MITSUBISHI ELECTRIC
Changes for the Better

FATEC

Motion Controller School Textbook
(Advanced Synchronous Control Edition)
Windows PC Compatible MT Works2

● Safety Precautions ●

(Always read before performing practical work.)

When designing systems, always read related manuals and give sufficient consideration to safety.
Pay due attention to the following points when performing practical work, and ensure correct handling of the product.

[Practical work precautions]

DANGER

- Do not touch terminals while the power is ON. Failure to observe this may result in electric shock.
- When removing the safety cover, either turn OFF the power, or ensure that sufficient attention is paid to safety.

CAUTION

- Carry out practical work in accordance with the instructions of your teacher.
- Do not remove the demonstration machine, or make changes to the wiring.
  Failure to observe this may result in a fault, malfunction, injury, or fire.
- Turn OFF the power before attaching or removing the module.
  Removing or attaching the module with the power ON may result in a module fault or electric shock.
- If the demonstration machine emits an abnormal odor or noise, press the [Power] button or [EMERGENCY STOP] button to stop the module.
- If an error occurs, notify your teacher immediately.

Revision History

* The text number is indicated in the lower left of the rear cover of this manual.

Print date	* Text No.	Revision details
Sep. 2015	SH-030148ENG-A	First print

This manual does not guarantee the enforcement of industrial property or other rights, and does not grant licensing rights. Furthermore, Mitsubishi accepts no responsibility for industrial property related problems arising through use of the content described in this manual.

© 2015 MITSUBISHI ELECTRIC CORPORATION

Contents

Chapter 5 Motion SFC Programs 5 – 1 to 5 - 23

Chapter 6 SV22 Servo Programs 6 - 1 to 6 - 27

Chapter 7 Operation Control Programs 7 – 1 to 7 - 4

Introduction

This document is a schooling text created for the purpose of helping users understand the motion controller developed to easily control multi-axis positioning.
This manual provides an overview of the Q motion controller, and describes how to specify data settings to perform positioning, and create servo programs, mechanical support languages, and sequence programs using a Windows computer and programming tool (MT Works2).
(Usable software packages and function specifications will differ depending on the model.)

The following related manuals are available.

	Model	Model code
(1) User's manual		
• Q172D(S)CPU/Q173D(S)CPU Describes the motion controller hardware (exterior, wiring, etc.).	IB(NA)-0300133	1XB927
(2) Programming manuals		
• Q172D(S)CPU・Q173D(S)CPU Common Edition	IB(NA)-0300134	1XB928
• SV13/22 (Q172D(S) / Q173D(S) Real Mode Edition)	IB(NA)-0300136	1XB930
• (Q172D(S)/Q173D(S) Advanced Synchronous Control Edition) for SV22 automatic machine	IB(NA)-0300198	1XB953
• Motion SFC Edition (Q172D(S)/ Q173D(S)) Describes parameters for positioning control, dedicated positioning devices, positioning methods, and motion SFC, etc.	IB(NA)-0300135	1XB929
(3) Software manual		
• MELSOFT MT Works2 Installation Instructions	BCN-B62008-364	―――
(4) Sequence programming manuals		
• QCPU (Q mode) Common Command Edition	SH(NA)-080809ENG	13JW10
• QnUCPU User's Manual Function Description, Program Basics Edition	SH(NA)-080807ENG	13JZ27
• QnUCPU User's Manual Hardware Design, Maintenance & Inspection Edition	SH(NA)-080483ENG	13JR73
• QnUCPU User's Manual Multi-CPU System Edition Describes devices and all commands required to create sequence programs.	SH(NA)-080485ENG	13JR75
(5) GX Works2 related manuals		
• GX Works2 Version1 Operating Manual (Common Edition)	SH(NA)-080779ENG	13JU63
• GX Works2 Version1 Operating Manual (Simple Project Edition)	SH(NA)-080780ENG	13JU64
• GX Works2 Version1 Operating Manual (Intelligent Function Unit Operation Edition)	SH(NA)-080921ENG	13JU69
(6) Technical document collections		
• MR-J4-□B Servo Amp Technical Document Collection Describes SSCNET III (/H) servo amp handling and error displays, etc.	SH(NA)-030106	1CW805
• MELSERVO-J4 Servo Amp Technical Document Collection (Troubleshooting Edition)	SH(NA)-030109	1CW808

SSCNET is an abbreviation of Servo System Controller Network.

Chapter 1 Overview

1.1 Motion Controller Features

The motion controller has the following features.

(1) Q PLC CPU and multiple CPU System
Processing loads can be balanced to realize a flexible system construction by using the Q motion CPU module for complex servo control, and the Q PLC CPU module for all other machine and information control.

(2) Full range of controllers for all applications
The following motion controller models are available to suit the scale of the systems required to perform multi-axis positioning.

• Q172DSCPU	(Multi-axis positioning function for 1 to 16 axes)	SSCNET III/H
• Q173DSCPU	(Multi-axis positioning function for 1 to 32 axes)	
• Q172DCPU	(Multi-axis positioning function for 1 to 8 axes)	SSCNET III
• Q173DCPU	(Multi-axis positioning function for 1 to 32 axes)	
• Q172HCPU	(Multi-axis positioning function for 1 to 8 axes)	
• Q173HCPU	(Multi-axis positioning function for 1 to 32 axes)	
• Q172CPU	(Multi-axis positioning function for 1 to 8 axes)	SSCNET
• Q173CPU	(Multi-axis positioning function for 1 to 32 axes)	

(3) Control is possible with an MR-J4-□B servo amplifier.
Servo motors can be controlled by externally connecting an MR-J4-□B servo amplifier with motion network SSCNET III/H.
(Using the Q172DSCPU or Q173DSCPU, up to 16 or 32 servo motors can be controlled, respectively.)

(4) High-speed serial communication with servo amplifiers is possible.
Servo data can be collected, changes can be made to servo parameters, servo tests can be carried out, servos can be monitored, and mechanical system programs can be monitored through motion network SSCNET III/H high-speed serial communication. Furthermore, SSCNET III/H communication offers a maximum communication speed of 150 Mbps, accelerated command communication synchronization of 0.22 ms, and high-speed, high-accuracy positioning.

(5) An absolute position system is possible.
An absolute position system is possible using servo motors equipped with absolute position detector. (Zeroing is unnecessary even in the event of a power outage.)

(6) A Windows computer is used as the programming tool for positioning.
Motion SFC programming, servo control programming, monitoring, and testing can be performed using a Windows computer and dedicated software package.
Windows computer peripheral software package: MT Works2

(7) Changes can be made to the operating system (OS).

A comprehensive range of software packages is available to suit all applications, and the applicable OS can be written directly to the CPU built-in Flash memory to realize a motion controller suitable for any machine.
Furthermore, functional upgrades to software packages are also possible.

1) SV13 for conveyance and assembly
   SV13 can perform tasks such as 1 to 4-axis linear interpolation with dedicated servo commands, 2-axis circular interpolation, 3-axis helical interpolation, CP control (constant speed control), speed control, and fixed-pitch feeding, making it ideal for equipment such as conveyors and assembly machines.
2) SV22 for automatic machines
   Multiple servo motors can be controlled simultaneously with a mechanical support language, and cam control is possible using software, making SV22 ideal for automatic machines and so on.

Motion controllers come preinstalled with SV22 when shipped.
Furthermore, the latest versions of the OS software for all motion controllers can be downloaded from the Mitsubishi Electric FA site and then installed.

(8) Mechanical support language (mechanical system program): valid only for SV22

In the past, synchronous motion and cooperative motion were required for industrial equipment and automatic machines, and these motions were combined as an implementation tool.
This method used transfer mechanisms such as main shafts, which were the driving forces, and gears, clutches, and cranks to drive output mechanisms such as rotational motions, linear motions, reciprocating motions, and feed motions,. This method was excellent in terms of synchronous and cooperative motions, but was lacking in flexibility.

Separating the mechanical support language from the previous mechanical combination, and using software to process machine mechanism motions has led to improvements in the functionality and performance of the positioning control used to control servo motors, and because this is an electrical method, there are few mechanical limitations, facilitating a logical design.

Transfer mechanisms from main shafts to gears, clutches, transmissions, and differential gears, and output mechanisms such as roller output, ballscrew output, rotary table output, and cam output are shown in diagrams on peripheral equipment screens, and simply by setting the respective module parameters, synchronous and cooperative motions can be realized, facilitating the easy construction of flexible control systems.
Consequently, mechanical parts such as main shafts, gears, clutches, cranks, transmissions, differential gears, and cams can be significantly eliminated or omitted, meaning lower costs and less wear.

(9) Software cam: valid only for SV22

By replacing the cam mechanism for which synchronous control was being performed mechanically with software, and then setting synchronous control parameters, the following features can be obtained by synchronizing control with input axes.

1) Cam curved line data can be created easily with cam curved line creation software, eliminating the need to manufacture cam parts.
2) Cams can be replaced easily by changing the cam No. from the motion SFC program or sequence program.
3) There is no need to consider the wear or short life characteristic of cams.

(10) Teaching function

Gauging servo programs can be created with the current value teaching function.

(11) Limit switch function

This function outputs ON/OFF signals corresponding to the data range for watch data set for each output device (X, Y, M, L, B).
Output devices for up to 32 signals can be set.

(12) Peripheral I/F (Ethernet)

With the peripheral I/F built-in motion CPU, connections can be made to a wide range of devices such as GOT and COGNEX vision systems via Ethernet.

(13) Support for 4 million pulse synchronous encoder as standard

The "Q171ENC-W8" 4 million pulse synchronous encoder is supported as standard, meaning significant improvements in synchronized operation accuracy (16 times higher than previous system). High-accuracy control can be achieved in combination with an MR-J4-B servo amplifier (standard motor resolution of 4 million (22-bit) pulses).

1.2 Control Overview

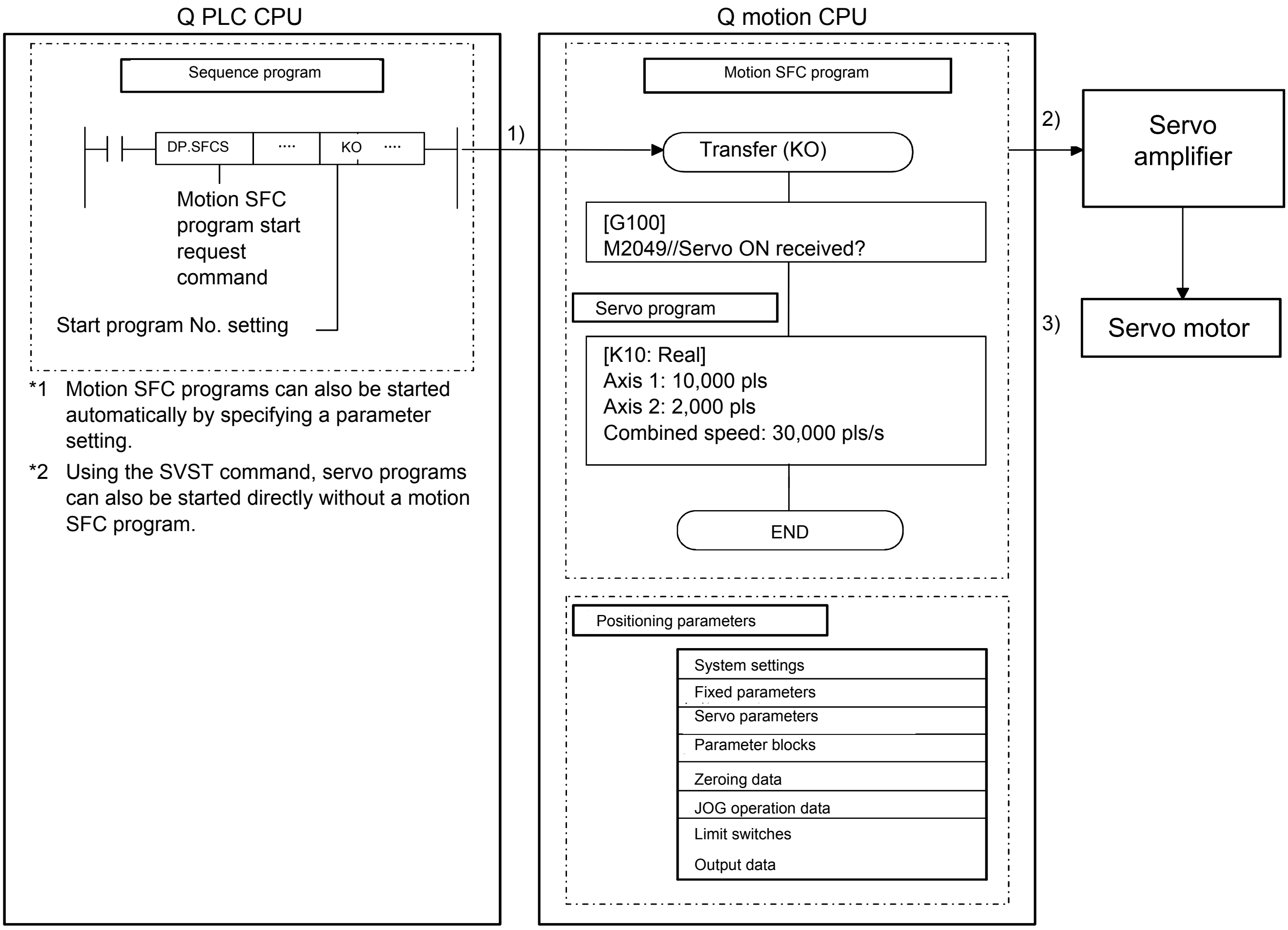
1.2.1 Real mode control for SV13 conveyance and assembly/SV22 automatic machines

(a) Systems using servo motors are controlled directly with a servo program.

(b) Positioning parameters must be set, and servo programs and motion SFC programs must be created.

(c) The procedure when performing positioning control is as follows.

1) Issue a motion SFC program start request with a sequence program SFCS command.
↓
2) Perform positioning control with the specified motion SFC program.
↓
3) Servo motors are controlled.



1.2.2 Advanced synchronous control for SV22 automatic machines

(a) Performs the same control by replacing the mechanism used to perform mechanical synchronous control using devices such as gears, shafts, transmissions, and cams with software.

(b) Synchronous control parameters are required in addition to the positioning parameters, servo programs, and motion SFC programs used in real mode.

(c) The procedure for positioning control with advanced synchronous control is as follows.

1) Issue an advanced synchronous control motion SFC program start request with a sequence program SFCS command.

↓

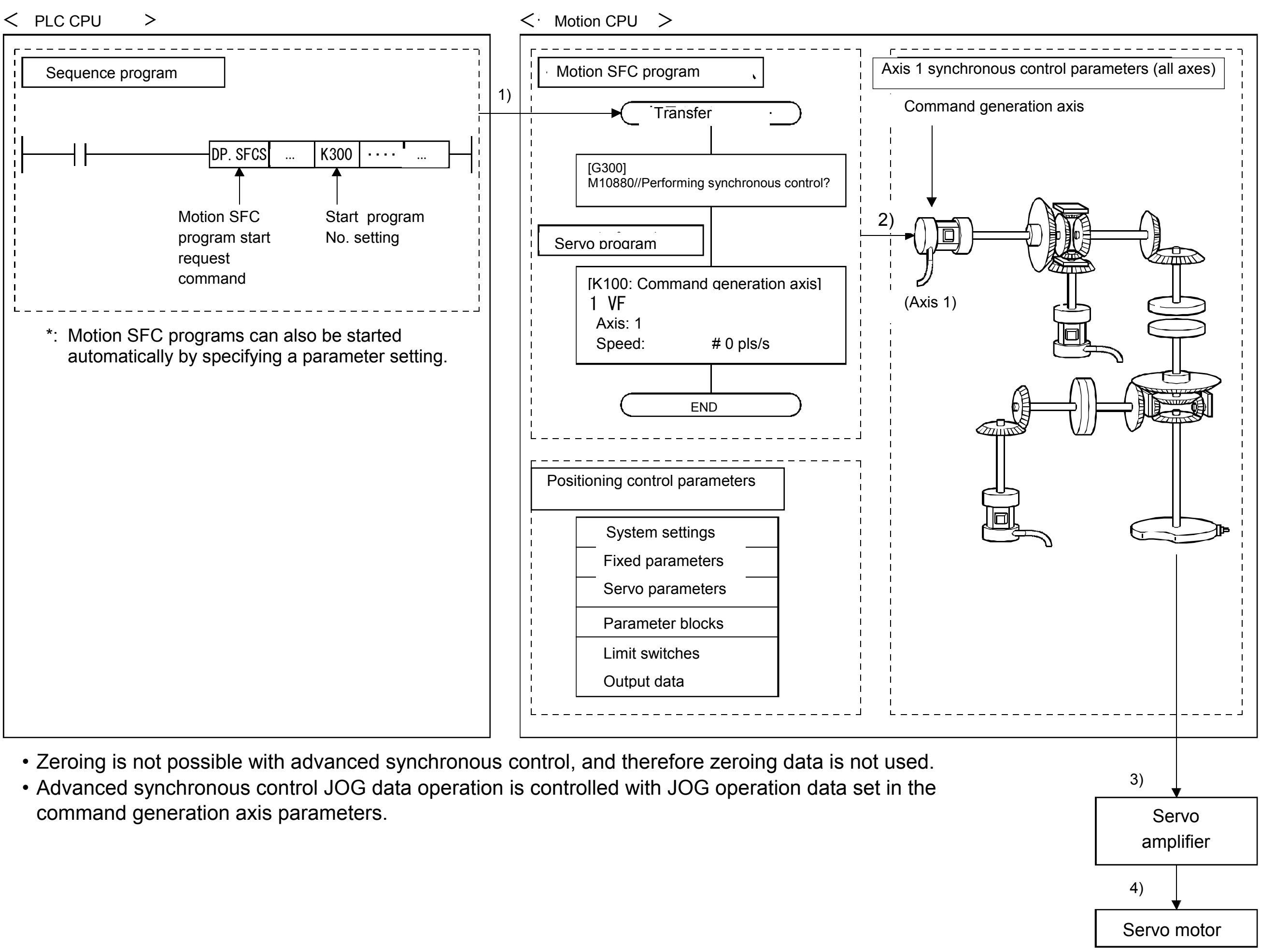
2) The advanced synchronous control command generation axis starts up.

↓

3) Output synchronous control parameters to the servo amplifier for each axis.

↓

4) Servo motors are controlled.



- Zeroing is not possible with advanced synchronous control, and therefore zeroing data is not used.
- Advanced synchronous control JOG data operation is controlled with JOG operation data set in the command generation axis parameters.

1.3 System Startup Requirements

The steps inside the boxes with unbroken lines must be carried out.

The steps inside the boxes with broken lines should be carried out as required.

Refer to Chapter 8 for details on system startup.

1	Motion controller device selection system assembly, wiring	Select devices such as the Q PLC base, power supply modules, Q motion CPU, Q PLC CPU, motion module, servo amplifiers, servo motors, and cables, and assemble and wire the system.
2	To Windows computer Software package registration	Register the software package (MT Works2, MR Configurator2, GX Works2).
3	Q PLC CPU multiple CPU settings	Create with GX Works2.
4	Sequence program creation	Create with GX Works2.
5	Data writing to the Q PLC CPU	Write the sequence program and computer parameters at the computer.
6	Cam creation	Create cams when SV22 is used, and using cams for the output module.
7	SV13, SV22 startup (new project creation)	Start the software package used, and then create a new project.
8	System settings creation	Create system basic settings, multiple CPU settings, the Q PLC base, motion module, servo amplifiers, servo motors, axis numbers and so on as the motion controller system.
9	Servo data creation • Fixed parameters • Servo parameters • Zeroing data • JOG operation data • Parameter blocks	• Set unit settings, travel value per pulse, stroke limit values, etc. • Set the rotation direction, auto tuning, etc. • Set the zeroing direction, method, address, speed, etc. • Set the JOG speed limit value, parameter block numbers, etc. • Set the speed limit values, acceleration/deceleration time, torque limit values, etc. (Set servo parameters at MR Configurator2 started from MT Works2.)
10	Servo data creation • Limit switch data	Set only when using the limit switch output function.
11	Motion SFC program creation	
12	Mechanical system program creation Synchronous control parameter setting	Create and set when using SV22.
13	Cable connection to Q motion CPU	Use Ethernet to connect to the Windows computer, and use Ethernet, RS-232C, or USB to connect to the Q PLC CPU.
14	Registering the OS in the Q motion CPU	Register the OS using the installation procedure at the servo menu screen. (Performed only once when constructing the system. SV22 comes preinstalled.)
15	Data writing to the Q motion CPU	Write the motion SFC program, servo data, servo program, mechanical system program, synchronous control parameters, and cam data.
16	Resetting the Q PLC CPU	Press the Q PLC CPU [RESET] button.
17	Running the Q PLC CPU, Q motion CPU	Press the Q PLC CPU, Q motion CPU [RUN] button.

Chapter 2 Function Description

This section describes the system functions.

2.1 Specifications List

2.1.1 Motion control specifications list (SV13/SV22)

Comparison item \ Model		Q173DSCPU	Q172DSCPU
External dimensions [mm]		120.5(H) × 27.4(W) × 120.3(D)	
Number of control axes		Max. 32 axes (Max. 16 axes per system × 2)	Max. 16 axes
No of equipped motion related modules		Up to 4 × Q172DLX modules can be used.	Up to 2 × Q172DLX modules can be used.
		Up to 6 × Q172DEX modules can be used.	
		Up to 4 × Q173DPX modules can be used. *1	
Operation cycle (default)	SV13	0. 22 ms/1 to 4 axes 0. 44 ms/5 to 10 axes 0. 88 ms/11 to 24 axes 1. 77 ms/25 to 32 axes	0. 22 ms/1 to 4 axes 0. 44 ms/5 to 10 axes 0. 88 ms/11 to 16 axes
	SV22	0. 44 ms/1 to 6 axes 0. 88 ms/7 to 16 axes 1. 77 ms/17 to 32 axes	0. 44 ms/1 to 6 axes 0. 88 ms/7 to 16 axes
Interpolation function		Linear interpolation (max. 4 axes), circular interpolation (2 axes), helical interpolation (3 axes)	
Control mode		PTP (Point To Point) control, speed control, speed position control, fixed feeding, constant speed control, fixed-pitch feed, fixed position stop speed control, speed change control, high-speed oscillation control, speed/torque control, synchronous control (SV22)	
Acceleration/deceleration processing		Trapezoidal acceleration/deceleration, S-curve acceleration/deceleration, advanced S-curve acceleration/deceleration	
Compensation function		Backlash compensation, electronic gear, phase compensation (SV22)	
Program language		Motion SFC, dedicated commands, mechanical support language (SV22)	
Servo program capacity		16 k steps	
Number of positioning points		3,200 points (indirect designation possible)	
Peripheral I/F		USB/RS-232/Ethernet (via PLC CPU), peripheral I/F (motion CPU control)	
Zeroing function		Proximity dog method (2 types), count method (3 types), data set method (2 types) dog cradle method, stopper stopping method (2 types), combined use with limit switch, scale home position signal detection method (Equipped with zeroing retry function, home position shift function)	
JOG operation function		Yes	
Manual pulse generator operation function		3 modules can be connected (when using Q173DPX) 1 module can be connected (when using motion CPU built-in interface)	
Synchronous encoder operation function		12 modules can be connected (when using SV22) (Q172DEX + Q173DPX + motion CPU built-in interface)	
M-code function		Equipped with M-code output function Equipped with await M-code completion function	
Limit switch output function	SV13	32 output points Watch data: motion control data/word device	
	SV22	64 output points x 2 settings Output timing compensation Watch data: motion control data/word device	
ROM operation function		Yes	
Absolute position system		Compatible by inserting battery in servo amplifier. (Absolute system/incremental system can be specified for each axis.)	
Number of SSCNETIII(/H) systems *2		2 systems *3	1 system *3

*1: This is the number of modules if using an INC synchronous encoder (when using SV22). Only one module can be used if connecting a manual pulse generator.
*2: SSCNET compatible servo amplifiers cannot be used.
*3: SSCNET III and SSCNET III/H cannot be used together within the same system. If using Q173DSCPU, SSCNET III and SSCNET III/H can be set for each system.

2.1.2 Motion SFC performance specifications list (SV13/SV22)

Item				Q173DSCPU/Q172DSCPU
Program capacity	Code total (SFC diagram + operation control + transition)			652 kb
	Text total (Operation control + transition)			668 kb
Motion SFC program	Number of motion SFC programs			256 (No. 0 to 255)
	SFC diagram size/program			Max. 64 kb (inc. SFC diagram comment)
	Number of SFC steps/program			Max. 4,094 steps
	No. of selection branches/branch			255
	No. of parallel branches/branch			255
	Parallel branch nest			Max. 4 types
Operation control Program (F/FS) / transition program (G)	Number of operation control programs			F (one-time execution type)/FS (scan execution type) 4,096 in total (F/FS0 to F/FS4095)
	Number of transition programs			4096 (G0 to G4095)
	Code size/program			Max. approx 64 kb (32,766 steps)
	Number of blocks (lines)/program			Max. 8,192 blocks (if 4 steps (min.)/block)
	Number of characters/block (line)			Max. 128 single-byte characters (inc. comment)
	Number of operands/block			Max. 64 (operand: constant, word device, bit device)
	( ) nests/block			Max. 32 types
	Running form	Operation control program		Calculation method, bit conditional expression, branch/iteration
		Transition program		Calculation method, bit conditional expression, comparison conditional expression
Execution specifications	Number of simultaneous execution programs			Max. 256
	Number of simultaneous active steps			Max. 256 steps/all programs
	Execution tasks	Normal tasks		Execution during motion main cycle
		Event tasks (Mask possible)	Fixed cycle	Execution every fixed cycle (0.22 ms, 0.44 ms, 0.88 ms, 1.77 ms, 3.55 ms, 7.11 ms, 14.2 ms)
			External interrupts	Execution when turning set inputs ON out of 16 interrupt module QI60 inputs
			PLC interrupts	Execution with interrupt command (D(P).GINT) from PLC
		NMI tasks		Execution when turning set inputs ON out of 16 interrupt module QI60 inputs
Number of inputs/outputs (X/Y)				8,192
Number of actual inputs/outputs (PX/PY)				256 (Motion CPU built-in interface (4 inputs) + I/O module)
Devices (Motion CPU built-in portion only) (inc. dedicated positioning devices)	Number of internal relays (M)			12288
	Number of link relays (B)			8192
	Number of annunciators (F)			2048
	Number of special relays (SM)			2256
	Number of data registers (D)			8192
	Number of link registers (W)			8192
	Number of special registers (SD)			2256
	Number of motion registers (#)			12288
	Number of coasting timers (FT)			1 (888 μs)
	Multiple CPU area devices			Max. 14336 types *

*: The number of devices that can be used differs depending on the system settings.

2.1.3 System configuration device list (SV13/SV22)

(1) Motion controller OS software

Application	Model	
	Q173DSCPU *1	Q172DSCPU *1
For conveyance and assembly (SV13)	SW8DNC-SV13QJ	SW8DNC-SV13QL
For automatic machines (SV22)	SW8DNC-SV22QJ	SW8DNC-SV22QL

*1: The motion controller OS software (SV22 (advanced synchronous control method)) is already installed when the product is shipped.)
The latest OS software can be downloaded from the Mitsubishi Electric FA site.

(2) Peripheral software package

Software name	Model
MELSOFT MT Works2 (MT Developer2 *1)	SW1DNC-MTW2-J

*1: This programming software is included in motion controller engineering environment "MELSOFT MT Works2".

(3) Related software packages

(a) PLC software packages

Software name	Software package name
GX Works2	SW1DNC-GXW2-J

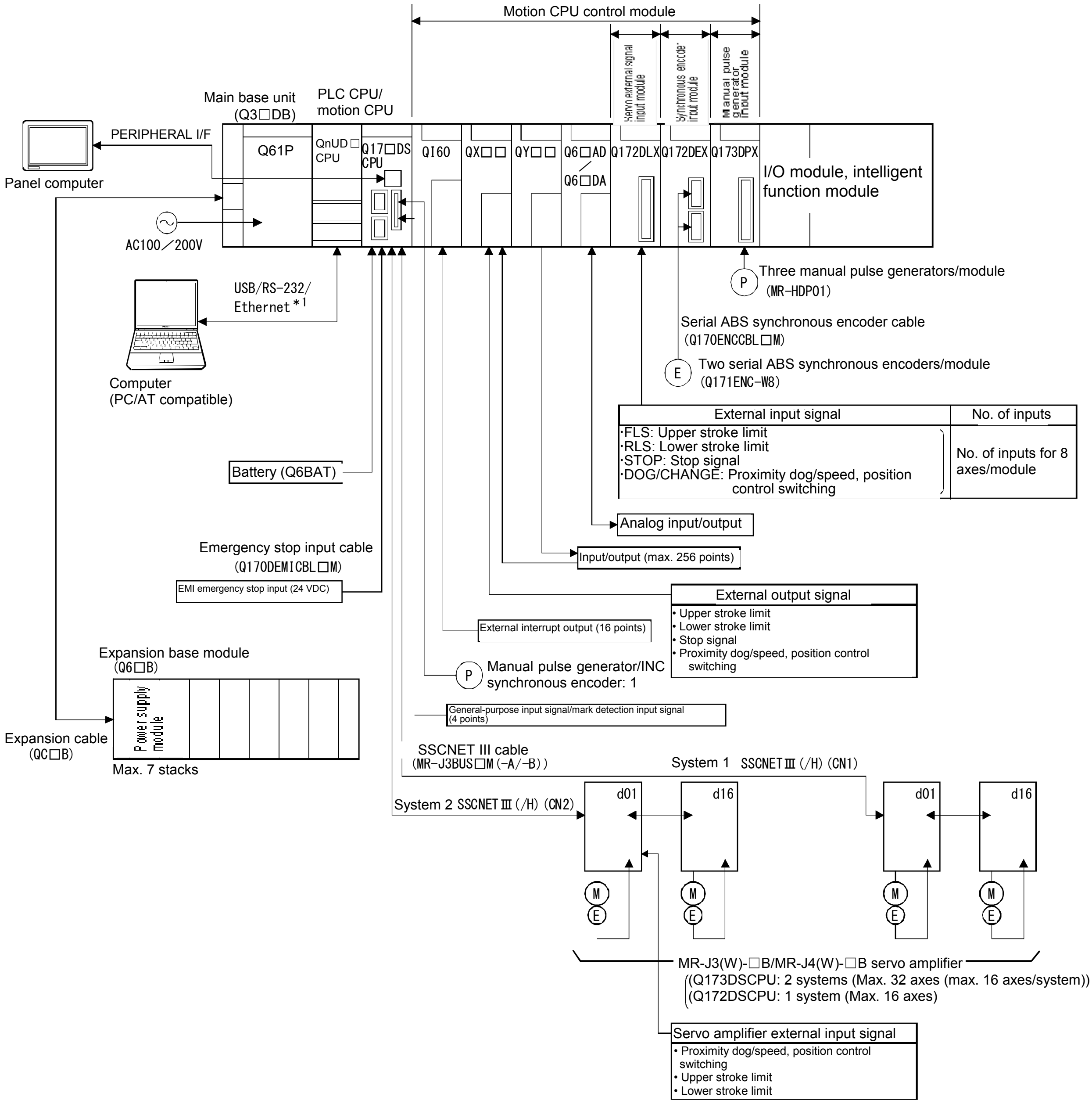
(b) Servo setup software package

Software name	Software package name
MR Configurator2	SW1DNC-MRC2-J

2.2 System Configuration Diagrams

Refer to the User's Manual for details on wiring.

