: 0001H) • LengthPerCycle is out of range. (Detail Code: 0002H) • StrokeAmount is out of range. (Detail Code: 0003H) • PositionPerCycle is out of range. (Detail Code: 0004H) • ReferenceSetPosition is out of range. (Detail Code: 0005H) • StartingPoint is out of range. (Detail Code: 0007H)	During operation	The execution of FB is interrupted.	• Set RingCountUpperValue within the range. (Detail Code: 0001H) • Set LengthPerCycle within the range. (Detail Code: 0002H) • Set StrokeAmount within the range. (Detail Code: 0003H) • Set the values so that PositionPerCycle < LengthPerCycle. (Detail Code: 0004H) • Set ReferenceSetPosition within the range. (Detail Code: 0005H) • Set StartingPoint within the range. (Detail Code: 0007H)
34AFH	System	Unable to Calculate Cam Position Under Advanced Synchronous Control	The corresponding current position per cycle could not be calculated during the current value per cycle calculation. (Occurs in reciprocating cam patterns.)	During operation	The execution of FB is interrupted.	Set "StrokeAmount", "ReferenceSetPosition", and "SetPosition" within the stroke range of the reciprocating cam pattern.

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
34B0H	System	Operation Profile Data Settings Incorrect for Cam Position Under Advanced Synchronous Control	Any of the following operation profile data settings have been set to values that cannot be used for calculating cam position. • Profile data type incorrect (Detail code: 0001H) • Operation profile data format incorrect (Detail code: 0002H) • The operation is not repetitive. (Detail code: 0003H) • The master axis (input) absolute coordinate is not set to relative coordinate. (Detail code: 0004H) • The slave axis (output) absolute coordinate is not set to relative coordinate. (Detail code: 0005H) • The starting point is not 0. (Detail code: 0006H) • The initial stroke is not 0. (Detail code: 0007H) • The cam starting point for cam position calculation (StartingPoint) exceeded the resolution. (Detail code: 0008H) • The target operation profile data is being operated by another FB. (Detail code: 0009H) • The output coordinate is set to a value exceeding 32 bits. (Detail code: 000AH) • An output value calculated by extending the line segment between two coordinates exceeded 32 bits. (Detail code: 000BH)	During operation	The execution of FB is interrupted.	• Set the profile data type to "1: Cam Data". (Detail code: 0001H) • Set the operation profile data format to "0: Linear Interpolation", "1: Section Interpolation", or "2: Spline Interpolation". (Detail code: 0002H) • Set repetitive operation to "1: Enable". (Detail code: 0003H) • Set master axis (input) absolute coordinate to "0: Relative Coordinate". (Detail code: 0004H) • Set slave axis (output) absolute coordinate to "0: Relative Coordinate". (Detail code: 0005H) • Set the starting point to "0". (Detail code: 0006H) • Set the initial stroke to "0". (Detail code: 0007H) • Set the cam starting point for cam position calculation (StartingPoint) to a value smaller than the resolution. (Detail code: 0008H) • End the FB that is operating the target operation profile data. (Detail code: 0009H) • If the operation profile data format is "0: Linear Interpolation", correct the cam data so that no output values will exceed 32 bits. (Detail code: 000AH) • Adjust the slope between the two coordinates so that the output value calculated by extending the line segment between them does not exceed 32 bits within the length per cycle. (Detail code: 000BH)
34B2H	System	MCv_MovePositioning Data Instruction Error	When MCv_MovePositioningData was executed or being executed, an error occurred. The detected positioning data No. (data No.) and the related error code (related event code) are stored in the detailed information (operation profile data control information).	When executing FB During FB execution	Does not execute FB. The execution of FB is interrupted.	• Review the settings of the positioning data No. where the error occurred. • Check the corrective actions for the related error code.
34F8H	System	Device/Label Specification Incorrect	A 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 Called 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.

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
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 is not correctly written.	When executing instruction	Interrupts importing the files.	Please write the program again.
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.

23

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
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 an 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 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 location.	At ready ON	The program stops.	Please check the memory capacity of ST add-on.
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	A 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.

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
3513H	System	Parameter Error	The "Fixed Scan Interval Setting" of the program whose "Execution Type" is set to "3: Fixed Scan" in "Program Execution Setting" is one of the following. • Values out of the range • Other than multiplication by n of the first operation cycle	At ready ON	The program stops.	• Set the program's "Fixed Scan Interval Setting" to satisfy all of the following. • From the first operation cycle to 60,000 ms. • Multiplication by n of the first operation cycle. • If you want the program to operate according to the "Fixed Scan Interval Setting" you set, review the "Communication Period Setting" in "Module Parameter (Network)".
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	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.
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 Motion module. If the same error occurs again even after taking the above, the possible cause is a hardware failure of the Motion 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 Motion module. If the same error occurs again even after taking the above, the possible cause is a hardware failure of the Motion 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 Motion module. If the same error occurs again even after taking the above, the possible cause is a hardware failure of the Motion module. Please consult your local Mitsubishi representative.

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
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 Motion module. If the same error occurs again even after taking the above, the possible cause is a hardware failure of the Motion module. Please consult your local Mitsubishi representative.
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 Motion module. If the same error occurs again even after taking the above, the possible cause is a hardware failure of the Motion 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 Motion module. If the same error occurs again even after taking the above, the possible cause is a hardware failure of the Motion 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 Motion module. If the same error occurs again even after taking the above, the possible cause is a hardware failure of the Motion module. Please consult your local Mitsubishi representative.
3C13H	System	Hardware Error	An error was detected in the hardware.	During operation	Stops the system.	• Please take measures to reduce noise. • Please reset the motion module. If the same error occurs again even after taking the above, the possible cause is a hardware failure of the motion 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 Motion module. If the same error occurs again even after taking the above, the possible cause is a hardware failure of the Motion module. Please consult your local Mitsubishi representative.

Error code	Error	Error name	Error detail and cause	Detection timing	Operation at error occurrence	Action
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. After that, please write all files and reset the Motion module. If the same error occurs again even after taking the above, the possible cause is a hardware failure of the Motion 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.
3F40H	System	Hardware Error	An error was detected in the hardware.	At initialization	Stops the system.	• Please take measures to reduce noise. • Please reset the Motion module. If the same error occurs again even after taking the above, the possible cause is a hardware failure of the Motion module. Please consult your local Mitsubishi representative.

23.7 List of Logging Error Codes

Error code	Cause	Action
4008H	The system is in the BUSY state.	• Execute it again after a certain period of time has elapsed.*1
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 it again after the module RUN when using local labels.*1
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.*1
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.*1
41CEH	The specified file is read only and write protected.	• The specified file is write protected. Check the attributes and then execute it again.*1
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.*1
41D2H	Unsupported function.	• Check if the related add-on is installed. • Add-on SignalIO is required to use devices or labels. • Add-on MotionEventHist 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.*1
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.*1
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.*1
4C00H	There is not enough free space on target memory for result file creation.	• Increase free space of target memory, then execute it again.*1
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.*1
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.*1
4C56H	• Failed to read (access) file. • The specified file does not exist. Or the specified sub directory does not exist.	• Check the file is deleted or not. • Check the file name or sub directory name and then execute it again.*1
4F01H	An error in the data type.	Recheck the data to be collected.

*1 For the method of re-executing the servo system recorder logging, refer to the following.
☞ Page 699 Restart from the logging interruption

23.8 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
06FB	Device station parameter automatic setting completion	Device station parameters were automatically set from the master station.
06FC	Device station parameter backup completion	Device station parameters were backed up to the master station.
06FD	Master-slave Operation - Axes Detection Started/Completed	• The detection of master-slave operation axes was started. (Detail Code: 0000H) • The detection of master-slave operation axes was completed. (Detail Code: 0001H)
06FF	Add-on Event 1	An add-on event occurred.
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 canceled.
07F0	Motion Control FB Start (Administrative)	A motion control FB (administrative) was started.
07F1	Motion Control FB Start (Motion)	A motion control FB (motion) was started or multiply started.
07F2	Motion Control FB (General)	A motion control FB (general) was started.
07F3	Motion Control FB Stop	A motion control FB was stopped by stop cause.
07F4	File Transfer Start	File transfer was started by script file/label.
07F5	Start when SW Stroke Limit is Disabled	Started when software stroke limit was disabled by Software Stroke Limit Override (AxisName.Cd.SwStrokeLimit_Override).
07F6	Start when HW Stroke Limit is Disabled	Started when hardware stroke limit was disabled by Hardware Stroke Limit Override (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 Temporarily Invalid	Follow-up was switched to invalid.
07FC	Servo ON when Follow-up is Invalid	Changed it to servo ON in follow-up invalid status during servo OFF.
07FD	Current Position Restoration Completion	Current position restoration was completed.
07FE	Homing Request OFF to ON	Homing request is ON. • Absolute position erased on driver side was detected. [Real drive axis only] (Detail Code: 0000H) • An absolute position data error in the Motion module was detected. (Detail Code: 0001H) • The axis type was changed. (Detail Code: 0002H) • Slave object "Polarity (607EH)" b7: position polarity was changed. (Detail Code: 0003H) • Change of the driver or motor encoder was detected (Detail Code: 0004H) • Machine homing was started. (Detail Code: 0005H) • Driver unit conversion (numerator/denominator) of the axis was changed. (Detail Code: 0006H) • Machine homing has never been executed after the 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 the encoder ring counter (encoder resolution) was changed. (Detail Code: 0009H)) • Slave object "Home offset (607CH)" was changed. (Detail Code: 000AH)
07FF	Add-on Event 2	An add-on event occurred.

23

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 an IP restricted with IP filter setting	It was accessed from an 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
20300	SD Memory Card Usable	The SD memory card was in usable status.
20301	SD Memory Card Forced Stop	The SD memory card was in removable (disabled) status due to SD memory card forced stop function.
27FFE	Software Reboot Execution	The software reboot was executed.
27FFF	Event History Clear	The event history was cleared.

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

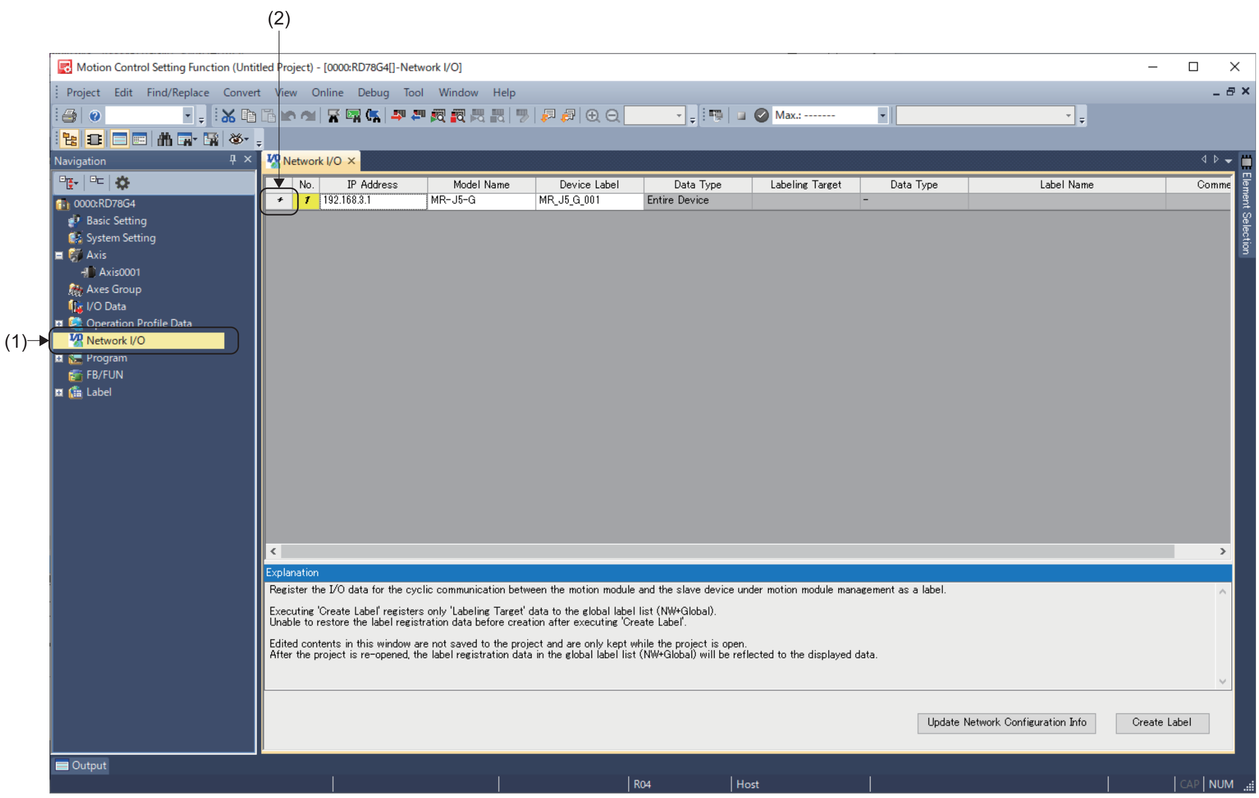
How to use the slave label

Labels assigned to the cyclic data of the device station in the motion control station are called "Slave labels". The slave labels are generated by setting unique device name (device label) for each device station in the engineering tool.

MR-J5-G usage example is used to describe the method of using the slave label. and Other devices can also be used it with the same method.

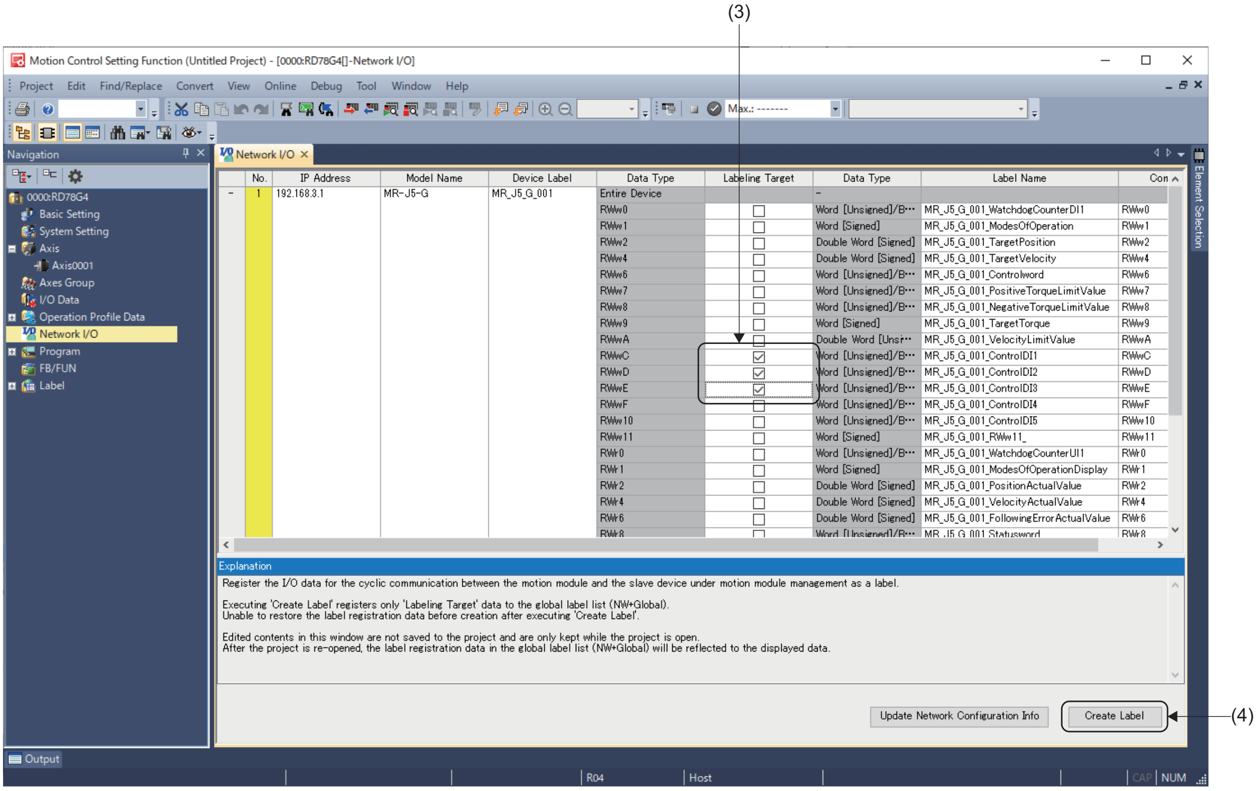
1. In Motion Control Setting Function, select Navigation window => [Network I/O](1) to open the "Network I/O" screen.

2. From the devices displayed on "Network I/O" screen, click the "+" on the left end of the device to generate labels. (Example: Click "+"(2) of No.1 device (MR-J5-G))

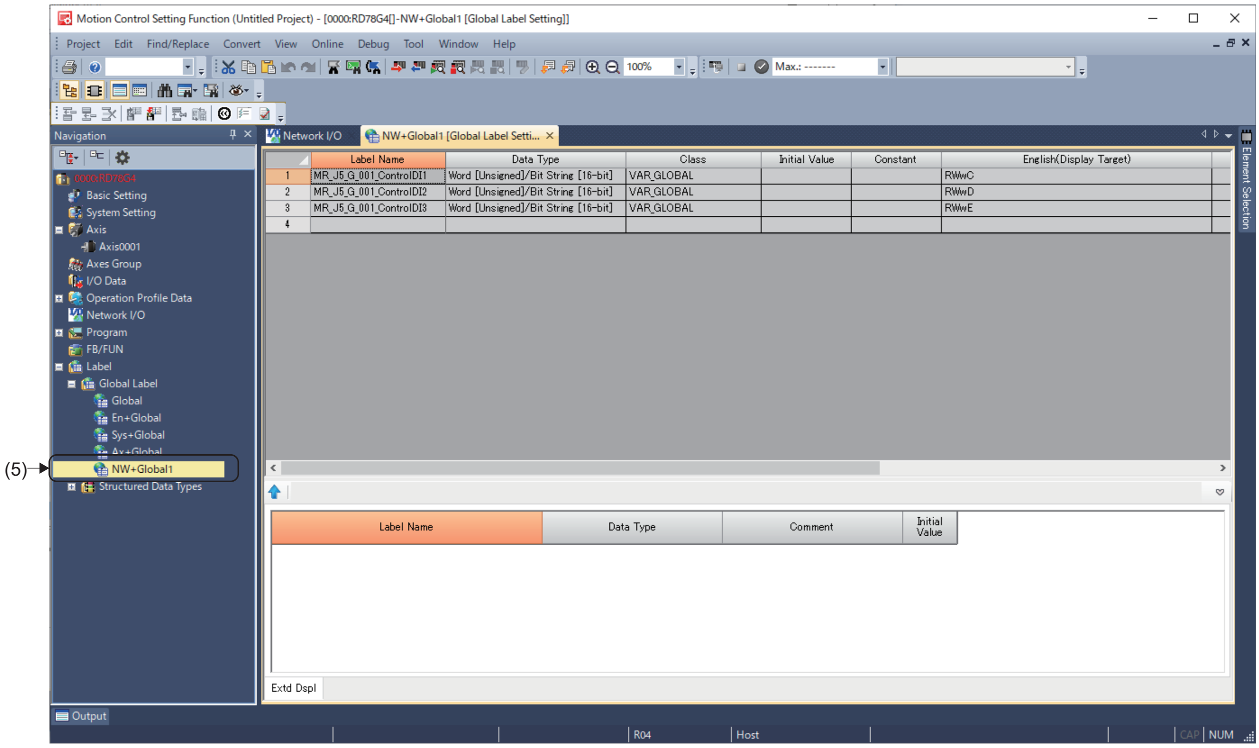


3. Open the object list of the selected device. Insert checks in labeling targets for the data types to label. (Example: Insert checks (3) in data type "RWwC to RWwE")

4. When checks have been inserted for all the data types to label, click the [Create Label] button (4).



5. [NW+Global1] (5) is registered in Navigation window ⇨ [Label] ⇨ [Global Label]. Select the registered [NW+Global1] to open the "NW+Global1" screen. A list of the created labels is displayed.



Point

- Slave labels are performed to refresh the values automatically (reflect to send data/import receive data) in operation cycle process. For details of operation cycle, refer to the following.
☞ Page 105 Operation Cycle
- When device stations are disconnected, the slave labels stored the corresponding receive data hold the value before disconnected.
- For RPDO (RWw) of stations in which axis assignment is executed, do not generate slave label for objects set in slave object settings.

Precautions

- The slave labels cannot be used properly when there are mismatches in the settings of network configuration in the settings of network configuration and network I/O in the engineering tool. Perform the procedure above again when the network configuration setting is changed.
- When the slave labels are assigned to 1 byte cyclic data (such as Modes of operation display object), the lower byte of Word [signed] type label is used, and The upper byte will always be 0. Therefore, even when the cyclic data is a negative value, it is displayed as a positive value.