2.2.1 Q173DSCPU/Q172DSCPU system



*1: Ethernet: QnUDE(H)CPU only
RS-232: QnUD(H)CPU only

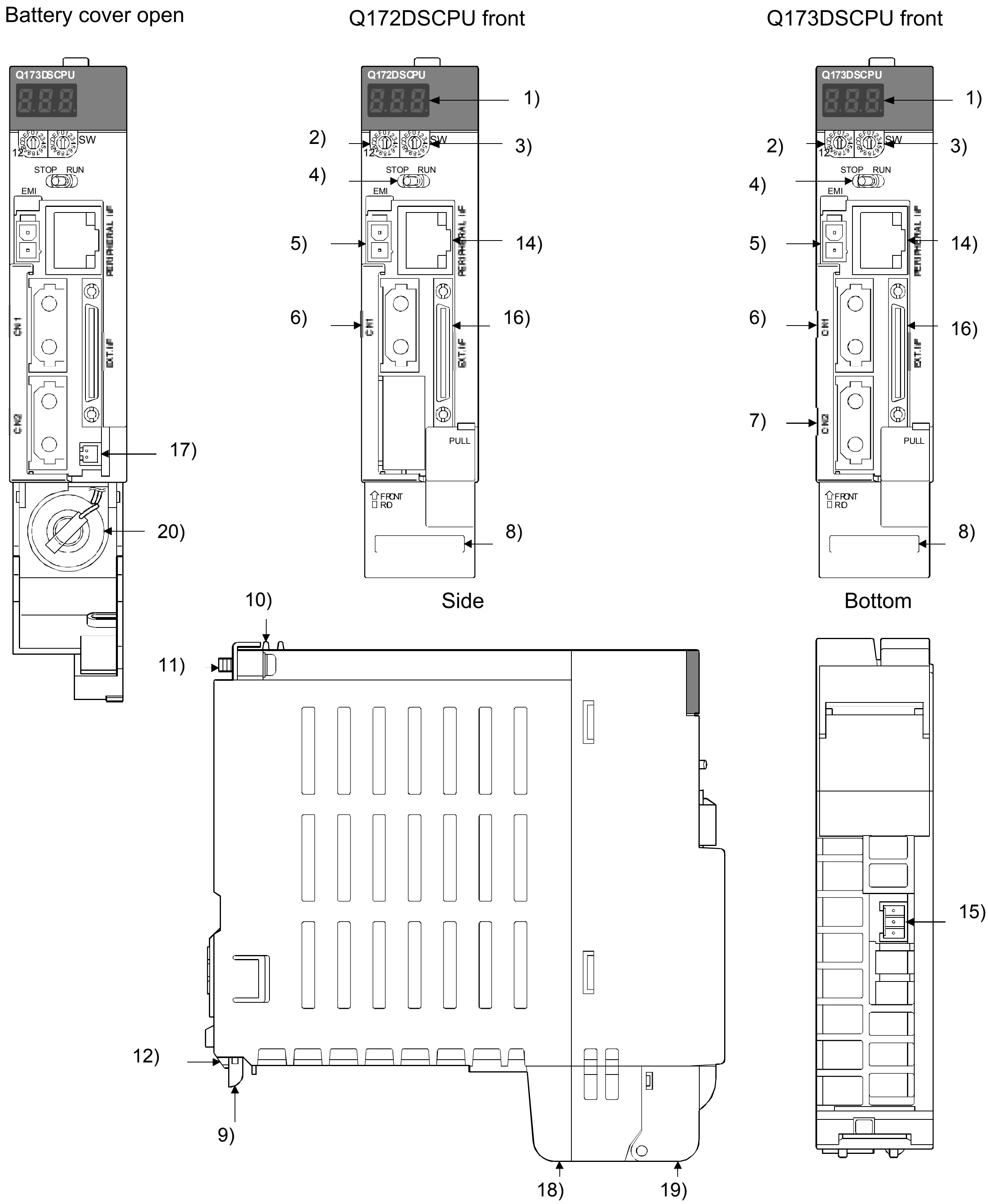
⚠ CAUTION

- If the operation performed when an error occurs and the system safe direction operation differs for the controller and servo amplifier, construct a countermeasure circuit outside the servo amplifier.
- Use parts used in the system (other than controller, servo amplifiers, servo motors) with rating and characteristics suited to the controller, servo amplifiers, and servo motors.
- Set parameter values applicable to the controller, servo amplifier, servo motor, regenerative resistor models, and system application. Safeguards may fail to function if settings are specified incorrectly.

2.3 Name of Each Part

This section describes the names and settings of all Q172DSCPU/ Q173DSCPU parts.

(1) Names of Q172DSCPU/Q173DSCPU parts



<table>
<tr><th>No.</th><th>Item</th><th>Function</th></tr>
<tr><td>1)</td><td>7-segment LED</td><td>Displays the operating status and error information.</td></tr>
<tr><td>2)</td><td>For function selection 1
Rotary switch (SW1)</td><td rowspan="2">• Sets the operation mode (normal operation mode, installation mode, ROM operation mode, etc.)
• Switch settings are specified with 0 to F. (Default: SW1 "0", SW2 "0")</td></tr>
<tr><td>3)</td><td>For function selection 2
Rotary switch (SW2)</td></tr>
<tr><td>4)</td><td>RUN/STOP switch</td><td>• Used for RUN/STOP. (Default: STOP)
RUN : Runs the motion SFC program (SV13/SV22).
STOP : Stops the motion SFC program (SV13/SV22).</td></tr>
<tr><td>5)</td><td>Emergency stop input connector (EMI) [*1]</td><td>• Performs an emergency stop for all servo amplifier axes together.
EMI ON (open) : Emergency stop
EMI OFF (24 VDC input) : Cancels emergency stop</td></tr>
<tr><td>6)</td><td>SSCNET III CN1 connector [*2]</td><td>Connector used to connect with the first system servo amplifier (for 16 axes).</td></tr>
<tr><td>7)</td><td>SSCNET III CN1 connector [*2,3]</td><td>Connector used to connect with the second system servo amplifier (for 16 axes).</td></tr>
<tr><td>8)</td><td>Serial No. indication</td><td>Indicates the serial No. on the rating plate.</td></tr>
<tr><td>9)</td><td>Module attachment lever</td><td>Used to attach modules to the base module.</td></tr>
<tr><td>10)</td><td>Module securing hook [*4]</td><td>Hook used to secure the module to the base module. (Helps when performing module attachment.)</td></tr>
<tr><td>11)</td><td>Module securing screw</td><td>Screw used to secure to the base module. (M3 × 13)</td></tr>
<tr><td>12)</td><td>Module securing protrusion</td><td>Protrusion used for securing to the base module.</td></tr>
<tr><td>13)</td><td>Battery connector (BAT) [*5]</td><td>Connector used to connect to battery holder module Q170DBATC.</td></tr>
<tr><td>14)</td><td>Peripheral I/F connector</td><td>For communication interface with peripheral devices
• Bottom LED
Flashing : Accessing peripheral devices
ON : Not accessing peripheral devices
• Top LED
Data transfer speed
ON : 100 Mbps
OFF : 10 Mbps
<table>
<tr><th colspan="2">Item</th><th>Specification</th></tr>
<tr><td rowspan="4">Transfer</td><td>Data transfer speed</td><td>100 Mbps/
10 Mbps</td></tr>
<tr><td>Communication mode</td><td>Full duplex/
half duplex</td></tr>
<tr><td>Transfer method</td><td>Base band</td></tr>
<tr><td>Cable length</td><td>Max. 30 m</td></tr>
</table></td></tr>
<tr><td>15)</td><td>RIO connector</td><td>Connector used to connect to safety signal module (Q173DSXY).</td></tr>
<tr><td>16)</td><td>Built-in interface connector</td><td>Connector used for manual pulse generator/INC synchronous encoder connection, and to input general-purpose input signals/mark detection input signals. (Voltage output/open collector type, differential output type)</td></tr>
<tr><td>17)</td><td>Battery connector</td><td>Connector used to connect to the battery (Q6BAT).</td></tr>
<tr><td>18)</td><td>Battery holder</td><td>Holder used to hold the battery (Q6BAT).</td></tr>
<tr><td>19)</td><td>Battery cover</td><td>Cover for battery (Q6BAT) protection</td></tr>
<tr><td>20)</td><td>Battery [*5]</td><td>Battery (Q6BAT) for program, parameter, motion device (#), latch range device, and absolute position data protection.</td></tr>
</table>

*1: Always use an external forced stop input cable (sold separately). If not used, it will not be possible to clear emergency stop conditions.
If preparing your own external forced stop input cable, ensure a cable length of 30 [m] or less.

*2: In order that the weight of the SSCNET III cable is not applied to the SSCNET III connector, store the cable in a duct, or secure the part near the motion CPU with a cable tie.

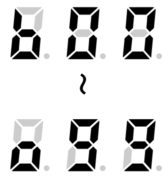
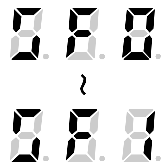
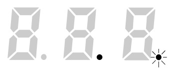
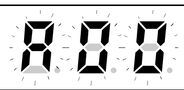
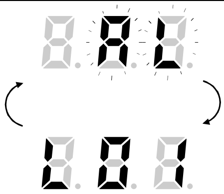
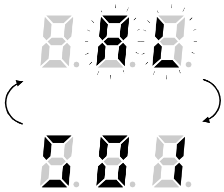
*3: Q173DSCPU only

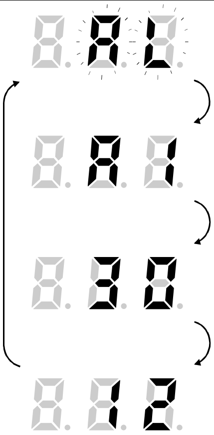
*4: This helps when attaching modules to the main base module. Always secure modules to the main base module with the screws provided.

*5: Always use a battery.
If the battery is not inserted properly, programs stored in the motion CPU built-in SRAM, parameters, motion device (#), latch range device, and absolute position data will not be retained.

(2) **7-segment LED display**

The mode display turns ON or flashes based on the combination with each error.

Item		7-segment LED		Remarks
When starting			Initialization item display	Initialization (until RUN/STOP is displayed) takes approximately 10 seconds. If stopped with the initialization display, turn the system power from OFF to ON. If the same condition occurs again, a motion CPU module hardware error is likely. Contact your nearest system service center, dealer, or branch, and describe the abnormal startup condition (LED indicator).
			Initialization item display (When using safety monitoring function)	With the power ON, initialize the safety monitoring function and perform self-diagnosis. This takes approximately 15 seconds.
When normal			"☀" flashes.	This flashing symbol indicates normal CPU operation.
Installation mode			"INS" lights up "☀" flashes.	This mode is used to install the motion controller OS software via the computer.
Operation mode	RAM operation mode		"☀" flashes.	This mode is used to perform operation with the user program and parameters stored in the motion CPU built-in RAM.
	ROM operation mode		"-" lights up "☀" flashes.	This mode is used to run the motion controller after booting the user program and parameters stored in the motion CPU built-in FLASH ROM to the motion CPU built-in SRAM.
STOP			"STP" lights up.	A "STOP" condition occurs when the PLC ready flag (M2000) turns OFF. Stops the motion SFC program (SV13/SV22).
RUN			"RUN" lights up	A "RUN" condition occurs when the PLC ready flag (M2000) turns ON. Runs the motion SFC program (SV13/SV22).
Battery error	Initial (2.7 V or less)		"BT1" lights up.	Displays when the battery voltage is 2.7 V or less.
	End of life (2.5 V or less)		"BT2" lights up.	Displays when the battery voltage is 2.5 V or less.
Motion controller OS software not installed			"A00" flashes.	The mode changes to installation mode when the motion controller OS software has not been installed.
System setting error			"AL" flashes 3 times. ↓ "L01" lights up.	Motion CPU system setting error Refer to the "Q173D(S)CPU/Q172D(S)CPU Motion Control Programming Manual (common edition)" for details.
Servo error			"AL" flashes 3 times. ↓ "S01" lights up.	Motion CPU servo error Refer to the programming manual for the OS software used.
WDT error			"..." lights up.	Hardware error or software error. Refer to the programming manual for the OS software used.

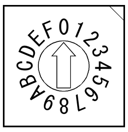
Item	7-segment LED	Remarks	Item
Self-diagnosis error (Multiple CPU related error)		"AL" flashes 3 times. ↓ "A1" lights up. (Self-diagnosis error) ↓ The 4-digit error code is split up and displayed twice. (The example on the left is for error code [3012].)	Multiple CPU system setting error Refer to the "Q173D(S)CPU/Q172D(S)CPU Motion Controller Programming Manual (common edition)" for details.

POINT

1) If an error is indicated at the 7-segment LED, check the error code and so on at MT Works2.
2) For error details, refer to the MT Works2 motion error monitor, or the error list in each programming manual.

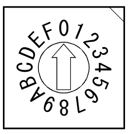
(3) Rotary switch allocation

(a) Function selection 1 rotary switch (SW1)

Rotary switch	Setting *	Mode	Details
	0	Normal mode	Normal operation mode
	A	Installation mode	Used to install the motion controller OS software from MT Works2.

*: Settings other than the above are prohibited.

(b) Function selection 2 rotary switch (SW2)

Rotary switch	Setting *	Mode	Details
	0	RAM operation mode	Normal operation mode (Functions with motion CPU built-in SRAM settings data and parameters.)
	6	ROM operation mode	Functions with settings data written to the motion CPU built-in FLASH ROM and parameters.
	8	Ethernet IP address Display mode	This mode displays the Ethernet IP address.
	C	SRAM clear	SRAM 0 clear

*: Settings other than the above are prohibited.

CAUTION

- If changing the rotary switch setting, always turn the multiple CPU System power OFF beforehand.

Chapter 3 Q PLC Multiple CPU

I/O unit and special function unit sequence control, and calculation with application commands and dedicated commands is performed with sequence programs.
Furthermore, they are also used to execute SFCS (motion SFC start request) commands used to start motion SFC programs, GINT commands used to perform interrupts for motion CPUs, DDRD and DDWR commands used to perform direct device reading and writing for Q motion CPUs, SVST commands used to issue servo program startup request, CHGA current value change commands, CHGV speed change commands, and CHGT torque limit value change commands.

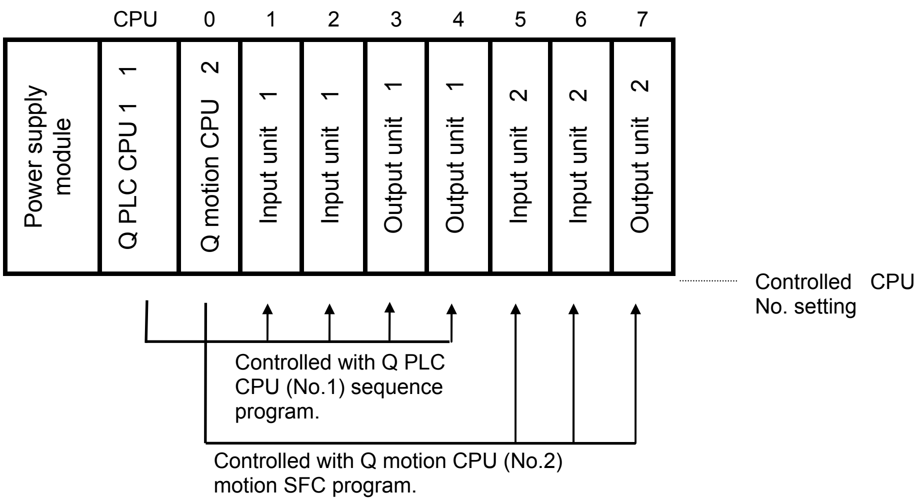
This is described as Q172DSCPU specifications in this chapter.
(Refer to Appendix 8 for details on GINT, DDRD, DDWR, CHGA, CHGV, and CHGT commands.)

3.1 Multiple CPU System

The multiple CPU system incorporates multiple (max. 4) Q PLC CPU/Q motion CPUs on a main base unit, and is used to control I/O units and intelligent function units with each Q PLC CPU/Q motion CPU.
Processing loads can be balanced by using the Q motion CPU unit for complex servo control, and the Q PLC CPU unit for all other machine and information control.

3.1.1 Multiple CPU system settings

With the multiple CPU system, it is necessary to set (control CPU settings) which I/O modules and intelligent function modules are to be controlled with which Q PLC CPU/Q motion CPU, and the number of installed Q PLC CPU/Q motion CPU units for all Q PLC CPU/Q motion CPUs.
(The multiple CPU setting method is described in section 8.3.2.)



Initially, the Q motion CPU compares the parameters in the following table against the No.1 Q PLC CPU. An error occurs if there is a mismatch, and therefore the following parameters must be made to match.

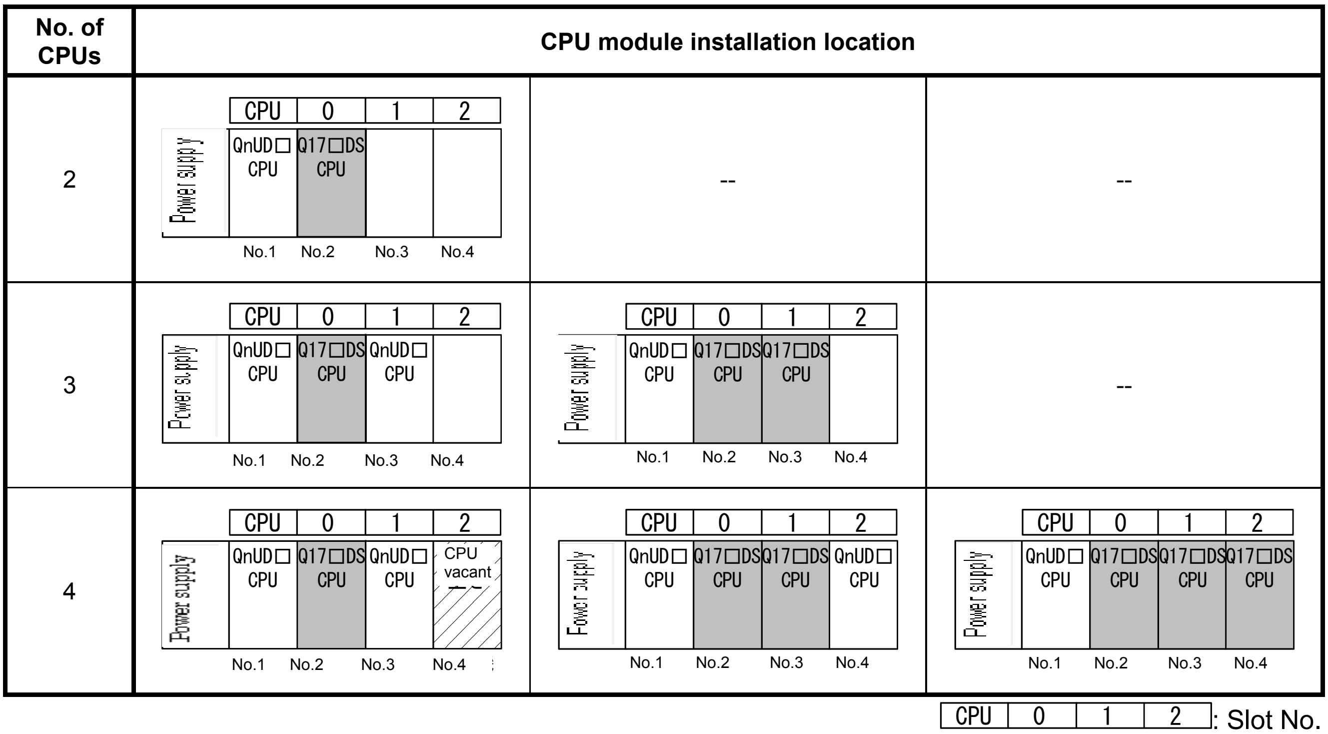
<table>
<tr><th rowspan="2">No.</th><th rowspan="2" colspan="2">Comparison item</th><th colspan="4">Parameter</th><th rowspan="2">Remarks</th></tr>
<tr><th colspan="2">Name at Q motion CPU</th><th colspan="2">Name at Q PLC CPU</th></tr>
<tr><td>1</td><td colspan="2">Unit control CPU No.</td><td colspan="2">Motion slot setting</td><td rowspan="4">I/O assignment settings</td><td>Control CPU</td><td>Compares only the unit No. set at Q motion CPU.</td></tr>
<tr><td>2</td><td colspan="2">Total base qty</td><td colspan="2" rowspan="3">Base settings</td><td rowspan="3">Basic settings</td><td rowspan="3">No comparison made if no settings specified at Q PLC CPU.</td></tr>
<tr><td rowspan="2">3</td><td rowspan="2">Base</td><td>Base No.</td></tr>
<tr><td>No. of base slots</td></tr>
<tr><td>4</td><td colspan="2">No. of CPU module</td><td rowspan="3">Multiple CPU settings</td><td>No. of multiple CPUs</td><td rowspan="3">Multiple CPU settings</td><td>No. of CPUs</td><td></td></tr>
<tr><td>5</td><td colspan="2">Operation mode when CPU stop error occurs</td><td>Operation mode</td><td>Operation mode</td><td></td></tr>
<tr><td>6</td><td colspan="2">No. of automatic refreshes</td><td>Automatic refresh settings</td><td>Refresh settings</td><td></td></tr>
</table>

3.1.2 Q PLC CPU, Q motion CPU installation locations

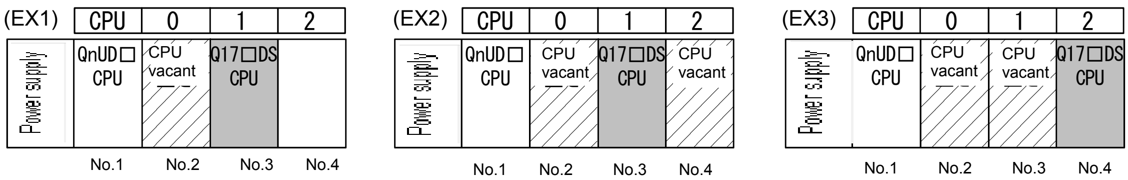
Up to four PLC CPU modules or motion CPU modules can be installed from the main base unit CPU slot (slot to right of power supply module) to slot 2.
Motion CPU modules cannot be installed in CPU slots.
With multiple CPU combinations, CPU No.1 must be a PLC CPU module. There are no restrictions in the installation order for CPU module No.2 to No.4.

***: If using in combination with high-performance model CPU modules, process CPU modules, computer CPU modules, or C language controller modules, refer to the manual for each CPU module.**

CPU module installation example



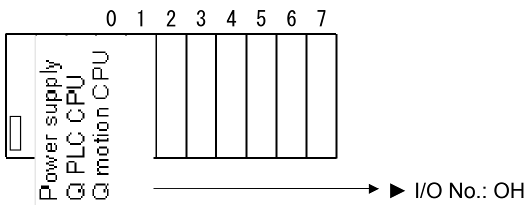
A vacant slot can be added for additional CPU modules in the future. Set the number of CPUs, including the vacant slot, in the multiple CPU settings, and set the type for the slot to be left vacant to "CPU (Vacant)" in the CPU settings.



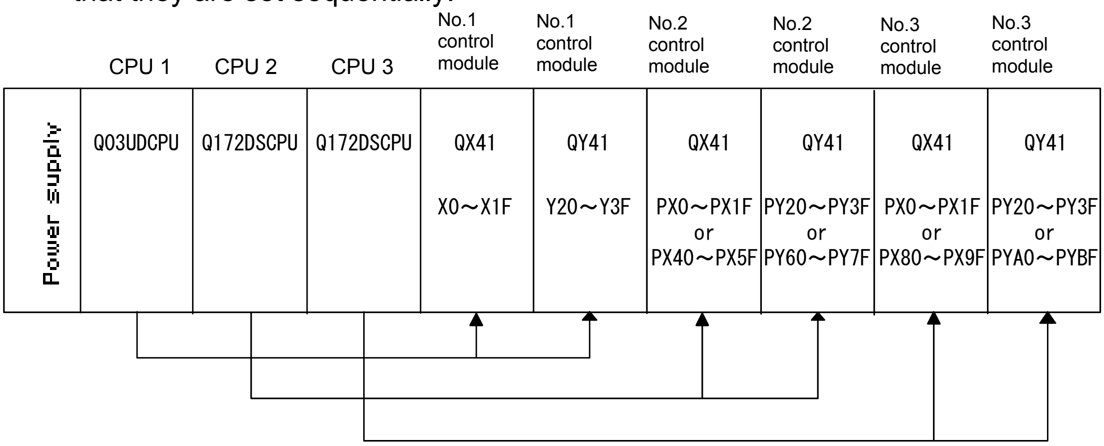
3.1.3 I/O numbers

With the multiple CPU system, the number of slots set in the computer parameter multiple CPU settings is occupied by Q PLC CPU/Q motion CPUs. The I/O numbers for I/O modules and intelligent function modules installed to the right of the slots occupied by Q PLC CPU/Q motion CPUs begin with "OH", and are numbered sequentially from left to right.

Q PLC CPU: If the number of CPUs is set to 2



Q motion CPU I/O numbers are unrelated to Q PLC CPU I/O numbers. The Q motion CPU I/O numbers are those set in the Q motion CPU system settings. (I/O numbers for modules controlled by Q motion CPUs are indicated by PX/PY.)
Allocating Q motion CPU control module I/O numbers to Q PLC CPUs is meaningless.
It is generally recommended that I/O numbers be common to all CPUS, and that they are set sequentially.

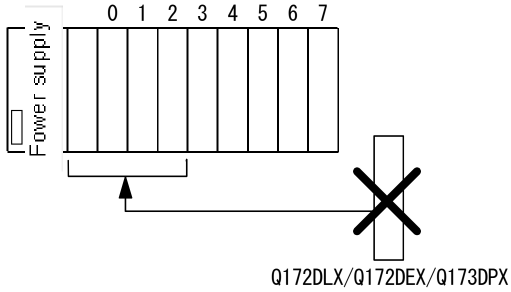


If setting Q motion CPU control modules when allocating Q PLC CPU I/O numbers, refer to the following table and set. (With the Q172DLX, Q172DEX, and Q173DPX, intelligent function modules occupy 32 points on Q PLC CPUs.)

Module	Type	No. of points	Remarks
Input module	Input	Set based on module.	• Set the control CPU No. applicable to the Q motion CPU. (Required) • Type and No. of points settings may be omitted.
Output module	Output	Set based on module.	
Mix of input/output modules	Mix of inputs/ outputs	Set based on module.	
Analog input module	Analog input	16	
Analog output module	Analog output		
Interrupt module (QI60)	Interrupt		
Q172DLX (servo external signal input)	Intelligent	32	
Q172DEX (synchronous encoder input)	Intelligent	32	
Q173DPX (manual pulse generator input)	Intelligent	32	

POINT

With the Q172DLX, Q172DEX, and Q173DPX, Q motion CPU modules cannot be installed in main base unit CPU slots or in I/O slots 0 to 2. If mistakenly installed, the main base unit may be damaged.



Please note that with Q172DLX/DPX, modules can be installed in expansion base units, however, this is not possible with the Q172DEX.

3.1.4 CPU shared memory

CPU shared memory is memory used to transfer date between CPUs in the multiple CPU system, and has 24,335 words from 0H to 5F0FH.

CPU shared memory has a "self CPU operation information area", "system area", "user setting area", and "multiple CPU high speed transmission area".

The CPU shared memory configuration, and whether or not data exchange from self CPUs using CPU shared memory with a program is performed is shown in the following table.

		CPU shared memory		Communication with self CPU		Communication with other CPU	
				Write	Read	Write	Read
(0H) to (1FFH)	0 to 511	Self CPU operation information area		×	○ *2	×	○ *2
(200H) to (7FFH)	512 to 2047	System area		×	×	×	○ *2
(800H) to (FFFH)	2048 to 4095	User setting area		○ *1	○ *2	×	○ *2
(1000H) to (270FH)	4096 to 9999	Use not possible		×	×	×	×
(2710H) to (5F0FH)	10000 to Max. 24335	Multiple CPU high speed transmission area (Size variable from 0 to 14 k [points]: 1 k word units)	High-speed bus between multiple CPUs	○ *3	○ *3	×	○ *3

Remarks

*1: With motion CPUs, use an MULTW command to write to the self CPU user setting area.
With PLC CPUs, use an S.TO command to write to the self CPU user setting area.

*2: With motion CPUs, use an MULTR command to read self CPU and other CPU shared memory.
To read motion CPU shared memory from a PLC CPU, use a FROM command/multiple CPU area device (U□\G□).

*3: Refer to section 3.1.5 for details on how to access the multiple CPU high speed transmission area.

(1) Self CPU operation information area (0H to 1FFH)

(a) The following self CPU information is stored as multiple CPU information.

Shared memory address	Name	Content	Content details *	Corresponding special register
0H(0)	Information presence	Information presence flag	Area used to confirm whether there is information stored in the self CPU operation information area (1H to 1FH). • 0: No information is stored in the self CPU operation information area. • 1: Information is stored in the self CPU operation information area.	-
1H(1)	Diagnostic error	Diagnostic error No.	The error No. when an error occurs during diagnosis is stored in BIN.	SD0
2H(2)	Diagnostic error date/time	Diagnostic error date/time	The year and month in which the error No. was stored in CPU shared memory address 1H are stored with a 2-digit BCD code. B15 to B8 B7 to B0 Year (0 - 99) Month (1 -12) (Example) Jan. 2006 H0601	SD1
3H(3)			The day and hour at which the error No. was stored in CPU shared memory address 1H are stored with a 2-digit BCD code. B15 to B8 B7 to B0 Day (1 - 31) Hour (0 -23) (Example) 25th at 10 am H2510	SD2
4H(4)			The minute and second at which the error No. was stored in CPU shared memory address 1H are stored with a 2-digit BCD code. B15 to B8 B7 to B0 Minute (0 - 59) Second (0 -59) (Example) 35 m, 48 s H3548	SD3
5H(5)	Error information category code	Error information category code	A category code used to judge what the error common information and error individual information contains.	SD4
6H(6) ~ 10H(16)	Error common information	Error common information	Common information corresponding to the error No. when an error occurs during diagnosis is stored.	SD5 ~ SD15
11H(17) ~ 1BH(27)	Error individual information	Error individual information	Individual information corresponding to the error No. when an error occurs during diagnosis is stored.	SD16 ~ SD26
1CH(28)	Vacant	-	Use not possible	-
1DH(29)	Status of switch	Operating status of CPU	The CPU module switch status is stored. B15 B12 B11 B8 B7 B4 B3 B0 Not used (1) (1): Operating status of CPU: 0: RUN, 1: STOP	SD200
1EH(30)	Vacant	-	Use not possible	-
1FH(31)	Operating status of CPU	Operating status of CPU	The CPU module operating status is stored.	SD203

*: Refer to the corresponding special register for details.

(b) The self CPU operation information area is updated during the main cycle when the corresponding register changes.

(c) Other PLC CPUs are able to read the self CPU operation information area data with an FROM command.
However, the data update process will be delayed, and therefore read data should be used for monitoring purposes.

(2) System area (200H to 7FFH)

This is an area used by the PLC CPU/motion CPU system (OS). The OS uses this area when executing dedicated communication commands between multiple CPUs.

- System area (204H to 20DH) used with dedicated motion sequence commands

  The completion status of each flag is stored in the following addresses.

<table>
<tr><th>Shared memory address</th><th>Name</th><th>Content details</th></tr>
<tr><td>204H(516)</td><td>Axis start accept flag (axes 1 to 16)</td><td rowspan="2">There are start accept flags for 32 axes, and they are stored corresponding to each bit.
(Bits are actually set in J1 to J32 for the Q173DSCPU, and J1 to J16 for the Q172DSCPU.)
OFF: Start accept possible
ON: Start accept not possible
b15
Address 204H(516) | J16 | •••••••• | J2 | J1 |
Address 205H(517) | J32 | •••••••• | J17 |</td></tr>
<tr><td>205H(517)</td><td>Axis start accept flag (axes 17 to 32)</td></tr>
</table>

(3) User setting area

This area is used to exchange data between each CPU unit in the multiple CPU system using the motion CPU MULTR and MULTW commands.

(With PLC CPUs, data is exchanged between CPUs using FROM and S.TO commands, and multiple CPU area devices.)

Refer to the programming manual for the OS software used for details on MULTR and MULTW commands.

3.1.5 Multiple CPU high speed transmission

(1) Multiple CPU high speed transmission

Multiple CPU high speed transmission is a function used to transfer data between multiple CPUs in fixed cycles (multiple CPU high speed transmission cycle: 0.88 [ms]).

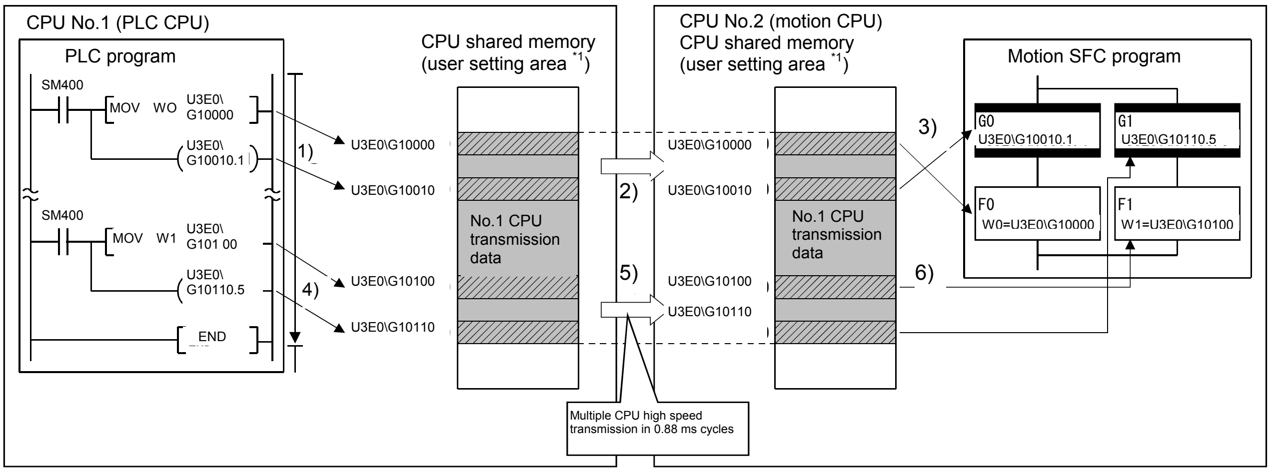
With data transfer between multiple CPUs through multiple CPU high speed transmission, processing is performed in parallel with sequence program, and motion SFC program/motion program execution, facilitating stable data transmission without being affected by the PLC CPU scan time or motion CPU main cycle.

The multiple CPU high speed transmission cycle is synchronized with the motion CPU operation cycle, and high-speed responses can be delivered between multiple CPUs.

The following methods can be used to transfer data between multiple CPUs using multiple CPU high speed transmission.

- Using a multiple CPU area device
  Specify a multiple CPU high speed transmission area using a direct multiple CPU area device (U□\G□) in the program.
- Using automatic refresh
  All CPU internal devices are refreshed automatically via the multiple CPU high speed transmission area.

(a) Example using a multiple CPU area device



1), 4): CPU No.1 writes to the user setting area *1 with a command using a multiple CPU area device.
3), 6): CPU No.2 reads from the user setting area *1 with a command using a multiple CPU area device.
2), 5): The content of the user setting area *1 is transferred to other CPUs in 0.88 ms cycles with multiple CPU high speed transmission.
*1: Area configured inside multiple CPU high speed transmission area
(Refer to "(3) Multiple CPU high speed transmission area memory configuration".)

1) Access to multiple CPU high speed transmission area
   a) Multiple CPU area device description method

Word device: U□ \ G□

- G□ → CPU shared memory address (decimal notation) (1000 to max. 24335)
- U□ → CPU module first I/O No.

CPU No.	CPU No.1	CPU No.2	CPU No.3	CPU No.4
First I/O No.	3E0(H)	3E1(H)	3E2(H)	3E3(H)

Bit device: U□ \ G□. □

- . □ → Bit designation (0 to F: hexadecimal notation)
- G□ → CPU shared memory address (decimal notation) (1000 to max. 24335)
- U□ → CPU module first I/O No.

CPU No.	CPU No.1	CPU No.2	CPU No.3	CPU No.4
First I/O No.	3E0(H)	3E1(H)	3E2(H)	3E3(H)

(Example)

- CPU No.2 multiple CPU high speed transmission memory address: 10002
  U3E1\G10002
- CPU No.3 multiple CPU high speed transmission memory address: 10200 bit14
  U3E2\G10200.E

b) Example of access with program

<Motion SFC program> *SV13/SV22

- Program substituting K12345678 for self CPU (No.2) multiple CPU high speed transmission memory 10200, 10201.
  U3E1\G10200L = K12345678
- Turns ON self CPU (No.3) multiple CPU high speed transmission memory 10301 bit12.
  Program
  SET U3E2\G10301.C

<Servo program> *SV13/SV22

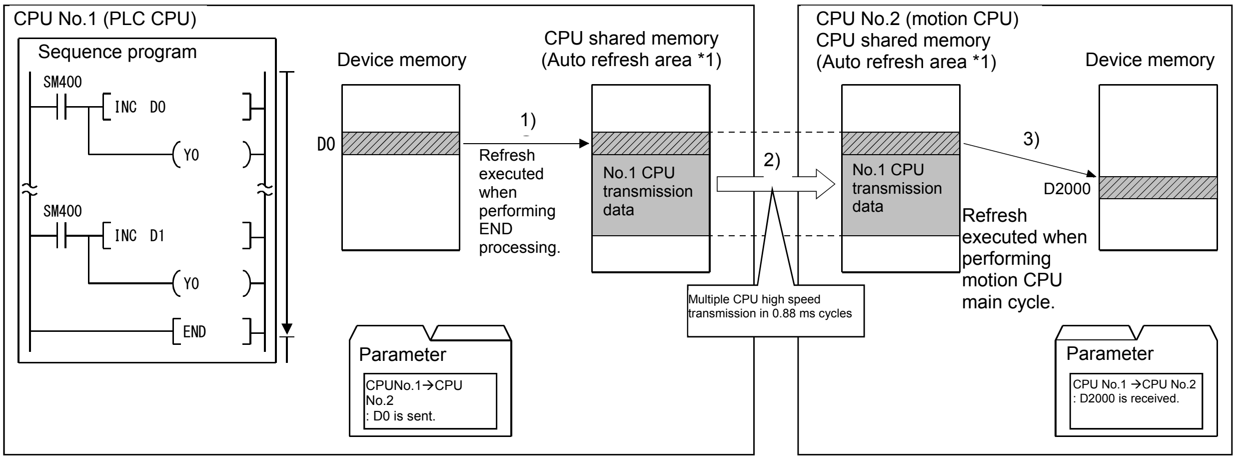
- Program used to position axis 1 at the position set in CPU No.1 multiple CPU high speed transmission memory 10400 and 10401, at speed set in CPU No.1 multiple CPU high speed transmission memory 10402 and 10403, and use the CPU No.1 multiple CPU high speed transmission memory 10404 bit1 as a cancel signal.

```
ABS-1
  Axis      1,      U3E0\G10400
  Speed             U3E0\G10402
  Cancel            U3E0\G10404.1
```

POINT

Only the CPU shared memory "multiple CPU high speed transmission area" can be accessed with this method. It cannot be used to access CPU shared memories 0 to 4095.

(b) Example using automatic refresh



1) By specifying in the parameters, the content of D0 is sent to the auto refresh area *1 when performing END processing.
2) The content of the auto refresh area *1 is transferred to other CPUs in 0.88 ms cycles by multiple CPU high speed transmission.
3) By specifying in the parameters, the content of the auto refresh area *1 is read and then transferred to D2000 when performing the motion CPU main cycle.

*1: Area configured inside multiple CPU high speed transmission area (Refer to "(3) Multiple CPU high speed transmission area memory configuration".)

(2) System configuration

Multiple CPU high speed transmission can only be used between multiple CPU high speed transmission compatible CPU modules installed on the multiple CPU high speed main base (Q3☐DB).

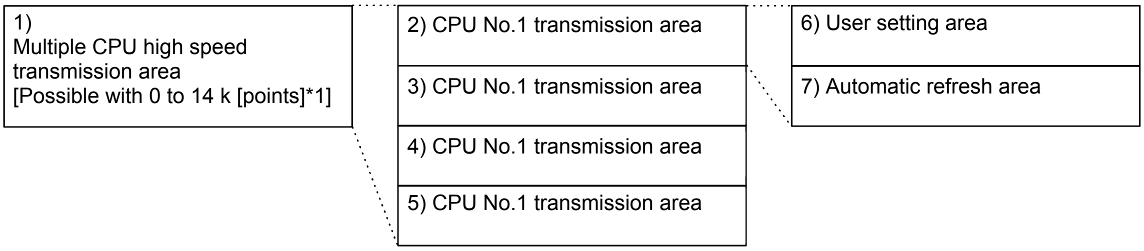
The system configuration specifications are shown in the following table.

Applicable module	Restriction details
Base module	Uses multiple CPU high speed main base (Q3☐DB).
CPU module	QnUD(E)(H) CPU is used for CPU No.1.
	Q173DSCPU/Q172DSCPU and QnUD(E)(H) CPUs are used for CPU Nos. 2 to 4.

If the multiple CPU system power is turned ON when the above specifications are not satisfied, a "MULTI EXE.ERROR (error code: 7011)" error occurs.

(3) Multiple CPU high speed transmission area memory configuration

The multiple CPU high speed transmission area memory configuration is shown below.



***1: The 14 k [points] in the multiple CPU high speed transmission area is the maximum value for two CPU modules.**
This value will be 13 k [points] for three CPU modules, and 12 k [points] for four CPU modules.

No.	Name	Description	Size	
			Setting range	Setting unit
1)	Multiple CPU high speed transmission area	• This area is used for data transfer between CPU modules in the multiple CPU system. • An area of up to 14 k [points] is distributed among each CPU module in the multiple CPU system.	0 to 14 k	1 k
2)	CPU No.n transmission area (n = 1 to 4)	• Area in which transmission data for each CPU module is stored. • Data stored in the self CPU transmission area is sent to other CPUs. • Data received from other CPU modules is stored in the other CPU transmission area.	0 to 14 k	1 k
3)				
4)				
5)				
6)	User setting area	• This areas is used to transfer data between other CPUs with a multiple CPU area device. • Accesses the transmission area with a user program using a Multiple CPU area device.	0 to 14 k	2
7)	Automatic refresh area	• This areas is used to transfer device data between other CPUs through exchange with automatic refresh. • Access with a user program is not possible.	0 to 14 k	2

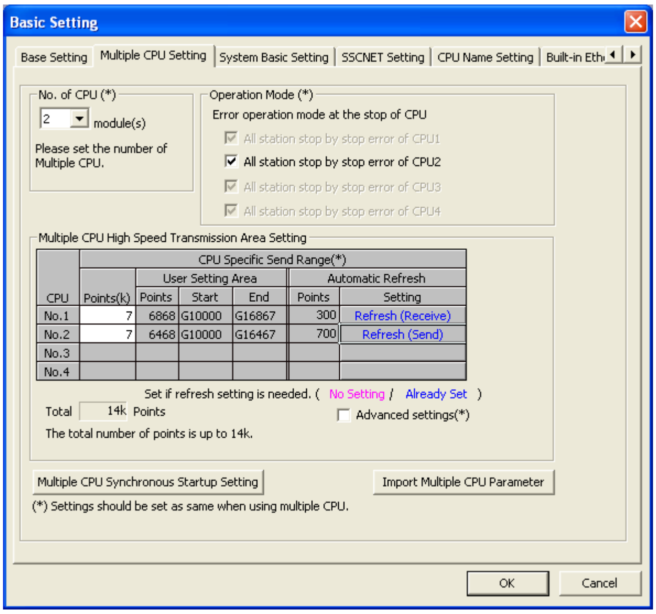
(4) Parameter settings

The parameter settings required to use multiple CPU high speed transmission are shown in the following table.

Parameter name	Details	Applicable CPU
Multiple CPU high speed transmission area settings	Sets the size of the multiple CPU high speed transmission area assigned to each CPU module in the multiple CPU system.	Required for all CPU modules
Automatic refresh settings	Sets the range for data transmission with the automatic refresh function from the user area inside the multiple CPU high speed transmission area.	

(a) Multiple CPU high speed transmission area settings

The Multiple CPU High Speed Transmission Area Setting screen and setting range are shown below.



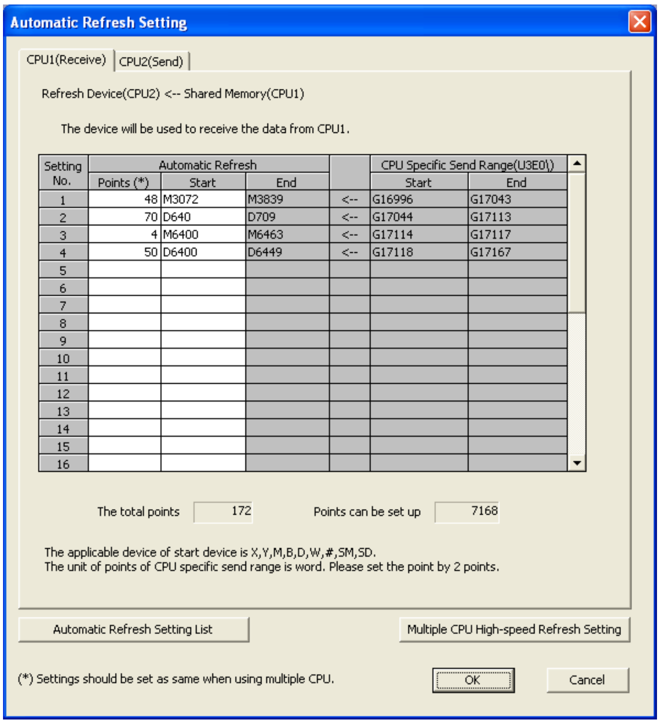
<table>
<tr><th>Item</th><th>Setting details</th></tr>
<tr><td>No. of CPU</td><td>Sets the number of CPUs including PLC CPUs.
• No. of CPUs: 2 to 4</td></tr>
<tr><td>Operation Mode</td><td>Sets the operation mode when a CPU stop error occurs.</td></tr>
<tr><td>Multiple CPU High Speed Transmission Area Settings</td><td>• Points
Sets the number of data points transmitted by each CPU.
Range: 0 to 14 [k points], unit: 1 [k points]
The default values assigned to each CPU are as follows.
<table>
<tr><th rowspan="2">No. of multiple CPUs</th><th colspan="4">Transmission area size for each CPU (words)</th></tr>
<tr><th>No.1</th><th>No.2</th><th>No.3</th><th>No.4</th></tr>
<tr><td>2</td><td>7 k</td><td>7 k</td><td>-</td><td>-</td></tr>
<tr><td>3</td><td>7 k</td><td>3 k</td><td>3 k</td><td>-</td></tr>
<tr><td>4</td><td>3 k</td><td>3 k</td><td>3 k</td><td>3 k</td></tr>
</table>
• User Setting Area
Displays the number of points used in the user setting area, and the used address start and end range.
The user setting area is the range used when performing automatic refresh subtracted from the number of points assigned to each CPU.
• Automatic Refresh
Displays the number of points set in the automatic refresh settings.
By clicking the [Automatic refresh] button, an automatic refresh settings dialog box appears.</td></tr>
<tr><td>Total</td><td>Displays the total number of [points] in the transmission range for each CPU.
Set so that the total for CPUs is equal to or less than the following number of points.
CPU No.2 configuration: 14 [k points]
CPU No.3 configuration: 13 [k points]
CPU No.4 configuration: 12 [k points]</td></tr>
</table>

(b) Automatic refresh settings

The settings required to use the automatic refresh function are shown below.

32 ranges can be set at each CPU module.

The Automatic Refresh Setting screen and setting ranges are shown below.



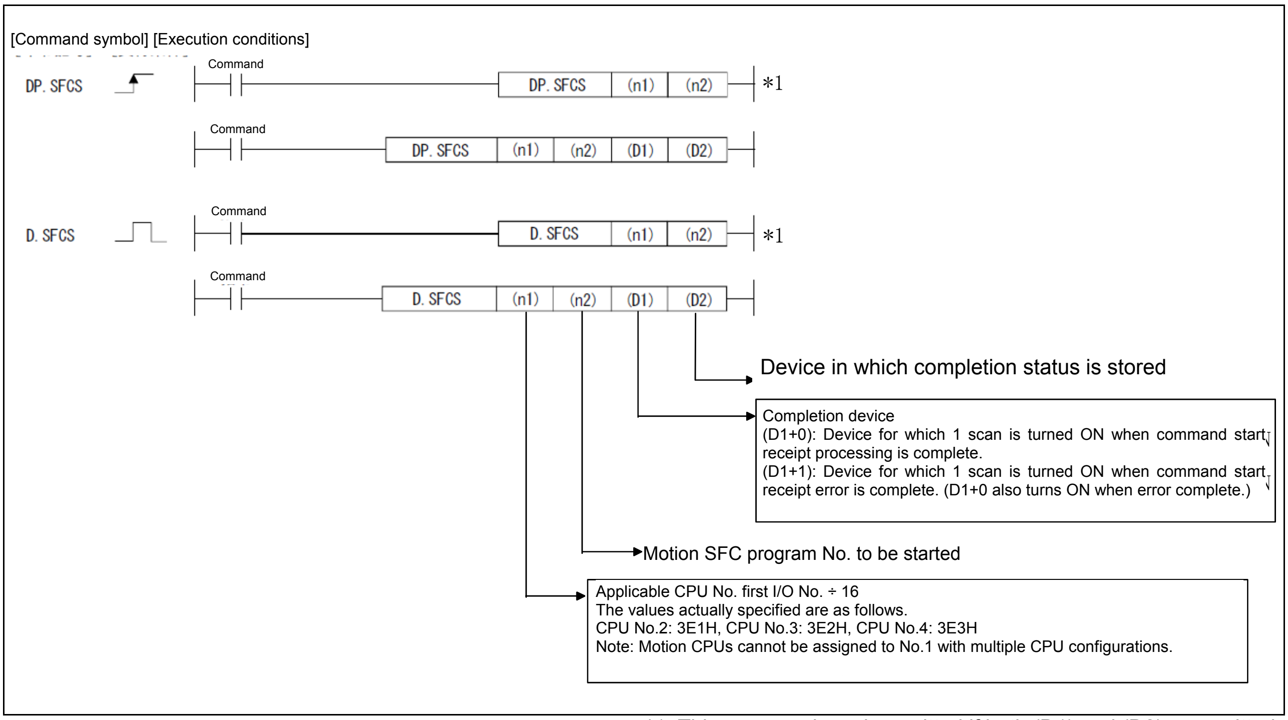
Item		Setting details
Tab		Select the CPU No. for which automatic refresh setting is to be specified.
Setting No.		Displays the transfer setting No. for each CPU module.
Automatic refresh	Points	Sets the number of points for which transfer is performed in word units. Setting range: 2 to 14336 Setting unit: 2
	Start	Sets the first device subject to transfer. Usable devices: X, Y, M, B, D, W, #, SM, SD
	End	Sets the last device subject to transfer. The last device is calculated from the [No. of points] and [First device].
Refresh direction		Displays the refresh direction. <--: Send -->: Receive ---: If the number of points has been entered, and the self CPU has not been set ×: If the device has not been set
CPU Specific Send Range		Displays the CPU transmission range used for automatic refresh.
The total points		Displays the total number of points.
Points can be set up		Displays the transmission range (k points) assigned to each CPU.

3.2 Dedicated Multiple CPU Motion Commands

This section describes dedicated commands (SFCS, SVST, CHGA, CHGV) for multiple CPUs.
However, refer to Appendix 7.5 (page, Appendix-54) for details on CHGA, and Appendix 7.6 (page, Appendix-57) for details on CHGV.

3.2.1 SFCS motion SFC program start command

This is an SFCS (SFC start) command used to start the specified motion SFC program.



*1: This command can be omitted if both (D1) and (D2) are omitted.

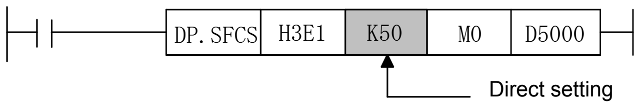
(1) Motion SFC program No. setting

The motion SFC program No. can be set directly or indirectly.

(a) Direct setting involves setting the motion SFC program No. directly with a numerical value (K0 to K255).

Example

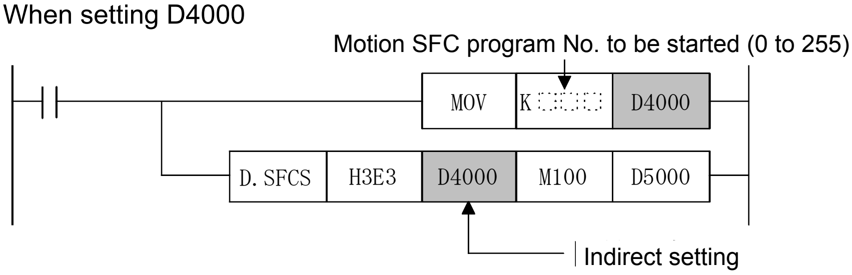
Motion SFC program No. 50 is set as follows.



(b) Indirect setting involves setting the motion SFC program No. with word device (D0 to D8191, W0 to W1FF) content.

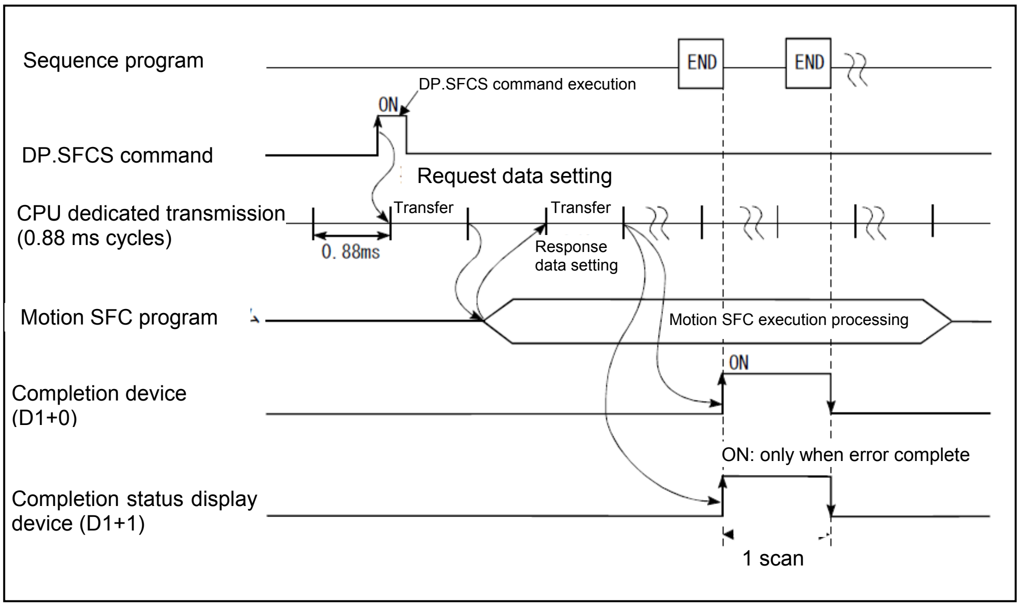
Example

When setting D4000



(2) Execution timing

A start request for the specified motion SFC program is made when the SFCS command execution command turns ON.
Motion SFC programs can be started regardless of whether the task setting is normal task execution or NMI task execution.
This is valid at any time, regardless of whether in real mode, virtual mode, or while changing mode.
The following is an overview of operation between CPUs when executing the DP.SFCS command.



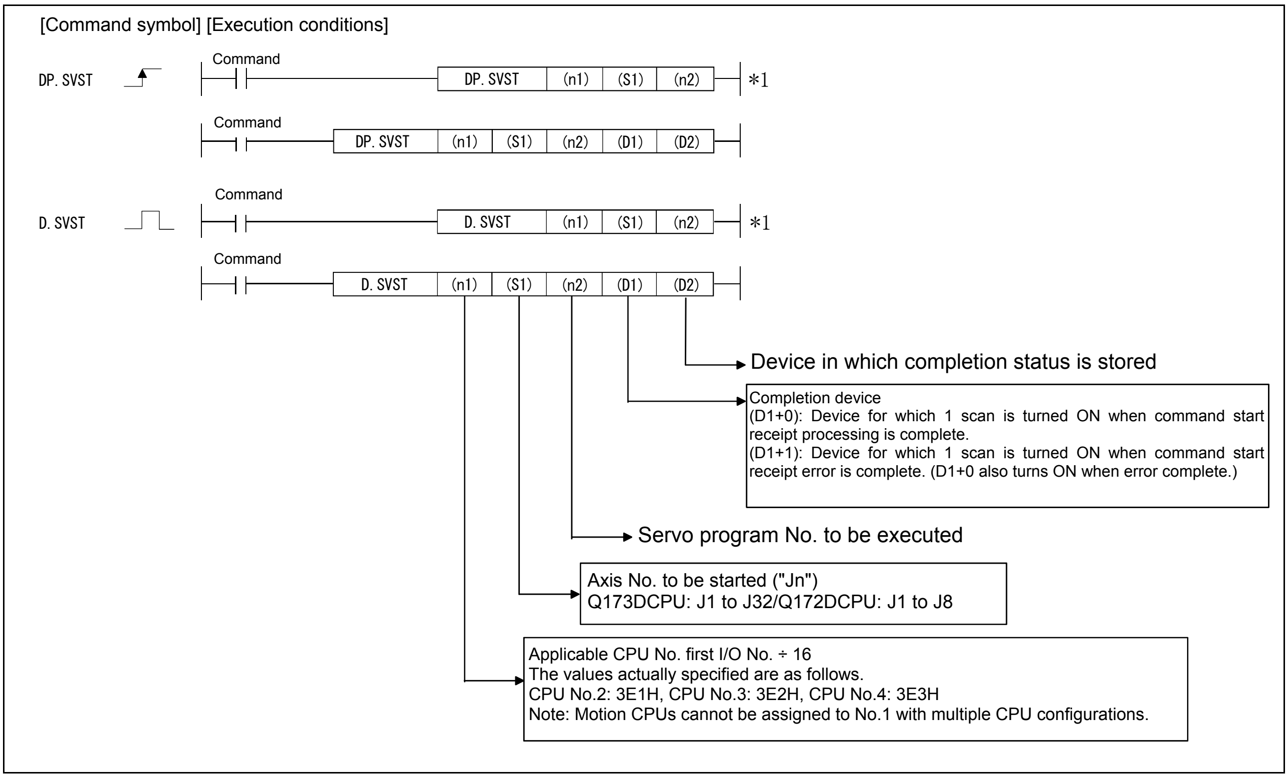
(3) Operation error conditions

In the following cases, an operation error occurs, and the SFCS command is not executed.

(a) When a CPU No. reserved with the applicable CPU No. first No. I/O No. ÷ 16(n1) is specified.
(b) When specified for the self CPU with the applicable CPU No. first No. I/O No. ÷ 16(n1).
(c) When a CPU other than a Q motion CPU is specified with the applicable CPU No. first No. I/O No. ÷ 16(n1).
(d) When the specified command name is incorrect.
(e) When the command is configured with a device other than a usable device.
(f) When 0 to 3DFH, or 3E4H and above is specified with the applicable CPU No. first No. I/O No. ÷ 16(n1).

3.2.2 SVST servo program start request command

This command is used to request the start of the specified servo program.

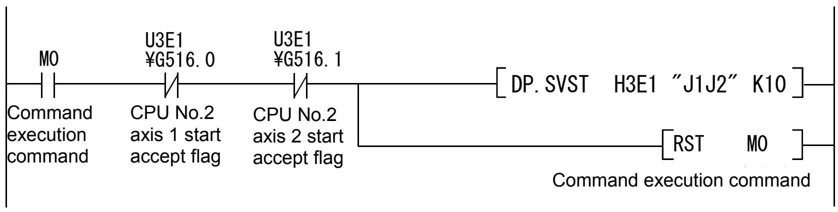


*1: This command can be omitted if both (D1) and (D2) are omitted.

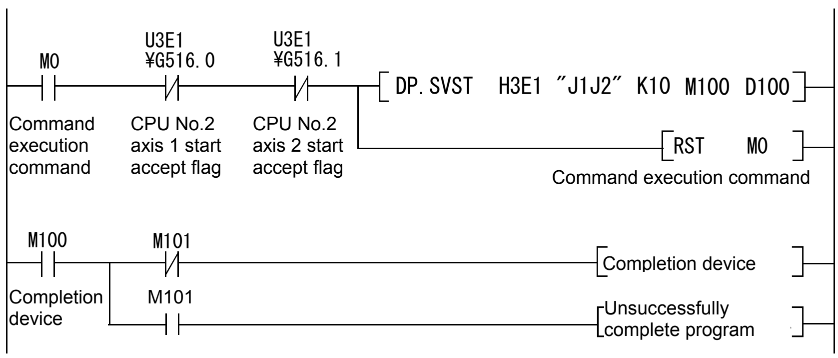
(1) SVST command program example

This program is used to issue a servo program No.10 start request for motion CPU (No.2) axis 1 and 2 when M0 is ON.

<Example 1> Program with Completion device, completion status omitted

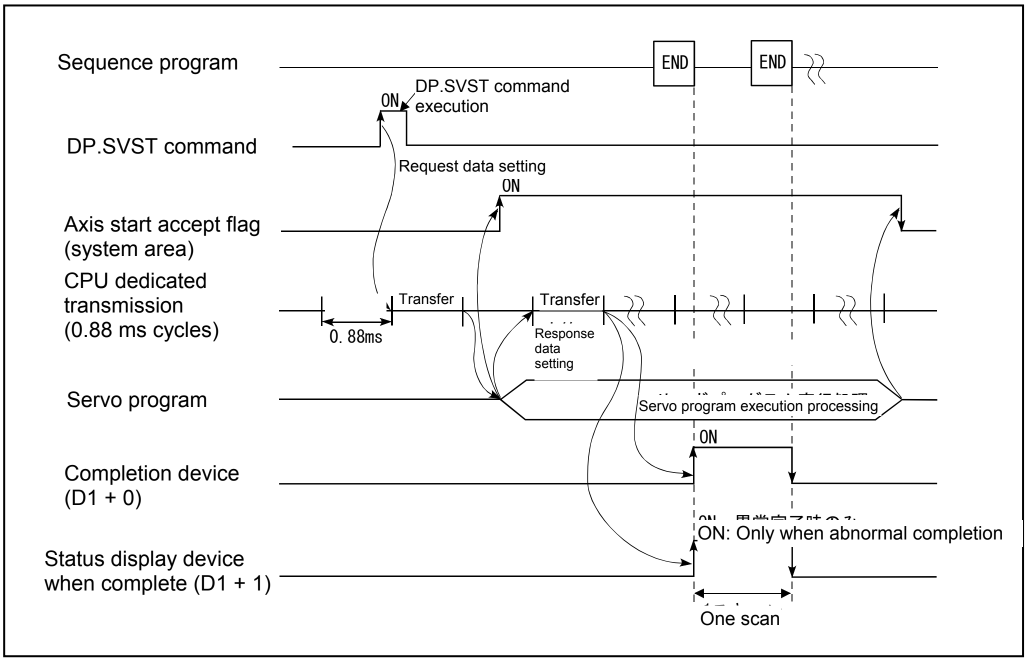


<Example 2> Program using Completion device, completion status



(2) Execution timing

A start request for the specified servo program is issued when the SVST command execution command turns ON.



(3) Error content

In the following cases, an abnormal termination occurs, and an error code is stored in the device specified at the completion status storage device (D2).
If the completion status storage device (D2) is omitted, no error is detected and processing is not performed, and therefore caution is advised.

Completion status * (Error code) (H)	Error cause	Remedy
0010	The command request from the PLC CPU to the motion CPU exceeds the permissible value.	Check the program, and then change to the correct sequence program.
2100	The number of command (D(P).SVST/D(P).CHGA combined) requests issued from the PLC CPU to the motion CPU simultaneously is 65 or more, and therefore the motion CPU is unable to process.	
2201	The No. of the servo program being executed lies outside the 0 to 4095 range.	
2202	The axis No. specified with the D(P).SVST command is illegal.	

*: 0000H (normal)

In the following cases, an operation error occurs, the diagnostic error flag (SM0) turns ON, and the error code is stored in the diagnostic error register (SD0).

Error code *	Error cause	Remedy
4350	The specified applicable CPU module is incorrect. (1) A reserved CPU No. was specified. (2) An uninstalled CPU No. was specified. (3) The applicable CPU module first No. I/O No. ÷ 16(n1) lies outside the 3E0H to 3E3H range.	Check the program, and then change to the correct sequence program.
4351	Cannot be executed at the specified applicable CPU module. (1) The command name is incorrect. (2) An unsupported command was specified at the applicable CPU module.	
4352	The number of specified command devices is incorrect.	
4353	A device that cannot be used with the specified command has been specified.	
4354	A character string that cannot be handled with the specified command has been specified.	

*: 0 (normal)

Chapter 4 Q Motion CPU

Q motion CPUs hold system settings data and servo data, and run the servo programs and mechanical support language required to perform multi-axis positioning.

Q motion CPUs hold the following types of data. The default values are set, and therefore it is necessary to make changes to the data to suit the system.
Data is stored in the motion CPU memory area (SRAM battery backup).

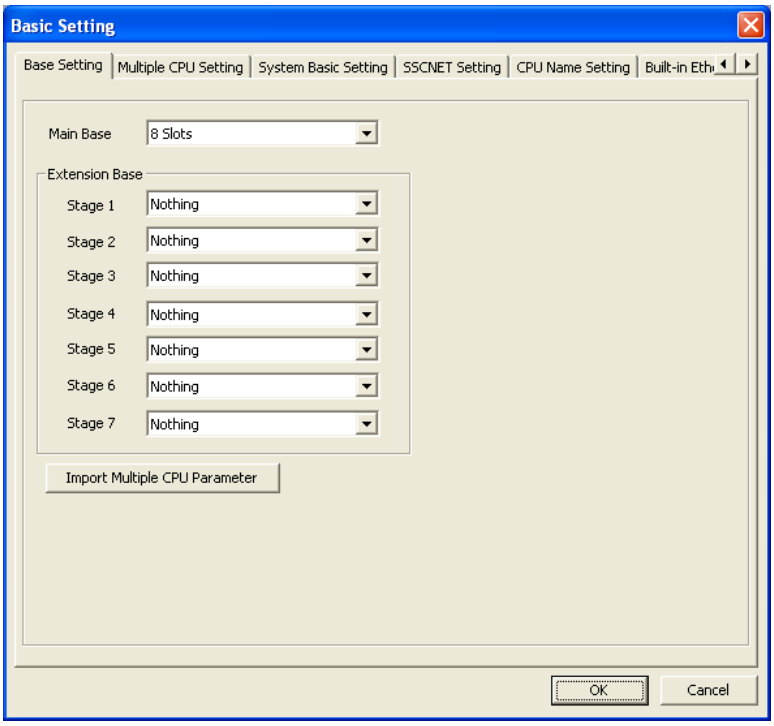
- Data
  - System settings
    - Basic settings: Specifies system basic settings such as main base and extension base settings, and multiple CPU system settings.
    - System configuration: Specifies module settings used for main bases and extension bases.
    - SSCNET configuration: Selects used servo amplifiers, servo motors, and regenerative resistors, etc.
  - Servo data settings
    - Servo data: Specifies settings relating to machine characteristic values, zeroing, and JOG operation.
    - Parameter blocks: (Block No.1 to 64) Data such as acceleration/deceleration time used in zeroing data, JOG operation data, and servo programs.
    - Servo parameters: (Axes 1 to 32) Sets servo characteristic values required for servo operation.