Ex.

When the cyclic data is "-1(FFH)"
The slave label is "255(00FFH)".

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 "Speed/acceleration/deceleration unit selection (PT01.1)" is set to "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 "Speed/acceleration/deceleration unit selection (PT01.1)" is set to "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 the 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 [Pr. PA06 Electronic gear - Numerator] and [Pr. PA07 Electronic gear - Denominator] 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 slave object "Velocity actual value (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	000000000h	000000000h
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 "Sensor input method selection (PD41.3)" is set to "1: input from 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>
Sets Driver Command Discard Detection Setting (AxisName.Pr.StopOption_DriverTargetIgnored) to 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.

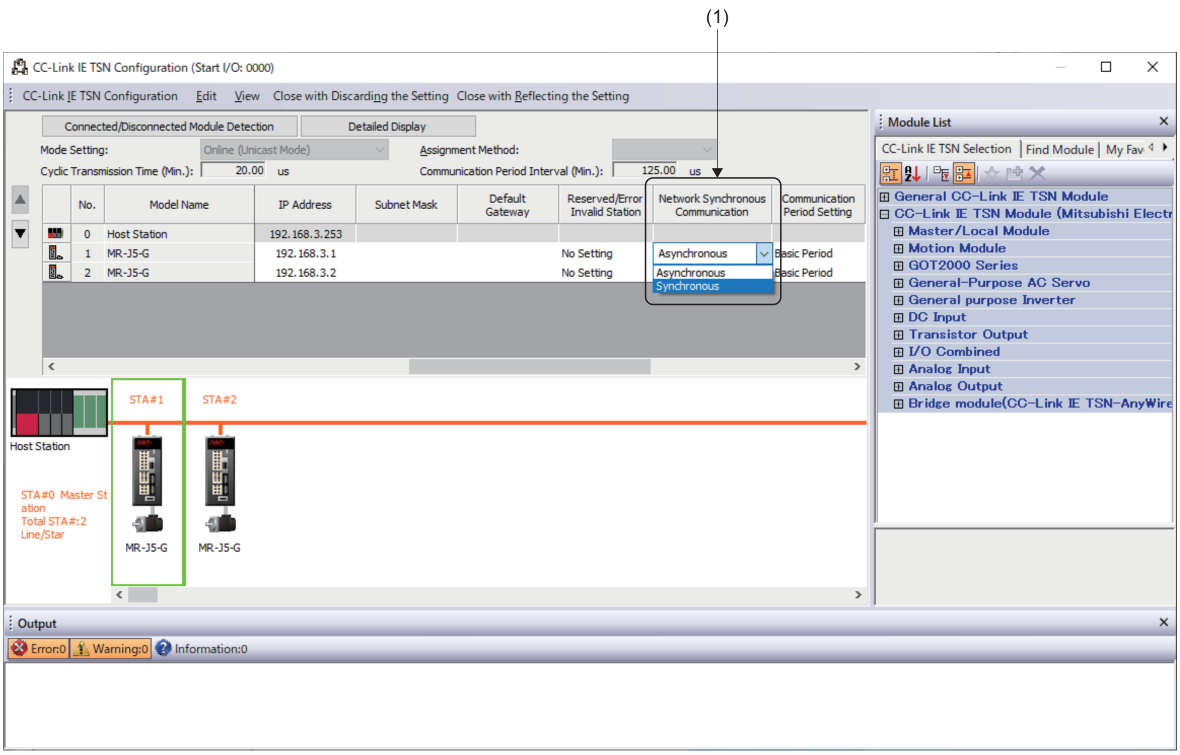
*2 When "1h" is set, wire the limit switch to the controller side since the input to MR-J5(W)-G becomes invalid. In the case of wiring to MR-J5(W)-G, the motor does not stop if the limit is detected.
Also, the labeling of the slave must not be carried out since [Control DI 5(2D05h)] of the RPDO mapping is used for sending signal to MR-J5(W)-G side.

■ Network parameter setting to use MR-J5(W)-G

1. In the "CC-Link IE TSN Configuration" screen, set the "Network Synchronous Communication" (1) of the target servo amplifier to be used in cyclic synchronous mode to "Synchronous"[*1].

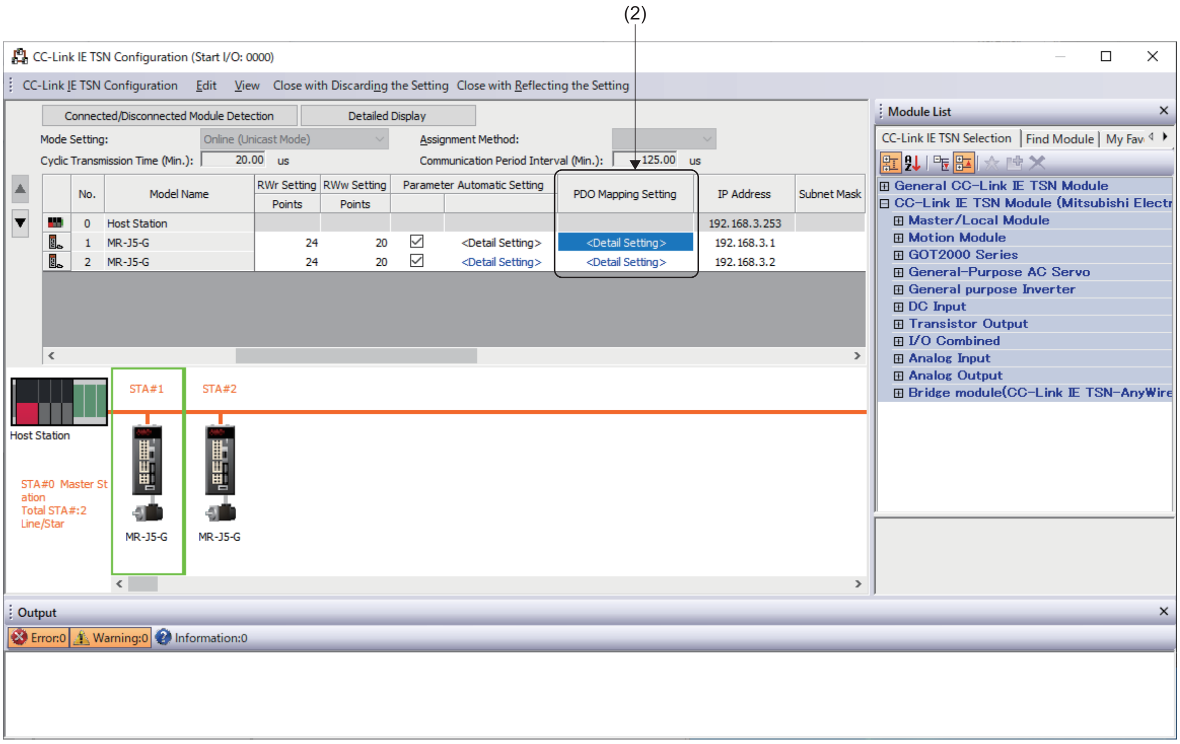
*1 The setting, "Synchronous" is available from version "1.5" or later of add-on baseSystem.
Set "Asynchronous" when using version "1.4" or earlier.

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ [Module model] ⇨ [Module Parameter (Network)] ⇨ "Parameter editor" screen ⇨ [Basic Settings] ⇨ [Network Configuration Settings] ⇨ [Detailed Setting]



2. Set PDO mapping.

In the "CC-Link IE TSN Configuration" screen, click "<Detail Setting>" (2) of the [PDO Mapping Setting] of the target servo amplifier to open the "PDO Mapping Pattern Selection (1/2)" screen.

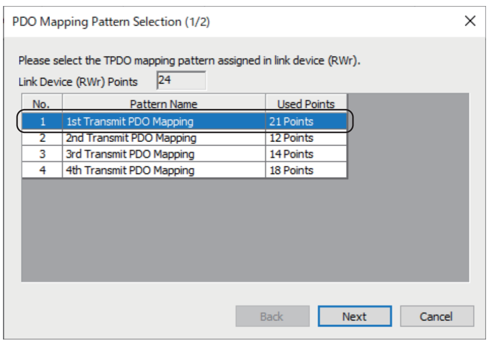


3. In the "PDO Mapping Pattern Selection (1/2)" screen, select the mapping pattern for "TPDO", and clicking the [Next] button will switch the screen to the "PDO Mapping Pattern Selection (2/2)" screen. Select the mapping pattern for "RPDO" and click the [OK] button to display the "PDO Mapping Setting" screen.

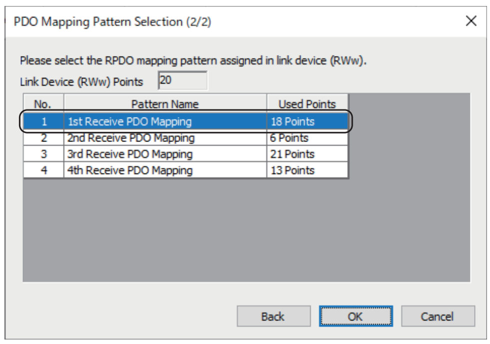
When using MR-J5(W)-G in cyclic synchronous mode, set the following mapping patterns.

Item	Mapping pattern
TPDO	1st Transmit PDO Mapping
RPDO	1st Receive PDO Mapping

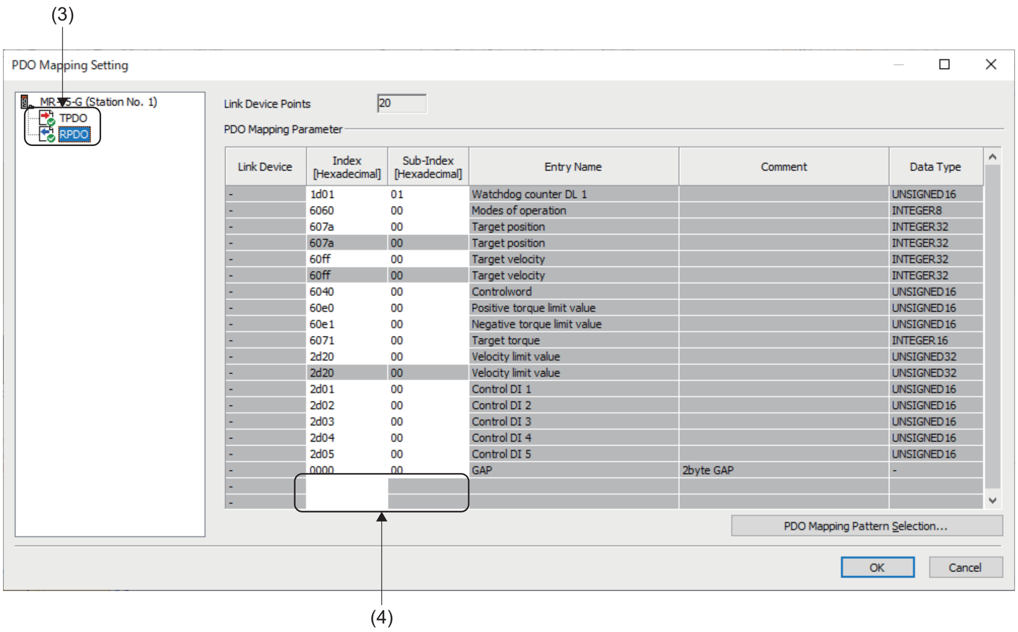
■TPDO Mapping Pattern



■RPDO Mapping Pattern



4. In the "PDO Mapping Setting" screen, add and change the PDO mapping according to the functions used.
Switch the setting screen by selecting TPDO and RPDO on the window (3).
To add objects to be exchanged using PDO, set "Index/Sub-Index" (4).
For the necessary of objects of each function, refer to each function.



5. When finished, click the [OK] button to close the "PDO Mapping Setting" screen.

6. On the "Module Parameter" screen, click the [Apply] button to confirm the changed objects.

A

Using methods

■ How to operate external signal of MR-J5(W)-G via communication

1. Preparation

Generates the required objects of the slave label according to the procedures described in common items for all devices (☞ Page 831 Common Items for all devices).

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 slave label is shown below. (Generate the label previously as Control DI1 and Status DO1 are used for the gain switching.)

```
[Global label]
bGainChange //Bit type label
```

```
[ST program]
//the gain switching request to MR-J5-G
MR_J5_G_001_ControlDI1.4 := bGainChange;

//check the gain of MR-J5-G is switched or not
IF(MR_J5_G_001_StatusD01.4 = TRUE) THEN
                    //describes the process after the gain switching
END_IF;
```

The bit4 of ControlDI1 is the gain switching request.
This program switches the gain of MR-J5-G as operating bit4 of ControlDI1 with bGain Change.

The bit4 of StatusD01 is the gain switching status.
The process which is performed only during the gain switching can be created by checking the bit4 status.

■ The connection with MR-J5(W)-G in fast operation mode

For details on the fast operation mode, refer to the following.

☞ Page 105 Operation Cycle

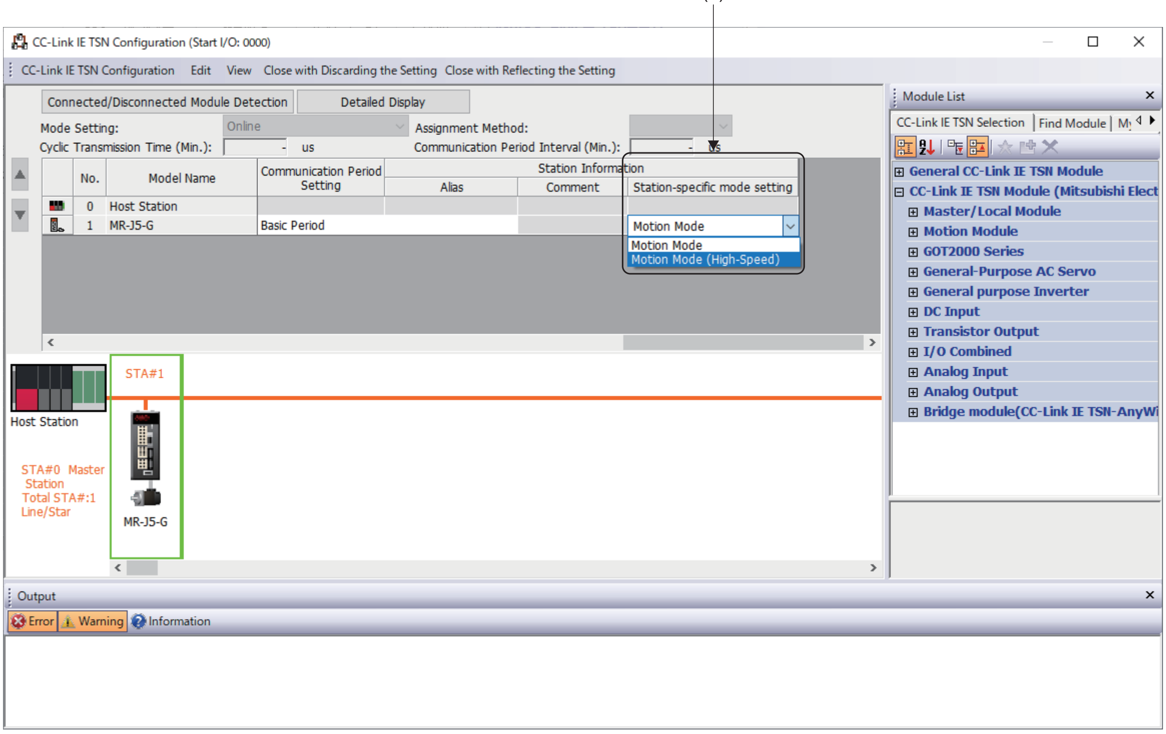
1. Preparation

• Setting of network (GX Works3)

Set [Motion Mode (High-Speed)] (1) to [Station-specific mode setting] on the "CC-Link IE TSN Configuration" screen.

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ [Module model] ⇨ [Module Parameter (Network)] ⇨ "Parameter editor" screen ⇨ [Basic Settings] ⇨ [Network Configuration Settings] ⇨ [Detailed Setting]

(1)



When Motion Mode (High-Speed) is set, select [2nd Transmit PDO Mapping] and [2nd Receive PDO Mapping] in the PDO Mapping Pattern. The following objects are mapped in each PDO.

■TPDO Mapping Pattern [2nd Transmit PDO Mapping]

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

■RPDO Mapping Pattern [2nd Receive PDO Mapping]

No.	Object name
1	Watch dog counter DL
2	Modes of operation
3	Target position
4	Controlword
5	Control DI4

- Motion system setting (Engineering tool)
  Set to the fast operation mode. For details, refer to the following.
  ☞ Page 105 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 slave label.

1. Map [Digital inputs(60fdh)] to TPDO.

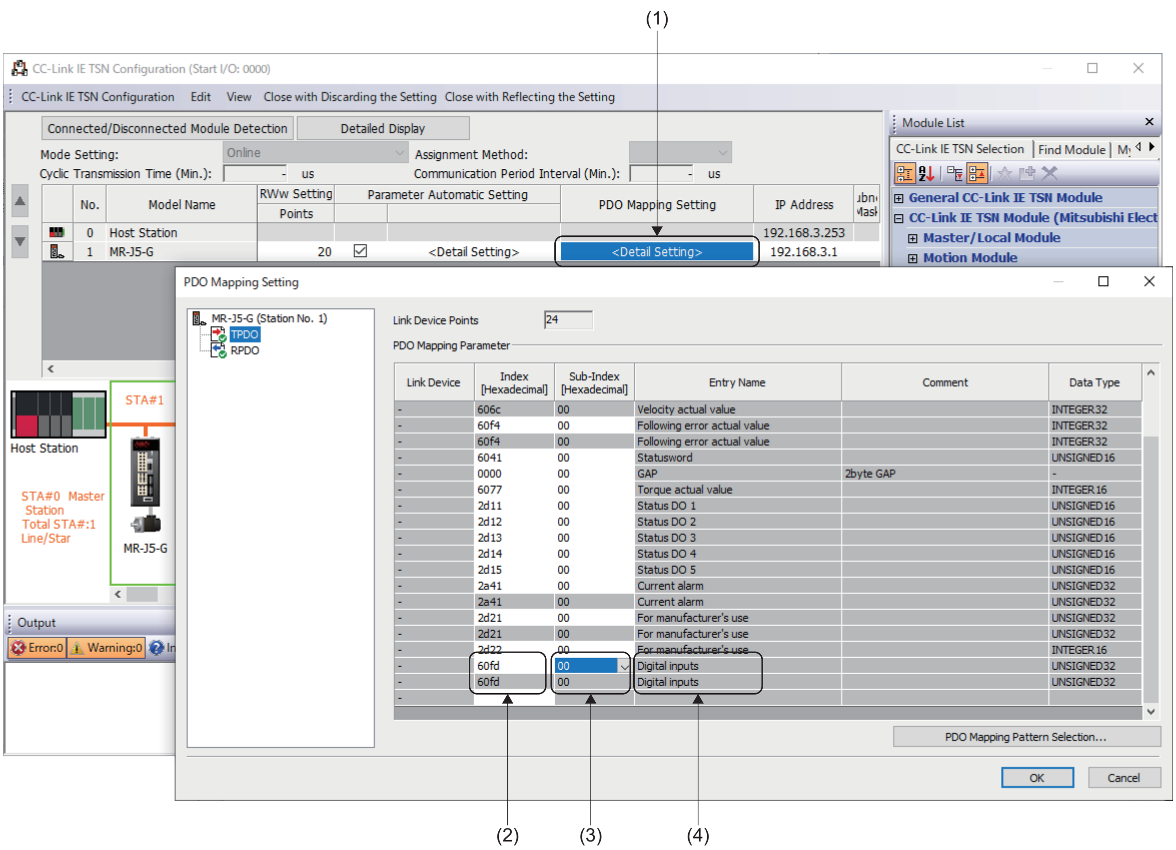
In the "CC-Link IE TSN Configuration" screen, click "<Detail Setting>" (1) of [PDO Mapping Setting] for the target servo amplifier to open the "PDO Mapping Setting" screen.

Selecting the following for [Index] (2) and [Sub-index] (3) in the "PDO Mapping Setting" screen adds "Digital inputs" to [Entry Name] (4).

- Index: 60fdh
- Sub-index: 00h

When completed, click the [OK] button to close the "PDO Mapping Setting" screen.

In the "Module Parameter" screen, click the [Apply] button to fix the changed objects.