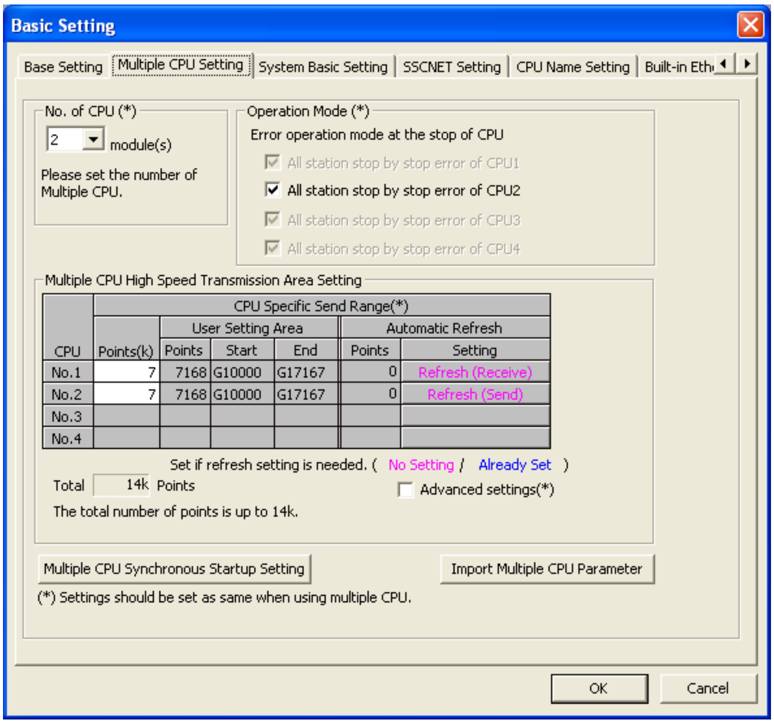
4.1 System Settings

System settings are used to select the bases and modules used, and to decide axis numbers, and the servo amplifier and servo motor types.

4.1.1 Basic settings

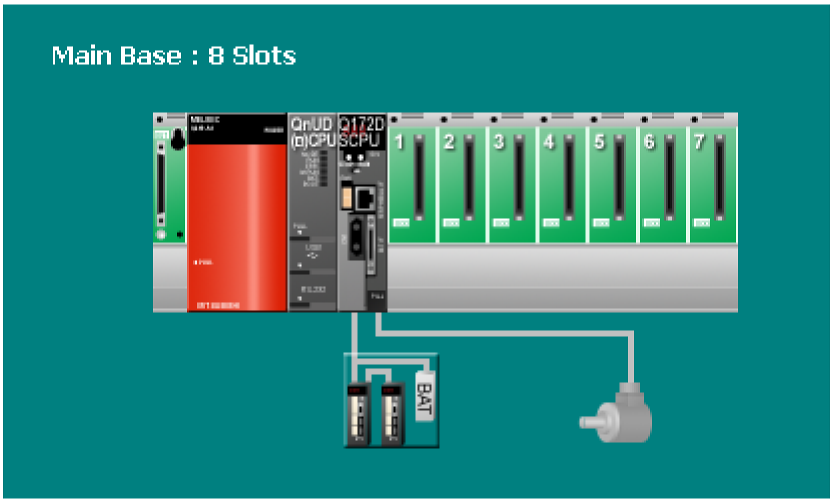
Basic settings are used to specify system basic settings such as main base and extension base settings, as well as multiple CPU system settings.





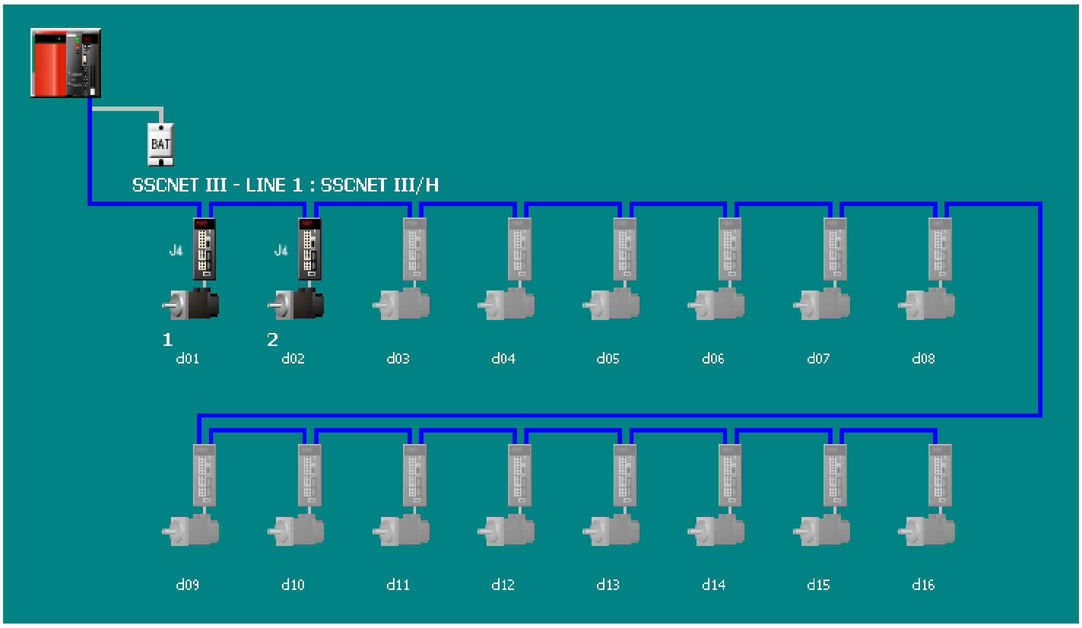
4.1.2 System configuration

The system configuration specifies module settings used for main bases and extension bases.



4.1.3 SSCNET configuration

The SSCNET configuration selects the servo amplifiers and servo motors used.



4.2 Servo Data Settings

Servo data settings are used to set servo data required to perform positioning control for the axes set in the system settings.

4.2.1 Servo data

Servo data is used to specify settings relating to machine characteristic values, zeroing, and JOG operation.

Item	Axis1	Axis2
Fixed Parameter	**Set the fixed parameters for each axis ...**	
Unit Setting	3:PLS	3:PLS
Number of Pulses/Rev.	20000[PLS]	20000[PLS]
Travel Value/Rev.	20000[PLS]	20000[PLS]
Backlash Compensation	0[PLS]	0[PLS]
Upper Stroke Limit	2147483647[PLS]	2147483647[PLS]
Lower Stroke Limit	0[PLS]	0[PLS]
Command In-position	100[PLS]	100[PLS]
Sp. Ctrl. 10x Mult. for Deg.	-	-
Home Position Return Data	**Set the data to execute the home position return.**	
OPR Direction	0:Reverse Direction	0:Reverse Direction
OPR Method	0:Proximity Dog Type 1	0:Proximity Dog Type 1
Home Position Address	0[PLS]	0[PLS]
OPR Speed	1[PLS/s]	1[PLS/s]
Creep Speed	1[PLS/s]	1[PLS/s]
Travel After Dog	-	-
Parameter Block Setting	1	1
OPR Retry Function	0:Invalid	0:Invalid
Dwell Time at OPR Retry	-	-
Home Position Shift Amount	0[PLS]	0[PLS]
Speed Set at Home Pos. Shift	0:OPR Speed	0:OPR Speed
Torque Limit Value at Creep Speed	-	-
Operation for OPR Incompletion	1:Not Execute Servo Program	1:Not Execute Servo Program
OPR Request Setting in Pulse Conversion Unit	-	-
Standby Time after Clear Signal Output in Pulse C...	-	-
JOG Operation Data	**Set the data to execute the JOG operati...**	
JOG Speed Limit Value	20000[PLS/s]	20000[PLS/s]
Parameter Block Setting	1	1

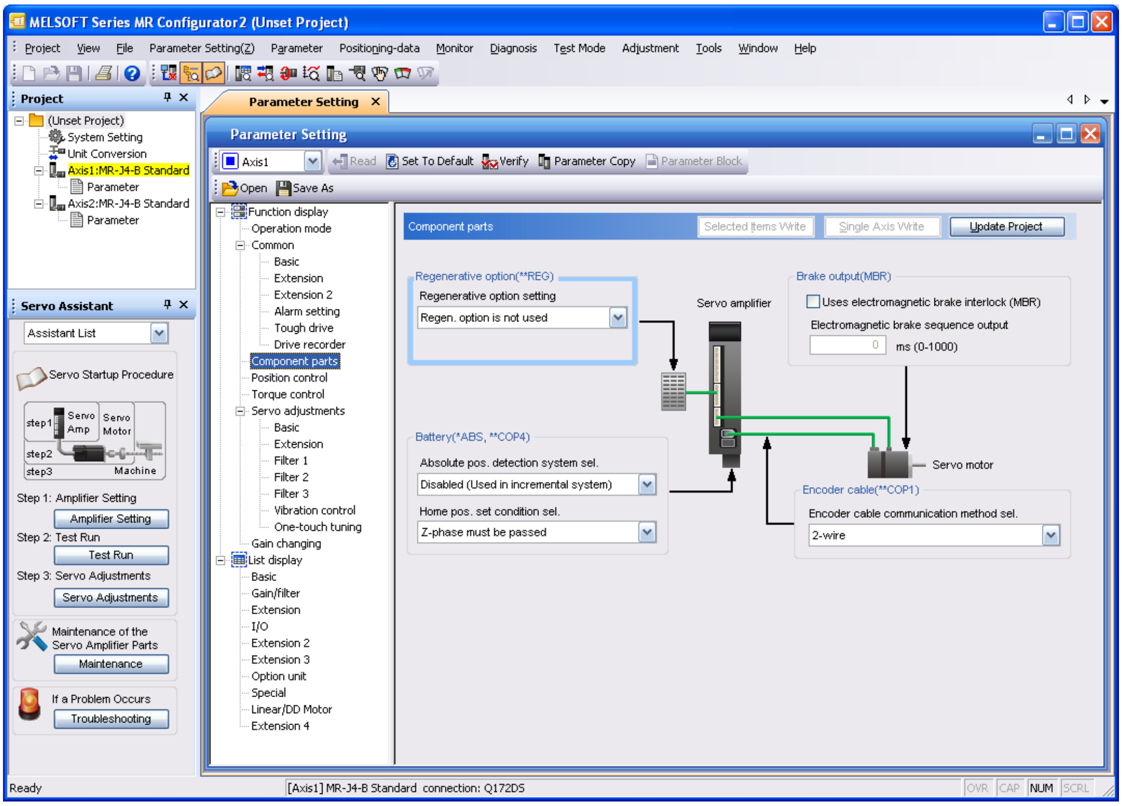
4.2.2 Parameter blocks

Parameter blocks contain data such as acceleration/deceleration time used in zeroing data, JOG operation data, and servo programs.

Item	Block No.1	Block No.2	Block No.3
Parameter Block	**Set the data such as the acceleration/deceleration control used for ea**		
Interpolation Control Unit	3:PLS	3:PLS	3:PLS
Speed Limit Value	200000[PLS/s]	200000[PLS/s]	200000[PLS/s]
Acceleration Time	1000[ms]	1000[ms]	1000[ms]
Deceleration Time	1000[ms]	1000[ms]	1000[ms]
Rapid Stop Deceleration Time	1000[ms]	1000[ms]	1000[ms]
S-curve Ratio	0[%]	0[%]	0[%]
Torque Limit Value	300[%]	300[%]	300[%]
Deceleration Process on STOP	0:Deceleration Stop	0:Deceleration Stop	0:Deceleration Stop
Allowable Error Range for Circular Interpolation	100[PLS]	100[PLS]	100[PLS]
Bias Speed at Start	0[PLS/s]	0[PLS/s]	0[PLS/s]
Acceleration/Deceleration System	0:Trapezoid/S-curve	0:Trapezoid/S-curve	0:Trapezoid/S-curve
Advanced S-curve Acceleration/Decelerat...	**Set the data of advanced S-curve acceleration/deceleration, which pe converting the speed smoothly.**		
Accel. Section 1 Ratio	-	-	-
Accel. Section 2 Ratio	-	-	-
Decel. Section 1 Ratio	-	-	-
Decel. Section 2 Ratio	-	-	-

4.2.3 Servo parameters

Servo parameters contain data determined by the specifications of servo amplifiers and servo motors controlled with parameters set for each axis, as well as data required to control servo motors.
Servo parameters are set with the setup software (MR-Configurator2).
Refer to the Servo amplifier Technical Document Collection for details on servo parameters.



POINT
If changes are made to parameters that require the servo amplifier control power to be rebooted, do so after resetting or rebooting the multiple CPU system.

4.3 Positioning Control Devices

Q motion CPUs are equipped with positioning control devices for positioning information.

Of the devices in the motion CPU, the following five devices are used for motion CPU internal signals.

If using SV13 (real mode)

• Internal relay (M):	M2000 to M3839	(1840 points)
• Special relay (SM):	SM0 to SM2255	(2256 points)
• Data register (D):	D0 to D799	(800 points)
• Motion register (#):	#8000 to #8735	(736 points)
• Special register (SD):	SD0 to SD2255	(2256 points)

If using SV22 (advanced synchronous control)

• Internal relay (M):	M2000 to M3839	(1840 points)
	M8192 to M12287	(4096 points)
• Special relay (SM):	SM0 to SM2255	(2256 points)
• Data register (D):	D0 to D799	(800 points)
	D10240 to D19823	(9584 points)
• Motion register (#):	#8000 to #8751	(752 points)
• Special register (SD):	SD0 to SD2255	(2256 points)

(1) Internal relay list

<table>
<tr><th colspan="2">SV13</th><th colspan="4">SV22</th></tr>
<tr><th rowspan="2">Device No.</th><th rowspan="2">Application type</th><th colspan="2">Virtual mode switching method</th><th colspan="2">Advanced synchronous control method</th></tr>
<tr><th>Device No.</th><th>Application type</th><th>Device No.</th><th>Application type</th></tr>
<tr><td>M0
~</td><td>User device
(2000 points)</td><td>M0
~</td><td>User device
(2000 points)</td><td>M0
~</td><td>User device
(2000 points)</td></tr>
<tr><td>M2000
~</td><td>Common device
(320 points)</td><td>M2000
~</td><td>Common device
(320 points)</td><td>M2000
~</td><td>Common device
(320 points)</td></tr>
<tr><td>M2320
~</td><td>Unusable
(80 points)</td><td>M2320
~</td><td>Unusable
(80 points)</td><td>M2320
~</td><td>Unusable
(80 points)</td></tr>
<tr><td>M2400
~</td><td>Axis status
(20 points × 32 axes)</td><td>M2400
~</td><td>Axis status
(20 points × 32 axes)
Real mode: all axes
Virtual mode: output modules</td><td>M2400
~</td><td>Axis status
(20 points × 32 axes)</td></tr>
<tr><td>M3040
~</td><td>Unusable
(32 points)</td><td>M3040
~</td><td>Unusable
(32 points)</td><td>M3040
~</td><td>Unusable
(32 points)</td></tr>
<tr><td>M3072
~</td><td>Common device (command signal)
(64 points)</td><td>M3072
~</td><td>Common device (command signal)
(64 points)</td><td>M3072
~</td><td>Common device (command signal)
(64 points)</td></tr>
<tr><td>M3136
~</td><td>Unusable
(64 points)</td><td>M3136
~</td><td>Unusable
(64 points)</td><td>M3136
~</td><td>Unusable
(64 points)</td></tr>
<tr><td>M3200
~</td><td>Axis command signal
(20 points × 32 axes)</td><td>M3200
~</td><td>Axis command signal
(20 points × 32 axes)
Real mode: all axes
Virtual mode: output modules</td><td>M3200
~</td><td>Axis command signal
(20 points × 32 axes)</td></tr>
<tr><td rowspan="7">M3840
~
M8191</td><td rowspan="7">User device
(4352 points)</td><td>M3840
~</td><td>Unusable
(160 points)</td><td rowspan="7">M3840
~
M8191</td><td rowspan="7">User device
(4352 points)</td></tr>
<tr><td>M4000
~</td><td>Virtual servo motor axis status *1
(20 points × 32 axes)</td></tr>
<tr><td>M4640
~</td><td>Synchronous encoder axis status
(4 points × 12 axes)</td></tr>
<tr><td>M4688
~</td><td>Unusable*1
(112 points)</td></tr>
<tr><td>M4800
~</td><td>Virtual servo motor axis command signal *1
(20 points × 32 axes)</td></tr>
<tr><td>M5440
~</td><td>Synchronous encoder axis command signal
(4 points × 12 axes)</td></tr>
<tr><td>M5488
~
M8191</td><td>User device
(2704 points)</td></tr>
</table>

SV13		SV22			
		Virtual mode switching method		Advanced synchronous control method	
Device No.	Application type	Device No.	Application type	Device No.	Application type
M8192 ~ M12287	System area (4096 points)	M8192 ~ M12287	System area (4096 points)	M8192 ~	System area (1608 points)
				M9800 ~	Command generation axis status (20 points × 32 axes)
				M10440 ~	Synchronous encoder axis status (10 points × 12 axes)
				M10560 ~	Output axis status (10 points × 32 axes)
				M10880 ~	Synchronous control signal[St.380] (32 points)
				M10912 ~	Synchronous analysis complete signal [St.381] (32 points)
				M10944 ~	Unusable (16 points)
				M10960 ~	Command generation axis command signal (20 points × 32 axes)
				M11600 ~	Synchronous encoder axis command signal (4 points × 12 axes)
				M11648 ~	Unusable (32 points)
				M11680 ~	Output axis command signal (10 points × 32 axes)
				M12000 ~	Synchronous control start signal [Rq.380] (32 points)
				M12032 ~	Synchronous analysis request signal [Rq.381] (32 points)
				M12064 ~ M12287	Unusable (224 points)

□ can be used with user devices.

*1: If using only in SV22 real mode, use with user devices is possible.

POINT

- Total number of user device points
  SV13: 6352 points (SV13), SV22 virtual mode switching method: 4704 points*,
  SV22 advanced synchronous control method: 6352 points
  *: If not used with virtual mode, up to 6096 points can be used.
- If using the Q172DCPU, devices for 16 axes are used.

(2) Data register list

<table>
<tr><th colspan="2">SV13</th><th colspan="4">SV22</th></tr>
<tr><th rowspan="2">Device No.</th><th rowspan="2">Application type</th><th colspan="2">Virtual mode switching method</th><th colspan="2">Advanced synchronous control method</th></tr>
<tr><th>Device No.</th><th>Application type</th><th>Device No.</th><th>Application type</th></tr>
<tr><td>D0
~</td><td>Axis monitor device
(20 points × 32 axes)</td><td>D0
~</td><td>Axis monitor device
(20 points × 32 axes)
Real mode: all axes
Virtual mode: output modules</td><td>D0
~</td><td>Axis monitor device
(20 points × 32 axes)</td></tr>
<tr><td>D640
~</td><td>Control change register
(2 points × 32 axes)</td><td>D640
~</td><td>Control change register
(2 points × 32 axes)</td><td>D640
~</td><td>Control change register
(2 points × 32 axes)</td></tr>
<tr><td>D704
~</td><td>Common device (command signal)
(54 points)</td><td>D704
~</td><td>Common device (command signal)
(54 points)</td><td>D704
~</td><td>Common device (command signal)
(54 points)</td></tr>
<tr><td>D758
~</td><td>Unusable
(42 points)</td><td>D758
~</td><td>Unusable
(42 points)</td><td>D758
~</td><td>Unusable
(42 points)</td></tr>
<tr><td rowspan="4">D800
~
D8191</td><td rowspan="4">User device
(7392 points)</td><td>D800
~</td><td>Virtual servo motor axis monitor Device *1
(10 points × 32 axes)</td><td rowspan="4">D800
~
D8191</td><td rowspan="4">User device
(7392 points)</td></tr>
<tr><td>D1120
~</td><td>Synchronous encoder axis monitor Device
(10 points × 12 axes)</td></tr>
<tr><td>D1240
~</td><td>Cam axis monitor device *1
(10 points × 32 axes)</td></tr>
<tr><td>D1560
~
D8191</td><td>User device
(6632 points)</td></tr>
<tr><td rowspan="11" colspan="2"></td><td rowspan="11" colspan="2"></td><td>D8192*2
~</td><td>User device
(2048 points)</td></tr>
<tr><td>D10240*2
~</td><td>System area
(2040 points)</td></tr>
<tr><td>D12280*2
~</td><td>Servo input axis monitor device
(10 points × 32 axes)</td></tr>
<tr><td>D12600*2
~</td><td>Command generation axis monitor device
(20 points × 32 axes)</td></tr>
<tr><td>D13240*2
~</td><td>Synchronous encoder axis monitor device
(20 points × 12 axes)</td></tr>
<tr><td>D13480*2
~</td><td>Unusable
(120 points)</td></tr>
<tr><td>D13600*2
~</td><td>Output axis monitor device
(30 points × 32 axes)</td></tr>
<tr><td>D14560*2
~</td><td>Unusable
(40 points)</td></tr>
<tr><td>D14600*2
~</td><td>Servo input axis control device
(2 points × 32 axes)</td></tr>
<tr><td>D14664*2
~</td><td>Unusable
(16 points)</td></tr>
<tr><td>D14680*2
~
D14807*2</td><td>Command generation axis control device
(4 points x 32 axes)</td></tr>
</table>

SV13		SV22			
		Virtual mode switching method		Advanced synchronous control method	
Device No.	Application type	Device No.	Application type	Device No.	Application type
				D14808 *2 ～	Unusable (12 points)
				D14820 *2 ～	Synchronous encoder axis control device (10 points × 12 axes)
				D14940 *2 ～	Unusable (60 points)
				D15000 *2 ～	Output axis control device (150 points × 32 axes)
				D19800 *2 ～ D19823 *2	Unusable (24 points)

☐ can be used with user devices.

*1: If using only in SV22 real mode, use with user devices is possible.

*2: If using the advanced synchronous control method, D8192 to D19823 cannot be set in the latch range.

POINT

- Total number of user device points
  SV13: 7392 points, SV22 virtual mode switching method: 6632 points*,
  SV22 advanced synchronous control method: 9440 points
  *: If not used with virtual mode, up to 7272 points can be used.
- If using the Q172DSCPU, devices for 16 axes are used.

4.3.1 Internal relays (status/command signals)

The Q17☐DSCPU is equipped with an internal relay with 12288 points from M0 to M12287.
Of these, M2400 to M5487 are used for data transfer for each axis, and the signal names and I/O Nos. for each axis are fixed as shown in the following tables.

(1) Axis status list

Axis No.	Device No.
1	M2400～M2419
2	M2420～M2439
3	M2440～M2459
4	M2460～M2479
5	M2480～M2499
6	M2500～M2519
7	M2520～M2539
8	M2540～M2559
9	M2560～M2579
10	M2580～M2599
11	M2600～M2619
12	M2620～M2639
13	M2640～M2659
14	M2660～M2679
15	M2680～M2699
16	M2700～M2719
17	M2720～M2739
18	M2740～M2759
19	M2760～M2779
20	M2780～M2799
21	M2800～M2819
22	M2820～M2839
23	M2840～M2859
24	M2860～M2879
25	M2880～M2899
26	M2900～M2919
27	M2920～M2939
28	M2940～M2959
29	M2960～M2979
30	M2980～M2999
31	M3000～M3019
32	M3020～M3039

Signal name

	Signal name		Refresh cycle	Load cycle	Signal type
0	Positioning start complete		Operation cycle		Status signal
1	Positioning complete				
2	In-position				
3	Command in-position				
4	Speed controlling				
5	Speed, position switching latch				
6	Zero pass				
7	Error detection		Immediate		
8	Servo error detection		Operation cycle		
9	Zeroing request		Main cycle		
10	Zeroing complete		Operation cycle		
11	External signals	FLS	Main cycle		
12		RLS			
13		STOP			
14		DOG/CHANGE			
15	Servo ready		Operation cycle		
16	Torque limiting				
17	Unusable		—	—	—
18	Virtual mode continued operation not possible warning (SV22 only) *1		When switching to virtual mode		Status signal
19	M-code outputting		Operation cycle		

*1: Unusable in SV13/SV22 real mode, SV22 advanced synchronous control.

POINT

(1) With the Q172DSCPU, the axis No.1 to 16 range is valid.
(2) With the Q172DSCPU, device areas of 17 axes or greater can be used as user devices.
However, if a Q172DSCPU project is replaced with a Q173DSCPU project, it will no longer be able to be used as a user device.

(2) Axis command signal list

Axis No.	Device No.
1	M3200～M3219
2	M3220～M3239
3	M3240～M3259
4	M3260～M3279
5	M3280～M3299
6	M3300～M3319
7	M3320～M3339
8	M3340～M3359
9	M3360～M3379
10	M3380～M3399
11	M3400～M3419
12	M3420～M3439
13	M3440～M3459
14	M3460～M3479
15	M3480～M3499
16	M3500～M3519
17	M3520～M3539
18	M3540～M3559
19	M3560～M3579
20	M3580～M3599
21	M3600～M3619
22	M3620～M3639
23	M3640～M3659
24	M3660～M3679
25	M3680～M3699
26	M3700～M3719
27	M3720～M3739
28	M3740～M3759
29	M3760～M3779
30	M3780～M3799
31	M3800～M3819
32	M3820～M3839

Signal name

	Signal name	Refresh cycle	Load cycle	Signal type
0	Stop command		Operation cycle	Command signal
1	Rapid stop command			
2	Forward rotation JOG start		Main cycle	
3	Reverse rotation JOG start			
4	Complete signal OFF command			
5	Speed, position switching enable command		Operation cycle	
6	Unusable	—	—	—
7	Error reset command		Main cycle	Command signal
8	Servo error reset command			
9	External stop input disable at start command		When starting	
10	Unusable	—	—	—
11				
12	Feed current value update command		When starting	Command signal
13	Address clutch reference setting command (SV22 only) *1		When switching to virtual mode	
14	Cam reference position setting command (SV22 only) *1			
15	Servo OFF command		Operation cycle	
16	Gain changing command		Operation cycle *2	
17	PI-PID changing command			
18	Control loop changing command		Operation cycle	
19	FIN signal			

*1: Unusable in SV13/SV22 real mode, SV22 advanced synchronous control.
*2: Every 3.5 [ms] if the operation cycle is 7.1 [ms] or longer.

POINT

(1) With the Q172DSCPU, the axis No.1 to 16 range is valid.
(2) With the Q172DSCPU, device areas of 17 axes or greater can be used as user devices.
However, if a Q172DSCPU project is replaced with a Q173DSCPU project, it will no longer be able to be used as a user device.

(3) Command generation axis status list

Axis No.	Device No.
1	M9800～M9819
2	M9820～M9839
3	M9840～M9859
4	M3260～M9879
5	M9880～M9899
6	M9900～M9919
7	M9920～M9939
8	M9940～M9959
9	M9960～M9979
10	M9980～M9999
11	M10000～M10019
12	M10020～M10039
13	M10040～M10059
14	M10060～M10079
15	M10080～M10099
16	M10100～M10119
17	M10120～M10139
18	M10140～M10159
19	M10160～M10179
20	M10180～M10199
21	M10200～M10219
22	M10220～M10239
23	M10240～M10259
24	M10260～M10279
25	M10280～M10299
26	M10300～M10319
27	M10320～M10339
28	M10340～M10359
29	M10360～M10379
30	M10380～M10399
31	M10400～M10419
32	M10420～M10439

Signal name

	Symbol	Signal name	Refresh cycle	Load cycle	Signal type
0	St. 340	Command generation axis positioning start complete	Operation cycle		Status signal
1	St. 341	Command generation axis positioning complete			
2	—	Unusable	—	—	—
3	St. 342	Command generation axis command in-position	Operation cycle		Status signal
4	St. 343	Command generation axis speed controlling			
5	—	Unusable	—	—	—
6					
7	St. 344	Command generation axis error detection	Immediate		Status signal
8	—	Unusable	—	—	—
9					
10	St. 345	Command generation axis start accept flag	Operation cycle		Status signal
11	St. 346	Command generation axis speed change accepting flag			
12	St. 347	Command generation axis speed change "0" accepting flag			
13	St. 348	Command generation axis automatic decelerating flag			
14	—	Unusable	—	—	—
15					
16					
17					
18					
19	St. 349	Command generation axis M-code outputting	Operation cycle		Status signal

POINT

(1) With the Q172DSCPU, the axis No.1 to 16 range is valid.
(2) With the Q172DSCPU, device areas of 17 axes or greater can be used as user devices.
However, if a Q172DSCPU project is replaced with a Q173DSCPU project, it will no longer be able to be used as a user device.

(4) Command generation axis command signal list

Axis No.	Device No.
1	M10960～M10979
2	M10980～M10999
3	M11000～M11019
4	M11020～M11039
5	M11040～M11059
6	M11060～M11079
7	M11080～M11099
8	M11100～M11119
9	M11120～M11139
10	M11140～M11159
11	M11160～M11179
12	M11180～M11199
13	M11200～M11219
14	M11220～M11239
15	M11240～M11259
16	M11260～M11279
17	M11280～M11299
18	M11300～M11319
19	M11320～M11339
20	M11340～M11359
21	M11360～M11379
22	M11380～M11399
23	M11400～M11419
24	M11420～M11439
25	M11440～M11459
26	M11460～M11479
27	M11480～M11499
28	M11500～M11519
29	M11520～M11539
30	M11540～M11559
31	M11560～M11579
32	M11580～M11599

Signal name

	Symbol	Signal name	Refresh cycle	Load cycle	Signal type
0	Rq. 341	Command generation axis stop command		Operation cycle	Command signal
1	Rq. 342	Command generation axis rapid stop command			
2	Rq. 343	Command generation axis forward rotation JOG start command		Main cycle	
3	Rq. 344	Command generation axis reverse rotation JOG start command			
4	Rq. 345	Command generation axis complete signal OFF command			
5	—	Unusable	—	—	—
6					
7	Rq. 346	Command generation axis error reset command		Main cycle	Command signal
8	—	Unusable	—	—	—
9					
10					
11					
12	Rq. 347	Command generation axis feed current value update request command		When starting	Command signal
13	—	Unusable	—	—	—
14					
15					
16					
17					
18					
19	Rq. 348	Command generation axis FIN signal		Operation cycle	Signal type

POINT

(1) With the Q172DSCPU, the axis No.1 to 16 range is valid.
(2) With the Q172DSCPU, device areas of 17 axes or greater can be used as user devices.
However, if a Q172DSCPU project is replaced with a Q173DSCPU project, it will no longer be able to be used as a user device.