2. Make [Digital inputs (60fdh)] the slave label.

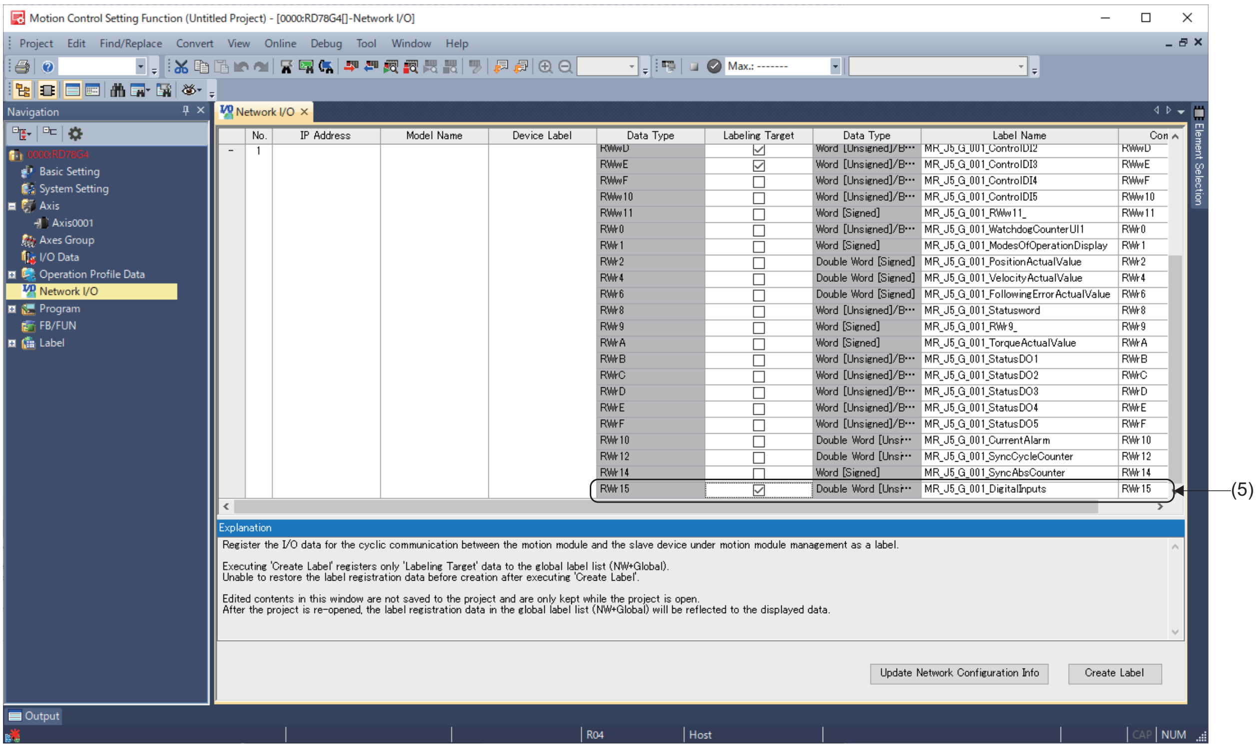
Make slave labels in the "Network I/O" screen of Motion Control Setting Function.

For the method of the slave labeling, refer to the following.

☞ Page 831 Common Items for all devices

Ex.

When the labeling is performed with the label name of "MR_J5_G_001_DigitalInputs" (5).

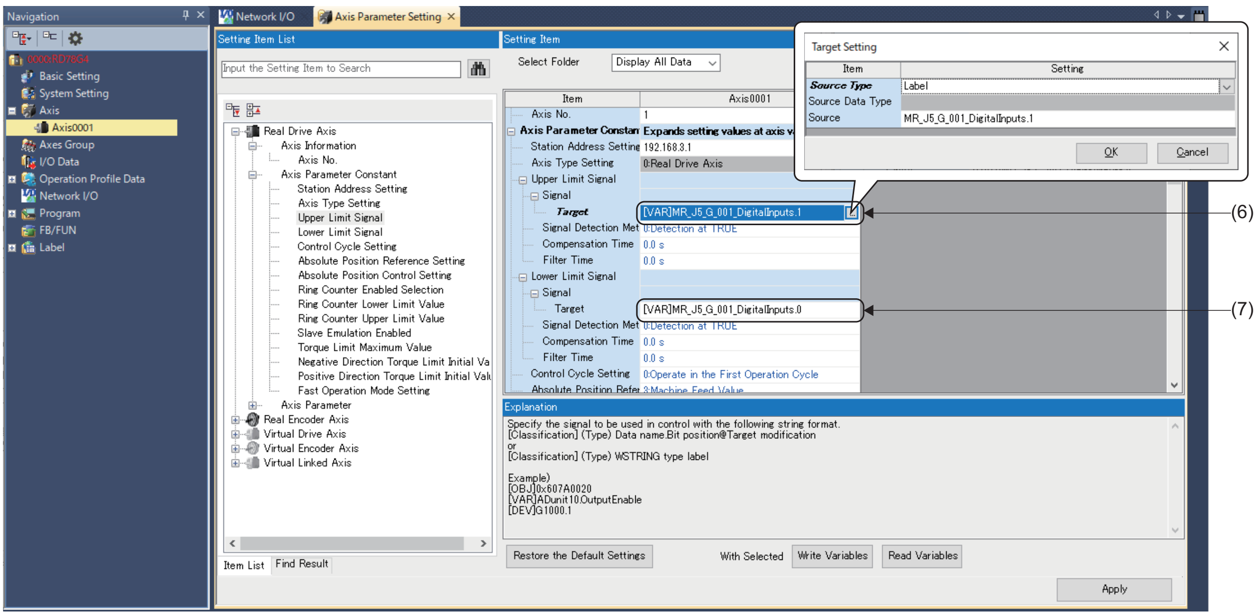


3. Specify the slave label to the home position switch of FLS/RLS or MC_Home (Homing).

- The upper limit/lower limit setting

  Set the target of the upper limit signal (6)/lower limit signal (7) of the "axis parameter setting" screen of Motion Control Setting Function.

Variable	Setting value (when "Travel direction selection (PA14)" is 0)
Target (AxisName.PrConst.HwStrokeLimit_FlsSignal.Source.Target)	[VAR]MR_J5_G_001_DigitalInputs.1
Target (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 (Homing).

Variable	Setting value (when "Travel direction selection (PA14)" is 0)
Target (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)".

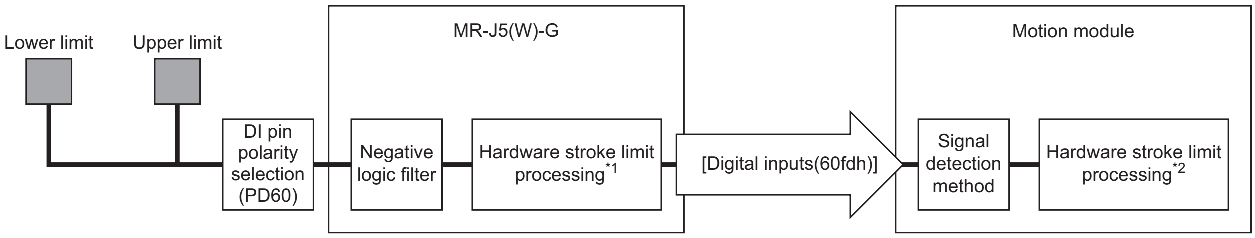
Precautions

- 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.

  For MR-J5(W)-G: MR-J5-G/MR-J5W-G User's Manual (Parameters)
- 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
Signal Detection Method (AxisName.PrConst.HwStrokeLimit_FlsSignal.Detection)	1: Detection at FALSE (LowLevel)
Signal Detection Method (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	
	"DI pin polarity selection (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	→	Negative logic (1: Detection at FALSE (LowLevel))	Not detect
	1		Detect		FALSE			Detect
OFF	0		Detect		FALSE			Detect
	1		Not detect		TRUE			Not detect
ON	0		Not detect	→	FALSE	→	Positive logic (0: Detection at TRUE (HighLevel))	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 Hardware Stroke Limit Override (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 slave 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, "Communication Cycle Unsupported Driver Connection (error code: 1C47H)" will occur at connection and it can not be connected with the driver.
- Do not assign a MR-J5(W)-G set to CC-Link IE TSN Class A to an axis. When a MR-J5(W)-G set to CC-Link IE TSN Class A is assigned to an axis, a servo alarm [AL. 19E.2_Control mode setting warning 2] occurs in the MR-J5(W)-G when connected.

MR-J5(W)-G (other than Cyclic synchronous mode) connection method

This section describes how to set when connecting MR-J5(W)-G (other than the cyclic synchronous 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 the mode other than the cyclic synchronous mode is shown below.

Model name	Mode	Version
MR-J5(W)-G	Profile mode (pp, pv, tq)	A4 or later
	Point table mode (pt)	B8 or later

Setting method

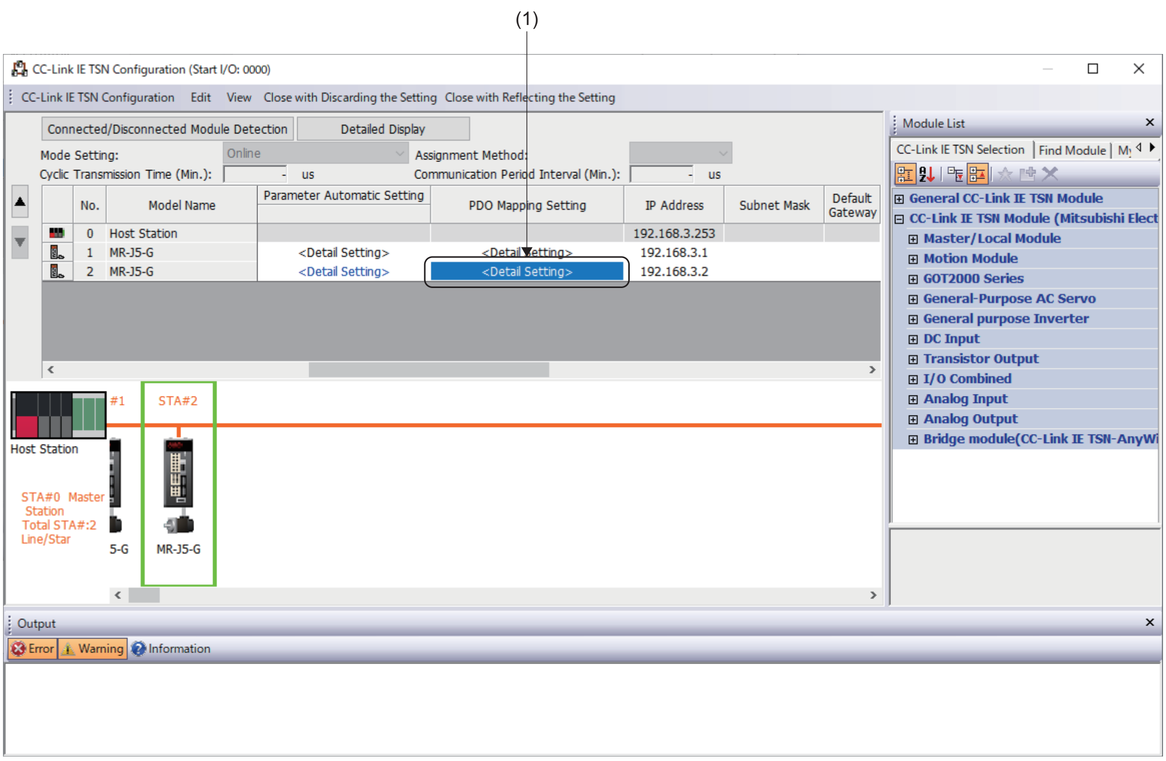
An example for using two MR-J5-G as the first station (192.168.3.1) set with the cyclic synchronous mode and the second station (192.168.3.2) set in other than the cyclic synchronous mode. For details of the objects used in each mode is shown below.

■ Setting of network (GX Works3)

Set PDO mapping and map required objects in the mode to use on the "CC-Link IE TSN Configuration" screen.

1. In order to set the second station as a mode other than the cyclic synchronous mode, conduct the PDO mapping settings for the second station.
In the "CC-Link IE TSN Configuration" screen, click "<Detail Setting>" (1) of the [PDO Mapping Setting] of the target servo amplifier to open the "PDO Mapping Pattern Selection (1/2)" screen.

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ [Module model] ⇨ [Module Parameter (Network)] ⇨ "Parameter editor" screen ⇨ [Basic Settings] ⇨ [Network Configuration Settings] ⇨ [Detailed Setting]



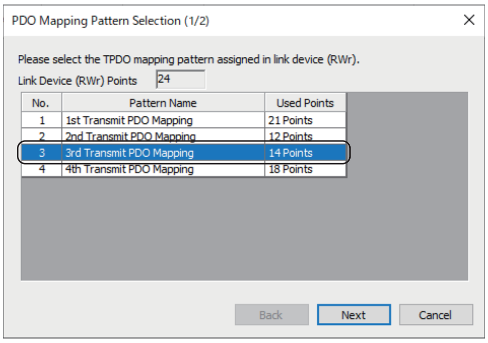
2. In the "PDO Mapping Pattern Selection (1/2)" screen, select the mapping pattern for "TPDO", and clicking the [Next] button will switch the screen to the "PDO Mapping Pattern Selection (2/2)" screen. Select the mapping pattern for "RPDO" and click the [OK] button to display the "PDO Mapping Setting" screen.

Set the following mapping patterns depending on the operation mode used.

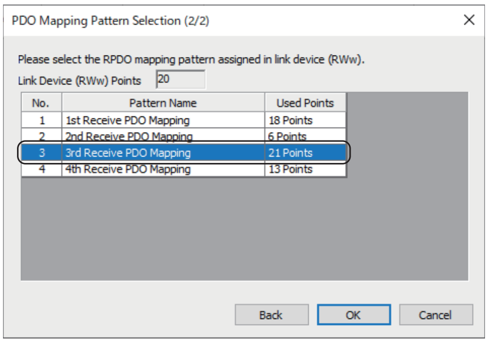
Mode	Mapping pattern	
	TPDO	RPDO
Profile mode (pp, pv, tq)	3rd Transmit PDO Mapping	3rd Receive PDO Mapping*1
Point table mode (pt)	4th Transmit PDO Mapping	4th Receive PDO Mapping

*1 When the "RWw Setting" points set on the "CC-Link IE TSN Configuration" screen is set to the initial value (20 points), the size of the objects to be mapped exceeds the number of points.
Set the "RWw Setting" points to 24 points or above in advance or delete any unnecessary objects from the "PDO Mapping Setting" screen.

■TPDO Mapping Pattern

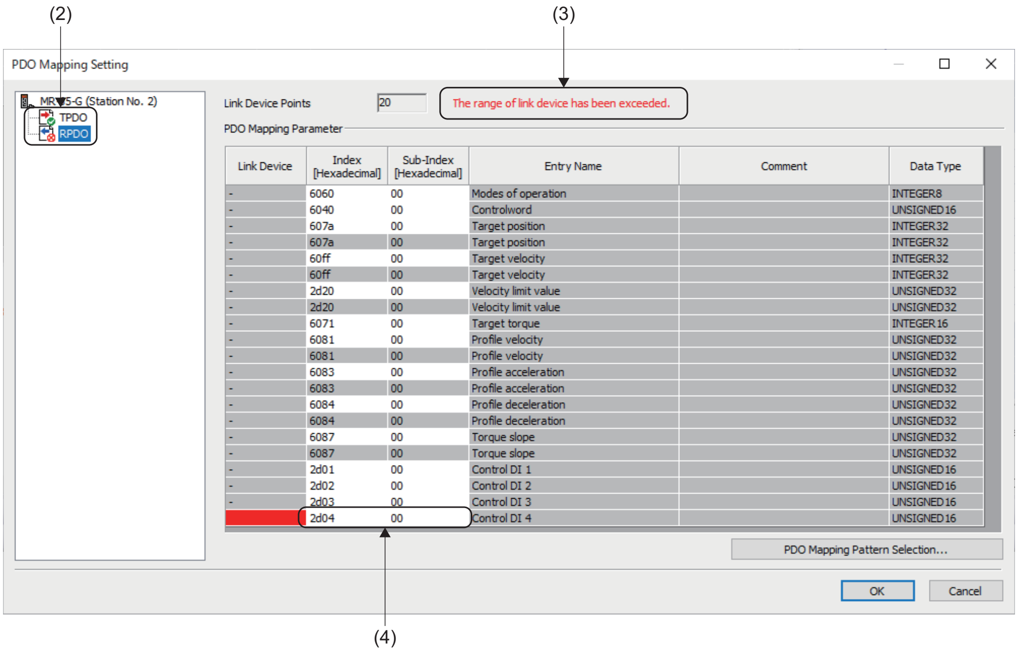


■RPDO Mapping Pattern



A

3. In the "PDO Mapping Setting" screen, add and change the objects required by the mode to be used.
Switch the setting screen by selecting TPDO and RPDO in the window (2).
When the size of the objects mapped to TPDO/RPDO is greater than the RWr/RWw points, "The range of link device has been exceeded" (3) is displayed. Delete the objects being mapped, or click [Cancel], and then try again after increasing the number of "RWw/RWr Setting" points on the "CC-Link IE TSN Configuration" screen.
To change a mapping object, select the "Index/Sub-Index" (4) from the drop-down list. To delete it, select "Index" (4) and press the [Delete] key.



4. After completing the settings, click the [OK] button to close the "PDO Mapping Setting" screen.

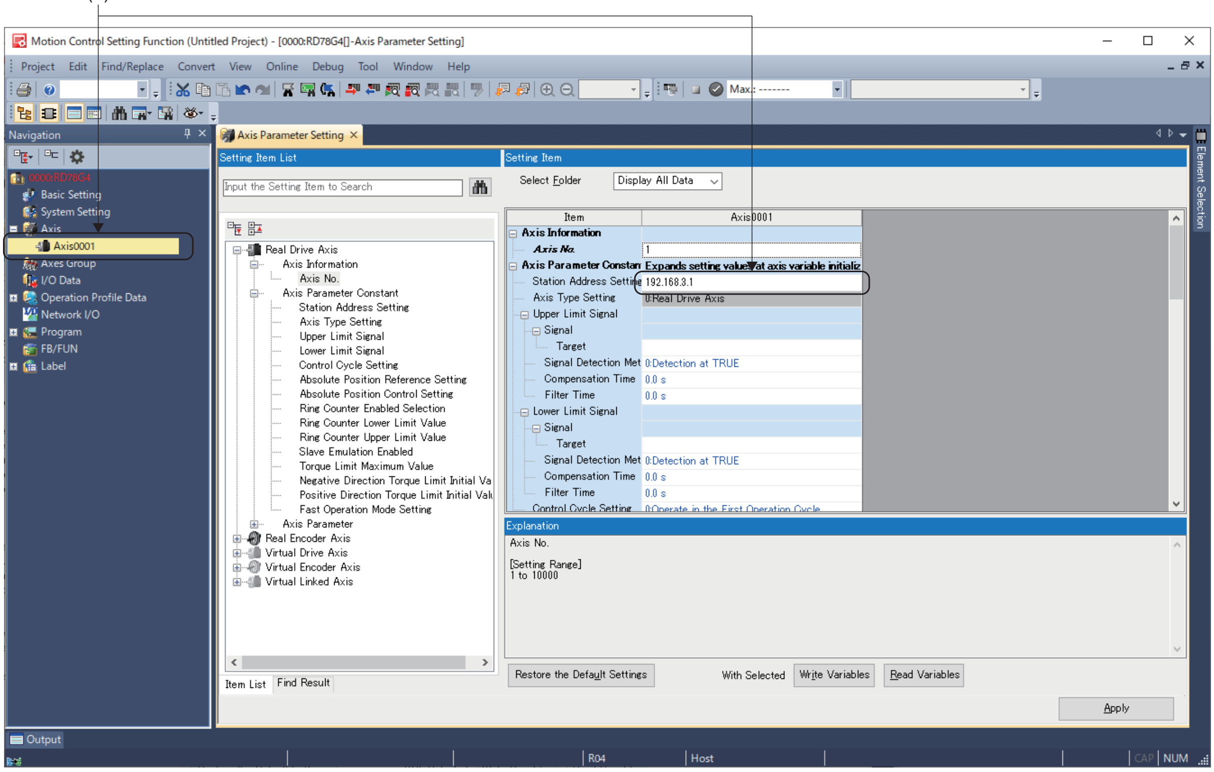
5. In the "Module Parameter" screen, click the [Apply] button to fix the changed objects.

■ The motion system setting [Motion control setting function]

The axis to be used in other than the cyclic synchronous mode must not be set.