(5) Synchronous encoder axis status list

Axis No.	Device No.
1	M10440～M10449
2	M10450～M10459
3	M10460～M10469
4	M10470～M10479
5	M10480～M10489
6	M10490～M10499
7	M10500～M10509
8	M10510～M10519
9	M10520～M10529
10	M10530～M10539
11	M10540～M10549
12	M10550～M10559

Signal name

	Symbol	Signal name	Refresh cycle	Load cycle	Signal type
0	St. 320	Synchronous encoder axis setting valid flag	When power turned ON		Status signal
1	St. 321	Synchronous encoder axis connecting valid flag	Operation cycle		Status signal
2	St. 322	Synchronous encoder axis counter enable flag	Operation cycle		Status signal
3	St. 323	Synchronous encoder axis current value setting request flag	Operation cycle		Status signal
4	St. 324	Synchronous encoder axis error detection flag	Immediate		Status signal
5	—	Unusable	—	—	—
6	St. 325	Synchronous encoder axis control complete flag	Immediate		Status signal
7	—	Unusable	—	—	—
8	—	Unusable	—	—	—
9	—	Unusable	—	—	—

(6) Synchronous encoder axis command signal list

Axis No.	Device No.
1	M11600～M11603
2	M11604～M11607
3	M11608～M11611
4	M11612～M11615
5	M11616～M11619
6	M11620～M11623
7	M11624～M11627
8	M11628～M11631
9	M11632～M11635
10	M11636～M11639
11	M11640～M11643
12	M11644～M11647

Signal name

	Symbol	Signal name	Refresh cycle	Load cycle	Signal type
0	Rq. 323	Synchronous encoder axis error reset		Main cycle	Command signal
1	Rq. 320	Synchronous encoder axis control request		Operation cycle	Command signal
2	Rq. 324	Connection command of synchronous encoder via device/master CPU		Main cycle	Command signal
3	—	Unusable	—	—	—

(7) Output axis status list

Axis No.	Device No.
1	M10560～M10569
2	M10570～M10579
3	M10580～M10589
4	M10590～M10599
5	M10600～M10609
6	M10610～M10619
7	M10620～M10629
8	M10630～M10639
9	M10640～M10649
10	M10650～M10659
11	M10660～M10669
12	M10670～M10679
13	M10680～M10689
14	M10690～M10699
15	M10700～M10709
16	M10710～M10719
17	M10720～M10729
18	M10730～M10739
19	M10740～M10749
20	M10750～M10759
21	M10760～M10769
22	M10770～M10779
23	M10780～M10789
24	M10790～M10799
25	M10800～M10809
26	M10810～M10819
27	M10820～M10829
28	M10830～M10839
29	M10840～M10849
30	M10850～M10859
31	M10860～M10869
32	M10870～M10879

Signal name

	Symbol	Signal name	Refresh cycle	Load cycle	Signal type
0	St. 420	Main shaft clutch ON/OFF status	Operation cycle		Status signal
1	St. 421	Main shaft clutch smoothing status			
2	St. 423	Auxiliary shaft clutch ON/OFF status			
3	St. 424	Auxiliary shaft clutch smoothing status			
4	—	Unusable	—	—	—
5					
6	St. 426	Control change complete	Operation cycle		Status signal
7	—	Unusable	—	—	—
8					
9					

POINT

(1) With the Q172DSCPU, the axis No.1 to 16 range is valid.
(2) With the Q172DSCPU, device areas of 17 axes or greater can be used as user devices.
However, if a Q172DSCPU project is replaced with a Q173DSCPU project, it will no longer be able to be used as a user device.

(8) Output axis command signal list

Axis No.	Device No.	Signal name
1	M11680～M11689	
2	M11690～M11699	
3	M11700～M11709	
4	M11710～M11719	
5	M11720～M11729	
6	M11730～M11739	
7	M11740～M11749	
8	M11750～M11759	
9	M11760～M11769	
10	M11770～M11779	
11	M11780～M11789	
12	M11790～M11799	
13	M11800～M11809	
14	M11810～M11819	
15	M11820～M11829	
16	M11830～M11839	
17	M11840～M11849	
18	M11850～M11859	
19	M11860～M11869	
20	M11870～M11879	
21	M11880～M11889	
22	M11890～M11899	
23	M11900～M11909	
24	M11910～M11919	
25	M11920～M11929	
26	M11930～M11939	
27	M11940～M11949	
28	M11950～M11959	
29	M11960～M11969	
30	M11970～M11979	
31	M11980～M11989	
32	M11990～M11999	

	Symbol	Signal name	Refresh cycle	Load cycle	Signal type
0	Rq. 400	Main shaft clutch command		Operation cycle	Command signal
1	Rq. 401	Main shaft clutch control invalid command			
2	Rq. 402	Main shaft clutch forced OFF command			
3	—	Unusable	—	—	—
4	Rq. 403	Auxiliary shaft clutch command		Operation cycle	Command signal
5	Rq. 404	Auxiliary clutch control invalid command			
6	Rq. 405	Auxiliary clutch forced OFF command			
7	—	Unusable	—	—	—
8	Rq. 406	Control change request command		Operation cycle	Command signal
9	—	Unusable	—	—	—

POINT

(1) With the Q172DSCPU, the axis No.1 to 16 range is valid.
(2) With the Q172DSCPU, device areas of 17 axes or greater can be used as user devices.
However, if a Q172DSCPU project is replaced with a Q173DSCPU project, it will no longer be able to be used as a user device.

(9) Synchronous control signal list

Axis No.	Device No.	Symbol	Signal name	Refresh cycle	Load cycle	Signal type
1	M10880	St.380	Synchronous controlling	Operation cycle		Status signal
2	M10881					
3	M10882					
4	M10883					
5	M10884					
6	M10885					
7	M10886					
8	M10887					
9	M10888					
10	M10889					
11	M10890					
12	M10891					
13	M10892					
14	M10893					
15	M10894					
16	M10895					
17	M10896					
18	M10897					
19	M10898					
20	M10899					
21	M10900					
22	M10901					
23	M10902					
24	M10903					
25	M10904					
26	M10905					
27	M10906					
28	M10907					
29	M10908					
30	M10909					
31	M10910					
32	M10911					

POINT

(1) With the Q172DSCPU, the axis No.1 to 16 range is valid.
(2) With the Q172DSCPU, device areas of 17 axes or greater can be used as user devices.
However, if a Q172DSCPU project is replaced with a Q173DSCPU project, it will no longer be able to be used as a user device.

(10) Synchronous analysis complete signal list

Axis No.	Device No.	Symbol	Signal name	Refresh cycle	Load cycle	Signal type
1	M10912	St.381	Synchronous analysis complete	Operation cycle		Status signal
2	M10913					
3	M10914					
4	M10915					
5	M10916					
6	M10917					
7	M10918					
8	M10919					
9	M10920					
10	M10921					
11	M10922					
12	M10923					
13	M10924					
14	M10925					
15	M10926					
16	M10927					
17	M10928					
18	M10929					
19	M10930					
20	M10931					
21	M10932					
22	M10933					
23	M10934					
24	M10935					
25	M10936					
26	M10937					
27	M10938					
28	M10939					
29	M10940					
30	M10941					
31	M10942					
32	M10943					

POINT

(1) With the Q172DSCPU, the axis No.1 to 16 range is valid.
(2) With the Q172DSCPU, device areas of 17 axes or greater can be used as user devices.
However, if a Q172DSCPU project is replaced with a Q173DSCPU project, it will no longer be able to be used as a user device.

(11) Synchronous control start signal list

Axis No.	Device No.	Symbol	Signal name	Refresh cycle	Load cycle	Signal type
1	M12000	Rq. 380	Synchronous control start		Operation cycle	Command signal
2	M12001					
3	M12002					
4	M12003					
5	M12004					
6	M12005					
7	M12006					
8	M12007					
9	M12008					
10	M12009					
11	M12010					
12	M12011					
13	M12012					
14	M12013					
15	M12014					
16	M12015					
17	M12016					
18	M12017					
19	M12018					
20	M12019					
21	M12020					
22	M12021					
23	M12022					
24	M12023					
25	M12024					
26	M12025					
27	M12026					
28	M12027					
29	M12028					
30	M12029					
31	M12030					
32	M12031					

POINT

(1) With the Q172DSCPU, the axis No.1 to 16 range is valid.
(2) With the Q172DSCPU, device areas of 17 axes or greater can be used as user devices.
However, if a Q172DSCPU project is replaced with a Q173DSCPU project, it will no longer be able to be used as a user device.

(12) Synchronous analysis request signal list

Axis No.	Device No.	Symbol	Signal name	Refresh cycle	Load cycle	Signal type
1	M12032	Rq. 381	Synchronous analysis request		When starting synchronous control	Command signal
2	M12033					
3	M12034					
4	M12035					
5	M12036					
6	M12037					
7	M12038					
8	M12039					
9	M12040					
10	M12041					
11	M12042					
12	M12043					
13	M12044					
14	M12045					
15	M12046					
16	M12047					
17	M12048					
18	M12049					
19	M12050					
20	M12051					
21	M12052					
22	M12053					
23	M12054					
24	M12055					
25	M12056					
26	M12057					
27	M12058					
28	M12059					
29	M12060					
30	M12061					
31	M12062					
32	M12063					

POINT

(1) With the Q172DSCPU, the axis No.1 to 16 range is valid.
(2) With the Q172DSCPU, device areas of 17 axes or greater can be used as user devices.
However, if a Q172DSCPU project is replaced with a Q173DSCPU project, it will no longer be able to be used as a user device.

4.3.2 Internal relays (common devices)

Q17□DSCPU is equipped with an internal relay with 12288 points from M0 to M12287.
Of these, M2000 to M2319 and M3072 to M3135 are used for positioning control, and their respective applications are fixed as shown in the following tables.

(1) Common devices

Device No.	Signal name
M2000	PLC ready flag
M2001 ⟆ M2032	Axis 1 start accept flag ⟆ Axis 32 start accept flag
M2033	Unusable
M2034	
M2035	Motion error history clear request flag
M2036	Unusable
M2037	
M2038	Motion SFC debugging flag
M2039	Motion error detection flag
M2040	Speed switching point specified flag
M2041	System setting error flag
M2042	All axis servo ON command
M2043	Real mode/virtual mode switching request (SV22) *1
M2044	Real mode/virtual mode switching request (SV22) *1
M2045	Real mode/virtual mode switching error (SV22) *1
M2046	Out-of-sync warning (SV22) *1
M2047	Motion slot module error
M2048	JOG simultaneous start command
M2049	All axes servo ON accept flag
M2050	Unusable
M2051	Manual pulse generator 1 enable flag
M2052	Manual pulse generator 2 enable flag
M2053	Manual pulse generator 3 enable flag
M2054	Operation cycle over flag
M2055 ⟆ M2060	Unusable
M2061 ⟆ M2092	Axis 1 speed change flag ⟆ Axis 32 speed change flag
M2093 ⟆ M2100	Unusable
M2101 ⟆ M2112	Axis 1 synchronous encoder current value changing flag ⟆ Axis 12 synchronous encoder current value changing flag *1 *2
M2113 ⟆ M2127	Unusable
M2128 ⟆ M2159	Axis 1 automatic decelerating flag ⟆ Axis 32 automatic decelerating flag
M2160 ⟆ M2239	Unusable
M2240 ⟆ M2271	Axis 1 speed change "0" accepting flag ⟆ Axis 32 speed change "0" accepting flag

Device No.	Signal name
M2272 ⟆ M2303	Axis 1 control loop monitor status ⟆ Axis 32 control loop monitor status
M2304 ⟆ M2319	Unusable

*1: Unusable when performing SV22 advanced synchronous control.
*2: Unusable in real mode.

(2) Common device (command signal) list

Device No.	Signal name	Refresh cycle	Load cycle	Signal type	Remarks *1, *2
M3072	PLC ready flag		Main cycle	Command signal	M2000
M3073	Speed switching point specified flag		When starting		M2040
M3074	All axes servo ON command		Operation cycle		M2042
M3075	Real mode/virtual mode switching Request (SV22) *3		When switching to virtual mode		M2043
M3076	JOG operation simultaneous start command		Main cycle		M2048
M3077	Manual pulse generator 1 enable flag				M2051
M3078	Manual pulse generator 2 enable flag				M2052
M3079	Manual pulse generator 3 enable flag				M2053
M3080	Motion error history clear request flag				M2035
M3081 ~ M3135	Unusable*4 (55 points)	-	-	-	-

*1: If the device in the Remarks field is turned ON/OFF directly, the device status will not match. Please note that if requests are issued simultaneously from the data register and the above devices, requests from these devices will be valid.
*2: Commands are possible even for devices in the Remarks field.
*3: Unusable when performing SV22 advanced synchronous control.
*4: Do not use as a user device. This will be a command signal spare area, and therefore can be used as a device to perform automatic refresh.

4.3.3 Data register (monitor device/control change register)

There are 19824 data registers in the Q17nDSCPU, from D0 to D19823.
Of these, 800 points from D0 to D799 are used for positioning control, and 9584 points from D10240 to D19823 are used for advanced synchronous control, and their respective applications are fixed as shown in the following tables.

(1) Axis monitor device list

Axis No.	Device No.
1	D0～D19
2	D20～D39
3	D40～D59
4	D60～D79
5	D80～D99
6	D100～D119
7	D120～D139
8	D140～D159
9	D160～D179
10	D180～D199
11	D200～D219
12	D220～D239
13	D240～D259
14	D260～D279
15	D280～D299
16	D300～D319
17	D320～D339
18	D340～D359
19	D360～D379
20	D380～D399
21	D400～D419
22	D420～D439
23	D440～D459
24	D460～D479
25	D480～D499
26	D500～D519
27	D520～D539
28	D540～D559
29	D560～D579
30	D580～D599
31	D600～D619
32	D620～D639

Signal name

	Signal name	Refresh cycle	Load cycle	Signal type
0	Feed current value	Operation cycle		Monitor device
1				
2	Real current value			
3				
4	Deviation counter value			
5				
6	Minor error code	Immediate		
7	Major error code			
8	Servo error code	Main cycle		
9	Zeroing retravel value	Operation cycle		
10	Travel value after proximity dog ON			
11				
12	Execute program No.	When starting		
13	M-code	Operation cycle		
14	Torque limit value			
15	Data set pointer for constant-speed control	When starting/started		
16	Unusable *	—	—	—
17				
18	Real current value at stop input	Operation cycle		Monitor device
19				

*1: Can be used as the travel value change register. The travel value change register can be set for the desired device in the servo program.

POINT

(1) With the Q172DSCPU, the axis No.1 to 16 range is valid.
(2) With the Q172DSCPU, device areas of 17 axes or greater can be used as user devices.
However, if a Q172DSCPU project is replaced with a Q173DSCPU project, it will no longer be able to be used as a user device.

(2) Control change register list

Axis No.	Device No.	Signal name
1	D640, D641	
2	D642, D643	
3	D644, D645	
4	D646, D647	
5	D648, D649	
6	D650, D651	
7	D652, D653	
8	D654, D655	
9	D656, D657	
10	D658, D659	
11	D660, D661	
12	D662, D663	
13	D664, D665	
14	D666, D667	
15	D668, D669	
16	D670, D671	
17	D672, D673	
18	D674, D675	
19	D676, D677	
20	D678, D679	
21	D680, D681	
22	D682, D683	
23	D684, D685	
24	D686, D687	
25	D688, D689	
26	D690, D691	
27	D692, D693	
28	D694, D695	
29	D696, D697	
30	D698, D699	
31	D700, D701	
32	D702, D703	

	Signal name	Refresh cycle	Load cycle	Signal type
0 1	JOG speed setting		When starting	Command device

POINT

(1) With the Q172DSCPU, the axis No.1 to 16 range is valid.
(2) With the Q172DSCPU, device areas of 17 axes or greater can be used as user devices.
However, if a Q172DSCPU project is replaced with a Q173DSCPU project, it will no longer be able to be used as a user device.

(3) Servo input axis monitor device list

Axis No.	Device No.
1	D12280～D12289
2	D12290～D12299
3	D12300～D12309
4	D12310～D12319
5	D12320～D12329
6	D12330～D12339
7	D12340～D12349
8	D12350～D12359
9	D12360～D12369
10	D12370～D12379
11	D12380～D12389
12	D12390～D12399
13	D12400～D12409
14	D12410～D12419
15	D12420～D12429
16	D12430～D12439
17	D12440～D12449
18	D12450～D12459
19	D12460～D12469
20	D12470～D12479
21	D12480～D12489
22	D12490～D12499
23	D12500～D12509
24	D12510～D12519
25	D12520～D12529
26	D12530～D12539
27	D12540～D12549
28	D12550～D12559
29	D12560～D12569
30	D12570～D12579
31	D12580～D12589
32	D12590～D12599

Signal name

	Symbol	Signal name	Refresh cycle	Load cycle	Signal type
0	Md. 300	Servo input axis current value	Operation cycle		Monitor device
1					
2	Md. 301	Servo input axis speed			
3					
4	Md. 302	Servo input axis phase compensation amount			
5					
6	Md. 303	Servo input axis rotation direction restriction amount			
7					
8	—	Unusable	—	—	—
9					

POINT

(1) With the Q172DSCPU, the axis No.1 to 16 range is valid.
(2) With the Q172DSCPU, device areas of 17 axes or greater can be used as user devices.
However, if a Q172DSCPU project is replaced with a Q173DSCPU project, it will no longer be able to be used as a user device.

(4) Servo input axis control device list

Axis No.	Device No.	Signal name
1	D14600, D14601	
2	D14602, D14603	
3	D14604, D14605	
4	D14606, D14607	
5	D14608, D14609	
6	D14610, D14611	
7	D14612, D14613	
8	D14614, D14615	
9	D14616, D14617	
10	D14618, D14619	
11	D14620, D14621	
12	D14622, D14623	
13	D14624, D14625	
14	D14626, D14627	
15	D14628, D14629	
16	D14630, D14631	
17	D14632, D14633	
18	D14634, D14635	
19	D14636, D14637	
20	D14638, D14639	
21	D14640, D14641	
22	D14642, D14643	
23	D14644, D14645	
24	D14646, D14647	
25	D14648, D14649	
26	D14650, D14651	
27	D14652, D14653	
28	D14654, D14655	
29	D14656, D14657	
30	D14658, D14659	
31	D14660, D14661	
32	D14662, D14663	

	Symbol	Signal name	Refresh cycle	Load cycle	Signal type
0 1	Pr. 302	Servo input axis phase compensation advance time		Operation cycle	Command device

POINT

(1) With the Q172DSCPU, the axis No.1 to 16 range is valid.
(2) With the Q172DSCPU, device areas of 17 axes or greater can be used as user devices.
However, if a Q172DSCPU project is replaced with a Q173DSCPU project, it will no longer be able to be used as a user device.

(5) Command generation axis monitor device list

Axis No.	Device No.
1	D12600～D12619
2	D12620～D12639
3	D12640～D12659
4	D12660～D12679
5	D12680～D12699
6	D12700～D12719
7	D12720～D12739
8	D12740～D12759
9	D12760～D12779
10	D12780～D12799
11	D12800～D12819
12	D12820～D12839
13	D12840～D12859
14	D12860～D12879
15	D12880～D12899
16	D12900～D12919
17	D12920～D12939
18	D12940～D12959
19	D12960～D12979
20	D12980～D12999
21	D13000～D13019
22	D13020～D13039
23	D13040～D13059
24	D13060～D13079
25	D13080～D13099
26	D13100～D13119
27	D13120～D13139
28	D13140～D13159
29	D13160～D13179
30	D13180～D13199
31	D13200～D13219
32	D13220～D13239

Signal name

	Symbol	Signal name	Refresh cycle	Load cycle	Signal type
0	Md. 340	Command generation axis feed current value	Operation cycle		Monitor device
1					
2	Md. 341	Command generation axis minor error code	Immediate		
3	Md. 342	Command generation axis major error code			
4	Md. 343	Command generation axis execute program No.	When starting		
5	Md. 344	Command generation axis M-code	Operation cycle		
6	Md. 345	Command generation axis accumulative current value			
7					
8	—	Unusable	—	—	—
9	Md. 346	Command generation axis data set pointer for constant-speed control	When starting/started		Monitor device
10	Md. 347	Command generation axis current value per cycle	Operation cycle		
11					
12	Md. 348	Command generation axis command speed			
13					
14	—	Unusable	—	—	—
15					
16					
17					
18					
19					

POINT

(1) With the Q172DSCPU, the axis No.1 to 16 range is valid.
(2) With the Q172DSCPU, device areas of 17 axes or greater can be used as user devices.
However, if a Q172DSCPU project is replaced with a Q173DSCPU project, it will no longer be able to be used as a user device.

(6) Command generation axis control device list

Axis No.	Device No.	Signal name
1	D14680～D14683	
2	D14684～D14687	
3	D14688～D14691	
4	D14692～D14695	
5	D14696～D14699	
6	D14700～D14703	
7	D14704～D14707	
8	D14708～D14711	
9	D14712～D14715	
10	D14716～D14719	
11	D14720～D14723	
12	D14724～D14727	
13	D14728～D14731	
14	D14732～D14735	
15	D14736～D14739	
16	D14740～D14743	
17	D14744～D14747	
18	D14748～D14751	
19	D14752～D14755	
20	D14756～D14759	
21	D14760～D14763	
22	D14764～D14767	
23	D14768～D14771	
24	D14772～D14775	
25	D14776～D14779	
26	D14780～D14783	
27	D14784～D14787	
28	D14788～D14791	
29	D14792～D14795	
30	D14796～D14799	
31	D14800～D14803	
32	D14804～D14807	

	Symbol	Signal name	Refresh cycle	Load cycle	Signal type
0 1	Cd. 340	Command generation axis JOG speed setting		When starting JOG operation	Command device
2	Pr. 348	Command generation axis JOG operation parameter block setting		When starting JOG operation	Command device
3	—	Unusable	—	—	—

POINT

(1) With the Q172DSCPU, the axis No.1 to 16 range is valid.
(2) With the Q172DSCPU, device areas of 17 axes or greater can be used as user devices.
However, if a Q172DSCPU project is replaced with a Q173DSCPU project, it will no longer be able to be used as a user device.

(7) Synchronous encoder axis monitor device list

Axis No.	Device No.
1	D13240～D13259
2	D13260～D13279
3	D13280～D13299
4	D13300～D13319
5	D13320～D13339
6	D13340～D13359
7	D13360～D13369
8	D13380～D13399
9	D13400～D13419
10	D13420～D13439
11	D13440～D13459
12	D13460～D13479

Signal name

	Symbol	Signal name	Refresh cycle	Load cycle	Signal type
0	Md. 320	Synchronous encoder axis current value	Operation cycle		Monitor device
1					
2	Md. 321	Synchronous encoder axis current value per cycle			
3					
4	Md. 322	Synchronous encoder axis speed			
5					
6	Md. 323	Synchronous encoder axis phase compensation amount			
7					
8	Md. 324	Synchronous encoder axis rotation direction restriction amount			
9					
10	Md. 327	Synchronous encoder axis minor error code	Immediate		
11	Md. 326	Synchronous encoder axis major error code			
12	—	Unusable	—	—	—
13					
14					
15					
16					
17					
18					
19					

(8) Synchronous encoder axis control device list

Axis No.	Device No.
1	D14820～D14829
2	D14830～D14839
3	D14840～D14849
4	D14850～D14859
5	D14860～D14869
6	D14870～D14879
7	D14880～D14889
8	D14890～D14899
9	D14900～D14909
10	D14910～D14919
11	D14920～D14929
12	D14930～D14939

Signal name

	Symbol	Signal name	Refresh cycle	Load cycle	Signal type
0	Pr. 326	Synchronous encoder axis phase compensation advance time		Operation cycle	Command device
1					
2	Cd. 320	Synchronous encoder axis control start condition		When starting synchronous encoder axis control	
3	Cd. 321	Synchronous encoder axis control method			
4	Cd. 322	Synchronous encoder axis current value setting address			
5					
6	Cd. 325	Input value for synchronous encoder via device		Operation cycle	
7					
8	—	Unusable	—	—	—
9					

(9) Output axis monitor device list

Axis No.	Device No.
1	D13600～D13629
2	D13630～D13659
3	D13660～D13689
4	D13690～D13719
5	D13720～D13749
6	D13750～D13779
7	D13780～D13809
8	D13810～D13839
9	D13840～D13869
10	D13870～D13899
11	D13900～D13929
12	D13930～D13959
13	D13960～D13989
14	D13990～D14019
15	D14020～D14049
16	D14050～D14079
17	D14080～D14109
18	D14110～D14139
19	D14140～D14169
20	D14170～D14199
21	D14200～D14229
22	D14230～D14259
23	D14260～D14289
24	D14290～D14319
25	D14320～D14349
26	D14350～D14379
27	D14380～D14409
28	D14410～D14439
29	D14440～D14469
30	D14470～D14499
31	D14500～D14529
32	D14530～D14559

Signal name

	Symbol	Signal name	Refresh cycle	Load cycle	Signal type
0 1	Md. 400	Current value after composite main shaft gear	Operation cycle		Monitor device
2 3	Md. 401	Current value per cycle after main shaft gear			
4 5	Md. 402	Current value per cycle after auxiliary shaft gear			
6 7	Md. 422	Main shaft clutch slippage (accumulative)			
8 9	Md. 425	Auxiliary shaft clutch slippage (accumulative)			
10 11	Md. 406	Cam axis phase compensation amount			
12 13	Md. 407	Cam axis current value per cycle			
14 15	Md. 408	Cam reference position			
16 17	Md. 409	Cam axis feed current value			
18	Md. 410	Execution cam No.			
19	—	Unusable	—	—	—
20 21	Md. 411	Execute cam stroke amount	Operation cycle		Monitor device
22 23	Md. 412	Execute Cam axis length per cycle			
24 25 26 27 28 29	—	Unusable	—	—	—

POINT

(1) With the Q172DSCPU, the axis No.1 to 16 range is valid.
(2) With the Q172DSCPU, device areas of 17 axes or greater can be used as user devices.
However, if a Q172DSCPU project is replaced with a Q173DSCPU project, it will no longer be able to be used as a user device.

(10) Output axis control device list

Axis No.	Device No.
1	D15000～D15149
2	D15150～D15299
3	D15300～D15449
4	D15450～D15599
5	D15600～D15749
6	D15750～D15899
7	D15900～D16049
8	D16050～D16199
9	D16200～D16349
10	D16350～D16499
11	D16500～D16649
12	D16650～D16799
13	D16800～D16949
14	D16950～D17099
15	D17100～D17249
16	D17250～D17399
17	D17400～D17549
18	D17550～D17699
19	D17700～D17849
20	D17850～D17999
21	D18000～D18149
22	D18150～D18299
23	D18300～D18449
24	D18450～D18599
25	D18600～D18749
26	D18750～D18899
27	D18900～D19049
28	D19050～D19199
29	D19200～D19349
30	D19350～D19499
31	D19500～D19649
32	D19650～D19799

Signal name

	Symbol	Signal name	Refresh cycle	Load cycle	Signal type
0	Pr. 400	Main input axis No.		When starting synchronous control	Command device
1	Pr. 401	Sub input axis No			
2	Pr. 402	Composite main shaft gear		Operation cycle	
3	—	Unusable	—	—	—
4	Pr. 403	Main shaft gear: Numerator		When starting synchronous control	Command device
5					
6	Pr. 404	Main shaft gear: Denominator			
7					
8	Pr. 405	Main shaft clutch control setting		Operation cycle	
9	Pr. 406	Main shaft clutch reference address setting		When starting synchronous control	
10	Pr. 407	Main shaft clutch ON address		Operation cycle	
11					
12	Pr. 408	Travel value before main shaft clutch ON		When clutch ON conditions established	
13					
14	Pr. 409	Main shaft clutch OFF address		Operation cycle	
15					
16	Pr. 410	Travel value before main shaft clutch OFF		When clutch OFF conditions established	
17					
18	Pr. 411	Main shaft clutch smoothing method		When starting synchronous control	
19	Pr. 412	Main shaft clutch smoothing time constant			
20	Pr. 413	Slippage amount at main shaft clutch ON		When starting clutch ON	
21					
22	Pr. 414	Slippage amount at main shaft clutch OFF		When starting clutch OFF	
23					
24	Pr. 418	Auxiliary shaft axis No.		When starting synchronous control	
25	Pr. 419	Composite auxiliary shaft gear		Operation cycle	
26	Pr. 420	Auxiliary shaft gear: Numerator		When starting synchronous control	
27					
28	Pr. 421	Auxiliary shaft gear: Denominator			
29					
30	Pr. 422	Auxiliary shaft clutch control setting		Operation cycle	
31	Pr. 423	Auxiliary shaft clutch reference address setting		When starting synchronous control	
32	Pr. 424	Auxiliary shaft clutch ON address		Operation cycle	
33					
34	Pr. 425	Travel value before auxiliary shaft clutch ON		When clutch ON conditions established	
35					
36	Pr. 426	Auxiliary shaft clutch OFF address		Operation cycle	
37					
38	Pr. 427	Travel value before auxiliary shaft clutch OFF		When clutch OFF conditions established	
39					
40	Pr. 428	Auxiliary shaft clutch smoothing method		When starting synchronous control	
41	Pr. 429	Auxiliary shaft clutch smoothing time constant			
42	Pr. 430	Slippage amount at auxiliary shaft clutch ON		When starting clutch ON	
43					

Output axis control device list (cont.)

Axis No.	Device No.
1	D15000～D15149
2	D15150～D15299
3	D15300～D15449
4	D15450～D15599
5	D15600～D15749
6	D15750～D15899
7	D15900～D16049
8	D16050～D16199
9	D16200～D16349
10	D16350～D16499
11	D16500～D16649
12	D16650～D16799
13	D16800～D16949
14	D16950～D17099
15	D17100～D17249
16	D17250～D17399
17	D17400～D17549
18	D17550～D17699
19	D17700～D17849
20	D17850～D17999
21	D18000～D18149
22	D18150～D18299
23	D18300～D18449
24	D18450～D18599
25	D18600～D18749
26	D18750～D18899
27	D18900～D19049
28	D19050～D19199
29	D19200～D19349
30	D19350～D19499
31	D19500～D19649
32	D19650～D19799

Signal name

	Symbol	Signal name	Refresh cycle	Load cycle	Signal type
44, 45	Pr. 431	Slippage amount at auxiliary shaft clutch OFF		When starting clutch OFF	Command device
46	Pr. 434	Speed change gear 1 allocation		When starting synchronous control	Command device
47	Pr. 435	Speed change gear 1 smoothing time constant		When starting synchronous control	Command device
48, 49	Pr. 436	Speed change ratio 1: Numerator		Operation cycle	Command device
50, 51	Pr. 437	Speed change ratio 1: Denominator		Operation cycle	Command device
52	Pr. 490	Speed change gear 2 allocation		When starting synchronous control	Command device
53	Pr. 491	Speed change gear 2 smoothing time constant		When starting synchronous control	Command device
54, 55	Pr. 492	Speed change ratio 2: Numerator		Operation cycle	Command device
56, 57	Pr. 493	Speed change ratio 2: Denominator		Operation cycle	Command device
58	Pr. 438	Cam axis cycle unit setting		When starting synchronous control	Command device
59	Pr. 442	Cam axis length per cycle change setting		When starting synchronous control	Command device
60, 61	Pr. 439	Cam axis length per cycle		When starting synchronous control	Command device
62	Pr. 440	Cam No.		When starting synchronous control, when passing cam data 0 point	Command device
63	—	Unusable	—	—	—
64, 65	Pr. 441	Cam stroke amount		When starting synchronous control, when passing cam data 0 point	Command device
66, 67	Pr. 444	Cam axis phase compensation advance time		Operation cycle	Command device
68	Pr. 445	Cam axis phase compensation time constant		When starting synchronous control	Command device
69	Pr. 448	Synchronous control parameter block No.		When starting synchronous control	Command device
70	Pr. 447	Output axis smoothing time constant		When starting synchronous control	Command device
71 to 81	—	Unusable	—	—	—

Output axis control device list (cont.)

Axis No.	Device No.
1	D15000～D15149
2	D15150～D15299
3	D15300～D15449
4	D15450～D15599
5	D15600～D15749
6	D15750～D15899
7	D15900～D16049
8	D16050～D16199
9	D16200～D16349
10	D16350～D16499
11	D16500～D16649
12	D16650～D16799
13	D16800～D16949
14	D16950～D17099
15	D17100～D17249
16	D17250～D17399
17	D17400～D17549
18	D17550～D17699
19	D17700～D17849
20	D17850～D17999
21	D18000～D18149
22	D18150～D18299
23	D18300～D18449
24	D18450～D18599
25	D18600～D18749
26	D18750～D18899
27	D18900～D19049
28	D19050～D19199
29	D19200～D19349
30	D19350～D19499
31	D19500～D19649
32	D19650～D19799

Signal name

	Symbol	Signal name	Refresh cycle	Load cycle	Signal type
82	—	Unusable	—	—	—
83					
84					
85					
86					
87					
88					
89					
90					
91					
92					
93					
94					
95					
96					
97					
98					
99					
100	Pr. 460	Setting method of current value per cycle after main shaft gear		When starting synchronous control	Command device
101	Pr. 461	Setting method of current value per cycle after auxiliary shaft gear			
102	Pr. 462	Cam axis position restoration object			
103	Pr. 463	Setting method of cam reference position			
104	Pr. 464	Cam axis 1 cycle current value setting method			
105	—	Unusable	—	—	—
106	Pr. 465	Current value per cycle after main shaft gear (Initial setting)		When starting synchronous control	Command device
107					
108	Pr. 466	Current value per cycle after auxiliary shaft gear (Initial setting)			
109					
110	Pr. 467	Cam reference position (Initial setting)			
111					
112	Pr. 468	Cam axis current value per cycle (Initial setting)			
113					
114	—	Unusable	—	—	—
115					
116					
117					
118					
119					
120					
121					
122					
123					
124					

Output axis control device list (cont.)