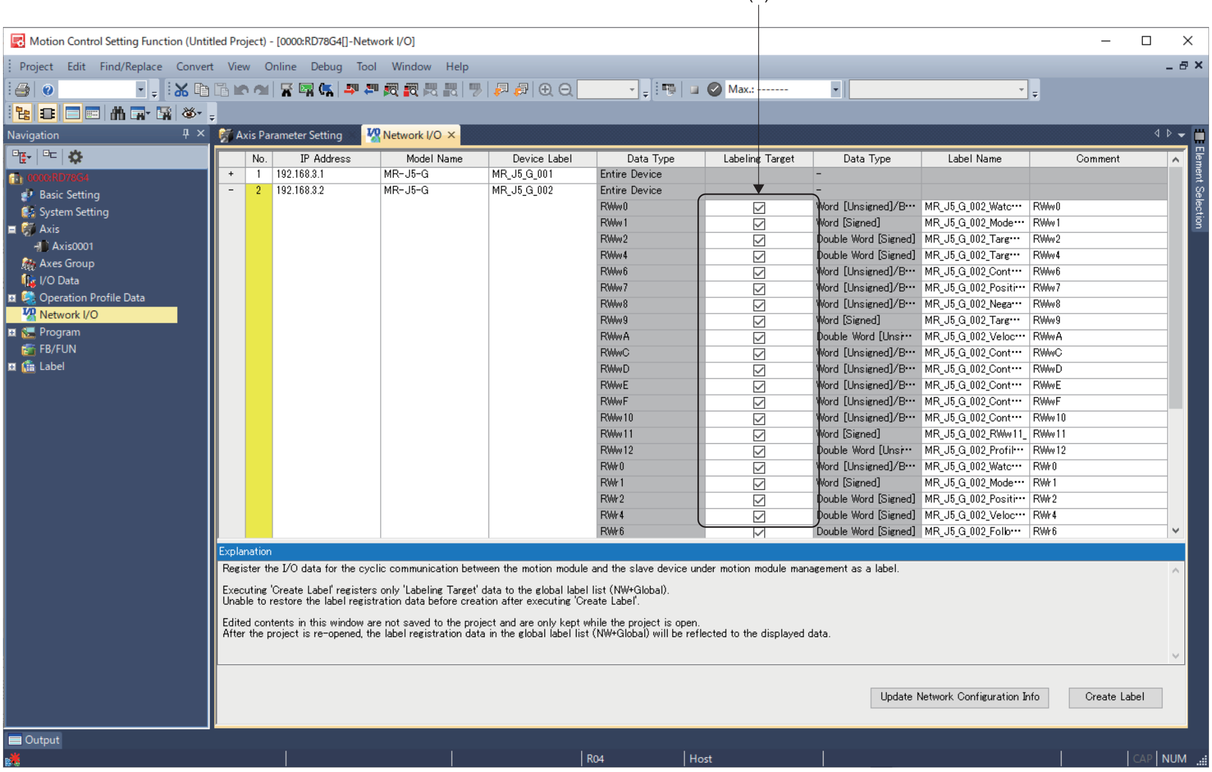
1. Set the axis used with cyclic synchronous mode (first station (192.168.3.1)) (1). The axis used with a mode other than the cyclic synchronous mode (second station (192.168.3.2)) is not set as it is used with a mode other than the cyclic synchronous mode.

(1)



2. Generate the slave label of MR-J5-G (second station (192.168.3.2)) used with a mode other than the cyclic synchronous mode by checking all the labeling targets (2) and clicking the [Create Label] button according to the procedure described in "Common Items for all devices". For using with a mode other than the cyclic synchronous mode, generate labels of the entire device as the labeling target.

(2)



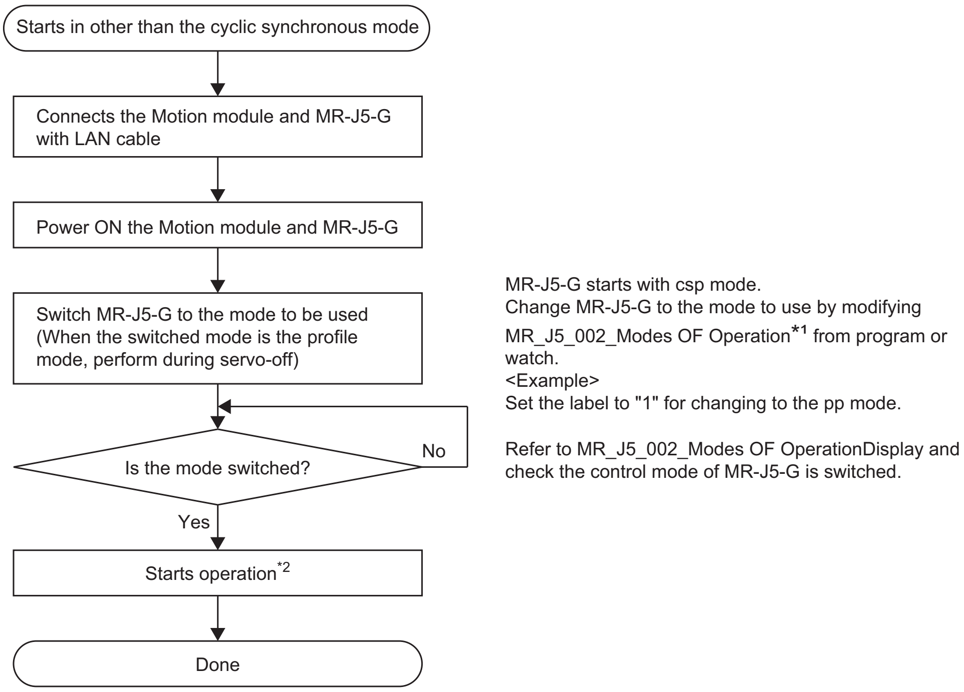
A

■ MR-J5-G setting (MR Configurator2)

Refer to the manual of MR-J5(W)-G because the required settings are different depending on the mode to use.

Setting method

The following figure describes the process for driving motors with other than the cyclic synchronous mode.



*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 Depending on the version of baseSystem, the following is required.

Version of baseSystem	Description
"1.14" or earlier	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 Set bit15 to TRUE. ■Method 2 Add 1 for each communication cycle. (As bit15 must always be TRUE, add 1 to keep bit15 from changing to FALSE.)
"1.16" or later	When Watch dog counter DL is mapped to PDO, Creating of an update program by users is not required because the WDC process is performed in the module

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 in other than the cyclic synchronous mode, the motion system does not perform the limit check of the command, or issuing the forced stop 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" or later. If its version is not compatible, "Communication Cycle Unsupported Driver Connection (error code: 1C47H)" will occur at connection and it can not be connected with the driver.

A

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

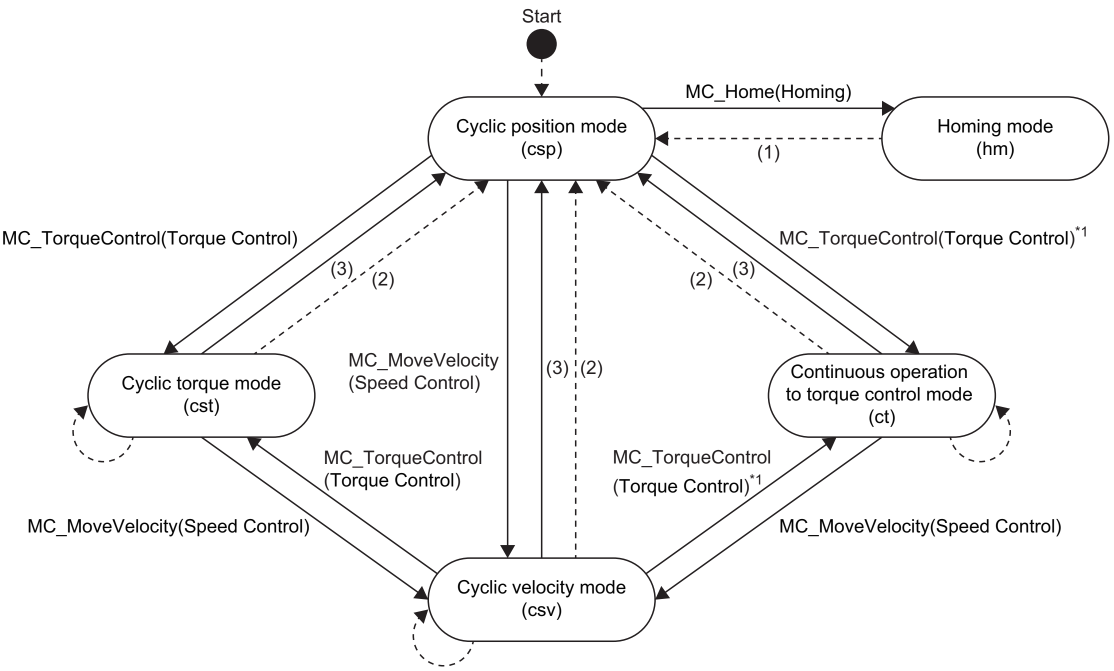
Driver control mode

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 195 Driver homing method
Cyclic position mode (csp)	8: csp	Executes control following the sequential positioning command from the controller in each communication cycle.	☞ Page 274 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 852 Continuous operation to torque control mode

Control mode switching of the driver is executed at the same time when the Motion control FB is executed.

The state transition is shown below.



*1 For continuous operation to torque control mode (ct) switch, refer to the following.
☞ Page 852 Continuous operation to torque control mode

No.	State transition description
(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 (Speed Control)/MC_TorqueControl (Torque Control) is being Aborting/Buffered.

Precautions

- When the control mode does not switch within 1 [s] after requesting the control mode switching to the driver, "Control Mode Switching Error (error code: 1A1DH)" occurs and the axis stops.
- When specified as a master axis for master-slave operation, switching to the continuous operation to torque control mode cannot be executed.

Continuous operation to torque control mode

For the drive unit which supports the continuous operation to torque control mode, when MC_TorqueControl (Torque Control) is started with selecting "1: ct (continuous operation to torque control mode)" by Control mode switching selection (Options (Options): Bit 19), 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.

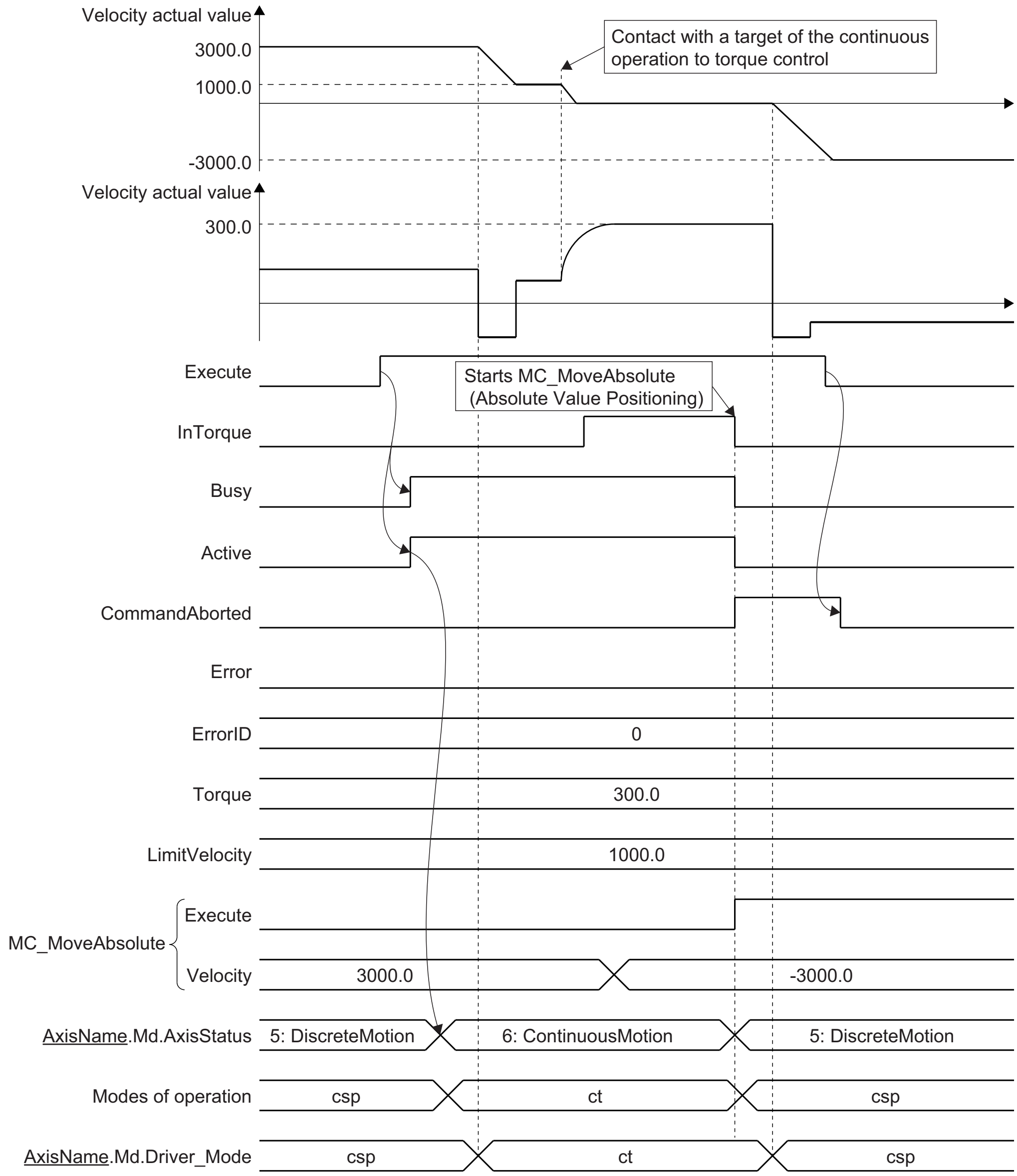
In the continuous operation to torque control mode, the driver executes the torque control with commanded torque set to Target torque (Torque) as accelerating or decelerating to the velocity set to Limit velocity (LimitVelocity).

To finish this FB, start MC_Stop (Forced Stop).

■ Timing chart

- When the FB is normally completed

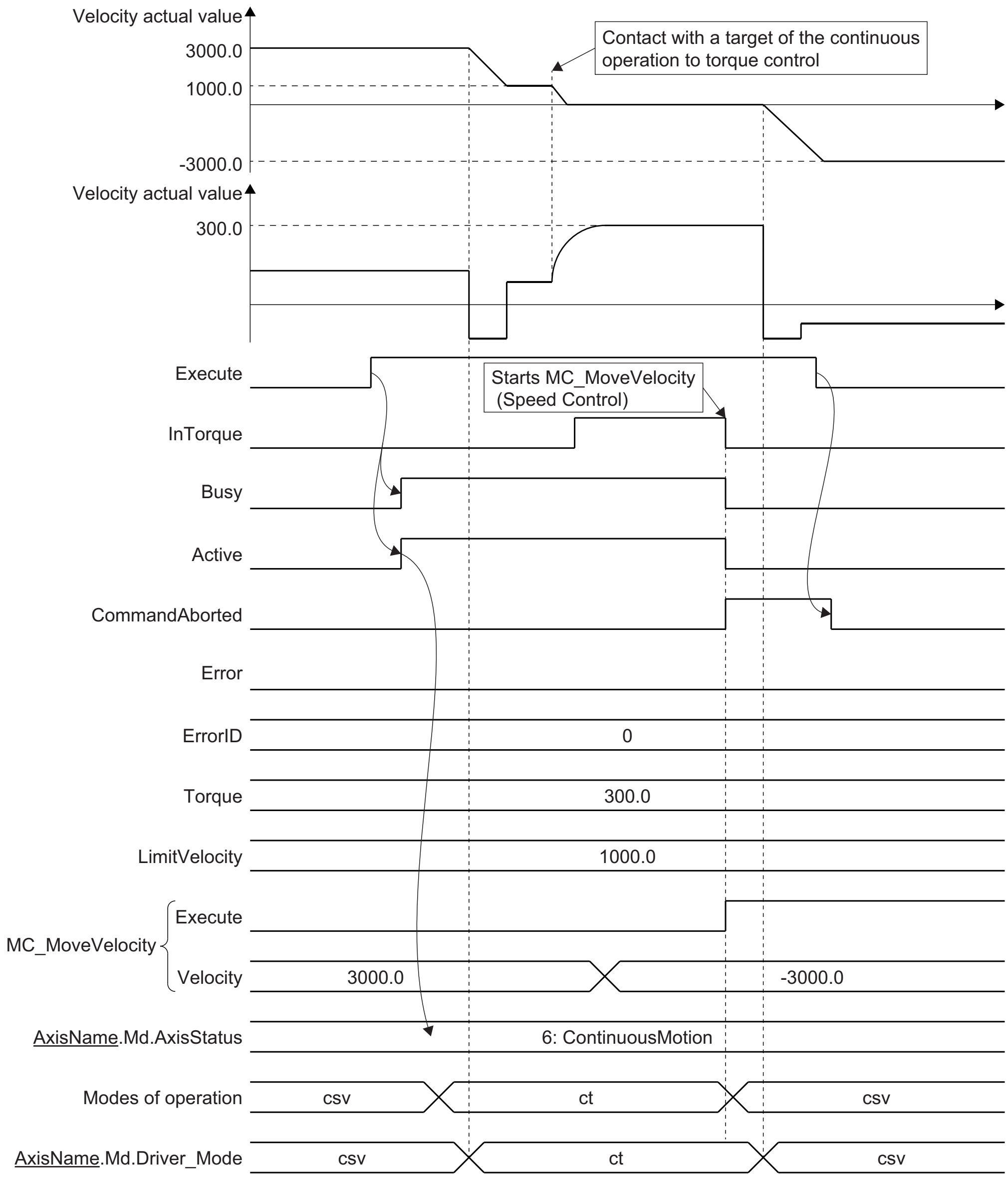
■Operation for "cyclic position mode ⇔ continuous operation to torque control mode switching"



Point

Set "1: Actual velocity" to Velocity initial value selection at continuous operation to torque control mode switching (Options (Options): Bit 20, 21) 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 (Speed Control) 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 "0: Acceleration/deceleration specification method (mcAccDec)" for Acceleration/deceleration method setting (Options (Options): Bit 0 to 2).)
- Change the torque limit value to the lower value.

- When an error occurs

For details when an error occurs, refer to "Basic operation of Execute command (Execute) type Motion control FBs" in the following manual.
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)

■ Relevant variables/FB

For details, refer to the following.

☞ Page 284 Relevant variables

■ Control details

For details, refer to the following.

☞ Page 284 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	"B0" or later
MR-JET-G	

- "The cyclic torque mode ⇔ the continuous operation to torque control mode switching" cannot be executed when multiple start MC_TorqueControl (Torque Control) is executed during MC_TorqueControl (Torque Control) start-up. If it is executed, "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 slave 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 device.

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.
- If vibration occurs during the continuous operation to torque control, decrease "Torque feedback loop gain (PB03)" and check if the issue has been solved.

Safety communication

A safety CPU and the firmware for connectable devices which can combine with RD78G(H) and perform the safety communication are shown below.

Model name	Version
RnSFCPU	☞ Page 874 Restrictions by the version
MR-J5-G-RJ	"B2" or later For details, refer to the following manual. MR-J5-G/MR-J5W-G User's Manual (Introduction)
MR-J5D-G	"C0" or later For details, refer to the following manual. MR-J5D-G User's Manual (Introduction)

■ 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 "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.

Touch probe

The touch probe function records (latches) arbitrary data at trigger input signal detection. By setting the trigger input signal to a signal that supports external signal high-accuracy input, high-accuracy recording (latching) with a device station signal detection time is enabled.

For details on the touch probe, refer to the following.

☞ Page 389 Touch Probe

■ Setting method

The following example shows how to set external signal high-accuracy for a touch probe when using TPR1 of MR-J5(W)-G series as the input signal.

- Required settings for external signal high-accuracy input

  External signal high-accuracy input uses a device station signal detection time, so that the setting to obtain the signal detection time is required. If the required setting is not executed to external signal high-accuracy input, the signals operate with the same accuracy as normal signals. For details on the setting method, refer to the following.

  ☞ Page 859 Required settings for external signal high-accuracy input

- How to check whether high-accuracy input is enabled

  Check whether a signal detection time is stored to the object or link device those stores a device station signal detection time at the signal detection using the device label, etc. For details on the check method, refer to the following.

  ☞ Page 863 How to check whether high-accuracy input is enabled

- Settings of the delay compensation time peculiar to the system

  A delay time peculiar to the system occurs in the Probe data (ProbeData) set for the touch probe. The delay can be compensated by setting the Compensation Time (CompensationTime) in the touch probe. Also, the delay time varies depending on the data set in Probe data (ProbeData). Set the delay compensation time by referring to the table below.

<A rough standard for the delay compensation time peculiar to the system>

Type	Data	Motion Synchronization Station Send/Receive Data Refresh Method (System.PrConst.Link_MotionStationRefreshType)	
		0: Response preferred method (EmphasisResponse)	**1: Operation cycle preferred method (EmphasisOperationCycle)**
OBJ	0x60640020 (Position actual value)	Operation cycle × 2	Operation cycle × 3
VAR	Axis.Md.SetPosition	Operation cycle × 1	
	Axis.Md.ActualPosition	Operation cycle × 3	Operation cycle × 4
	Axis.Md.Io_TargetPos	Operation cycle × 1	
	Axis.Md.Io_PosActualValue	Operation cycle × 3	Operation cycle × 4
	MR_J5_XXX_PositionActualValue[*1]	Operation cycle × 2	Operation cycle × 3
AXIS	mcLatestSetValue	0	
	mcSetValue	Operation cycle × 1	
	mcLatestActualValue	Operation cycle × 2	Operation cycle × 3
	mcActualValue	Operation cycle × 3	Operation cycle × 4

*1 The device label name varies depending on the model. Set the data according to the created device label name.

■ Example programs

- When TPR1 of MR-J5-G-RJ (station address: 192.168.3.1) is used as Trigger input signal (TriggerInput) of MC_TouchProbe (Touch Probe Enabled) at the rising edge detection

<Setting of MR-J5-G-RJ>

1. Set a connector pin to assign TPR1 of the input device to be used.

Set "2Ch" to Pr.PD38.0-1. (Assign TPR1 to the connector pin CN3-10.)

<Setting of the Motion system>

1. Set the following slave objects to TPDO of PDO mapping. For the method of mapping, refer to "CC-Link IE TSN Configuration Window" of the following manual.

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

Index	SubIndex	Object name
60B9h	00h	Touch probe status 1
60D1h	00h	Touch probe time stamp 1 positive value

2. Generate a label to be used as Trigger input signal (TriggerInput) of MC_TouchProbe (Touch Probe Enabled). Create a new data of MC_TRIGGER_REF type and set Toggle status (Touch probe status 1 (Bit6)) for latch completion of TPR1 to Target (Target).

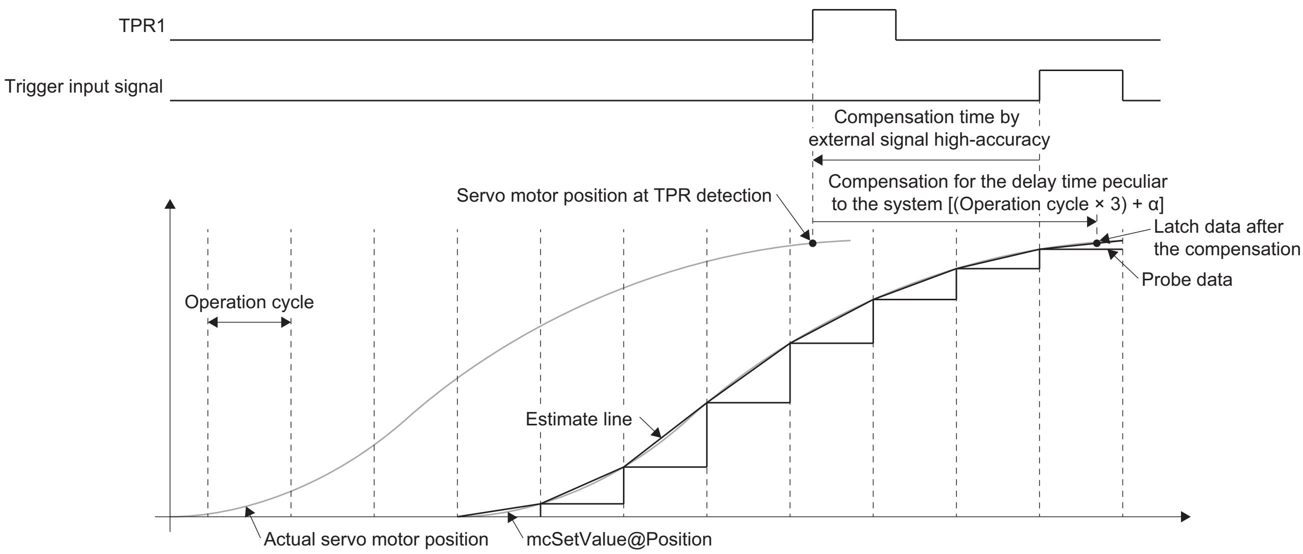
Data name	Setting of MC_TRIGGER_REF type data			
	Target (Target)	Signal Detection Method (Detection)	Compensation Time (CompensationTime)	Filter Time (FilterTime)
SignalData0001	[OBJ]0x60B90010.6@192.168.3.1	4: Detection at Rising Edge/ Falling Edge (BothEdges)	0.0	0.0

3. Generate a label to be used as Probe data (ProbeData) of MC_TouchProbe (Touch Probe Enabled). Create a new data of TARGET_REF type and set Probe data (ProbeData). As an example, the case to set axis data is shown below.

Data name	Setting of TARGET_REF type data
	Target (Target)
SignalData0002	[AXIS]mcSetValue@Position

4. As the correction value for setting the compensation time according to the data set in Probe data (ProbeData) will change depending on the actual equipment configuration, make adjustments based on "Settings of the delay compensation time peculiar to the system" below.

☞ Page 856 Setting method



Ex.

Compensation time settings

Setting item	Setting value
Communication period interval setting	1000 [μs]
Operation Cycle Setting (System.PrConst.OperationCycle[1])	0: Synchronizes with the basic cycle of network
Motion Synchronization Station Send/Receive Data Refresh Method (System.PrConst.Link_MotionStationRefreshType)	1: Operation cycle preferred method (EmphasisOperationCycle)
Target (MC_TouchProbe.ProbeData.Target)	[AXIS] mcLatestActualValue@Position

With the above setting, Compensation time (CompensationTime) becomes as follows.

Setting item	Setting value
Compensation time (MC_TouchProbe.CompensationTime)	-0.004 [s] (-(0.001 [s] (operation cycle) × 4))

External signal high-accuracy input

For some functions that have external signal input such as touch probe and advanced synchronous control, when the following signals that support external signal high-accuracy input are used for the function that has Trigger input signal (TriggerInput), high-accuracy control with a device station signal detection time is enabled.

For the accuracy of signal detection time, refer to the manual of the device station to be used.

When the following signals are specified for the function that does not support high-accuracy input, the signals operate with the same accuracy as normal signals.

- Signals that can be used to external signal high-accuracy input

  Signals that can be used to external signal high-accuracy input differs depending on the device station to be used.

Device station	Signal name	Description
MR-J5(W)-G series	[OBJ]60B90010.6	Toggle status for latch completion at the rising edge of touch probe 1
	[OBJ]60B90010.7	Toggle status for latch completion at the falling edge of touch probe 1
	[OBJ]60B90010.E	Toggle status for latch completion at the rising edge of touch probe 2
	[OBJ]60B90010.F	Toggle status for latch completion at the falling edge of touch probe 2
	[OBJ]2DE90010.6	Toggle status for latch completion at the rising edge of touch probe 3
	[OBJ]2DE90010.7	Toggle status for latch completion at the falling edge of touch probe 3

■ Relevant variables

I/O signals are expressed by the TARGET_REF structure. For details, refer to the following.

☞ Page 381 External Signal Selection

■ Control details

For the control details of each function when using external signal high-accuracy input, refer to the specification of each function.

For the list of functions that support external signal high-accuracy input, refer to the following.

☞ Page 381 External Signal Selection

Precautions

- When MR-J5(W)-G series is used as external signal high-accuracy input, the touch probe function of MR-J5(W)-G series is required. The touch probe function of MR-J5(W)-G series has restrictions of the device and the version that can be used. For details, refer to the manual of MR-J5(W)-G series.

  MR-J5-G/MR-J5W-G: 📖MR-J5-G/MR-J5W-G User's Manual (Introduction)

  MR-J5D-G: 📖MR-J5D-G User's Manual (Introduction)

  - Connected devices and versions that support the touch probe function

Device name	Supported version
MR-J5-G-RJ	"B6" or later
MR-J5(W)-G	"B6" or later
MR-J5D-G	"C0" or later

■ Required settings for external signal high-accuracy input

External signal high-accuracy input uses a device station signal detection time, so that the setting to obtain the signal detection time is required.

If the required setting is not executed to external signal high-accuracy input, the signals operate with the same accuracy as normal signals.

<When using MR-J5(W)-G series>

When using MR-J5(W)-G series, the touch probe function of MR-J5(W)-G series is used. For the details of parameters and objects of MR-J5(W)-G series, refer to the manual of MR-J5(W)-G.

The touch probe function of MR-J5(W)-G series has restrictions of the device and the version that can be used. For details, refer to the manual of MR-J5(W)-G series.

- Setting of MR-J5(W)-G series

1. Set connector pins to assign TPR1, TPR2, and TPR3 of the input device to be used.

<Setting example>

When using TPR1, TPR2, and TPR3 with MR-J5-G-RJ

- Set "2Ch" to Pr.PD38.0-1. (Assign TPR1 to connector the pin CN3-10.)
- Set "2Dh" to Pr.PD39.0-1. (Assign TPR2 to connector the pin CN3-1.)
- Set "63h" to Pr.PD05.0-1. (Assign TPR3 to connector the pin CN3-19.)

- Setting of the Motion system

1. Set the following slave objects to TPDO of PDO mapping.

For the method of mapping, refer to "CC-Link IE TSN Configuration Window" of the following manual.

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

- When using TPR1

Index	Sub Index	Object name	Description
60B9h	00h	Touch probe status 1	Set to use TPR1 and 2.
60D1h	00h	Touch probe time stamp 1 positive value	Set to detect the rising edge of TPR1. (The setting is not required for detecting the falling edge only.)
60D2h	00h	Touch probe time stamp 1 negative value	Set to detect the falling of TPR1. (The setting is not required for detecting the rising only.)

- When using TPR2

Index	Sub Index	Object name	Description
60B9h	00h	Touch probe status 1	Set to use TPR1 and 2.
60D3h	00h	Touch probe time stamp 2 positive value	Set to detect the rising edge of TPR2. (The setting is not required for detecting the falling edge only.)
60D4h	00h	Touch probe time stamp 2 negative value	Set to detect the falling of TPR2. (The setting is not required for detecting the rising only.)

- When using TPR3

Index	Sub Index	Object name	Description
2DE9h	00h	Touch probe status 2	Set to use TPR3.
2DF8h	00h	Touch probe time stamp 3 positive value	Set to detect the rising edge of TPR3. (The setting is not required for detecting the falling edge only.)
2DF9h	00h	Touch probe time stamp 3 negative value	Set to detect the falling edge of TPR3. (The setting is not required for detecting the rising edge only.)

2. Set high-accuracy input to data of MC_TRIGGER_REF type.

Detection signal	Detection direction	Setting of MC_TRIGGER_REF type data			
		Target (Target)	Signal Detection Method (Detection)*1	Compensation Time (CompensationTime)*2	Filter Time (FilterTime)*3
TPR1	Rising edge	[OBJ]0x60B90010.6@station address	4: Detection at Rising Edge/ Falling Edge (BothEdges)	Arbitrary value	0.0
	Falling edge	[OBJ]0x60B90010.7@station address	4: Detection at Rising Edge/ Falling Edge (BothEdges)	Arbitrary value	0.0
TPR2	Rising edge	[OBJ]0x60B90010.E@station address	4: Detection at Rising Edge/ Falling Edge (BothEdges)	Arbitrary value	0.0
	Falling edge	[OBJ]0x60B90010.F@station address	4: Detection at Rising Edge/ Falling Edge (BothEdges)	Arbitrary value	0.0
TPR3	Rising edge	[OBJ]0x2DE90010.6@station address	4: Detection at Rising Edge/ Falling Edge (BothEdges)	Arbitrary value	0.0
	Falling edge	[OBJ]0x2DE90010.7@station address	4: Detection at Rising Edge/ Falling Edge (BothEdges)	Arbitrary value	0.0

*1 Toggle status for latch completion at the rising edge/the falling edge which is set to Target (Target) differs the status (0 and 1) every detection at rising/falling edge. Therefore, set "4: Detection at Rising Edge/Falling Edge (BothEdges)".
*2 For the setting range, refer to the specification of each function.
*3 To use as the external signal high-accuracy input, set 0.0.

3. Set to the input of function that uses the external signal with MC_TRIGGER_REF type data which has been set.
(Example) Trigger input signal (TriggerInput) of MC_TouchProbe (Touch Probe Enabled)

4. Enable the touch probe function of MR-J5(W)-G series.

Enable the touch probe function of MR-J5(W)-G series using the following objects.

- When using TPR1 and TPR2

Index	Sub Index	Size (number of bits)	Object name
60B8h	00h	10h	Touch probe function 1

- When using TPR3

Index	Sub Index	Size (number of bits)	Object name
2DE8h	00h	10h	Touch probe function 2

Set the following value to the objects above. There are two methods to set the value to the object, one is using the transient transmission and the other one is using the cyclic transmission.

Detection signal	Detection direction	Object name	Value to be set	Description
TPR1	Rising edge	Touch probe function 1	Set "1" to Bit0, 1, 4.	Bit0: Touch probe 1 enabled Bit1: Continuous trigger mode Bit4: Start sampling the rising edge of touch probe 1 Bit5: Start sampling the falling edge of touch probe 1
	Falling edge	Touch probe function 1	Set "1" to Bit0, 1, 5.	
TPR2	Rising edge	Touch probe function 1	Set "1" to Bit8, 9, 12.	Bit8: Touch probe 2 enabled Bit9: Continuous trigger mode Bit12: Start sampling the rising edge of touch probe 2 Bit13: Start sampling the falling edge of touch probe 2
	Falling edge	Touch probe function 1	Set "1" to Bit8, 9, 13.	
TPR3	Rising edge	Touch probe function 2	Set "1" to Bit0, 1, 4.	Bit0: Touch probe 3 enabled Bit1: Continuous trigger mode Bit4: Start sampling the rising edge of touch probe 1 Bit5: Start sampling the falling edge of touch probe 1
	Falling edge	Touch probe function 2	Set "1" to Bit0, 1, 5.	

Point

If the device station that is used is disconnected from the network, reconnect the device station and execute the procedure from 4. again.

●How to use the transient transmission

Write to the following object using MC_WriteParameter (Parameter Write).

For details of MC_WriteParameter (Parameter Write), refer to the following.

☞ Page 730 Parameter Read/Write Function

Point

Object writing by MC_WriteParameter (Parameter Write) can be executed in the device station in which axis is assigned only. For device stations in which the axis is not assigned, use the method for cyclic transmission.

(Program example)

When using TPR1 of MR-J5-G-RJ (When using both of rising edge and falling edge)

ST program

```
// Enable the touch probe function of MR-J5-G-RJ.
// writeValue is a label of DINT
// writeParameterExecute and writeParameterDone are labels of BOOL type.
writeValue := DINT#16#00000033;  //Set Bit0, 1, 4, and 5.
MC_WriteParameter_1(
   Axis:= Axis0001.AxisRef ,
   Execute:= writeParameterExecute ,
   ParameterNumber:= UDINT#16#60B80010 , //Touch probe function 1
   Value:= DINT_TO_LREAL( writeValue ) ,
   ExecutionMode:= 0 ,
   Options:= UDINT#16#000000010000, //Set the setting value of integer data to the integer value without a sign.
   Done => writeParameterDone);
```

●How to use the cyclic transmission

(a) Set the following object to RPDO of PDO mapping.

For the method of mapping, refer to "CC-Link IE TSN Configuration Window" of the following manual.

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

• When using TPR1 and TPR2

Index	Sub Index	Object name
60B8h	00h	Touch probe function 1

• When using TPR3

Index	Sub Index	Object name
2DE8h	00h	Touch probe function 2

(b) Generate a device label of the above object.

For how to generate a device label, refer to the following.

☞ Page 831 Common Items for all devices

(c) Execute the following operation to the generated device label.

(Program example)

When using TPR1 of MR-J5-G-RJ (When using both of rising edge and falling edge)

ST program

```
// Enable the touch probe function of MR-J5-G-RJ.
MR_J5_G_RJ_001_ TouchProbeFunction:= UINT#16#0033; //(Set 1 to BIT0, 1, 4, and 5.)
```

5. Check whether the touch probe function of MR-J5(W)-G series is enabled.

Check whether the touch probe function of MR-J5(W)-G series is enabled using the following object.

• When using TPR1 and TPR2

Index	Sub Index	Size (number of bits)	Object name
60B9h	00h	10h	Touch probe status 1

• When using TPR3

Index	Sub Index	Size (number of bits)	Object name
2DE9h	00h	10h	Touch probe status 1

Check the following value is stored to the above object. There are two methods to check the object, one is using the transient transmission and the other one is using the cyclic transmission.

Detection signal	Detection direction	Object name	Value at touch probe enabled	Description
TPR1	Rising edge	Touch probe Status1	Bit0 is set to 1.	Bit0: Touch probe 1 enabled
	Falling edge	Touch probe Status1	Bit0 is set to 1.	
TPR2	Rising edge	Touch probe Status1	Bit8 is set to 1.	Bit8: Touch probe 2 enabled
	Falling edge	Touch probe Status1	Bit8 is set to 1.	
TPR3	Rising edge	Touch probe Status2	Bit0 is set to 1.	Bit0: Touch probe 3 enabled
	Falling edge	Touch probe Status2	Bit0 is set to 1.	

●How to use the transient transmission

Read the following object using MC_ReadParameter (Parameter Read).

For details of MC_ReadParameter (Parameter Read), refer to the following.

☞ Page 730 Parameter Read/Write Function

(Program example)

When using TPR1 of MR-J5-G-RJ

ST program

```
// Obtain the status of touch probe of MR-J5-G-RJ.
// readValue is a label of LREAL.
// tempValue is a label of INT.
// readParameterEnable and readParameterValid are labels of BOOL type.
MC_ReadParameter_1(
    Axis:= Axis0001.AxisRef ,
    Enable:= readParameterEnable ,
    ParameterNumber:= UDINT#16#60B90010 , //Touch probe status 1
    Options:= UDINT#16#000000010000, //Set the setting value of integer data to the integer value without a sign.
    Valid => readParameterValid,
    Value => readValue);

// Check whether touch probe of MR-J5-G-RJ is enabled.
tempValue := LREAL_TO_INT(readValue);
IF tempValue.0 THEN
  //Start the function that uses external signal high-accuracy input.
END_IF;
```

●How to use the cyclic transmission

(a) Generate a device label of the following object.

For how to generate a device label, refer to the following.

☞ Page 831 Common Items for all devices

• When using TPR1 and TPR2

Index	Sub Index	Object name
60B9h	00h	Touch probe status 1

• When using TPR3

Index	Sub Index	Object name
2DE9h	00h	Touch probe status 2

(b) Execute the following operation to the generated device label.
(Program example)
When using TPR1 of MR-J5-G-RJ (When using both of rising edge and falling edge)

ST program

// Check whether touch probe of MR-J5-G-RJ is enabled.
IF MR_J5_G_RJ_001_ TouchProbeStaus.0 THEN
  //Start the function that uses external signal high-accuracy input.
END_IF;

■ How to check whether high-accuracy input is enabled

Check whether a signal detection time is stored to the object or link device those stores a device station signal detection time at the signal detection using the device label, etc.

- When using MR-J5(W)-G series
  - When using TPR1
    - To detect the rising edge of signal: Touch probe time stamp 1 positive value(60D1h)
    - To detect the falling edge of signal: Touch probe time stamp 1 negative value(60D2h)
  - When using TPR2
    - To detect the rising edge of signal: Touch probe time stamp 2 positive value(60D3h)
    - To detect the falling edge of signal: Touch probe time stamp 2 negative value(60D4h)
  - When using TPR3
    - To detect the rising edge of signal: Touch probe time stamp 3 positive value(2DF8h)
    - To detect the falling edge of signal: Touch probe time stamp 3 negative value(2DF9h)

■ Program example (When using MR-J5-G-RJ series)

- When TPR1 of MR-J5-G-RJ (station address: 192.168.3.1) is used as Trigger input signal (TriggerInput) of MC_TouchProbe (Touch Probe Enabled) at the rising edge detection

<Setting of MR-J5-G-RJ>

1. Set a connector pin to assign TPR1 of the input device to be used.
Set "2Ch" to Pr.PD38.0-1. (Assign TPR1 to the connector pin CN3-10.)

<Setting of the Motion system>

1. Set the following slave objects to TPDO of PDO mapping.
For the method of mapping, refer to "CC-Link IE TSN Configuration Window" of the following manual.
MELSEC iQ-R Motion Module User's Manual (Network)

Index	Sub Index	Object name
60B9h	00h	Touch probe status 1
60D1h	00h	Touch probe time stamp 1 positive value

2. Generate a label to be used as Trigger input signal (TriggerInput) of MC_TouchProbe (Touch Probe Enabled).
Create a new data of MC_TRIGGER_REF type and set Toggle status (Touch probe status 1 (Bit6)) for latch completion of TPR1 to Target (Target).

Data name	Setting of MC_TRIGGER_REF type data			
	Target (Target)	Signal Detection Method (Detection)	Compensation Time (CompensationTime)	Filter Time (FilterTime)
SignalData0001	[OBJ]0x60B90010.6@192.168.3.1	4: Detection at RisingEdge/Falling Edge (BothEdges)	0.0	0.0

3. Generate a label to be used as Probe data (ProbeData) of MC_TouchProbe (Touch Probe Enabled).
Create a new data of TARGET_REF type and set Probe data (ProbeData). As an example, the case to set axis data is shown below.