Axis No.	Device No.
1	D15000～D15149
2	D15150～D15299
3	D15300～D15449
4	D15450～D15599
5	D15600～D15749
6	D15750～D15899
7	D15900～D16049
8	D16050～D16199
9	D16200～D16349
10	D16350～D16499
11	D16500～D16649
12	D16650～D16799
13	D16800～D16949
14	D16950～D17099
15	D17100～D17249
16	D17250～D17399
17	D17400～D17549
18	D17550～D17699
19	D17700～D17849
20	D17850～D17999
21	D18000～D18149
22	D18150～D18299
23	D18300～D18449
24	D18450～D18599
25	D18600～D18749
26	D18750～D18899
27	D18900～D19049
28	D19050～D19199
29	D19200～D19349
30	D19350～D19499
31	D19500～D19649
32	D19650～D19799

Signal name

	Symbol	Signal name	Refresh cycle	Load cycle	Signal type
125–129	—	Unusable	—	—	—
130	Cd. 407	Synchronous control change		When requesting simultaneous control change	Command device
131	Cd. 409	Synchronous control change reflection time			
132–133	Cd. 408	Synchronous control change value			
134–149	—	Unusable	—	—	—

POINT

(1) With the Q172DSCPU, the axis No.1 to 16 range is valid.
(2) With the Q172DSCPU, device areas of 17 axes or greater can be used as user devices.
However, if a Q172DSCPU project is replaced with a Q173DSCPU project, it will no longer be able to be used as a user device.

(12) Common device list

Device No.	Signal name	Refresh cycle	Load cycle	Signal type
D704	PLC ready flag request		Main cycle	Command device
D705	Speed switching point specified flag request			
D706	All axes servo ON command request			
D707	Real mode/virtual mode switching request (SV22) *1			
D708	JOG operation start command request			
D709	Unusable	—	—	—
D710	JOG operation simultaneous start axis setting register		When starting	Command device
D711				
D712				
D713				
D714	Axis No. setting register controlled by manual pulse generator 1		When manual pulse generator enable flag	
D715				
D716	Axis No. setting register controlled by manual pulse generator 2			
D717				
D718	Axis No. setting register controlled by manual pulse generator 3			
D719				
D720	Axis 1 — Manual pulse generators 1 pulse input magnification setting register *2, *3			
D721	Axis 2			
D722	Axis 3			
D723	Axis 4			
D724	Axis 5			
D725	Axis 6			
D726	Axis 7			
D727	Axis 8			
D728	Axis 9			
D729	Axis 10			
D730	Axis 11			
D731	Axis 12			
D732	Axis 13			
D733	Axis 14			
D734	Axis 15			
D735	Axis 16			
D736	Axis 17			
D737	Axis 18			
D738	Axis 19			
D739	Axis 20			
D740	Axis 21			
D741	Axis 22			
D742	Axis 23			
D743	Axis 24			
D744	Axis 25			
D745	Axis 26			
D746	Axis 27			
D747	Axis 28			
D748	Axis 29			
D749	Axis 30			
D750	Axis 31			
D751	Axis 32			

Device No.	Signal name	Refresh cycle	Load cycle	Signal type
D752	Manual pulse generator 1 smoothing magnification setting register		When manual pulse generator enable flag	Command device
D753	Manual pulse generator 2 smoothing magnification setting register			
D754	Manual pulse generator 3 smoothing magnification setting register			
D755	Manual pulse generator 1 enable flag request		Main cycle	
D756	Manual pulse generator 2 enable flag request			
D757	Manual pulse generator 3 enable flag request			
D758	Unusable (42 points)	—	—	—
D759				
D760				
D761				
D762				
D763				
D764				
D765				
D766				
D767				
D768				
D769				
D770				
D771				
D772				
D773				
D774				
D775				
D776				
D777				
D778				
D779				
D780				
D781				
D782				
D783				
D784				
D785				
D786				
D787				
D788				
D789				
D790				
D791				
D792				
D793				
D794				
D795				
D796				
D797				
D798				
D799				

*1: Unusable with SV22 advanced synchronous control.
*2: With the Q172DSCPU, the axis No. 1 to 16 range is valid.
*3: With the Q172DSCPU, devices areas for axis 17 and above are unusable.

4.3.4 Special relays

The Q17nDSCPU has 2256 special relays from SM0 to SM2255.
Nine of these are used for positioning control, and their respective applications are fixed as shown in the following tables.

Device No.	Signal name	Refresh cycle	Signal type
SM500	PCPU READY complete flag	Main cycle	Status signal
SM501	Test mode flag		
SM502	External forced stop input flag	Operation cycle	
SM503	Digital oscilloscope executing flag	Main cycle	
SM506	External forced stop input ON latch flag	Operation cycle	
SM508	Amplifier-less operation status flag	Main cycle	
SM510	TEST mode request error flag		
SM512	PCPU WDT error flag		
SM513	Manual pulse generator axis setting error flag		
SM516	Servo program setting error flag		

4.3.5 Special Registers

There are 2256 special registers in the Q17nDSCPU, from SD0 to SD2255.
In addition to special registers used for positioning control, 23 data registers are used as special registers, and their respective applications are fixed as shown in the following tables.

Device No.	Signal name	Refresh cycle	Load cycle	Signal type
SD200	Switch status	Main cycle		Monitor device
SD500 SD501	Real mode axis information register (SV22) *1			
SD502	Servo amplifier loading information	When power turned ON and when performing operation cycle		
SD503				
SD504	Real mode/virtual mode switching error information (SV22) *1	When switching to virtual mode		
SD505				
SD506				
SD508	SSCNET control (Status)	Main cycle		
SD510	Test mode request error information	When making test mode request		
SD511				
SD512	Motion CPU WDT error cause	When motion CPU WDT error occurs		
SD513	Manual pulse generator axis setting error information	When manual pulse generator enable flag ↑		
SD514				
SD515				
SD516	Error program No.	When starting		
SD517	Error item information			
SD522	Motion operation cycle	Operation cycle		
SD523	Operation cycle of the motion CPU setting	When power turned ON		
SD524	Maximum motion operation cycle	Operation cycle		
SD550	System setting error information	System settings When an error occurs		
SD551				
SD560	Operation method	When power turned ON		
SD803	SSCNET control (Command)		Main cycle	Command device

*1: Unusable when performing SV22 advanced synchronous control.

4.4 Motion Devices

Motion registers (#0 to #12287) and a coasting timer (FT) are used as dedicated motion CPU devices.
These can be used for operation control (F/FS) programs or transition (G) programs.
(Direct access is not possible from PLCs, and therefore motion CPUs should be accesses after substituting the PLC device if using at the PLC side.)

4.4.1 Motion registers (#0 to #12287)

	Item	Q173DSCPU/Q172DSCPU
Motion register (#)	No. of points	12288 points (#0 to #12287)
	Data size	16 bits/point
	Latch	Only user devices are latched. (All points are cleared with the latch clear operation.)
	Usable tasks	Normal, event, NMI
	Access	Complete range Read, Write possible

(1) Motion register list

These OS is common for all registers.

Device No.	Application type	Remarks
#0 ⟆	User device (8000 points)	Cleared with the latch clear operation.
#8000 ⟆	Monitor device (640 points)	Cleared only when the power is turned ON or when reset.
#8640 ⟆	Motion error history device (96 points)	Cleared with motion error history clear request flag ON. (Retained when power turned ON, or when reset.)
#8736 ⟆	Product information list device (16 points)	Set when the power is turned ON or when reset.
#8752 ⟆ #12287	System area (3536 points)	Cleared only when the power is turned ON or when reset.

(2) Monitor devices (#8000 to #8639)

Monitor devices store information for each axis. Details of the stored data are as follows.

Axis No.	Device No.	Signal name
1	#8000 to #8019	
2	#8020 to #8039	
3	#8040 to #8059	
4	#8060 to #8079	
5	#8080 to #8099	
6	#8100 to #8119	
7	#8120 to #8139	
8	#8140 to #8159	
9	#8160 to #8179	
10	#8180 to #8199	
11	#8200 to #8219	
12	#8220 to #8239	
13	#8240 to #8259	
14	#8260 to #8279	
15	#8280 to #8299	
16	#8300 to #8319	
17	#8320 to #8339	
18	#8340 to #8359	
19	#8360 to #8379	
20	#8380 to #8399	
21	#8400 to #8419	
22	#8420 to #8439	
23	#8440 to #8459	
24	#8460 to #8479	
25	#8480 to #8499	
26	#8500 to #8519	
27	#8520 to #8539	
28	#8540 to #8559	
29	#8560 to #8579	
30	#8580 to #8599	
31	#8600 to #8619	
32	#8620 to #8639	

	Signal name	Refresh cycle	Signal type
0	Servo amplifier type	When amplifier power turned ON	Monitor device
1	Motor current	Operation cycle of 1.7 [ms] or shorter: operation cycle Operation cycle of 3.5 [ms] or longer: 3.5 [ms]	
2	Motor speed		
3			
4	Command speed	Operation cycle	
5			
6	Zeroing re-travel value	When performing zeroing re-travel	
7			
8	Servo amplifier display Servo error code	Main cycle	
9	Parameter error No.		
10	Servo status 1	Operation cycle of 1.7 [ms] or shorter: operation cycle Operation cycle of 3.5 [ms] or longer: 3.5 [ms]	
11	Servo status 2		
12	Servo status 3		
13	Unusable	-	-
14			
15			
16			
17			
18			
19			

(3) Motion error history devices (#8640 to #8735)

Motion error history devices are shown below.

Device No.	Signal name		Signal direction: Status	Signal direction: Command	Refresh cycle	Load cycle
#8640 to #8651	Information on past 7 errors (Oldest error information)	Motion error history (8 times) (96 points)	○	-	When an error occurs	-
#8652 to #8663	Information on past 6 errors					
#8664 to #8675	Information on past 5 errors					
#8676 to #8687	Information on past 4 errors					
#8688 to #8699	Information on past 3 errors					
#8700 to #8711	Information on past 2 errors					
#8712 to #8723	Information on past 1 errors					
#8724 to #8735	Newest error information					

(4) **Motion error history device error information**

Information for the past eight errors after turning ON the CPU power is stored as history. Numbers #8724 to #8735 contain the latest errors.

Errors when performing SFC control, and all existing minor and major errors, servo errors, servo program errors, and mode switching errors, etc. are tabulated in the history.

When an error occurs, "Motion error detection flag M2039" is also set.

Error information is as follows.

<table>
<tr><th rowspan="2">No.</th><th rowspan="2">Signal name</th><th colspan="2">Details</th></tr>
<tr><th>If error occurs when performing motion SFC control</th><th>If motion control error</th></tr>
<tr><td>+0</td><td>Motion SFC Error program No.</td><td>0 to 255: Motion SFC program No. for which error occurred
-1: If unrelated to motion SFC program</td><td>-1</td></tr>
<tr><td>+1</td><td>Error type</td><td>20: When F/FS
21: When G
22: When K or other (when neither F/FS nor G)
23: When motion SFC diagram</td><td>2: Minor, major errors (command generation axis) (SV22 advanced synchronous control method)
3: Minor, major errors
4: Minor, major errors (virtual servo motor axis) (SV22 virtual mode switching method)
5: Minor, major errors (synchronous encoder axis) (SV22)
6: Errors detected by servo amplifier (MR-J3-B)
7: Servo program setting errors
8: Mode switching errors (SV22 virtual mode switching method)
9: Manual pulse generator axis setting errors
10: Test mode request errors
11: WDT errors
13: Self-diagnostic errors (error code 10000 and below)
14: System setting errors/motion slot error detection
15: Errors detected by servo amplifier (MR-J4-B)
42: Errors detected by SSCNET III/H head unit
50: Safety monitor errors
51: Safety monitor warnings</td></tr>
<tr><td>+2</td><td>Error program No.</td><td>0 to 4095: F/FS, G, K program No.
0 to 255: GSUB program No.
-1: If unrelated to F/FS, G, K, GSUB</td><td>• If error type is "2", "3", "4", or "7"
0 to 4095: Servo program No.
FFFFH: JOG operation
FFFEH: Manual pulse generator
FFFDH: Test mode (zeroing, servo diagnosis, servo startup)
FFEFH: Synchronous control
FFDFH: Speed control
FFDEH: Torque control
FFDDH: Push control
FF00H: All other cases
• If error type is other than "2", "3", "4", or "7"
-1</td></tr>
<tr><td>+3</td><td>Error block No./Motion SFC list/ Line No./Axis No.</td><td>0 to 8191: If error type is "20"or "21", F/FS or G program block No. (line No.)
0to 8188: If error type is "23", motion SFC list line No.
-1: If error type is "22", or error type is "20" or "21" and unrelated to block</td><td>1 to 32: If error type is "2" to "6", relevant axis No.
1 to 8: If error type is "42", relevant SSCNET III/H head unit axis No.
-1: All other cases</td></tr>
</table>

(Go to next page)

(From previous page)

No.	Signal name		Details: If error occurs when performing motion SFC control	Details: If motion control error
+4	Error code		16000 and above	• If error type is as follows "2": D12602+20n or D12603+20n storage error code "3": D6+20n or D7+20n storage error code "4": D802+10n or D803+10n storage error code "5": D1122+10n or D1123+10n storage error code (SV22 virtual mode switching method) D13250+20n or D13251+20n storage error code (SV22 advanced synchronous control method) "6": D8+20n storage error code "7": SD517 storage error code "8": SD504 storage error code "9", "10": -1 "11": SD512 storage error code "13", "14": SD0 storage error code "15": #8008+20n storage error code "42": SSCNET III/H head unit monitor device alarm/warning No. storage error code "50", "51": SD32 storage error code
+5	Error Occur-rence date/ time	YY/ MM	Sets the clock data (SD210, SD211, SD212) at the time of the error. (BCD code, year is last 2 digits of calendar year)	
+6		DD/Hr		
+7		Min/ Sec		
+8	Error setting data info.		b15 b14 b13 b12 b11 b10 b9 b8 b7 b6 b5 b4 b3 b2 b1 b0 Error setting data presence 0: No 1: Yes	b15 b14 b13 b12 b11 b10 b9 b8 b7 b6 b5 b4 b3 b2 b1 b0 Degree axis speed x 10 setting 0: Disable 1: Enable • Sets the status when an error occurs when the error setting data unit is "11: Control unit (speed data)", and the control unit is "10: degree". Control unit/display method • Sets the control unit when the error setting data units are "01: Axis unit, output module unit", "10: Control unit (address data, radius arc interpolation error tolerance error", and "11: Control unit (speed data)". 00: mm 01: inch 10: degree 11: PLS * The virtual servo motor axis is "11: PLS fixed". • Sets the display method when the error setting data unit is "00: none". 00: Decimal notation with symbol 01: Decimal notation with no symbol 10: Hexadecimal notation (lower bit 4-digit display) 11: Hexadecimal notation (8-digit display) Error setting data unit 00: None 01: Axis unit, output module unit (when output module error occurs) 10: Control unit (address data, radius arc interpolation error tolerance error) 11: Control unit (speed data) Error setting data presence 0: No 1: Yes
+9	Unusable		-	
+10 +11	Error setting data		• Stores error details code. *1 • Fixed at 0 if error with no details code.	• Setting data that caused error • If error type is one of the following "15": #8009＋20n storage parameter error No. (hexadecimal notation) "42": Fixed at 0 "50", "51": SD33 (safety signal monitor error details data) is store.

*1: If command execution fails for motion SFC program synchronous control dedicated functions, a details code is output to both the motion error history device error code and the error setting data.

(5) Motion error detection flag (M2039)

The motion error detection flag (M2039) turns ON when all errors detected by the motion CPU occur.
When an error occurs, set the motion error detection flag (M2039) for the error device with the following procedure.

(a) Set the error code for each axis or each error device.

(b) Turn ON the error detection signal for each axis or each error.

(c) Set the motion error detection flag (M2039) for the above "motion error history devices (#8640 to #8735)".

(d) Turn the motion error detection flag (M2039) ON.

After reading error history with the "Motion error detection flag (M2039)" ON, reset the "Motion error detection flag (M2039)" in the user program. The "Motion error detection flag (M2039)" will turn ON again for subsequent new errors.

POINT

- If turning the "Motion error detection flag (M2039)" OFF, check the error content, eliminate the cause of the error, and then turn it OFF at the user side.
  Turning M2039 OFF clears self-diagnostic error information other than for stop errors.
- Set clock data and the clock data read request (SM801) in the user program.

(6) Error setting when servo warnings occur

Set whether to output an error to the MT Developer2 motion error history and self-diagnostic errors when a servo warning occurs.
Set in the system basic settings in the system settings.
Refer to the "Q173D(S)CPU/Q172D(S)CPU Motion Controller Programming Manual (Common Edition)" for details.

4.5 Coasting Timer (FT)

Motion device	Item	Specification
Coasting timer (FT) *1, *2	No. of points	1 point (FT)
	Data size	32 bits/point (-2147483648 to 2147483647)
	Latch	No latch. The timer is reset to 0 when the power is turned ON, and counting is continued.
	Usable tasks	Normal, event, NMI
	Access	Read only possible
	Timer specifications	888 μs timer (1 is added to the current value (FT) every 888 μs.)

*1: Use devices SD720 or SD721 for the 444 μs coasting timer.
*2: Use devices SD722 or SD723 for the 222 μs coasting timer.

Chapter 5 Motion SFC Programs

This section describes the configuration and each element of motion SFC programs.
Previously, machine operations were managed at the PLC CPU side, and the starting and stopping of motion SFC programs was controlled at the motion CPU side with start and stop commands from the PLC. Consequently, the time taken from the point command conditions were established until commands were issued was delayed by at most the number of sequences taken to perform a single scan, and the resultant variations in this time restricted applications which demanded responsiveness and short tact time.
With the Q Series motion controller, programs at the motion side are described with an SFC (Sequential Function Chart), enabling the control of machine operations. Furthermore, it is now also possible to control events that require program execution when interrupts are input from external sensors.

5.1 Features

(1) By breaking up machine sequential operations into individual steps, anyone can create easy-to-understand programs in flowchart format, resulting in improved maintenance.

(2) Transition conditions are identified and positioning started at the motion CPU side, meaning no variations in the response time that can be influence sequence scan time.

(3) With the motion SFC step processing method (active steps only executed), high-speed processing, and high-speed response processing can be realized.

(4) In addition to positioning control, numerical operations and device SET/RST, etc. can also be processed at the motion CPU side, leading to reduced tact time without involving the PLC CPU.

(5) Commands can be issued to servo amplifiers when start conditions are established with a transition conditions description unique to motion SFC.

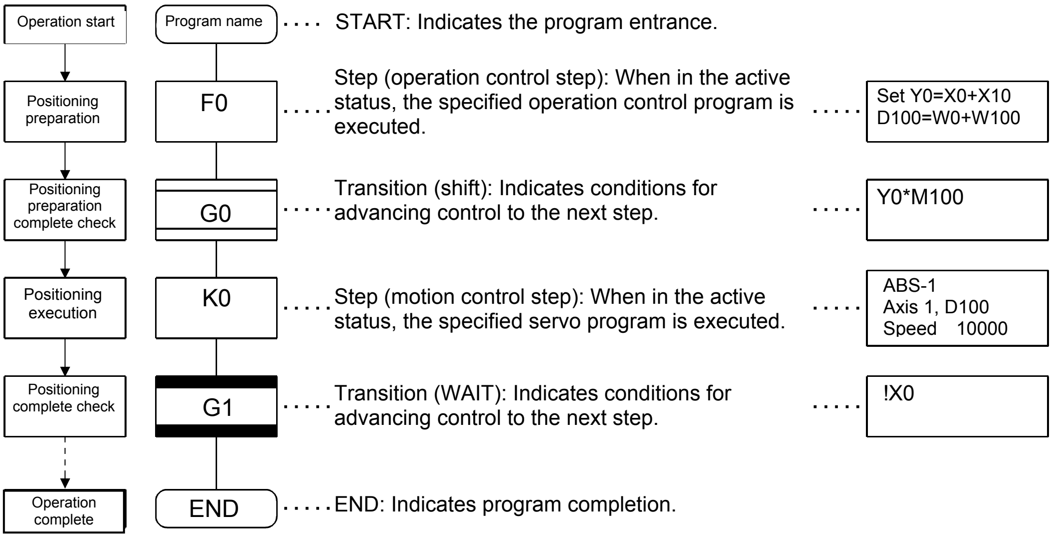
(6) Operation can proceed to the next step without waiting for positioning to be completed after starting with a transition condition description unique to motion SFC.

(7) Motion SFC programs that respond to interrupt inputs from external sources can be executed.

(8) Motion SFC programs can be executed at regular intervals (min. 0.22 ms: when using Q17☐DSCPU) by synchronizing with the motion operation cycle.

5.2 Motion SFC Program Configuration

Motion SFC programs are configured by START, step, transition, and END components and so on as shown below.



Operation for the above motion SFC program when started is as follows.

(1) The step (F0) status becomes active, and the operation specified at the step (F0) is executed (positioning preparation). An active status step is known as an active step.

(2) A check is carried out to determine whether the conditions specified at the transition (G0) have been established (whether the positioning program can be started), the active step (F0) becomes inactive when conditions are established, and the next step (K0) becomes active (servo program K0 is started).

(3) A check is carried out at the transition (G1) to ensure that step (K0) operation is complete (servo program K0 positioning complete), and control advances to the next step when operation is complete (conditions established).

(4) As the active step advances as described in (1) to (3) above, control is executed and then completed with END.

POINT

The number of steps that can simultaneously be active steps in all motion SFC programs is 256 or less.
If 256 is exceeded, a Motion SFC error [16120] occurs.
The motion SFC program symbols are as follows.
F/FS: operation control, K: positioning control, G: judgment

5.3 SFC Diagram Symbol List

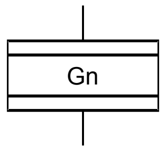
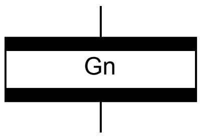
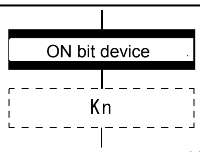
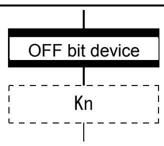
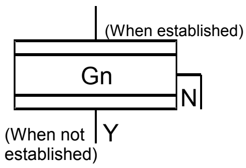
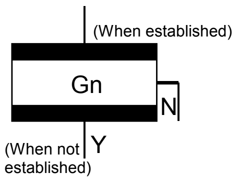
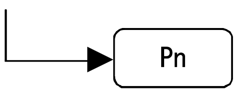
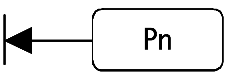
The parts that form the component elements of the motion SFC program are as follows. The motion SFC program expresses the operation order and transition control by joining these parts with a directed line.

Category	Name	Symbol (Code size (bytes))	List expression	Function
Program Start/end	START	Program name (0)	Program name	• Indicates the program entrance with the program name. • This program name is specified when calling sub-routines. • Limited to one per program.
	END	END (8)	END	• Indicates the end (exit) of the program. • When a sub-routine is called, operation returns to the program from which the sub-routine was called. • Multiple ENDs can be set within a single program, and can be set even if none.
Steps	Motion control step	Kn (8)	CALL Kn	• Starts servo program Kn (K0 to K4095).
	Single execution type Operation control step	Fn (8)	CALL Fn	• Executes operation control program Fn (F0 to F4095) once.
	Scan execution type operation control step	FSn (8)	CALL FSn	• Executes operation control program FSn (FS0 to FS4095) repeatedly until the next transition condition is established.
	Sub-routine call/start step	Program name (8)	GSUB program name	• If WAIT follows GSUB, a "Sub-routine call" condition occurs, and control advances to the specified program. Control is returned to the program from which the sub-routine is called when END is executed. • If GSUB is followed by other than WAIT, a "Sub-routine start" condition occurs, the specified program is started, and control advances to the next (below) program. The start source program and start destination program are executed simultaneously, and the start destination program is exited when END is executed.
	Clear step	CLR Program name (8)	CLR program name	• Execution of the specified program currently running is stopped, and the program is exited. By restarting the program after exiting, it starts from the initial step (start step). • If the specified program is currently "calling a sub-routine", execution of the sub-routine program is also stopped. • If the specified program is at a point after "starting the sub-routine", execution of the sub-routine program is not stopped. • If a clear is performed for the "called sub-routine", execution of the specified sub-routine is stopped, control returns to the program from which the sub-routine was called, and then proceeds to the next.

POINT

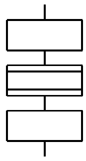
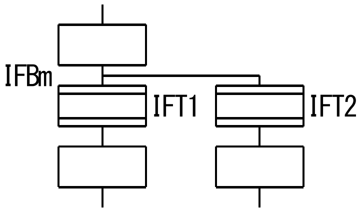
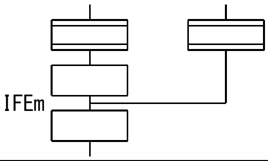
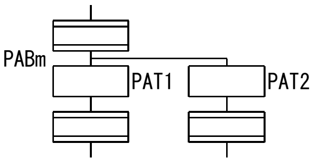
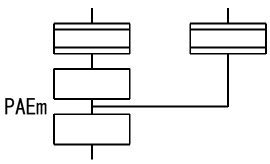
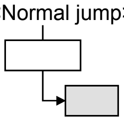
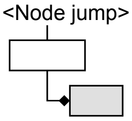
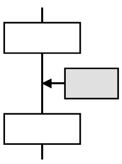
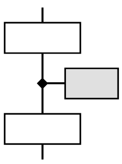
Comments can be set for each symbol in SFC diagram steps, transitions, etc.

- Program start/end comments cannot be set.
- Step/transition comments: max. 80 half-width (40 full-width) characters, 20 characters displayed in 4 lines
- Jump/pointer comments: max. 64 half-width (32 full-width) characters, 16 characters displayed in 4 lines

Category	Name	Symbol (Code size (bytes))	List expression	Function
Transi- tion	Shift (Read-ahead transition)	 (8)	SFT Gn	• If the previous step is a motion control step, processing proceeds to the next step without waiting for the completion of motion operation when transition condition Gn (G0 to G4095) is established. • If the previous step is an operation control step, processing proceeds to the next step following operation execution when the transition condition is established. • If the previous step is a sub-routine call/start step, processing proceeds to the next step without waiting for the completion of sub-routine operation when the transition condition is established.
	WAIT	 (8)	WAIT Gn	• If the previous step is a motion control step, processing proceeds to the next step without waiting for the completion of motion operation when transition condition Gn (G0 to G4095) is established. • If the previous step is an operation control step, processing proceeds to the next step following operation execution when the transition condition is established. (Same operation as shift) • If the previous step is a sub-routine call/start step, processing waits for completion of sub-routine operation, and then proceeds to the next when the transition condition is established.
	WAITON	 (14)	WAITON bit device	• Start preparations are carried out for the next motion control step, and a command is issued immediately when the specified bit device turns ON. • Always set a one-to-one pair with the motion control step.
	WAITOFF	 (14)	WAITOFF bit device	• Start preparations are carried out for the next motion control step, and a command is issued immediately when the specified bit device turns OFF. • Always set a one-to-one pair with the motion control step.
	Shift Y/N		IFBm IFT1 SFT Gn : JMP IFEm IFT2 SFT Gn+? : JMP IFEm IFEm	• If the previous step is a motion control step, processing proceeds to the step below without waiting for the completion of motion operation when transition condition Gn (G0 to G4095) is established, and when the condition is not established, processing proceeds to the step connected from the right. • If the previous step is an operation control step, processing proceeds to the step below following operation execution when the transition condition is established, and when the condition is not established, processing proceeds to the step connected from the right. • If the previous step is a sub-routine call/start step, processing proceeds to the step below without waiting for the completion of the sub-routine operation when the transition condition is established, and when the condition is not established, processing proceeds to the step connected from the right.
	WAIT Y/N		IFBm IFT1 WAIT Gn : JMP IFEm IFT2 WAIT Gn+? : JMP IFEm IFEm	• If the previous step is a motion control step, processing waits for completion of motion operation, and proceeds to the next step when transition condition Gn (G0 to G4095) is established, and when the condition is not established, processing proceeds to the step connected from the right. • If the previous step is an operation control step, processing proceeds to the step below following operation execution when the transition condition is established, and when the condition is not established, processing proceeds to the step connected from the right. (Same operation as shift) • If the previous step is a sub-routine call/start step, control waits for completion of the sub-routine, and proceeds to the step below when the transition condition is established, and when the condition is not established, processing proceeds to the step connected from the right.
Jump	Jump	 (14)	JMP Pn	• Controls jumps to specified pointer Pn (P0 to P16383) inside the self program.
Pointer	Pointer	 (8)	Pn	• Indicates the jump destination pointer (label). • Pointers can be set for steps, transitions, branch points, and nodes. • P0 to P16383 can be set for a single program. Numbers may overlap with those in other programs.

5.4 Branch and Node Diagram List

SFC diagrams show branch and node patterns used to specify the flow of steps and transitions.

	Name (code size) (bytes))	SFC symbol		List expression	Function
Basic shape	Series transitions (Size of each symbol)			Based on list expression corresponding to SFC diagram symbols shown in 5.2.	• Processes each step, transition connected in series in order from the top. • Steps and transitions do not have to be aligned alternately. • If transitions are omitted, unconditional shift transition processing is performed.
	Selection branches ((No. of branches + 2) × 10)			CALL Kn IFBm IFT1 SFT Gn CALL Fn : JMP IFEm IFT2 SFT Gn’ CALL Fn’ : (JMP IFEm) IFEm CALL Fn’’	• After executing the step or branch immediately before the branch, the route for which transition conditions are established first is executed. • The start of the branch destination name for the selection branch must be a transition, is limited to all SHIFTs or all WAITs. (If SHIFTs and WAITs are mixed, the branch will be a parallel branch.)
	Selection nodes (8)				• Following processing of the branched route with the selection branch, processing proceeds to the node. • Either steps or transitions may be used immediately before or after nodes.
	Parallel branches (No. of branches × 22 + No. of nodes × 2 + 12)			SFT Gn PABm PAT1 CALL Fn SFT Gn’ : JMP PAEm PAT2 CALL Fn’ SFT Gn’’ : (JMP PAEm) PAEm CALL Fn’’ :	• Multiple routes (steps) connected in parallel are executed simultaneously. • The start of the branch destination name for the parallel branch may be either a step or a transition.
	Parallel node (8)				• Completion of execution of each branched route with a parallel branch waits at a node, and processing proceeds to the next step when execution is complete for all routes. • Either steps or transitions may be used immediately before or after nodes. • If the step immediately before the node is an FS step, scanning is performed even while waiting. Scanning is not performed after waiting is complete.
	Jump transition (Size of each symbol)	<Normal jump> 	<Node jump> 	CALL Fn JMP Pn	(1) Normal jump • After executing the previous step or transition, control jumps to execution of the specified pointer Pn inside the self program. • It is possible to jump to either a step or a transition. • Even if jumping from an FS step to a transition, scanning is performed while the jump destination transition condition is established. (2) Node jump • If jumping to another route within a parallel branch after a parallel branch, a "node jump" is made, and the system awaits execution at the jump destination.
				CALL Fn’ Pn CALL Kn	

5.5 Motion SFC Program Name

The "motion SFC program name" is set individually for motion SFC program No. 0 to No. 255.
The motion SFC program name is set within 16 half-width characters (8 full-width characters). Specify this motion SFC program name in "sub-routine call/ start steps (GSUB)", and "clear steps (CLR)".

POINT

(1) The motion SFC program can be set to a random number between 0 and 255.
(2) "$ (half-width)" cannot be set for the first character of the motion SFC program name.
(3) "\ / : ; ,. * ? " < > | (half-width)" cannot be set in the motion SFC program name.

5.6 Steps

5.6.1 Motion control steps

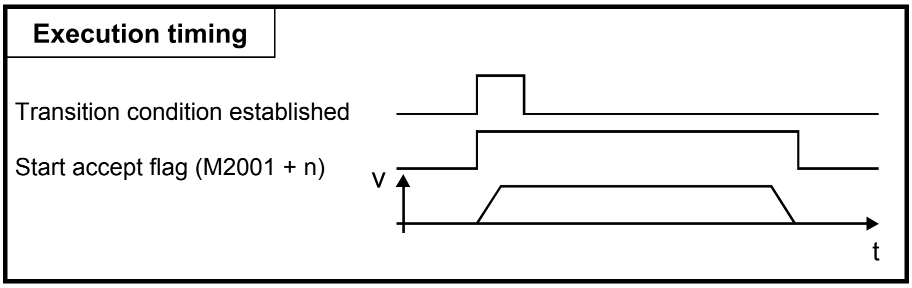
Motion control steps are used to start servo program Kn.

Name	Symbol	Function
Motion control step	Kn	Motion control steps are used to start servo program Kn. Specification range: K0 to K4095

(1) Operation description

(a) The start accept flag for the axis specified in the specified servo program Kn turns ON.

(b) The specified servo program Kn is started.



(2) Error

A Motion SFC error [16200] occurs when the specified servo program Kn does not exist, and execution of the motion SFC program is stopped the moment this error is detected.

(3) Precautions

(a) If changes are made to the current values in the motion SFC program, specify the CHGA command in the servo program, and then call it with the motion control step.

(b) Even if a minor error/major error occurs and an error stop condition occurs at the servo program when the servo program specified with the motion control step is started or while starting, execution of the motion SFC program continues. If wishing to stop the motion SFC program when an error is detected, insert an error detection condition in the transition (transition condition).