Data name	Setting of TARGET_REF type data
	Taeget (Target)
SignalData0002	[AXIS]mcSetValue@Position

Point

When setting the data which is based on device information in Probe data (ProbeData), a delay peculiar to the system occurs. Set the compensation time by referring to "Settings of the delay time peculiar to the system" below and compensate the delay. As the delay time changes depending on the actual equipment configuration, make adjustments based on reference values of "Settings of the delay time peculiar to the system" below.

☞ Page 856 Setting method

4. Enable the touch probe function of MR-J5-G-RJ by ST program and start MC_TouchProbe (Touch Probe Enabled).

ST program

```
// Enable the touch probe function of MR-J5-G-RJ.
// writeValue is a label of DINT.
// writeParameterExecute, writeParameterDone, and touchProbeExecute are labels of BOOL type.
writeValue := DINT#16#00000013;  //Set Bit0, 1, 4.
MC_WriteParameter_1(
   Axis:= Axis0001.AxisRef ,
   Execute:= writeParameterExecute ,
   ParameterNumber:= UDINT#16#60B80010 , //Touch probe function 1
   Value:= DINT_TO_LREAL( writeValue ) ,
   ExecutionMode:= 0 ,
   Options:= UDINT#16#000000010000, //Set the setting value of integer data to the integer value without a sign.
   Done => writeParameterDone);

// Obtain the status of touch probe of MR-J5-G-RJ.
// readValue is a label of LREAL. tempValue is a label of INT.
// readParameterEnable and readParameterValid are labels of BOOL type.
MC_ReadParameter_1(
   Axis:= Axis0001.AxisRef ,
   Enable:= readParameterEnable ,
   ParameterNumber:= UDINT#16#60B90010 , //Touch probe status 1
   Options:= UDINT#16#000000010000, //Set the setting value of integer data to the integer value without a sign.
   Valid => readParameterValid,
   Value => readValue );

tempValue := LREAL_TO_INT(readValue); // When touch probe of MR-J5-G-RJ is enabled
IF tempValue.0 THEN
  touchProbeExecute := TRUE; //Set TRUE to Execute of MC_TouchProbe.
  readParameterEnable := FALSE; //Stop readParameter.
END_IF;

// Set external signal high-accuracy input to TriggerInput of MC_TouchProbe.
// Latch the data set to ProbeData at the rising edge of TPR1 of MR-J5-G-RJ.
// touchProbeRecordedPos is a label of LREAL type.
MC_TouchProbe_1(
   Axis:= Axis0001.AxisRef ,
   Execute := touchProbeExecute,
   TriggerInput := SignalData0001 ,
   ProbeData := SignalData0002 ,
   RecordedPosition => touchProbeRecordedPos);
```

■ Precautions

- When using external signal high-accuracy input, set "0.0" to Filter Time (FilterTime) of the external signal selection.
  If the device station has the input filter, use the filter of the device station.
  When using external signal high-accuracy input and setting other than "0.0" to Filter Time (FilterTime) of the external signal selection, a signal detection time cannot be detected correctly.
- Note that the signal to be input to the device station as high-accuracy input should not be turned ON/OFF several times within a communication cycle. If the signal turns ON/OFF several times within a communication cycle, the signal cannot be detected correctly.

Master-slave operation

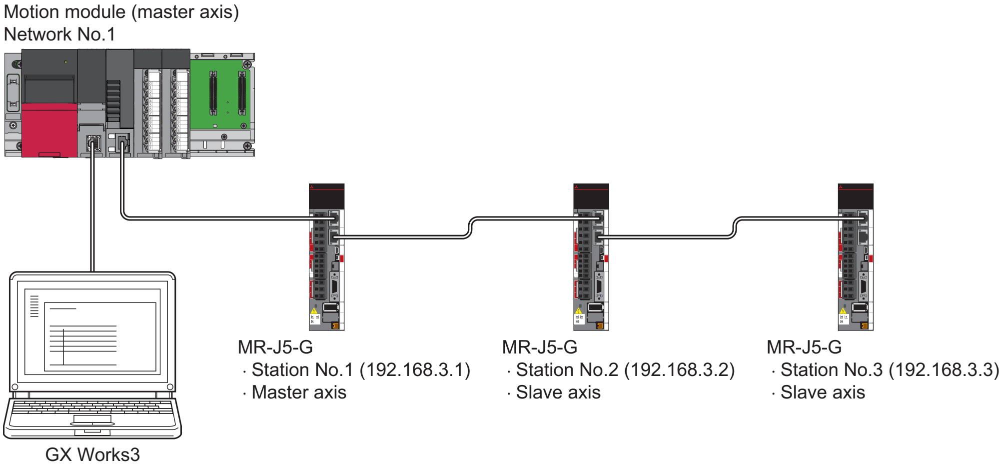
This section describes an example of how to use master-slave operation.

For details on master-slave operation, refer to the following.

☞ Page 407 Master-Slave Operation

■ System configuration

This section uses the following system configuration example to describe the master-slave operation.



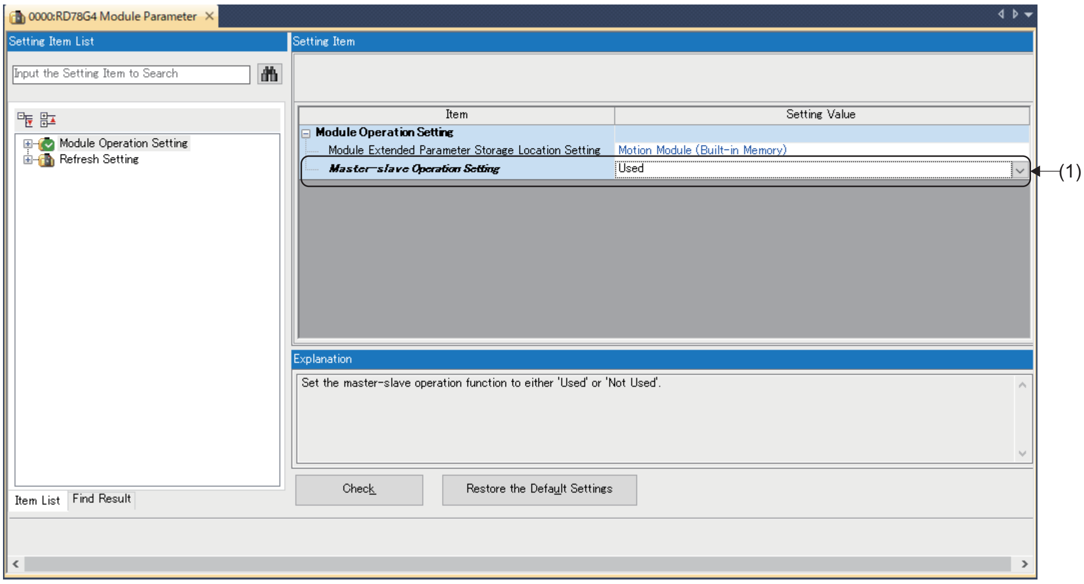
■ Parameter setting

This section describes each parameter setting.

<Master-slave operation setting>

- Set "Master-slave Operation Setting"(1) of the module parameter (motion) to "Used".

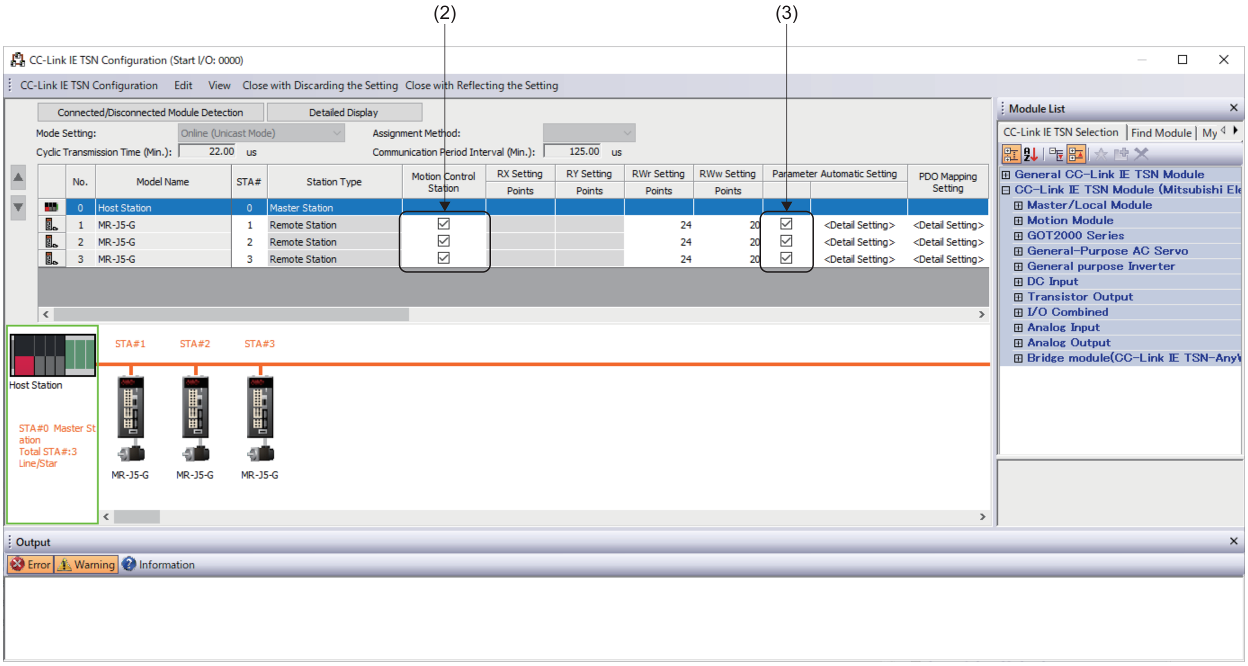
Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ [Module model] ⇨ [Module Parameter (Motion)] ⇨ "Parameter editor" screen ⇨ [Module Operation Setting] ⇨ [Master-slave Operation Setting]



<Network configuration setting>

- Check "Motion Control Station"(2) and "Parameter Automatic Setting"(3) of the stations specified as the master axis and the slave axis.

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ [Module model] ⇨ [Module Parameter (Network)] ⇨ "Parameter editor" screen ⇨ [Basic Settings] ⇨ [Network Configuration Settings] ⇨ [Detailed Setting]



<Servo parameter setting>

Set the servo parameters for the station in which master-slave operation is enabled.

For details on the servo parameters used for master-slave operation, refer to the following.

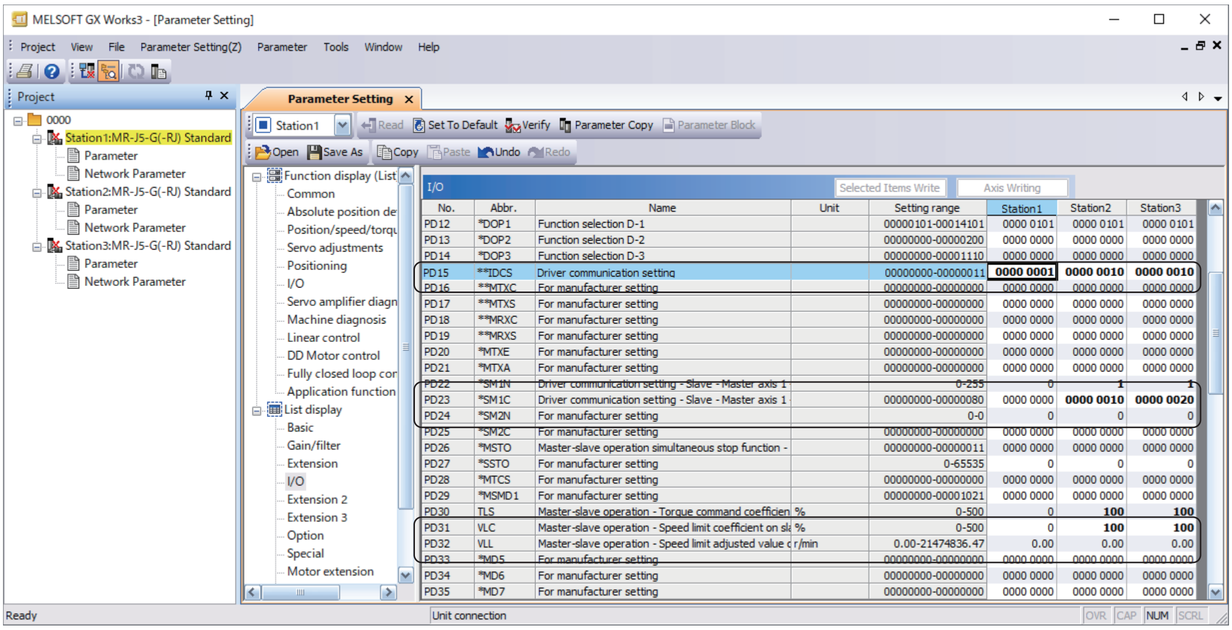
☞ Page 415 Parameter automatic setting

- Set the following servo parameters for the station in which master-slave operation is enabled.

No.	Name	Station 1	Station 2	Station 3
PD15.0	Master axis operation selection	1: Enabled	0: Disabled	0: Disabled
PD15.1	Slave axis operation selection	0: Disabled	1: Enabled	1: Enabled
PD22	Driver communication setting - Slave - Master axis 1 - Station No. setting	0	1	1
PD23.1	Driver communication setting - Slave - Master axis 1 - Control slave axis No. setting	0	1	2
PD30	Master-slave operation - Slave-side torque command coefficient	0	100*1	100*1
PD31	Master-slave operation - Slave-side speed limit coefficient	0	100*1	100*1

*1 Set the parameters according to your environment.

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ [Module model] ⇨ [Module Parameter (Network)] ⇨ "Parameter editor" screen ⇨ [Basic Settings] ⇨ [Network Configuration Settings] ⇨ [Detailed Setting] ⇨ "CC-Link IE TSN Configuration" screen ⇨ [Parameter Automatic Setting] ⇨ [Detailed Setting]



Set other servo parameters according to your environment.

<Communication period interval setting>

Set the communication period interval setting (4) and the cyclic transmission time (5).

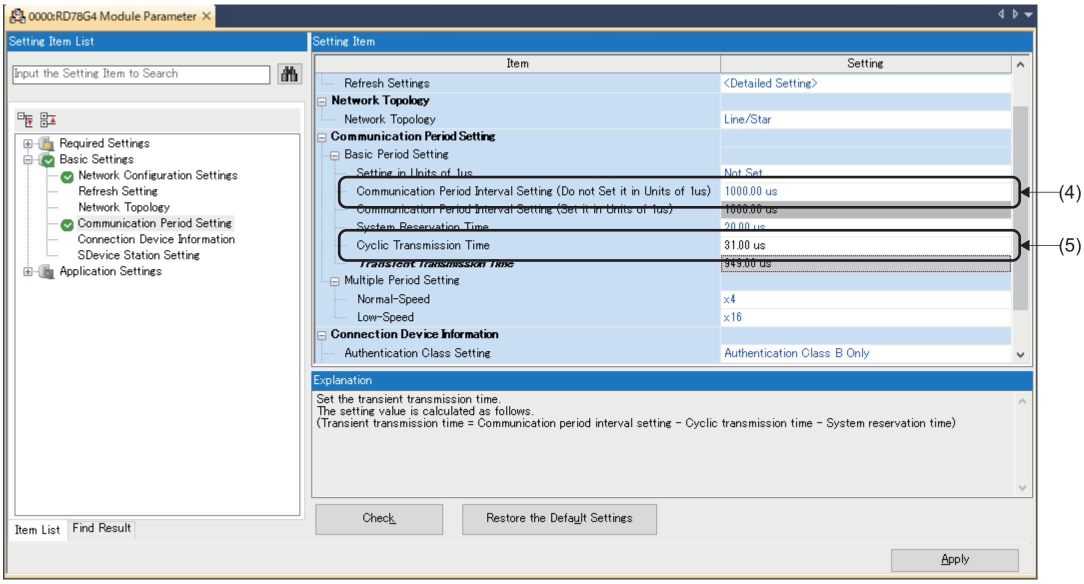
For the setting details, refer to the following.

☞ Page 414 Communication period setting

Set the following values.

Item	Setting value
Communication period interval setting	1000.00 [µs]
Cyclic transmission time	31.00 [µs]

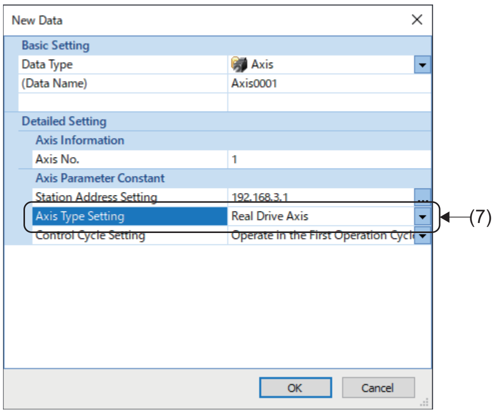
Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ [Module model] ⇨ [Module Parameter (Network)] ⇨ "Parameter editor" screen ⇨ [Basic Settings] ⇨ [Communication Period Setting]



<Module extended parameter>

Create a new axis for the axis specified as the master axis and set the axis type setting to "Real Drive Axis" (7).

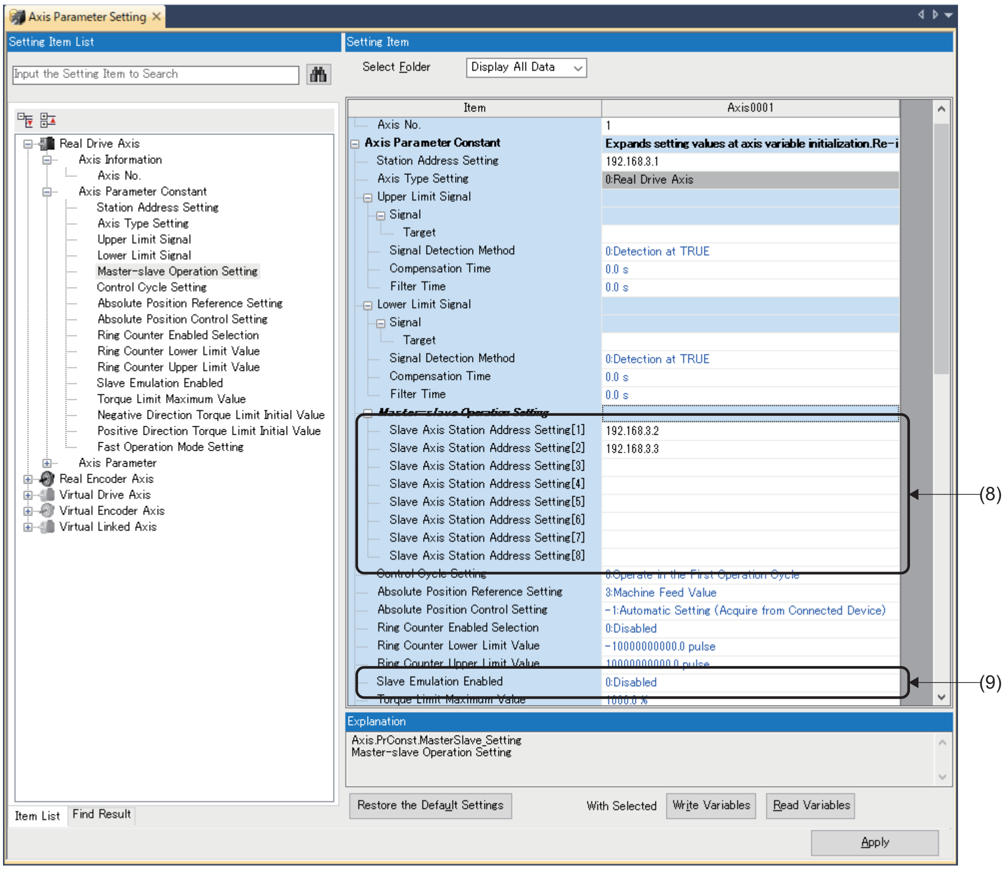
Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ [Module model] ⇨ [Module Extended Parameter] ⇨ [Motion Control Setting Function] ⇨ Navigation window ⇨ [Axis] ⇨ Right-click ⇨ [Add New Data]



Set the station addresses for the slave axes in "Slave Axis Station Address Setting [1 to 8]" (8) in the master-slave operation setting of the axis parameter constant.

In addition, set "Axis Emulation Enabled" (9) of the axis parameter constant to "0: Disabled".

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ [Module model] ⇨ [Module Extended Parameter] ⇨ [Motion Control Setting Function] ⇨ Navigation window ⇨ [Axis] ⇨ [Target axis] ⇨ [Axis Parameter Constant]



■ Program example

<Slave labels>

The slave labels used in the program example are shown below.

For how to use slave labels, refer to the following.

☞ Page 831 How to use the slave label

Master axis/slave axis	Device label	Label name
Master axis	MR_J5_G_001	—
Slave axis 1	MR_J5_G_002	MR_J5_G_002_Controlword
		MR_J5_G_002_PositiveTorqueLimitValue
		MR_J5_G_002_NegativeTorqueLimitValue
		MR_J5_G_002_Statusword
Slave axis 2	MR_J5_G_003	MR_J5_G_003_Controlword
		MR_J5_G_003_PositiveTorqueLimitValue
		MR_J5_G_003_NegativeTorqueLimitValue
		MR_J5_G_003_Statusword

<Labels>

The labels used in the program example are shown below.

Label name	Data type	Comment
MC_Power_1	MC_Power	Master axis servo ON FB
bSlaveAxis1_ServoOnReq	Bit	Slave axis 1 servo ON request
bSlaveAxis1_ServoOnSts	Bit	Slave axis 1 servo ON status
bSlaveAxis2_ServoOnReq	Bit	Slave axis 2 servo ON request
bSlaveAxis2_ServoOnSts	Bit	Slave axis 2 servo ON status
bSlaveAxis1_ErrorReset	Bit	Slave axis 1 error reset request
bSlaveAxis2_ErrorReset	Bit	Slave axis 2 error reset request

<Program for servo ON>

Execute servo ON of the slave axis and the master axis when Master-slave operation - axes group ready (Axis0001.Md.MasterSlave_ReadyOnGroup) of the master axis is TRUE.

- Master axis

ST program

```
//-----Axes Group Ready-----
IF ( Axis0001.Md.MasterSlave_ReadyOnGroup = TRUE ) THEN
  //-----Master axis servo ON-----
  MC_Power_1(Axis:= Axis0001.AxisRef, Enable:= TRUE, ServoON:= TRUE);
  //-----Slave axis 1, 2 servo ON-----
  bSlaveAxis1_ServoOnReq := TRUE;
  bSlaveAxis2_ServoOnReq := TRUE;
ELSE
  //-----Master axis servo OFF-----
  MC_Power_1(Axis:= Axis0001.AxisRef, Enable:= FALSE ,ServoON:= FALSE );
  //-----Slave axis 1, 2 servo OFF-----
  bSlaveAxis1_ServoOnReq := FALSE;
  bSlaveAxis2_ServoOnReq := FALSE;
END_IF;
```

- Slave axis 1

ST program

```
//-----Data link check-----
IF NOT SW0B0.1 THEN
  //----- Servo ON-----
  IF bSlaveAxis1_ServoOnReq THEN
    //-----PDS state transitions
    //-----For MR-J5, servo ON (transition to the Operation enabled state) can be executed directly by sending
    // the "Enable operation" command in the "Switch on disabled" state.-----
    IF MR_J5_G_002_Statusword.6 THEN //-----PDS state: Switch on disabled-----
      //-----Command: Shutdown-----
      MR_J5_G_002_Controlword := MR_J5_G_002_Controlword AND HFFF0 OR H0006;
      bSlaveAxis1_ServoOnSts := FALSE;
    ELSE
        CASE WORD_TO_INT(MR_J5_G_002_Statusword AND H000F) OF
        1:    //-----PDS state: Ready to switch on-----
              //-----Command: Switch on-----
              MR_J5_G_002_Controlword := MR_J5_G_002_Controlword AND HFFF0 OR H0007;
              bSlaveAxis1_ServoOnSts := FALSE;
        3:    //-----PDS state: Switched on-----
              //-----Command: Enable operation-----
              MR_J5_G_002_Controlword := MR_J5_G_002_Controlword AND HFFF0 OR H000F;
              bSlaveAxis1_ServoOnSts := FALSE;
        7:    //-----PDS state: Operation enabled-----
              bSlaveAxis1_ServoOnSts := TRUE;
        ELSE //-----PDS state: Fault , Fault reaction active-----
              bSlaveAxis1_ServoOnSts := FALSE;
        END_CASE;

    END_IF;
  ELSE
    //-----Command: Shutdown-----
    MR_J5_G_002_Controlword := MR_J5_G_002_Controlword AND HFFF0 OR H0006;
    bSlaveAxis1_ServoOnSts := FALSE;
  END_IF;

  //-----Torque limit-----
  MR_J5_G_002_PositiveTorqueLimitValue := 1000; //-----100%-----
  MR_J5_G_002_NegativeTorqueLimitValue := 1000; //-----100%-----
END_IF;
```

A

- Slave axis 2

ST program

```
//-----Data link check-----
IF NOT SW0B0.2 THEN
    //-----Servo ON-----
    IF bSlaveAxis2_ServoOnReq THEN
        //-----PDS state transitions-----
        //-----For MR-J5, servo ON (transition to the Operation enabled state) can be executed directly
        //by sending the "Enable operation" command in the "Switch on disabled" state.-----
        IF MR_J5_G_003_Statusword.6 THEN // PDS state: Switch on disabled
            bSlaveAxis2_ServoOnSts := FALSE;
            //-----Command: Shutdown-----
            MR_J5_G_003_Controlword := MR_J5_G_003_Controlword AND HFFF0 OR H0006;
            ELSE
            CASE WORD_TO_INT(MR_J5_G_003_Statusword AND H000F) OF
            1:      //-----PDS state: Ready to switch on-----
                    //-----Command: Switch on-----
                    MR_J5_G_003_Controlword := MR_J5_G_003_Controlword AND HFFF0 OR H0007;
                    bSlaveAxis2_ServoOnSts := FALSE;
            3:      //-----PDS state: Switched on-----
                    //-----Command: Enable operation-----
                    MR_J5_G_003_Controlword := MR_J5_G_003_Controlword AND HFFF0 OR H000F;
                    bSlaveAxis2_ServoOnSts := FALSE;
            7:      //-----PDS state: Operation enabled-----
                    bSlaveAxis2_ServoOnSts := TRUE;
            ELSE //-----PDS state: Fault , Fault reaction active-----
                    bSlaveAxis2_ServoOnSts := FALSE;
            END_CASE;

        END_IF;
    ELSE
        //-----Command: Shutdown-----
        MR_J5_G_003_Controlword := MR_J5_G_003_Controlword AND HFFF0 OR H0006;
        bSlaveAxis2_ServoOnSts := FALSE;
    END_IF;

    //-----Torque limit-----
    MR_J5_G_003_PositiveTorqueLimitValue := 1000; //-----100%-----
    MR_J5_G_003_NegativeTorqueLimitValue := 1000; //-----100%-----
END_IF;
```

<Program for slave axis error reset>

The following is a program example for error reset when an error occurs on the slave axis.

- Slave axis 1

ST program

```
//-----Data link check-----
IF NOT SW0B0.1 THEN
    //-----Error reset-----
    IF bSlaveAxis1_ErrorReset THEN
        IF MR_J5_G_003_Statusword.3 THEN //-----PDS state: Fault , Fault reaction active-----
            //-----Command: Fault Reset-----
            MR_J5_G_002_Controlword.7 := TRUE;
        ELSE
            MR_J5_G_002_Controlword.7 := FALSE;
            bSlaveAxis1_ErrorReset := FALSE;
        END_IF;
    END_IF;
END_IF;
```

- Slave axis 2

ST program

```
//-----Data link check-----
IF NOT SW0B0.2 THEN
    //-----Error reset-----
    IF bSlaveAxis2_ErrorReset THEN
        IF MR_J5_G_003_Statusword.3 THEN //-----PDS state: Fault , Fault reaction active-----
            //-----Command: Fault Reset-----
            MR_J5_G_002_Controlword.7 := TRUE;
        ELSE
            MR_J5_G_002_Controlword.7 := FALSE;
            bSlaveAxis2_ErrorReset := FALSE;
        END_IF;
    END_IF;
END_IF;
```

A

Precautions

When using MR-J5(W)-G with old firmware, "Watchdog Counter Error (error code: 1C41H)" may occur at connecting.

When this symptom occurs, update the firmware of MR-J5(W)-G.

Appendix 2 Restrictions by the version

Compatible versions of the engineering tool

The CPU module and engineering tool versions compatible with each Motion module are shown below.

Motion module		CPU module		Engineering tool version	Motion control FB Library version
Model	Software version	Model	Firmware version	GX Works3	
RD78G	"04" or earlier	R00CPU, R01CPU, R02CPU	"12" or later	"1.056J" or later	"1.00A"
		R04CPU, R08CPU, R16CPU, R32CPU, R120CPU	"44" or later		
		R04ENCPU, R08ENCPU, R16ENCPU, R32ENCPU, R120ENCPU	"44" or later		
	"05" or later	R00CPU, R01CPU, R02CPU	"12" or later	"1.060N" or later (The motion control setting function "1.005F" or later)	"1.00A" or later
		R04CPU, R08CPU, R16CPU, R32CPU, R120CPU	"44" or later		
		R04ENCPU, R08ENCPU, R16ENCPU, R32ENCPU, R120ENCPU	"44" or later		
RD78GH	—	R00CPU, R01CPU, R02CPU	"14" or later	"1.060N" or later	"1.00A" or later
		R04CPU, R08CPU, R16CPU, R32CPU, R120CPU	"46" or later		
		R04ENCPU, R08ENCPU, R16ENCPU, R32ENCPU, R120ENCPU	"46" or later		

■ Compatible versions of the Motion control FB library

The engineering tool versions ompatible with the Motion control FB Library are shown below.

Motion control FB Library version	Engineering tool version	
	GX Works3	Motion control setting function
"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

Point

The engineering tools other than those mentioned in this section do not support the Motion module.

■ Compatible versions for each Motion module function

The compatible CPU module and engineering tool versions have restrictions depending on the functions of the Motion module.

The Motion module function and each compatible version are shown below.

Point

The add-on versions and software versions of the Motion module combinations are shown below.

☞ Page 879 List of Add-on Library

—: There is no restriction by the version.

Function	Add-on library		CPU module		Engineering tool version		Motion control FB Library version	
	Library name	Version	Model	Firmware version	GX Works3	Motion control setting function	FB name	Version
Supports communication cycle 31.25 μs, 62.5 μs and 8000.0 μs	baseSystem	1.5	—	—	1.060N	—	—	—
Supports network synchronous communication setting	baseSystem	1.5	—	—	1.060N	—	—	—
Supports PDO mapping configuration	baseSystem	1.5	—	—	1.060N	—	—	—
	ServoDriver_CANopen	1.4	—	—				
Supports safety communication function	baseSystem	1.8	R08SFCPU/ R16SFCPU/ R32SFCPU/ R120SFCPU	20	1.065T	—	—	—
	NetworkDriver_CCIETSN	1.7						
Supports communication speed 100 Mbps	baseSystem	1.8	—	—	1.065T	—	—	—
Supports Inter-module Synchronization Function	baseSystem	1.16	R00CPU/ R01CPU/ R02CPU	24	1.075D	—	—	—
			R04CPU/ R08CPU/ R16CPU/ R32CPU/ R120CPU	57				
			R04ENCPU/ R08ENCPU/ R16ENCPU/ R32ENCPU/ R120ENCPU	57				
Supports the Simple motion mode	baseSystem	1.16	—	—	1.075D	—	—	—
	Axis	1.15						
	ServoDriver_CANopen	1.15						
	NetworkDriver_CCIETSN	1.15						
	SignalIO	1.15						
	SimpleMotion	1.15						
Supports station information list	baseSystem	1.16	—	—	1.075D	—	—	—
Supports up to 120 network stations	baseSystem	1.5	—	—	—	—	—	—
Supports general purpose connection of CANopen devices	NetworkDriver_CCIETSN	1.4	—	—	—	—	—	—

Function	Add-on library		CPU module		Engineering tool version		Motion control FB Library version	
	Library name	Version	Model	Firmware version	GX Works3	Motion control setting function	FB name	Version
Supports partial read and write of operation profile	ProfileControl	1.4	—	—	—	—	—	—
	PackagingApp	1.4						
Adds file transfer commands (copy, move, file, compress, extract and attrib)	FileTransfer	1.4	—	—	—	—	—	—
Supports Continuous operation to torque control mode	MotionEngine	1.7	—	—	—	—	—	—
	ServoDriver_CANopen	1.7						
Supports watchdog counter	baseSystem	1.16	—	—	—	—	—	—
	NetworkDriver_CCIETSN	1.15						
Supports MR-J5D-G	baseSystem	1.17	—	—	—	—	—	—
Adds fast axis operation mode	Axis	1.4	—	—	—	1.005F	—	—
	ServoDriver_CANopen	1.4						
	NetworkDriver_CCIETSN	1.4						
Adds encoder axis types	Axis	1.4	—	—	—	1.005F	—	—
	ServoDriver_CANopen	1.4						
Adds override function, MC_SetOverride and MC_GroupSetOverride	MotionEngine	1.4	—	—	—	1.005F	MC_SetOverride, MC_GroupSetOverride	00A
	Axis	1.4						
	MotionControl_General	1.4						
Adds MC_ReadParameter and MC_WriteParameter	Axis	1.4	—	—	—	1.005F	MC_ReadParameter, MC_WriteParameter	00A
	MotionControl_General	1.4						
Adds MC_Reset, MC_GroupReset and MCv_MotionErrorReset	MotionControl_General	1.4	—	—	—	1.005F	MCv_MotionErrorReset, MC_GroupReset, MC_Reset	00A
Adds MC_TouchProbe and MC_AbortTrigger	MotionControl_General	1.4	—	—	—	1.005F	MC_TouchProbe, MC_AbortTrigger	00A
Supports unit label	baseSystem	1.5	—	—	1.060N	1.005F	—	—
	Axis	1.4						
	Program_ST	1.4						
Supports TargetIgnored status of the drive unit	Axis	1.4	—	—	—	1.005F	—	—
	ServoDriver_CANopen	1.4						
Adds file transfer access control	FileTransfer	1.4	—	—	—	1.005F	—	—
Adds MCv_BacklashCompensationFilter, MCv_DirectionFilter and MCv_SpeedLimitFilter	MotionControl_AxisFilter	1.4	—	—	—	1.005F	MCv_BacklashCompensationFilter, MCv_DirectionFilter, MCv_SpeedLimitFilter	00A
Adds purged pool monitor in MCv_SmoothingFilter	MotionControl_AxisFilter	1.4	—	—	—	1.005F	MCv_SmoothingFilter	01B

Function	Add-on library		CPU module		Engineering tool version		Motion control FB Library version	
	Library name	Version	Model	Firmware version	GX Works3	Motion control setting function	FB name	Version
Supports RD78GH	baseSystem	1.5	R00CPU/ R01CPU/ R02CPU	14	1.060N	1.005F	—	—
			R04CPU/ R08CPU/ R16CPU/ R32CPU/ R120CPU	46				
			R04ENCPU/ R08ENCPU/ R16ENCPU/ R32ENCPU/ R120ENCPU	46				
Adds memory usage monitor for BaseSystem	—	—	—	—	—	1.005F	—	—
Adds clearing command for motion event history	MotionEventHist	1.4	—	—	—	1.005F	—	—
Adds logging realtime monitor	Logging	1.4	—	—	—	1.005F	—	—
Supports initial stroke of operation profile	ProfileControl	1.4	—	—	—	1.005F	—	—
Adds Servo System Recorder	Axis	1.7	—	—	—	1.010L	—	—
	Logging	1.7						
Supports Position data history	Axis	1.7	—	—	—	1.010L	—	—
	MotionEventHist	1.7						
Supports Response Preferred Method	NetworkDriver_CCIETSN	1.19	—	—	—	1.030G	—	—
Supports security key authentication function	baseSystem	1.24	—	—	—	1.035M	—	—
Supports LB/LW	baseSystem	1.24	—	—	1.085P	1.035M	—	—
	SignalIO	1.23						
	Program_ST	1.23						
Supports Advanced synchronous control	baseSystem	1.26	—	—	—	1.040S	MCv_AdvancedSync, MCv_AdvPosition PerCycleCalc, MCv_AdvCamSetPositionCalc	06G
	MotionEngine	1.25						
	Axis	1.25						
	MotionControl_General	1.25						
	MotionControl_AdvancedSync	1.25						
Supports Co-recording function	baseSystem	1.26	R00CPU/ R01CPU/ R02CPU/ R04CPU/ R08CPU/ R16CPU/ R32CPU/ R120CPU/ R04ENCPU/ R08ENCPU/ R16ENCPU/ R32ENCPU/ R120ENCPU	65	1.090U	1.040S	—	—
	Logging	1.25						
	ServoSystemRecorder	1.25						
Supports selective write function	baseSystem	1.26	—	—	—	1.040S*1	—	—
Supports Multiple Axes Positioning Data Operation (MCv_PositioningData)	MotionEngine	1.27	—	—	—	1.042U	MCv_PositioningData	1.07H
	MotionControl_General	1.27						
	ProfileControl	1.27						

Function	Add-on library		CPU module		Engineering tool version		Motion control FB Library version	
	Library name	Version	Model	Firmware version	GX Works3	Motion control setting function	FB name	Version
Supports master-slave operation function	baseSystem	1.28	—	—	1.095Z*2	1.042U	—	—
	Axis	1.27						
	ServoDoriver_CANopen	1.27						
	NetworkDriver_CCIETSN	1.27						
Supports operation profile (positioning data format)	ProfileControl	1.27	—	—	—	1.042U	—	—
Supports Multiple Axes Positioning Data Operation (MCv_MovePositioningData) conditional JUMP	MotionEngine	1.27	—	—	—	1.050C	—	—
	MotionControl_General	1.27						
	ProfileControl	1.27						
Adds operation profile (positioning data format) condition signal	ProfileControl	1.27	—	—	—	1.050C	—	—
Supports IP address setting function	baseSystem	1.32	—	—	—	—	—	—
Supports operation cycle settings subdivision	baseSystem	1.32	—	—	—	—	—	—

*1 When using the selective write function with baseSystem version "1.26", the project data must be written in advance with motion control setting function version "1.040S" or later.
After writing the project data using baseSystem version "1.26" or later and motion control setting function version "1.040S" or later, restoring to a baseSystem version earlier than "1.26" will cause an error in the motion module. In this case, the project data must be written again.

*2 Registering the profile version "09C" for software version "28" of the Motion module or using GX Works3 version "1.095Z" or later is required.
The profile can be downloaded from MITSUBISHI ELECTRIC FA Global Website.
For how to register profiles, refer to the following.
📖GX Works3 Operating Manual

Appendix 3 List of Add-on Library

Each add-on version which is compatible with the software version of the Motion module is shown below.