5.6.2 Operation control steps

Operation control steps are used to execute operation control program Fn/FSn.

Name	Symbol	Function
Operation control step	Fn/FSn	Operation control steps are used to execute operation control program Fn/FSn. Specification range: F0 to F4095/FS0 to FS4095

(1) Operation description

(a) One-time execution type operation control step Fn
Executes the specified operation control program Fn (n = F0 to F4095) once.

(b) Scan execution type operation control step FSn
Executes the specified operation control program FSn (n = 0 to 4095) repeatedly until the next transition condition is established.

(2) Error

A Motion SFC error [16201] occurs when the specified operation control program Fn/FSn does not exist, and execution of the motion SFC program is stopped the moment this error is detected.

(3) Precautions

(a) Even if an operation error, etc. occurs during operation control program execution, execution of the motion SFC program continues.

5.6.3 Sub-routine call/start steps

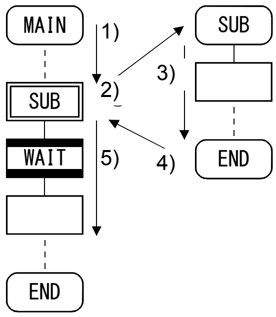
Sub-routine call/start steps are used to call or start motion SFC programs for the specified program name.

Name	Symbol	Function
Sub-routine Call/start step	Program name	Sub-routine call/start steps are used to call motion SFC programs for the specified program name.

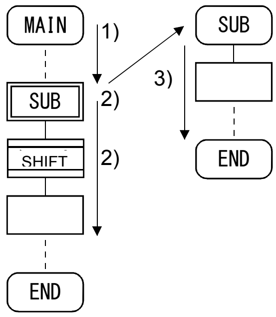
(1) Operation description

(a) Sub-routine call/start steps are used to call or start motion SFC programs for the specified program name.

(b) Control differs depending on the type of the transition linked after the sub-routine call/start step.
- If WAIT: The sub-routine is called.
- If other than WAIT: The sub-routine is started.



If WAIT (sub-routine call)



If other than WAIT (sub-routine start)

(2) Errors

An error occurs in the following cases and execution of the motion SFC program is stopped.

(a) A Motion SFC error [16005] occurs if the specified motion SFC program does not exist when a sub-routine is called/started, and execution of the motion SFC program from which the call/start originated is stopped the moment this error is detected.

(b) A Motion SFC error [16006] occurs if the called/started motion SFC program has already been started when a sub-routine is called/started, and execution of the motion SFC program from which the call/start originated is stopped the moment this error is detected.

(c) A Motion SFC error [16110] occurs if a self program is called/started when a sub-routine is called/started, and execution of the motion SFC program from which the call/start originated is stopped the moment this error is detected.

(d) When the sub-routine called/started when calling/starting a sub-routine is motion SFC program 1 (called/start program) in motion SFC program 2 called/started from motion SFC program 1, Motion SFC error [16111] occurs, and motion SFC program 2 from which the called/started originated is stopped the moment this error is detected.

(3) Precautions

(a) There are no restrictions on sub-routine call/start nesting depth.

(b) With sub-routine starting, processing of the motion SFC program from which the start originated continues even if an error stop occurs for the start destination motion SFC program.

(c) With sub-routine calling, when an error stop occurs for the call destination motion SFC program, execution of the motion SFC program from which the call originated is also stopped at the same time.

5.6.4 Clear Steps

Clear steps are used to stop execution of motion SFC programs for the specified program name.

Name	Symbol	Function
Clear step	CLR Program name	Clear steps are used to stop execution of motion SFC programs for the specified program name.

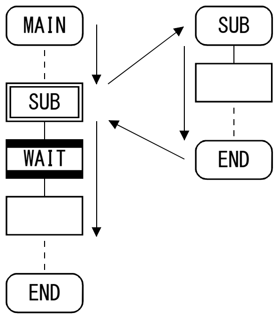
(1) Operation description

(a) Execution of the specified program currently running is stopped.

(b) Even if the motion SFC program for which the clear step is specified is set to start automatically, it will not automatically start again after stopping.

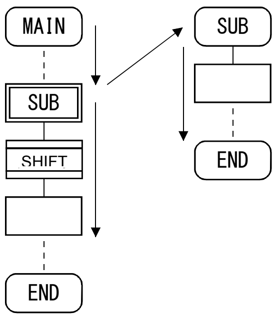
(c) The specified program can also be a self program.

(d) If the specified program is currently calling a sub-routine, execution of the sub- routine program being called is also stopped.



If a "sub-routine is called" as shown in the diagram on the left:
- When MAIN is cleared:
  Execution of the both MAIN and sub-routine (SUB) is stopped even if the sub-routine (SUB) is currently running.
- When the sub-routine (SUB) is cleared:
  Execution of the sub-routine (SUB) is stopped if currently running, and processing returns to MAIN.

(e) If the specified program is at a point after starting the sub-routine, processing of the started sub-routine program continues.



If a "sub-routine is started" as shown in the diagram on the left:
- When MAIN is cleared:
  Execution of MAIN is stopped even if the sub-routine (SUB) is currently running, however, processing of the started sub-routine (SUB) continues.
- When the sub-routine (SUB) is cleared:
  Execution of only the sub-routine (SUB) is stopped if currently running.

(f) If the servo program started from the specified program is currently being started, processing of the servo program continues.

(g) If waiting for conditions to be established at the WAITON/WAITOFF+ motion control step, the system waits for conditions to be established and then executes the servo program. If the servo program is not executed, enter a stop command separately for the relevant axis.

(2) Error

If the motion SFC program specified in the clear step does not exist, a Motion SFC error [16203] occurs.

(3) Precautions

(a) When the motion SFC program specified in the clear step has not been started, no error occurs and the condition is ignored.

(b) Even if execution of the motion SFC program is stopped with the clear step, output is maintained.

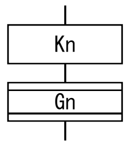
(c) If stopping the axis that is currently operating in conjunction with execution of the clear step, enter a stop command for the relevant axis separately.

5.7 Transition

Conditional expressions and operational expressions can be described in transitions. The operational expression described here is executed repeatedly until the transitional condition is established.

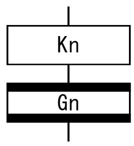
(1) Operation description

(a) Motion control step + SHIFT



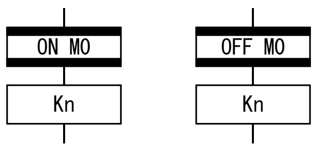
- Processing proceeds to the next step when transition condition Gn is established without waiting for the completion of operation of servo program Kn started with the motion control step.

(b) Motion control step + WAIT



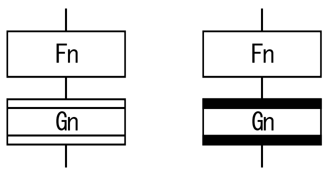
- Processing waits for the completion of operation of servo program Kn started with the motion control step, and then proceeds to the next step when transition condition Gn is established.
- No condition for the completion of operation of servo program Kn is required in transition condition Gn.
- Even if an error stop occurs when the started servo program Kn is started or while it is starting, the system deems that operation is complete.

(c) WAITON/WAITOFF + motion control step



- Processing starts immediately when the specified bit device for WAITON/WAITOFF turns ON/OFF.

(d) Combination with operation control step



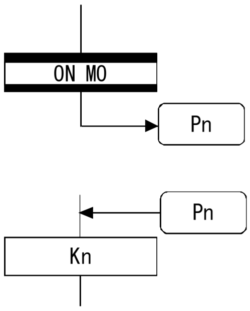
- The same operation is performed for both WAIT and SHIFT, and after executing operation control program Fn, processing proceeds to the next step when transition condition Gn is established.
- In the case of operation control steps, the same operation is performed for both WAIT and SHIFT, and after executing operation control program Fn, processing proceeds to the next step when transition condition Gn is established.

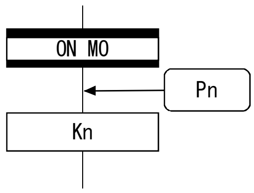
(2) Precautions

(a) Always set a one-to-one pair with the motion control step. If the step after WAITON/WAITOFF is not a motion control step, execution of the motion SFC program is stopped the moment an error is detected.

(b) When the jump destination immediately after WAITON/WAITOFF is a motion control step, no error occurs. (See lower left diagram.)

(c) It is possible for a pointer to exist immediately after WAITON/WAITOFF. (See lower right diagram.)

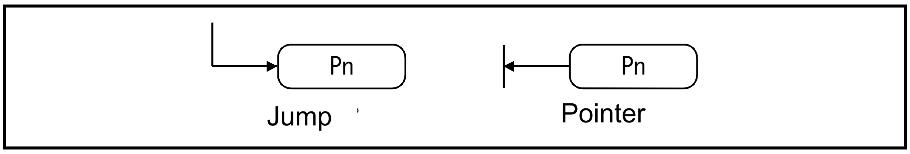




(d) If a minor/major error occurs when starting the servo program specified in the motion control step, preventing the program being started, execution of the motion SFC program continues regardless of the WAITON/WAITOFF bit device status, and processing proceeds to the next step. If wishing to stop the motion SFC program when an error is detected, insert an error detection condition in the next transition (transition condition).

(e) The following commands can be used with motion control steps used in combination with WAITON/WAITOFF.
(Linear interpolation control, circular interpolation control, helical interpolation control, speed switching control, fixed-pitch feed control, constant speed control, high-speed oscillating, fixed position stop speed control)

5.8 Jumps and Pointers

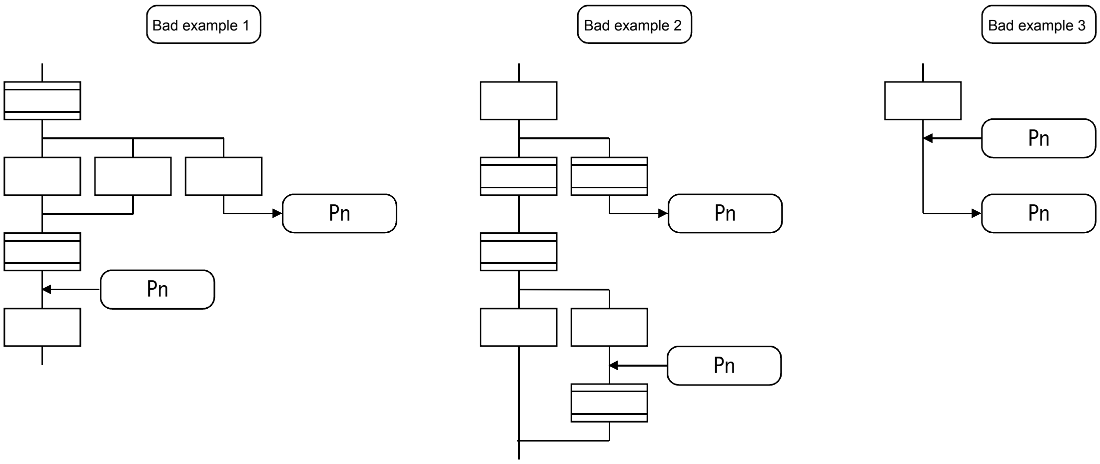


(1) Operation description

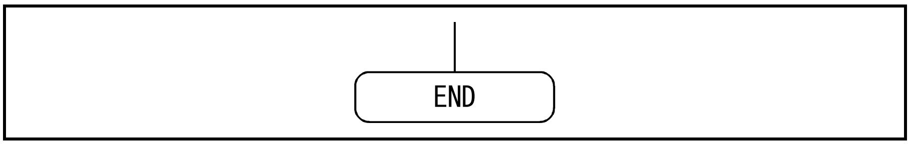
(a) Jumps are used to jump to specified pointer Pn inside the self program.
(b) Pointers can be set for steps, transitions, branch points, and nodes.
(c) Pointer Pn can be set from P0 to P16383 for a single program.

(2) Precautions

(a) It is not possible to set the kind of jumps that break from inside parallel branches to parallel nodes.
(Bad example 1 below)
(b) It is not possible to set jumps inside parallel branches to parallel nodes from outside parallel branches to parallel nodes. (Bad example 2 below)
(c) Labels and jumps cannot be set consecutively. (Bad example 3 below)



5.9 END



(1) Operation description

(a) END is used to exit the program.
(b) When a sub-routine is called, processing returns to the motion SFC program from which the sub-routine was called.

(2) Precautions

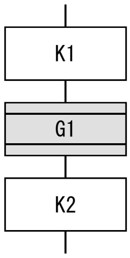
(a) Multiple ENDs can be set within a single program.
(b) An END cannot be set between a parallel branch and node.
(c) Output is maintained even after exiting a motion SFC program with END.

5.10 Branches and Nodes

5.10.1 Series transitions

Series transitions are used to execute steps or transitions directly below those connected in series.

(1) If wishing to start a servo program or sub-routine, and proceed to the next step without waiting for the completion of operation:
Set a SHIFT in the transition.
In such cases, the transition (SHIFT) can be omitted.
If transitions are omitted, unconditional shift transition processing is performed.

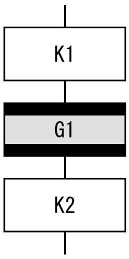


Servo program K1 is started.

Proceeds to the next step when the condition set in transition G1 is established without waiting for the completion of the servo program K1 operation.

Servo program K2 is started.

(2) If wishing to start servo program or sub-routine, and proceed to the next step upon the completion of operation:
Set a WAIT in the transition.



Servo program K1 is started.

Proceeds to the next step when the start axis in servo program K1 stops (start accept flag OFF), and condition set in transition G1 is established.

Servo program K2 is started.

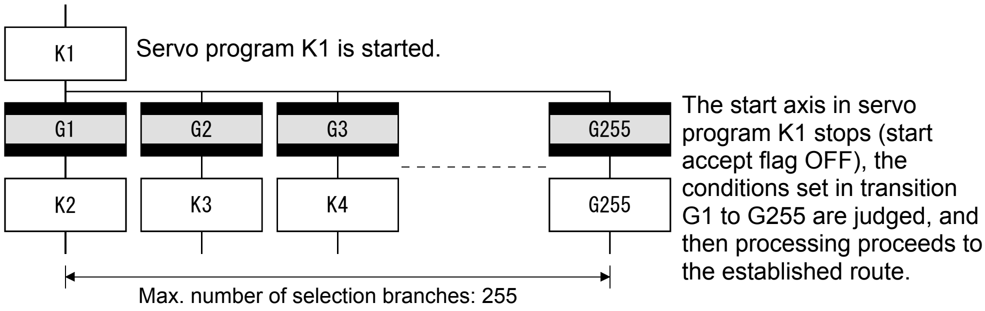
5.10.2 Selection branches and selection nodes

(1) Selection branches

Selection branches are used to judge the conditions for multiple transitions connected in series, and execute only the route for which conditions are established quickest.

Transitions are restricted to all SHIFT or all WAIT.

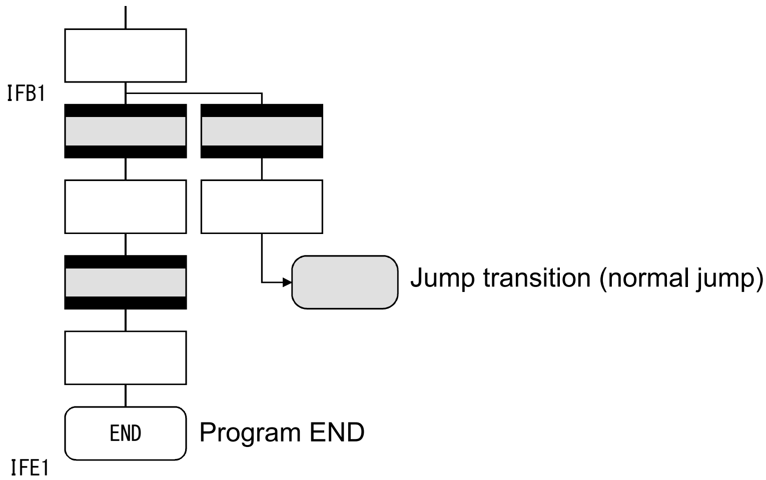
(Example) If WAIT



POINT
(1) The judgment of transition conditions is not necessarily performed in order from left to right.

(2) Selection nodes

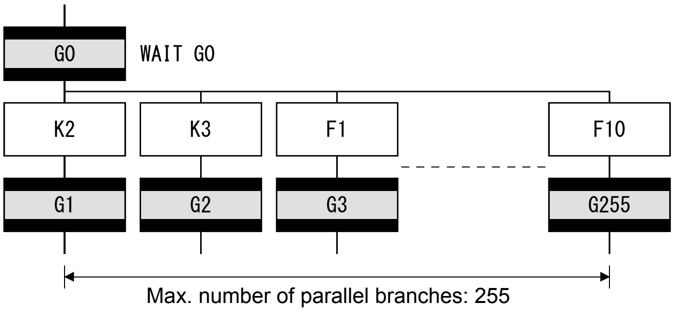
Selection nodes are used after selection branches if connecting to a single route again after completing the processing of each route, however, it is also possible to set not to be joined as shown below.



5.10.3 Parallel branches and parallel nodes

(1) Parallel branches

Multiple steps connected in parallel are executed simultaneously. The start of the parallel branch destination may be either a step or a transition.

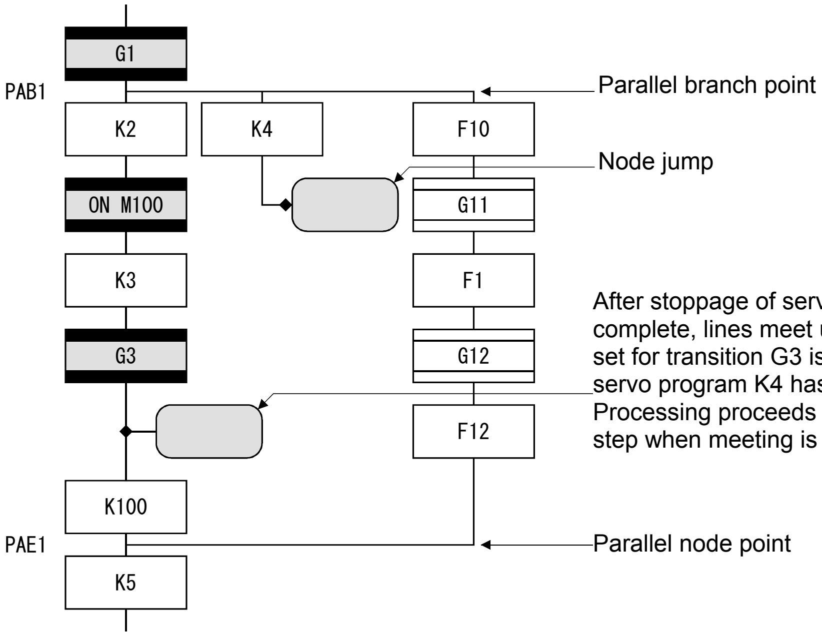


After operation for the previous step is complete, steps K2 to F10 connected in parallel are executed when the condition set for transition G0 is established, and each route is then executed up to the parallel node point.

POINT
A "SHIFT" or "WAIT" may also be set for transitions immediately before parallel branches. Neither "WAITON" nor "WAITOFF" can be set.

(2) Parallel nodes

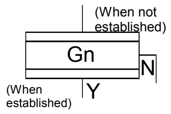
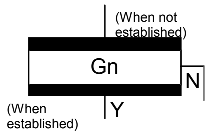
If using parallel branches, always connect them to parallel nodes.
Jumps to other branch routes can be set between parallel branches and parallel nodes.
In such cases, the jump destination is a midway parallel node point (node jump).
It is not possible to set jumps that break from between parallel branches and parallel nodes.



After stoppage of servo program K3 is complete, lines meet until the condition set for transition G3 is established, and servo program K4 has been started. Processing proceeds to the next (below) step when meeting is complete.

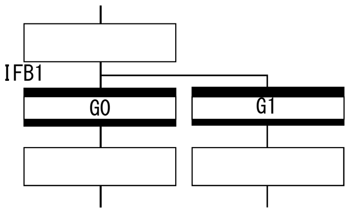
5.11 Y/N Transitions

If branching a route when transition conditions have or have not been established, it is helpful to use a "SHIFT Y/N transition" or "WAIT Y/N transition"

Name	Symbol	Function
SHIFT Y/N transition		• Processing proceeds to the step below when the transition condition set in Gn is established, and when the condition is not established, processing proceeds to the step connected from the right. • The difference between "SHIFT Y/N" and "WAIT Y/N" is the same as the difference between "SHIFT" and "WAIT".
WAIT Y/N transition		

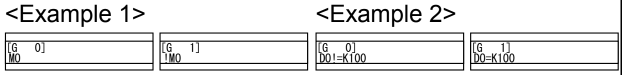
In this example, it has been made easy to describe a selection branch program for two routes as follows.

<Y/N transition not used>

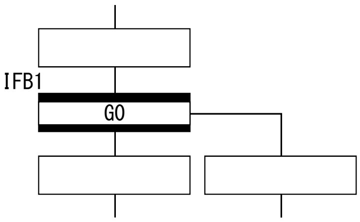


- G0 and G1 programs are restricted to cases in which the affirmation and negation in the conditional expression part differs only.

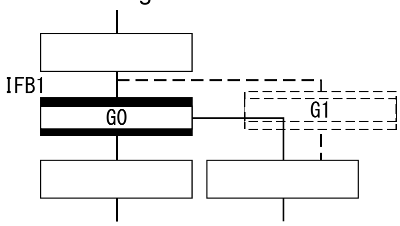
<Example 1> <Example 2>



<Y/N transition used>



- Set the G0 program shown in <Example 1> and <Example 2> above in the G0 program.
- The motion SFC program list and code following conversion will be the same as with the previous description. (The motion SFC diagram expression will differ only.) Consequently, when editing the program, "an automatic search for a vacant G No. is performed, a program for which the conditional expression portion is logically negated is generated automatically", and two G programs are occupied.
  If a Y/N transition is deleted when "editing the program", the automatically generated G program (G1 below) is not deleted. If deletion is necessary, delete at the "Program use list".

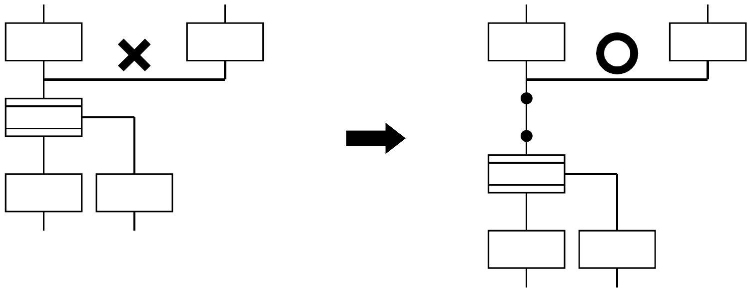


(2) Precautions

(a) If linking immediately before "SHIFT Y/N" or "WAIT Y/N", place a "consecutive node - branch" in between.

• It is not possible link directly to "SHIFT Y/N" or "WAIT Y/N".

• Place a "consecutive node - branch" in between.



5.12 Task Operation

The timing at which motion SFC programs are executed can be set for each program in the program parameters with a single task. Tasks are largely divided into three types as shown in the following table.

Task type	Details
Normal tasks	Executed during motion CPU main cycle (spare time).
Event tasks	1. Executed at fixed cycles (0.22 ms, 0.44 ms, 0.88 ms, 1.77 ms, 3.55 ms, 7.11 ms, 14.2 ms). 2. Executed when the input set for the event task factor from among external interrupts (16 in QI60) is turned ON. 3. Executed with interrupt from PLC.
NMI tasks (Non-Maskable Interrupt)	Executed when the input set for the NMI task factor from among external interrupts (16 for QI60) is turned ON.

POINT

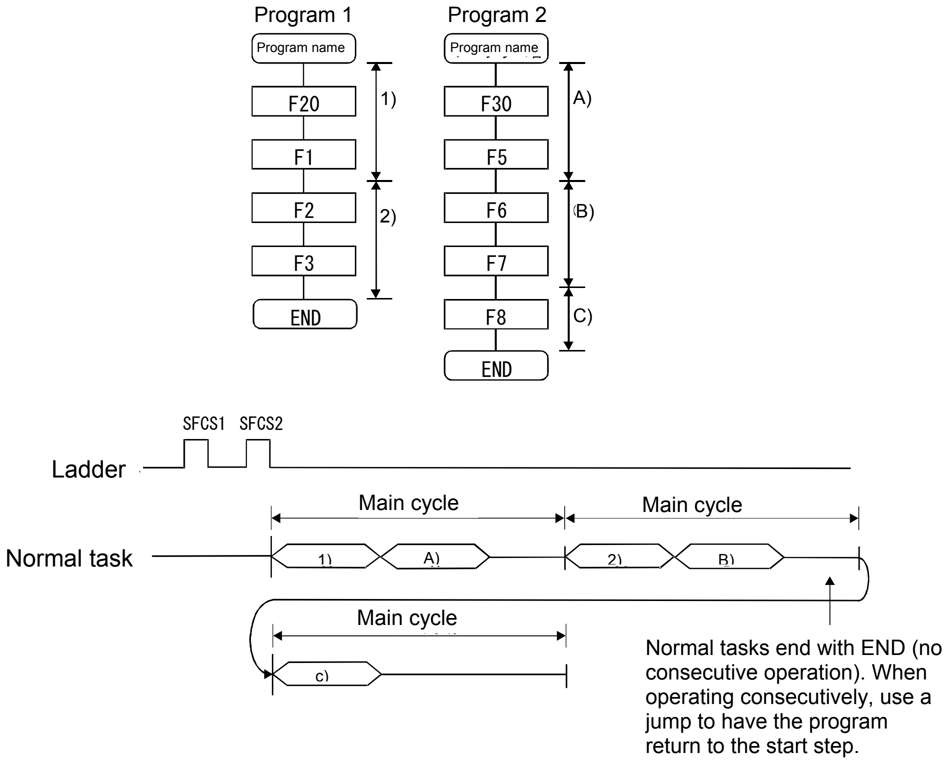
If executing event tasks in 0.22 ms fixed cycles, set "0.2 ms" for the operation cycle time in the MT Developer2 system basic settings.

(1) Normal tasks

[Operation description]

Motion SFC programs are executed during motion CPU processing main cycles (spare time). The following is an overview of processing.

* Example of motion SFC parameter "No. of consecutive transitions setting 2"



[Point]

(a) Set motion SFC programs containing motion control steps for normal tasks.

(b) Execution of normal tasks is aborted while executing event tasks and NMI tasks.
However, with normal tasks, event task prohibition commands (DI) can be specified in operation control steps, and therefore event task interrupts can be prohibited in parts enclosed with an event task prohibition command (DI) and event task enable command (EI).

(2) Event tasks

Event tasks trigger the execution of motion SFC programs when events occur. There are three types of events as follows.

(a) Fixed cycle

Fixed cycle events regularly trigger the execution of motion SFC programs in a 0.22 ms, 0.44 ms, 0.88 ms, 1.77 ms, 3.55 ms, 7.11 ms, or 14.2 ms cycle.

(b) External interrupt (16 points from I0 to I15)

A motion SFC program is executed when the input set for the event task from the 16 points of the QI60 (16 point interrupt unit) installed in the motion slot turns ON.

(c) Sequence interrupt

A motion SFC program is executed when a GINT command is executed for a sequence program for another Q PLC CPU.

POINT

(1) Multiple events can be set for a single motion SFC program. However, it is not possible to set multiple fixed cycles.
(2) It is also possible to execute multiple motion SFC programs with a single event.
(3) Motion control steps cannot be executed inside event tasks.
(4) If event tasks are prohibited with a normal task, it will not be possible to execute event tasks. If an event occurs while event tasks are prohibited, they are executed the moment event tasks are enabled.

(3) NMI tasks

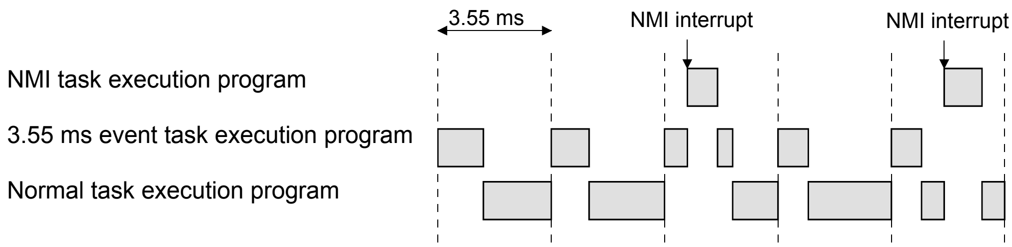
Motion SFC programs are executed when the input set for the NMI task factor from among external interrupts (16 for QI60) is turned ON.

POINT

(1) NMI tasks are given the highest priority among normal tasks, event tasks, and NMI tasks.
(2) Even if event tasks are prohibited (DI) in a normal task, NMI task interrupts are performed without masking.

(4) Execution status example

The following diagram displays an example of the execution status for each motion SFC program when motion SFC programs are executed with multiple tasks.



If there is a program executed with an NMI task, program executed with a 3.55 ms fixed cycle event task, and a program executed with a normal task, as shown in the above diagram,

(a) 3.55 ms fixed cycle event tasks are executed every 3.55 ms,
(b) If an NMI interrupt is entered, priority is given to execution of the NMI task,
(c) And the normal task is executed during spare time.

5.13 SFC Parameters

There are two types of SFC parameters, "task parameters" used to control tasks (normal tasks, event tasks, NMI tasks), and "program parameters" set for each motion SFC program.

5.13.1 Task parameters

<table>
<tr><th>No.</th><th colspan="2">Item</th><th>Setting range</th><th>Default value</th><th>Remarks</th></tr>
<tr><td>1</td><td>No. of consecutive transitions</td><td>Normal tasks
(Common to normal tasks)</td><td>1 to 30</td><td>3</td><td rowspan="5">This parameter reads values when the PLC ready flag (M2000) turns from OFF to ON, and then performs control.
If setting or changing this parameter, turn the PLC ready flag (M2000) OFF.</td></tr>
<tr><td>2</td><td colspan="2">Interrupt setting</td><td>Sets an event task or NMI task for external interrupt input (I0 to I15).</td><td>Event task</td></tr>
<tr><td rowspan="3">3</td><td rowspan="3">Repeat control restriction count</td><td>Normal task</td><td>1 to 100000</td><td>1000</td></tr>
<tr><td>Event task</td><td>1 to 10000</td><td>100</td></tr>
<tr><td>NMI task</td><td>1 to 10000</td><td>100</td></tr>
</table>

5.13.2 Program parameters

The following parameters are set for each motion SFC program.

<table>
<tr><th>No.</th><th>Item</th><th>Setting range</th><th>Default value</th><th>Remarks</th></tr>
<tr><td>1</td><td>Start setting</td><td>Sets whether to Start/Not start automatically.</td><td>Not start</td><td rowspan="7">This parameter reads values when the PLC ready flag (M2000) is ON, and then performs control. If setting or changing this parameter, turn the PLC ready flag (M2000) OFF.</td></tr>
<tr><td rowspan="3">2</td><td rowspan="3">Execution task</td><td>Only one from normal task, event task, NMI task</td><td>Normal task</td></tr>
<tr><td>If an event task is set, set another event to be enabled.
One of the follow 1 to 3 must be set.
1. Fixed cycle
One from 0.22 ms, 0.44 ms, 0.88 ms, 1.77 ms, 3.55 ms, 7.11 ms, or 14.2 ms,
or none.
2. External interrupt (selected from those set for event task)
Multiple interrupts can be set from I0 to I15.
3. PLC interrupt
Multiple interrupts can be set from I0 to I15.
Multiple tasks can be set from 1 to 3.
This is possible even if the same event is shared with multiple motion SFC programs.</td><td rowspan="2">None</td></tr>
<tr><td>If an NMI task is set, set another interrupt input to be enabled.
1. External interrupt (selected from those set for NMI task) Multiple interrupts can be set from I0 to I15.</td></tr>
<tr><td>3</td><td>No. of consecutive transitions</td><td>1 to 10
Set the No. of consecutive transitions for programs set for event tasks or NMI tasks.</td><td>1</td></tr>
<tr><td>4</td><td>END operation</td><td>End/continue
Set the END step operation mode for programs set for event tasks or NMI tasks.</td><td>End</td></tr>
<tr><td>5</td><td>Executing flag</td><td>None/bit device
Set the bit device to be turned ON during motion SFC program execution.
The following devices can be used.
X0 to X1FFF *1
Y0 to Y1FFF
M0 to M8191
B0 to B1FFF
U□\G10000.0 to U□\G(10000+p-1).F (self CPUs only) *2</td><td>None</td></tr>
</table>

*1: With input devices (PXn+0 to PXn+F) allocated to the motion CPU built-in interface (DI), the PXn+4 to PXn+F range is fixed at 0, and cannot be used. (n = first input No.)
*2: p is the number of user setting area points for each CPU multiple CPU high speed transmission area.