Software version	Add-on library name						
	baseSystem	MotionEngine	Axis	MotionControl_General	MotionControl_Sync	AbsSystem	ServoDriver_CANopen
01	1.0.1.0	1.0.1.0	1.0.1.0	1.0.1.0	1.0.1.0	1.0.1.0	1.0.1.0
02	1.2.1.0	1.1.1.0	1.1.1.0	1.1.1.0	1.1.1.0	1.1.1.0	1.1.1.0
03	1.3.1.0	1.2.1.0	1.2.1.0	1.2.1.0	1.2.1.0	1.2.1.0	1.2.1.0
04	1.4.1.0	1.3.1.0	1.3.1.0	1.3.1.0	1.3.1.0	1.3.1.0	1.3.1.0
05	1.5.1.0	1.4.1.0	1.4.1.0	1.4.1.0	1.4.1.0	1.4.1.0	1.4.1.0
06	1.6.1.0	1.5.1.0	1.5.1.0	1.5.1.0	1.5.1.0	1.5.1.0	1.5.1.0
07	1.7.1.0	1.6.1.0	1.6.1.0	1.6.1.0	1.6.1.0	1.6.1.0	1.6.1.0
08	1.8.1.0	1.7.1.0	1.7.1.0	1.7.1.0	1.7.1.0	1.7.1.0	1.7.1.0
09	1.9.1.0	1.8.1.0	1.8.1.0	1.8.1.0	1.8.1.0	1.8.1.0	1.8.1.0
10	1.10.1.0	1.9.1.0	1.9.1.0	1.9.1.0	1.9.1.0	1.9.1.0	1.9.1.0
12	1.12.1.0	1.11.1.0	1.11.1.0	1.11.1.0	1.11.1.0	1.11.1.0	1.11.1.0
13	1.13.1.0	1.12.1.0	1.12.1.0	1.12.1.0	1.12.1.0	1.12.1.0	1.12.1.0
14	1.14.1.0	1.13.1.0	1.13.1.0	1.13.1.0	1.13.1.0	1.13.1.0	1.13.1.0
16	1.16.1.0	1.15.1.0	1.15.1.0	1.15.1.0	1.15.1.0	1.15.1.0	1.15.1.0
17	1.17.1.0	1.16.1.0	1.16.1.0	1.16.1.0	1.16.1.0	1.16.1.0	1.16.1.0
18	1.18.1.0	1.17.1.0	1.17.1.0	1.17.1.0	1.17.1.0	1.17.1.0	1.17.1.0
20	1.20.1.0	1.19.1.0	1.19.1.0	1.19.1.0	1.19.1.0	1.19.1.0	1.19.1.0
21	1.21.1.0	1.20.1.0	1.20.1.0	1.20.1.0	1.20.1.0	1.20.1.0	1.20.1.0
22	1.22.1.0	1.21.1.0	1.21.1.0	1.21.1.0	1.21.1.0	1.21.1.0	1.21.1.0
24	1.24.1.0	1.23.1.0	1.23.1.0	1.23.1.0	1.23.1.0	1.23.1.0	1.23.1.0
26	1.26.1.0	1.25.1.0	1.25.1.0	1.25.1.0	1.25.1.0	1.25.1.0	1.25.1.0
28	1.28.1.0	1.27.1.0	1.27.1.0	1.27.1.0	1.27.1.0	1.27.1.0	1.27.1.0
30	1.30.1.0	1.29.1.0	1.29.1.0	1.29.1.0	1.29.1.0	1.29.1.0	1.29.1.0
31	1.31.1.0	1.30.1.0	1.30.1.0	1.30.1.0	1.30.1.0	1.30.1.0	1.30.1.0
32	1.32.1.0	1.31.1.0	1.31.1.0	1.31.1.0	1.31.1.0	1.31.1.0	1.31.1.0

Software version	Add-on library name						
	NetworkDriver_CCIETSN	PlcInstruction	SignalIO	ExternalSignal	Program_ST	Logging	MotionEventHist
01	1.0.1.0	1.0.1.0	1.0.1.0	1.0.1.0	1.0.1.0	1.0.1.0	1.0.1.0
02	1.1.1.0	1.1.1.0	1.1.1.0	1.1.1.0	1.1.1.0	1.1.1.0	1.1.1.0
03	1.2.1.0	1.2.1.0	1.2.1.0	1.2.1.0	1.2.1.0	1.2.1.0	1.2.1.0
04	1.3.1.0	1.3.1.0	1.3.1.0	1.3.1.0	1.3.1.0	1.3.1.0	1.3.1.0
05	1.4.1.0	1.4.1.0	1.4.1.0	1.4.1.0	1.4.1.0	1.4.1.0	1.4.1.0
06	1.5.1.0	1.5.1.0	1.5.1.0	1.5.1.0	1.5.1.0	1.5.1.0	1.5.1.0
07	1.6.1.0	1.6.1.0	1.6.1.0	1.6.1.0	1.6.1.0	1.6.1.0	1.6.1.0
08	1.7.1.0	1.7.1.0	1.7.1.0	1.7.1.0	1.7.1.0	1.7.1.0	1.7.1.0
09	1.8.1.0	1.8.1.0	1.8.1.0	1.8.1.0	1.8.1.0	1.8.1.0	1.8.1.0
10	1.9.1.0	1.9.1.0	1.9.1.0	1.9.1.0	1.9.1.0	1.9.1.0	1.9.1.0
12	1.11.1.0	1.11.1.0	1.11.1.0	1.11.1.0	1.11.1.0	1.11.1.0	1.11.1.0
13	1.12.1.0	1.12.1.0	1.12.1.0	1.12.1.0	1.12.1.0	1.12.1.0	1.12.1.0
14	1.13.1.0	1.13.1.0	1.13.1.0	1.13.1.0	1.13.1.0	1.13.1.0	1.13.1.0
16	1.15.1.0	1.15.1.0	1.15.1.0	1.15.1.0	1.15.1.0	1.15.1.0	1.15.1.0
17	1.16.1.0	1.16.1.0	1.16.1.0	1.16.1.0	1.16.1.0	1.16.1.0	1.16.1.0
18	1.17.1.0	1.17.1.0	1.17.1.0	1.17.1.0	1.17.1.0	1.17.1.0	1.17.1.0
20	1.19.1.0	1.19.1.0	1.19.1.0	1.19.1.0	1.19.1.0	1.19.1.0	1.19.1.0
21	1.20.1.0	1.20.1.0	1.20.1.0	1.20.1.0	1.20.1.0	1.20.1.0	1.20.1.0
22	1.21.1.0	1.21.1.0	1.21.1.0	1.21.1.0	1.21.1.0	1.21.1.0	1.21.1.0
24	1.23.1.0	1.23.1.0	1.23.1.0	1.23.1.0	1.23.1.0	1.23.1.0	1.23.1.0
26	1.25.1.0	1.25.1.0	1.25.1.0	1.25.1.0	1.25.1.0	1.25.1.0	1.25.1.0
28	1.27.1.0	1.27.1.0	1.27.1.0	1.27.1.0	1.27.1.0	1.27.1.0	1.27.1.0
30	1.29.1.0	1.29.1.0	1.29.1.0	1.29.1.0	1.29.1.0	1.29.1.0	1.29.1.0
31	1.30.1.0	1.30.1.0	1.30.1.0	1.30.1.0	1.30.1.0	1.30.1.0	1.30.1.0
32	1.31.1.0	1.31.1.0	1.31.1.0	1.31.1.0	1.31.1.0	1.31.1.0	1.31.1.0

Software version	Add-on library name						
	FileTransfer	MotionControl_AxisFilter	ProfileControl	PackagingApp	ServoSystem Recorder	SimpleMotion	MotionControl_AdvancedSync
01	1.0.1.0	1.0.1.0	1.0.1.0	1.0.1.0	—	—	—
02	1.1.1.0	1.1.1.0	1.1.1.0	1.1.1.0	—	—	—
03	1.2.1.0	1.2.1.0	1.2.1.0	1.2.1.0	—	—	—
04	1.3.1.0	1.3.1.0	1.3.1.0	1.3.1.0	—	—	—
05	1.4.1.0	1.4.1.0	1.4.1.0	1.4.1.0	—	—	—
06	1.5.1.0	1.5.1.0	1.5.1.0	1.5.1.0	—	—	—
07	1.6.1.0	1.6.1.0	1.6.1.0	1.6.1.0	—	—	—
08	1.7.1.0	1.7.1.0	1.7.1.0	1.7.1.0	1.7.1.0	—	—
09	1.8.1.0	1.8.1.0	1.8.1.0	1.8.1.0	1.8.1.0	—	—
10	1.9.1.0	1.9.1.0	1.9.1.0	1.9.1.0	1.9.1.0	—	—
12	1.11.1.0	1.11.1.0	1.11.1.0	1.11.1.0	1.11.1.0	—	—
13	1.12.1.0	1.12.1.0	1.12.1.0	1.12.1.0	1.12.1.0	—	—
14	1.13.1.0	1.13.1.0	1.13.1.0	1.13.1.0	1.13.1.0	—	—
16	1.15.1.0	1.15.1.0	1.15.1.0	1.15.1.0	1.15.1.0	1.15.1.0	—
17	1.16.1.0	1.16.1.0	1.16.1.0	1.16.1.0	1.16.1.0	1.16.1.0	—
18	1.17.1.0	1.17.1.0	1.17.1.0	1.17.1.0	1.17.1.0	1.17.1.0	—
20	1.19.1.0	1.19.1.0	1.19.1.0	1.19.1.0	1.19.1.0	1.19.1.0	—
21	1.20.1.0	1.20.1.0	1.20.1.0	1.20.1.0	1.20.1.0	1.20.1.0	—
22	1.21.1.0	1.21.1.0	1.21.1.0	1.21.1.0	1.21.1.0	1.21.1.0	—
24	1.23.1.0	1.23.1.0	1.23.1.0	1.23.1.0	1.23.1.0	1.23.1.0	1.23.1.0
26	1.25.1.0	1.25.1.0	1.25.1.0	1.25.1.0	1.25.1.0	1.25.1.0	1.25.1.0
28	1.27.1.0	1.27.1.0	1.27.1.0	1.27.1.0	1.27.1.0	1.27.1.0	1.27.1.0
30	1.29.1.0	1.29.1.0	1.29.1.0	1.29.1.0	1.29.1.0	1.29.1.0	1.29.1.0
31	1.30.1.0	1.30.1.0	1.30.1.0	1.30.1.0	1.30.1.0	1.30.1.0	1.30.1.0
32	1.31.1.0	1.31.1.0	1.31.1.0	1.31.1.0	1.31.1.0	1.31.1.0	1.31.1.0

Add-on library details

The details of the add-on library are shown below.

—: 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	[baseSystem] [Axis]	—
Axis	Axis control	Add-on for axis control	[baseSystem] [MotionEngine]	[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.	[baseSystem] [MotionEngine] [Axis]	[SignalIO]
MotionControl_Sync	PLCopen motion control synchronous control	Enables PLCopen® motion control synchronous control FB.	[baseSystem] [MotionEngine] [Axis] [MotionControl_General]	—
AbsSystem	Absolute position control	Enables absolute position control.	[baseSystem] [Axis]	—
ServoDriver_CANopen	Servo driver (CANopen)	CANopen servo driver	[baseSystem] [Axis] [NetworkDriver_CCIETSN] [SignalIO]	—
NetworkDriver_CCIETSN	Network driver (CC-Link IE TSN)	Network driver of CC-Link IE TSN	[baseSystem]	—
PlcInstruction	External sequence dedicated instructions	Enables external sequence dedicated instructions.	[baseSystem]	[PLCopenMC_General] [ST] [SignalIO]
SignalIO	I/O signal	Provides the signal input/output function.	[baseSystem]	[Axis]
ExternalSignal	External signal	Enables external signals.	[baseSystem] [SignalIO] [Axis]	—
Program_ST	ST language execution	Add-on that executes ST language	[baseSystem]	[PLCopenMC_General]
Logging	Logging	Provides the logging function.	[baseSystem] [SignalIO]	—
MotionEventHist	Motion event history	Add-on for motion event history.	[baseSystem]	—
FileTransfer	File transfer	Provides the file transfer function.	[baseSystem]	—
MotionControl_AxisFilter	Command filter	Enables PLCopen® motion control command filter FB.	[baseSystem] [MotionEngine] [Axis] [MotionControl_General] [MotionControl_Sync]	—
ProfileControl	Operation profile control	Enables operation profile control.	[baseSystem] [MotionEngine] [Axis] [MotionControl_General] [MotionControl_Sync]	[SignalIO]
PackagingApp	Packaging	Provides functions for packaging machines.	[baseSystem] [ProfileControl]	—
ServoSystemRecorder	Servo System Recorder	Provides the servo system recorder function.	[baseSystem] [Logging] [Axis]	[ServoDriver_CANopen]
SimpleMotion	SimpleMotion	Add-on for simple motion.	[baseSystem] [Axis] [ServoDriver_CANopen] [NetworkDriver_CCIETSN] [SignalIO]	—

Add-on library name	Name	Description	Dependent add-on (Necessary)	Dependent add-on (Restricted)
MotionControl_AdvancedSync	Advanced Synchronous Control	Provides the advanced synchronous control function.	[baseSystem] [Axis] [MotionEngine] [MotionControl_General]	[AbsSystem] [SignalIO] [ProfileControl]

Point

baseSystem is necessary.

Appendix 4 List of Boot software

The versions of each boot which is compatible with the software version of the Motion module are shown below.

Software version	Boot name	
	Network boot software	Boot software
01	1	1
02	2	1
03	2	1
04	2	2
05	3	3
06	3	3
07	3	3
08	4	4
09	4	4
10	5	5
12	5	5
13	5	5
14	6	6
16	7	7
17	8	8
18	8	8
20	9	9
21	9	9
22	10	10
24	11	11
26	12	12
28	13	13
30	14	14
31	14	14
32	15	15

MEMO

A

INDEX

T

V

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, 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
August 2021	IB(NA)-0300411ENG-D	■Added or modified parts SAFETY PRECAUTIONS, RELEVANT MANUALS, TERMS, GENERIC TERMS AND ABBREVIATIONS, HOW TO READ THIS MANUAL, FUTURE SUPPORT PLANNED, Chapter 7, 20, 23, 24, 25, 27, 28, 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, 3.4, 4.1, 4.2, 4.3, 4.4, 4.5, 5.1, 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, 10.1, 10.2, 11.1, 11.2, 11.3, 12.1, 12.2, 12.3, 12.4, 13.1, 13.2, 13.3, 13.4, 13.5, 13.6, 13.7, 13.8, 13.9, 13.10, 14.1, 14.2, 15.1, 15.2, 15.3, 16.1, 16.2, 16.3, 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, 22.6, 22.7, 22.8, Appendix 1, 2, 3, WARRANTY, TRADEMARKS
January 2022	IB(NA)-0300411ENG-E	■Added or modified parts INTRODUCTION, TERMS, GENERIC TERMS AND ABBREVIATIONS, Chapter 5, 7, Section 1.1, 1.2, 1.3, 1.4, 1.5, 2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 3.1, 3.2, 3.4, 4.1, 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, 10.1, 10.2, 11.1, 11.2, 11.3, 12.1, 12.2, 12.3, 13.1, 13.7, 13.8, 13.9, 14.1, 14.2, 15.1, 15.3, 16.1, 16.2, 16.3, 18.2, 19.1, 19.2, 19.3, 19.4, 20.1, 21.1, 21.2, 21.3, 22.1, 22.2, 22.3, 22.5, 22.6, Appendix 1, 2, 3, 4
August 2022	IB(NA)-0300411ENG-F	■Added or modified parts RELEVANT MANUALS, TERMS, GENERIC TERMS AND ABBREVIATIONS, HOW TO READ THIS MANUAL, FUTURE SUPPORT PLANNED, Chapter 5, 21, 24, 26, 27, 28, Section 1.1, 1.2, 1.3, 1.4, 2.1, 2.3, 2.4, 2.5, 2.6, 3.1, 3.2, 3.4, 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, 9.3, 10.2, 11.2, 12.1, 12.2, 12.4, 13.1, 13.2, 13.7, 13.8, 13.9, 14.1, 14.2, 15.1, 15.2, 15.3, 16.3, 18.1, 18.2, 19.1, 19.2, 19.3, 21.1, 21.2, 21.3, 22.2, 22.3, 22.5, 22.6, 22.8, Appendix 1, 3, INDEX
May 2023	IB(NA)-0300411ENG-G	■Added functions Multiple axes positioning data operation, Master-slave operation, Advanced synchronous control, Co-recording function ■Added or modified parts SAFETY PRECAUTIONS, TERMS, GENERIC TERMS AND ABBREVIATIONS, Chapter 15, 18, 22, Section 1.2, 1.3, 1.4, 2.1, 2.2, 2.5, 3.1, 3.2, 3.4, 4.3, 4.4, 4.5, 5.1, 6.4, 6.5, 7.1, 8.3, 8.4, 9.1, 11.2, 12.1, 12.2, 12.4, 12.5, 13.1, 13.7, 14.1, 14.2, 15.1, 15.2, 15.3, 15.4, 15.5, 15.6, 15.7, 15.8, 16.1, 17.3, 18.1, 18.2, 19.1, 19.2, 20.1, 20.2, 22.1, 23.3, 23.5, 23.6, 23.7, 23.8, Appendix 1, 2, 3, 4
August 2023	IB(NA)-0300411ENG-H	■Added or modified parts TERMS, Section 1.1, 1.2, 1.3, 1.4, 2.5, 2.6, 3.1, 4.4, 4.5, 7.2, 8.4, 12.1, 12.3, 12.5, 13.7, 14.1, 17.1, 17.3, 20.1, 20.2, 22.1, 23.5, 23.6, 23.8, Appendix 1, 2, 3, 4
January 2024	IB(NA)-0300411ENG-J	■Added or modified parts SAFETY PRECAUTIONS, 1.1, 1.2, 1.3, 1.4, 2.1, 2.4, 2.5, 2.6, 3.1, 4.3, 4.5, 6.4, 6.5, 7.1, 7.2, 8.3, 9.1, 12.1, 12.2, 12.3, 13.7, 14.1, 14.2, 15.7, 20.4, 22.1, 22.2, 22.3, 23.1, 23.3, 23.5, 23.6, 23.7, 23.8, Appendix 1, 2, 3, 4

Japanese manual number: IB-0300410-N

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 is repaired or replaced.

[Term]

For terms of warranty, please contact your original place of purchase.

[Limitations]

(1) You are requested to conduct an initial failure diagnosis by yourself, as a general rule.
It can also be carried out by us or our service company upon your request and the actual cost will be charged. However, it will not be charged if we are responsible for the cause of the failure.

(2) This limited warranty applies only when the condition, method, environment, etc. of use are in compliance with the terms and conditions and instructions that are set forth in the instruction manual and user manual for the Product and the caution label affixed to the Product.

(3) Even during the term of warranty, the repair cost will be charged on you in the following cases;

1. a failure caused by your improper storing or handling, carelessness or negligence, etc., and a failure caused by your hardware or software problem
2. a failure caused by any alteration, etc. to the Product made on your side without our approval
3. a failure which may be regarded as avoidable, if your equipment in which the Product is incorporated is equipped with a safety device required by applicable laws and has any function or structure considered to be indispensable according to a common sense in the industry
4. a failure which may be regarded as avoidable if consumable parts designated in the instruction manual, etc. are duly maintained and replaced
5. any replacement of consumable parts (battery, fan, smoothing capacitor, etc.)
6. a failure caused by external factors such as inevitable accidents, including without limitation fire and abnormal fluctuation of voltage, and acts of God, including without limitation earthquake, lightning and natural disasters
7. a failure generated by an unforeseeable cause with a scientific technology that was not available at the time of the shipment of the Product from our company
8. any other failures which we are not responsible for or which you acknowledge we are not responsible for

2. Term of warranty after the stop of production

(1) We may accept the repair at charge for another seven (7) years after the production of the product is discontinued. The announcement of the stop of production for each model can be seen in our Sales and Service, etc.

(2) Please note that the Product (including its spare parts) cannot be ordered after its stop of production.

3. Service in overseas countries

Our regional FA Center in overseas countries will accept the repair work of the Product. However, the terms and conditions of the repair work may differ depending on each FA Center. Please ask your local FA center for details.

4. Exclusion of loss in opportunity and secondary loss from warranty liability

Regardless of the gratis warranty term, Mitsubishi shall not be liable for compensation to:

(1) Damages caused by any cause found not to be the responsibility of Mitsubishi.

(2) Loss in opportunity, lost profits incurred to the user by Failures of Mitsubishi products.

(3) Special damages and secondary damages whether foreseeable or not, compensation for accidents, and compensation for damages to products other than Mitsubishi products.

(4) Replacement by the user, maintenance of on-site equipment, start-up test run and other tasks.

5. Change of Product specifications

Specifications listed in our catalogs, manuals or technical documents may be changed without notice.

6. Application and use of the Product

(1) For the use of our Motion module, its applications should be those that may not result in a serious damage even if any failure or malfunction occurs in the Motion module, and a backup or fail-safe function should operate on an external system to the Motion module when any failure or malfunction occurs.

(2) Our Motion module is designed and manufactured as a general purpose product for use at general industries.
Therefore, applications substantially influential on the public interest for such as atomic power plants and other power plants of electric power companies, and also which require a special quality assurance system, including applications for railway companies and government or public offices are not recommended, and we assume no responsibility for any failure caused by these applications when used
In addition, applications which may be substantially influential to human lives or properties for such as airlines, medical treatments, railway service, incineration and fuel systems, man-operated material handling equipment, entertainment machines, safety machines, etc. are not recommended, and we assume no responsibility for any failure caused by these applications when used.
We will review the acceptability of the abovementioned applications, if you agree not to require a specific quality for a specific application. Please contact us for consultation.

(3) Mitsubishi shall have no responsibility or liability for any problems involving programmable controller trouble and system trouble caused by DoS attacks, unauthorized access, computer viruses, and other cyberattacks.

INFORMATION AND SERVICES

For further information and services, please contact your local Mitsubishi Electric sales office or representative.
Visit our website to find our locations worldwide.

MITSUBISHI ELECTRIC Factory Automation Global Website
Locations Worldwide
www.MitsubishiElectric.com/fa/about-us/overseas/

TRADEMARKS

Microsoft and Windows are trademarks of the Microsoft group of companies.
Unicode is either a registered trademark or a trademark of Unicode, Inc. in the United States and other countries.
PLCopen is a registered trademark of PLCopen®.
The company names, system names and product names mentioned in this manual are either registered trademarks or trademarks of their respective companies.
In some cases, trademark symbols such as '™' or '®' are not specified in this manual.

IB(NA)-0300411ENG-J

IB(NA)-0300411ENG-J(2401)MEE
MODEL: RD78-U-APP-E
MODEL CODE: 1XB039

MITSUBISHI ELECTRIC CORPORATION

HEAD OFFICE: TOKYO BLDG., 2-7-3, MARUNOUCHI, CHIYODA-KU, TOKYO 100-8310, JAPAN
NAGOYA WORKS: 1-14, YADA-MINAMI 5-CHOME, HIGASHI-KU, NAGOYA 461-8670, JAPAN