5.14 Motion SFC Program Start Method

Motion SFC programs run while PLC ready flag M2000 is ON.
There are three ways of starting motion SFC programs as follows.
(1) Automatic start
(2) Start from motion SFC program
(3) Start from PLC
The start method is set in the program parameters for each motion SFC program.

(1) Automatic start
Motion SFC programs are started automatically by turning the PLC ready flag M2000 ON.

(2) Start from motion SFC program
Motion SFC programs are started by executing a sub-routine call/start step in the motion SFC program.

(3) Start from PLC
Motion SFC programs are started by executing a D(P).SFCS command with a PLC program.

5.15 Motion SFC Program Exit Method

There are three ways of exiting motion SFC programs as follows.
(1) Motion SFC programs are exited by executing an END set in the motion SFC program.
(2) Motion SFC programs are stopped by turning PLC ready flag M2000 OFF.
(3) Motion SFC programs are exited with a clear step.

Point

(1) Multiple ENDs can be set for a single motion SFC program.
(2) Motion SFC programs are exited even if set to start automatically.

Memo

Chapter 6 SV22 Servo Programs

6.1 Servo Programs

A servo program is used to specify the type of positioning control required to control positioning, as well as positioning data. This section describes the servo program configuration and specification method.

> SV13 and SV22 control servo motors with this servo program, and the applicable servo commands are shown in the "Servo command lists".

6.1.1 Servo program configuration

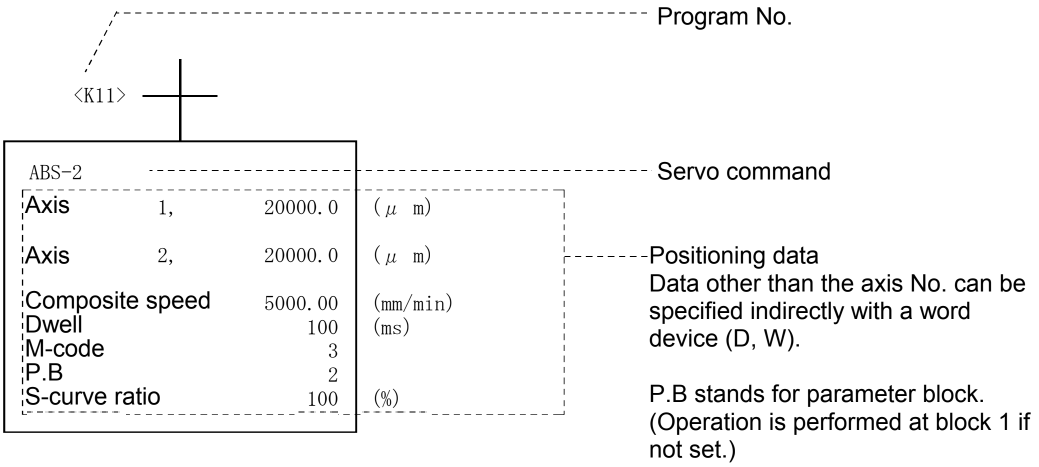
A single servo program consists of the following (1) to (3).

(1) Program No. .... This number is used to specify start requests in the sequence program, and a random number can be set from 0 to 4095.

0 to 4095

(2) Servo command .... Indicates the positioning control type.

(3) Positioning data .... This is data required to execute servo commands. The data required to execute the commands is fixed in each servo command.



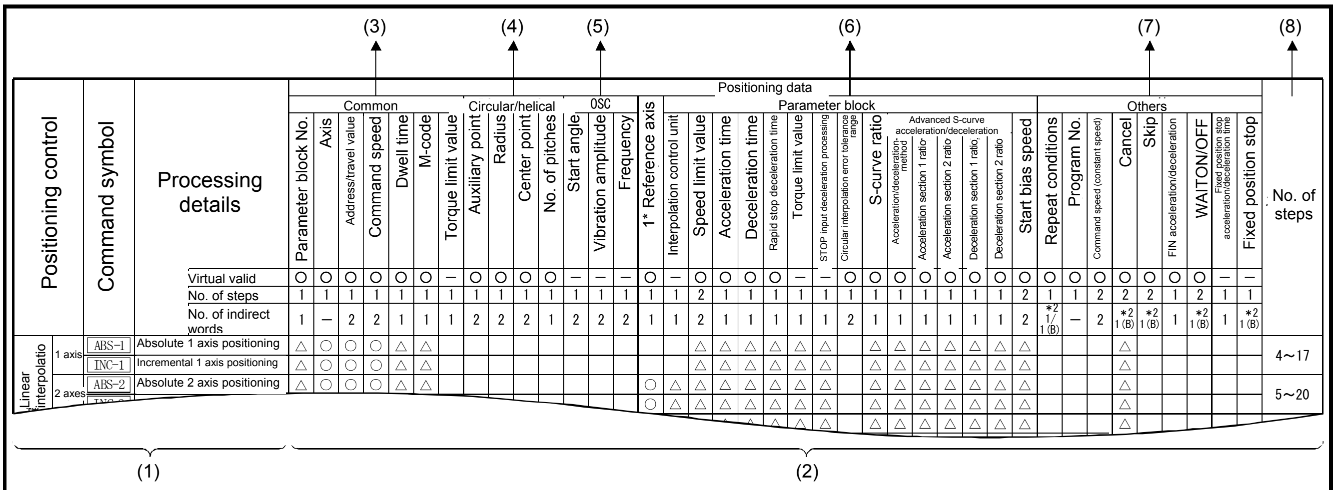
(4) Servo program area

1) The positioning CPU internal memory used to store servo programs created with peripheral equipment has a capacity of 14,334 steps (14 k steps), and the servo program area is used as a backup for the SRAM battery.

6.1.2 Servo command lists

Lists of servo commands used in servo programs are shown on the following pages.

(1) Viewing the command lists



No.	Content	
(1)	Command symbol	Describes servo commands that can be used in servo programs.
	Processing details	Provides an overview of servo command processing.
(2)	(1) Shows positioning data that can be set in servo commands. (a) ○: Items that must be set (Data for which it will not be possible to execute servo commands if not set) (b) Δ: Items set when required (Data controlled with default value if not set)	
	(2) Direct/indirect designation possible (exc. axis No.) (a) Direct designation: Set with numerical value. (b) Indirect designation: Set with word device. • Servo program execution is controlled with the content of the set word device. • 1 word data or 2 word data is used depending on the setting item. • If 2 word data, the first device No. is set.	
	(3) No. of steps The greater the number of setting items, the greater the number of command steps. (The number of steps is displayed when creating servo programs.) (A command + ○ item will result in the minimum step, and steps will increase one at a time with an Δ item 1 item.	
(3)	Items common to all servo commands	
(4)	Items set in circular interpolation start servo programs	
(5)	High-speed oscillating setting items	
(6)	Set when performing control after changing parameter block (default values used if not set) data set at the servo program. (Parameter block data is not changed.)	
(7)	Setting items other than common, circular, parameter block (The items set will differ depending on the servo command.)	
(8)	Indicates the number of steps in each servo command.	

(2) Servo command lists

Lists of servo commands that can be used with servo programs and positioning data set with servo commands are shown on the following table.

Positioning control		Command symbol	Processing details	Positioning data														
				Common							Circular /helical				OSC			
				Parameter block No.	Axis	Address/travel value	Command speed	Dwell time	M-code	Torque limit value	Auxiliary point	Radius	Center point	No. of pitches	Start angle	Vibration amplitude	Frequency	1* Reference axis No.
			Virtual valid	○	○	○	○	○	○	—	○	○	○	○	○	○	○	○
			No. of steps	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
			No. of indirect words	1	—	2	2	1	1	1	2	2	2	1	2	2	2	1
Linear interpolation control	1 axis	ABS−1	Absolute 1 axis positioning	△	○	○	○	△	△									
		INC−1	Incremental 1 axis positioning	△	○	○	○	△	△									
	2 axes	ABS−2	Absolute 2 axis positioning	△	○	○	○	△	△									○
		INC−2	Incremental 2 axis positioning	△	○	○	○	△	△									○
	3 axes	ABS−3	Absolute 3 axis positioning	△	○	○	○	△	△									○
		INC−3	Incremental 3 axis positioning	△	○	○	○	△	△									○
	4 axes	ABS−4	Absolute 4 axis positioning	△	○	○	○	△	△									○
		INC−4	Incremental 4 axis positioning	△	○	○	○	△	△									○
Circular interpolation control	Auxiliary point designation	ABS	Absolute auxiliary point designation circular interpolation	△	○	○	○	△	△		○							
		INC	Incremental auxiliary point designation circular interpolation	△	○	○	○	△	△		○							
	Radius designation	ABS	Absolute radius designation circular interpolation Less than CW 180°	△	○	○	○	△	△			○						
		ABS	Absolute radius designation circular interpolation CW 180° or greater	△	○	○	○	△	△			○						
		ABS	Absolute radius designation circular interpolation Less than CCW 180°	△	○	○	○	△	△			○						
		ABS	Absolute radius designation circular interpolation CCW 180° or greater	△	○	○	○	△	△			○						
		INC	Incremental radius designation circular interpolation Less than CW 180°	△	○	○	○	△	△			○						
		INC	Incremental radius designation circular interpolation CW 180° or greater	△	○	○	○	△	△			○						
		INC	Incremental radius designation circular interpolation Less than CCW 180°	△	○	○	○	△	△			○						
		INC	Incremental radius designation circular interpolation CCW 180° or greater	△	○	○	○	△	△			○						

Positioning data																								No. of steps
Parameter block															Other									
Interpolation control unit	Speed limit value	Acceleration time	Deceleration time	Rapid stop deceleration time	Torque limit value	STOP input deceleration processing	Circular interpolation error tolerance range	S-curve ratio	Advanced S-curve acceleration/deceleration: Acceleration/deceleration method	Advanced S-curve acceleration/deceleration: Acceleration section 1 ratio	Advanced S-curve acceleration/deceleration: Acceleration section 2 ratio	Advanced S-curve acceleration/deceleration: Deceleration section 1 ratio	Advanced S-curve acceleration/deceleration: Deceleration section 2 ratio	Start bias speed	Repeat conditions	Program No.	Command speed (constant speed)	Cancel	Skip	FIN acceleration/deceleration	WAITON/OFF	Fixed position stop acceleration/deceleration time	Fixed position stop	
—	○	○	○	○	—	—	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	—	—	
1	2	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	2	2	2	1	2	1	1	
1	2	1	1	1	1	1	2	1	1	1	1	1	1	2	*2 1/ 1 (B)	—	2	*2 1 (B)	*2 1 (B)	1	*2 1 (B)	1	*2 1 (B)	
	△	△	△	△	△	△		△	△	△	△	△	△	△				△						4～17
	△	△	△	△	△	△		△	△	△	△	△	△	△				△						
△	△	△	△	△	△	△		△	△	△	△	△	△	△				△						5～20
△	△	△	△	△	△	△		△	△	△	△	△	△	△				△						
△	△	△	△	△	△	△		△	△	△	△	△	△	△				△						7～21
△	△	△	△	△	△	△		△	△	△	△	△	△	△				△						
△	△	△	△	△	△	△		△	△	△	△	△	△	△				△						8～22
△	△	△	△	△	△	△		△	△	△	△	△	△	△				△						
△	△	△	△	△	△	△	△	△	△	△	△	△	△	△				△						7～22
△	△	△	△	△	△	△	△	△	△	△	△	△	△	△				△						
△	△	△	△	△	△	△	△	△	△	△	△	△	△	△				△						6～21
△	△	△	△	△	△	△	△	△	△	△	△	△	△	△				△						
△	△	△	△	△	△	△	△	△	△	△	△	△	△	△				△						
△	△	△	△	△	△	△	△	△	△	△	△	△	△	△				△						
△	△	△	△	△	△	△	△	△	△	△	△	△	△	△				△						
△	△	△	△	△	△	△	△	△	△	△	△	△	△	△				△						
△	△	△	△	△	△	△	△	△	△	△	△	△	△	△				△						
△	△	△	△	△	△	△	△	△	△	△	△	△	△	△				△						

○: Items that must be set, △: Items set when required
*1: Only when reference axis speed specified
*2: (B) indicates bit device.

Positioning control		Command symbol	Processing details	Parameter block No.	Axis	Address/travel value	Command speed	Dwell time	M-code	Torque limit value	Auxiliary point	Radius	Center point	No. of pitches	Start angle	Vibration amplitude	Frequency	1* Reference axis No.
			Positioning data:	Common							Circular /helical				OSC			
			Virtual valid	○	○	○	○	○	○	—	○	○	○	○	○	○	○	○
			No. of steps	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
			No. of indirect words	1	—	2	2	1	1	1	2	2	2	1	2	2	2	1
Circular interpolation control	Center point designation	ABS	Absolute center point designation circular interpolation CW	△	○	○	○	△	△				○					
Circular interpolation control	Center point designation	ABS	Absolute center point designation circular interpolation CCW	△	○	○	○	△	△				○					
Circular interpolation control	Center point designation	INC	Incremental center point designation circular interpolation CW	△	○	○	○	△	△				○					
Circular interpolation control	Center point designation	INC	Incremental center point designation circular interpolation CCW	△	○	○	○	△	△				○					
Helical interpolation control	Auxiliary point designation	ABH	Absolute auxiliary point designation helical interpolation	△	○	○	○	△	△		○			○				
Helical interpolation control	Auxiliary point designation	INH	Incremental auxiliary point designation helical interpolation	△	○	○	○	△	△		○			○				
Helical interpolation control	Radius designation	ABH	Absolute radius designation helical interpolation Less than CW 180°	△	○	○	○	△	△			○		○				
Helical interpolation control	Radius designation	ABH	Absolute radius designation helical interpolation CW 180° or greater	△	○	○	○	△	△			○		○				
Helical interpolation control	Radius designation	ABH	Absolute radius designation helical interpolation Less than CCW180°	△	○	○	○	△	△			○		○				
Helical interpolation control	Radius designation	ABH	Absolute radius designation helical interpolation CCW 180° or greater	△	○	○	○	△	△			○		○				
Helical interpolation control	Radius designation	INH	Incremental radius designation helical interpolation Less than CW 180°	△	○	○	○	△	△			○		○				
Helical interpolation control	Radius designation	INH	Incremental radius designation helical interpolation CW 180° or greater	△	○	○	○	△	△			○		○				
Helical interpolation control	Radius designation	INH	Incremental radius designation helical interpolation Less than CCW 180°	△	○	○	○	△	△			○		○				
Helical interpolation control	Radius designation	INH	Incremental radius designation helical interpolation CCW 180° or greater	△	○	○	○	△	△			○		○				
Helical interpolation control	Center point designation	ABH	Absolute center point designation helical interpolation CW	△	○	○	○	△	△				○	○				
Helical interpolation control	Center point designation	ABH	Absolute center point designation helical interpolation CCW	△	○	○	○	△	△				○	○				
Helical interpolation control	Center point designation	INH	Incremental center point designation helical interpolation CW	△	○	○	○	△	△				○	○				
Helical interpolation control	Center point designation	INH	Incremental center point designation helical interpolation CCW	△	○	○	○	△	△				○	○				

Positioning data																								
Parameter block															Other									No. of steps
Interpolation control unit	Speed limit value	Acceleration time	Deceleration time	Rapid stop deceleration time	Torque limit value	STOP input deceleration processing	Circular interpolation error tolerance range	S-curve ratio	Advanced S-curve acceleration/deceleration: Acceleration/deceleration method	Advanced S-curve acceleration/deceleration: Acceleration section 1 ratio	Advanced S-curve acceleration/deceleration: Acceleration section 2 ratio	Advanced S-curve acceleration/deceleration: Deceleration section 1 ratio	Advanced S-curve acceleration/deceleration: Deceleration section 2 ratio	Start bias speed	Repeat conditions	Program No.	Command speed (constant speed)	Cancel	Skip	FIN acceleration/deceleration	WAITON/OFF	Fixed position stop acceleration/deceleration time	Fixed position stop	
—	○	○	○	○	—	—	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	—	—	
1	2	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	2	2	2	1	2	1	1	
1	2	1	1	1	1	1	2	1	1	1	1	1	1	2	*2 1/1(B)	—	2	*2 1(B)	*2 1(B)	1	*2 1(B)	1	*2 1(B)	
△	△	△	△	△	△	△	△	△	△	△	△	△	△	△				△						7～22
△	△	△	△	△	△	△	△	△	△	△	△	△	△	△				△						
△	△	△	△	△	△	△	△	△	△	△	△	△	△	△				△						
△	△	△	△	△	△	△	△	△	△	△	△	△	△	△				△						
△	△	△	△	△	△	△		△	△	△	△	△	△	△				△						10～27
△	△	△	△	△	△	△		△	△	△	△	△	△	△				△						
△	△	△	△	△	△	△		△	△	△	△	△	△	△				△						9～26
△	△	△	△	△	△	△		△	△	△	△	△	△	△				△						
△	△	△	△	△	△	△		△	△	△	△	△	△	△				△						
△	△	△	△	△	△	△		△	△	△	△	△	△	△				△						
△	△	△	△	△	△	△		△	△	△	△	△	△	△				△						
△	△	△	△	△	△	△		△	△	△	△	△	△	△				△						
△	△	△	△	△	△	△		△	△	△	△	△	△	△				△						
△	△	△	△	△	△	△		△	△	△	△	△	△	△				△						
△	△	△	△	△	△	△		△	△	△	△	△	△	△				△						10～27
△	△	△	△	△	△	△		△	△	△	△	△	△	△				△						
△	△	△	△	△	△	△		△	△	△	△	△	△	△				△						
△	△	△	△	△	△	△		△	△	△	△	△	△	△				△						

○: Items that must be set, Δ: Items set when required
*1: Only when reference axis speed specified
*2: (B) indicates bit device.

Positioning control		Command symbol	Processing details	Parameter block No.	Axis	Address/travel value	Command speed	Dwell time	M-code	Torque limit value	Auxiliary point	Radius	Center point	No. of pitches	Start angle	Vibration amplitude	Frequency	1* Reference axis No.
			Positioning data:	Common							Circular /helical				OSC			
			Virtual valid	○	○	○	○	○	○	—	○	○	○	○	○	○	○	○
			No. of steps	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
			No. of indirect words	1	—	2	2	1	1	1	2	2	2	1	2	2	2	1
Fixed feed	1 axis	FEED−1	1 axis fixed feed rate start	△	○	○	○	△	△									
	2 axis	FEED−2	2 axis linear interpolation fixed feed rate start	△	○	○	○	△	△									
	3 axis	FEED−3	3 axis linear interpolation fixed feed rate start	△	○	○	○	△	△									
Speed control (I)	Forward	VF	Speed control (I) forward rotation start	△	○		○		△									
	Reverse	VR	Speed control (I) reverse rotation start	△	○		○		△									
Speed control (II)	Forward	VVF	Speed control (II) forward rotation start	△	○		○		△	△								
	Reverse	VVR	Speed control (II) reverse rotation start	△	○		○		△	△								
Speed, position control	Forward	VPF	Speed, position control forward rotation start	△	○	○	○	△	△	△								
	Reverse	VPR	Speed, position control reverse rotation start	△	○	○	○	△	△	△								
	Restart	VPSTART	Speed, position control restart		○													
Speed switching control		VSTART	Speed switching control start	△														
		VEND	Speed switching control end															
		ABS−1	Speed switching control End point address		○	○	○	△	△	△								
		ABS−2			○	○	○	△	△	△								
		ABS−3			○	○	○	△	△	△								
		INC−1	Speed switching control Travel value to end point		○	○	○	△	△	△								
		INC−2			○	○	○	△	△	△								
		INC−3			○	○	○	△	△	△								
		VABS	Speed switching point absolute designation			○	○		△	△								
		VINC	Speed switching point incremental designation			○	○		△	△								

Positioning data: Parameter block: Interpolation control unit	Speed limit value	Acceleration time	Deceleration time	Rapid stop deceleration time	Torque limit value	STOP input deceleration processing	Circular interpolation error tolerance range	S-curve ratio	Advanced S-curve acceleration/deceleration: Acceleration/deceleration method	Acceleration section 1 ratio	Acceleration section 2 ratio	Deceleration section 1 ratio	Deceleration section 2 ratio	Start bias speed	Other: Repeat conditions	Program No.	Command speed (constant speed)	Cancel	Skip	FIN acceleration/deceleration	WAITON/OFF	Fixed position stop acceleration/deceleration time	Fixed position stop	No. of steps
—	○	○	○	○	—	—	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	—	—	
1	2	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	2	2	2	1	2	1	1	
1	2	1	1	1	1	1	2	1	1	1	1	1	1	2	*2 1/ 1(B)	—	2	*2 1(B)	*2 1(B)	1	*2 1(B)	1	*2 1(B)	
	△	△	△	△	△	△		△	△	△	△	△	△	△				△						4～17
△	△	△	△	△	△	△		△	△	△	△	△	△	△				△						5～19
△	△	△	△	△	△	△		△	△	△	△	△	△	△				△						7～21
	△	△	△	△	△	△		△	△	△	△	△	△	△				△						3～15
	△	△	△	△	△	△		△	△	△	△	△	△	△				△						
	△	△	△	△	△	△		△	△	△	△	△	△	△				△						3～16
	△	△	△	△	△	△		△	△	△	△	△	△	△				△						
	△	△	△	△	△	△		△	△	△	△	△	△	△				△						4～18
	△	△	△	△	△	△		△	△	△	△	△	△	△				△						
																		△						2～4
△	△	△	△	△	△	△		△	△	△	△	△	△	△				△						1～13
																								1
																		△						4～9
																		△						5～10
																		△						7～12
																		△						4～9
																		△						5～10
																		△						7～12
																								4～6

○: Items that must be set, △: Items set when required
*1: Only when reference axis speed specified
*2: (B) indicates bit device.

Positioning control	Command symbol	Processing details	Positioning data														
			Common							Circular /helical				OSC			
			Parameter block No.	Axis	Address/travel value	Command speed	Dwell time	M-code	Torque limit value	Auxiliary point	Radius	Center point	No. of pitches	Start angle	Vibration amplitude	Frequency	1* Reference axis No.
		Virtual valid	○	○	○	○	○	○	—	○	○	○	○	○	○	○	○
		No. of steps	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
		No. of indirect words	1	—	2	2	1	1	1	2	2	2	1	2	2	2	1
Fixed position stop speed control Forward rotation	PVF	Fixed position stop speed control absolute designation	△	○	○	○	△	△									
Fixed position stop speed control Reverse rotation	PVR	Fixed position stop speed control absolute designation	△	○	○	○	△	△									
Fixed-pitch feed	PFSTART	Fixed-pitch feed start	△	○	○	○		△									
Constant speed control	CPSTART1	1 axis constant speed control start	△	○		○											
	CPSTART2	2 axis constant speed control start	△	○		○											
	CPSTART3	3 axis constant speed control start	△	○		○											
	CPSTART4	4 axis constant speed control start	△	○		○											
	ABS−1	Constant speed control pass point absolute designation		○	○			△	△								
	ABS−2			○	○			△	△								
	ABS−3			○	○			△	△								
	ABS−4			○	○			△	△								
	ABS			○	○			△	△	○							
	ABS			○	○			△	△		○						
	ABS			○	○			△	△		○						
	ABS			○	○			△	△		○						
	ABS			○	○			△	△		○						
	ABS			○	○			△	△			○					
	ABS			○	○			△	△			○					
	ABH	Constant speed control pass point helical absolute designation		○	○			△	△	○			○				
	ABH				○			△	△		○		○				
	ABH				○			△	△		○		○				
	ABH				○			△	△		○		○				
	ABH				○			△	△		○		○				
	ABH			○	○			△	△			○	○				
	ABH			○	○			△	△			○	○				

Positioning data																								No. of steps
Parameter block															Other									
									Advanced S-curve acceleration/deceleration															
Interpolation control unit	Speed limit value	Acceleration time	Deceleration time	Rapid stop deceleration time	Torque limit value	STOP input deceleration processing	Circular interpolation error tolerance range	S-curve ratio	Acceleration/deceleration method	Acceleration section 1 ratio	Acceleration section 2 ratio	Deceleration section 1 ratio	Deceleration section 2 ratio	Start bias speed	Repeat conditions	Program No.	Command speed (constant speed)	Cancel	Skip	FIN acceleration/deceleration	WAITON/OFF	Fixed position stop acceleration/deceleration time	Fixed position stop	
—	○	○	○	○	—	—	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	—	—	
1	2	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	2	2	2	1	2	1	1	
1	2	1	1	1	1	1	2	1	1	1	1	1	1	2	*2 1/ 1(B)	—	2	*2 1(B)	*2 1(B)	1	*2 1(B)	1	*2 1(B)	
	△		△	△	△	△		△	△	△	△	△	△	△				△				○	○	6~19
	△		△	△	△	△		△	△	△	△	△	△	△				△				○	○	
	△	△	△	△	△	△		△	△	△	△	△	△	△				△						4~16
	△	△	△	△	△	△		△	△	△	△	△	△	△				△		△				3~15
△	△	△	△	△	△	△	△	△	△	△	△	△	△	△				△		△				3~17
△	△	△	△	△	△	△	△	△	△	△	△	△	△	△				△		△				4~17
△	△	△	△	△	△	△	△	△	△	△	△	△	△	△				△		△				
																	△		△		△			2~10
																	△		△		△			3~11
																	△		△		△			4~12
																	△		△		△			5~13
																	△		△		△			5~14
																	△		△		△			4~13
																	△		△		△			
																	△		△		△			
																	△		△		△			
																	△		△		△			5~14
																	△		△		△			
																	△		△		△			9~14
																	△		△		△			8~13
																	△		△		△			
																	△		△		△			
																	△		△		△			
																	△		△		△			9~14
																	△		△		△			

○: Items that must be set, Δ: Items set when required
*1: Only when reference axis speed specified
*2: (B) indicates bit device.

Positioning control	Com-mand symbol	Processing details	Positioning data: Common: Parameter block No.	Axis	Address/travel value	Command speed	Dwell time	M-code	Torque limit value	Circular /helical: Auxiliary point	Radius	Center point	No. of pitches	OSC: Start angle	Vibration amplitude	Frequency	1* Reference axis No.
		Virtual valid	○	○	○	○	○	○	—	○	○	○	○	○	○	○	○
		No. of steps	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
		No. of indirect words	1	—	2	2	1	1	1	2	2	2	1	2	2	2	1
Constant speed control	INC−1	Constant speed control pass point incremental designation		○	○			△	△								
	INC−2			○	○			△	△								
	INC−3			○	○			△	△								
	INC−4			○	○			△	△								
	INC			○	○			△	△	○							
	INC			○	○			△	△		○						
	INC			○	○			△	△		○						
	INC			○	○			△	△		○						
	INC			○	○			△	△		○						
	INC			○	○			△	△			○					
	INC			○	○			△	△			○					
	INH	Constant speed control pass point helical incremental designation		○	○			△	△	○			○				
	INH				○			△	△		○		○				
	INH				○			△	△		○		○				
	INH				○			△	△		○		○				
	INH				○			△	△		○		○				
	INH			○	○			△	△			○	○				
	INH			○	○			△	△			○	○				
	CPEND	Constant speed control end					△										

Interpolation control unit	Speed limit value	Acceleration time	Deceleration time	Rapid stop deceleration time	Torque limit value	STOP input deceleration processing	Circular interpolation error tolerance range	S-curve ratio	Advanced S-curve acceleration/deceleration: Acceleration/deceleration method	Advanced S-curve acceleration/deceleration: Acceleration section 1 ratio	Advanced S-curve acceleration/deceleration: Acceleration section 2 ratio	Advanced S-curve acceleration/deceleration: Deceleration section 1 ratio	Advanced S-curve acceleration/deceleration: Deceleration section 2 ratio	Start bias speed	Repeat conditions	Program No.	Command speed (constant speed)	Cancel	Skip	FIN acceleration/deceleration	WAITON/OFF	Fixed position stop acceleration/deceleration time	Fixed position stop	No. of steps
Positioning data — Parameter block															Positioning data — Other									
—	○	○	○	○	—	—	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	—	—	
1	2	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	2	2	2	1	2	1	1	
1	2	1	1	1	1	1	2	1	1	1	1	1	1	2	*2 1/ 1(B)	—	2	*2 1(B)	*2 1(B)	1	*2 1(B)	1	*2 1(B)	
																	△		△		△			2～10
																	△		△		△			3～11
																	△		△		△			4～12
																	△		△		△			5～13
																	△		△		△			5～14
																	△		△		△			4～13
																	△		△		△			
																	△		△		△			
																	△		△		△			
																	△		△		△			5～14
																	△		△		△			
																	△		△		△			9～14
																	△		△		△			8～13
																	△		△		△			
																	△		△		△			
																	△		△		△			
																	△		△		△			9～14
																	△		△		△			
																								1～2

○: Items that must be set, Δ: Items set when required
*1: Only when reference axis speed specified
*2: (B) indicates bit device.

Positioning control	Com-mand symbol	Processing details	Positioning data														
			Common							Circular /helical				OSC			
			Parameter block No.	Axis	Address/travel value	Command speed	Dwell time	M-code	Torque limit value	Auxiliary point	Radius	Center point	No. of pitches	Start angle	Vibration amplitude	Frequency	1* Reference axis No.
		Virtual valid	○	○	○	○	○	○	—	○	○	○	○	○	○	○	○
		No. of steps	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
		No. of indirect words	1	—	2	2	1	1	1	2	2	2	1	2	2	2	1
Same control repetition (Used with speed switching control, constant speed control)	FOR-TIMES	Repeat range start setting															
	FOR-ON																
	FOR-OFF																
	NEXT	Repeat range end setting															
Simultaneous start	START	Simultaneous start															
Zeroing	ZERO	Zeroing start		○													
High-speed oscillating	OSC	High-speed oscillating	△	○				△						○	○	○	
Current value change	CHGA	Serco/virtual servo current value change		○	○												
	CHGA-E	Encoder current value change		○	○												
	CHGA-C	Cam axis current value change		○	○												

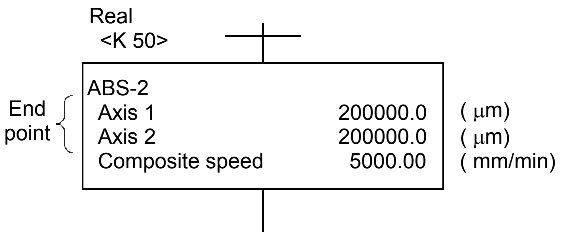
	Positioning data																								No. of steps
	Parameter block															Other									
										Advanced S-curve acceleration/deceleration															
	Interpolation control unit	Speed limit value	Acceleration time	Deceleration time	Rapid stop deceleration time	Torque limit value	STOP input deceleration processing	Circular interpolation error tolerance range	S-curve ratio	Acceleration/deceleration method	Acceleration section 1 ratio	Acceleration section 2 ratio	Deceleration section 1 ratio	Deceleration section 2 ratio	Start bias speed	Repeat conditions	Program No.	Command speed (constant speed)	Cancel	Skip	FIN acceleration/deceleration	WAITON/OFF	Fixed position stop acceleration/deceleration time	Fixed position stop	
	—	○	○	○	○	—	—	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	—	—	
	1	2	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	2	2	2	1	2	1	1	
	1	2	1	1	1	1	1	2	1	1	1	1	1	1	2	*2 1/ 1(B)	—	2	*2 1(B)	*2 1(B)	1	*2 1(B)	1	*2 1(B)	
																○									
																○									2
																○									
																									3
																	○								2～3
																									2
						△													△						5～10
																									3

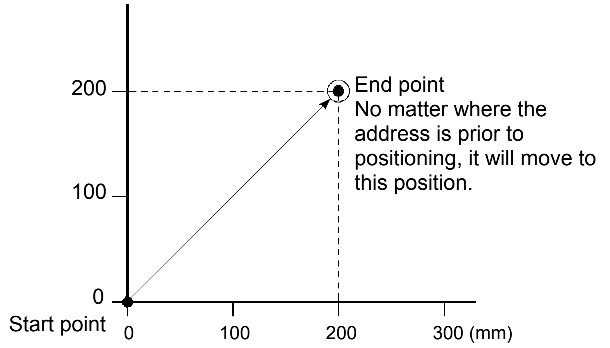
○: Items that must be set, Δ: Items set when required
*1: Only when reference axis speed specified
*2: (B) indicates bit device.

6.1.3 Linear control

Control of 1 to 4 axes with ABS-1 to ABS-4 (absolute method)

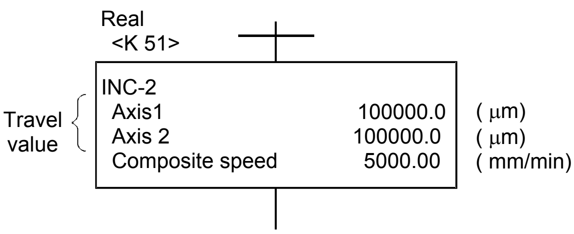
(1) Controls positioning from the current stop address (address prior to positioning) with home position as reference to the specified address.
(2) The movement direction is determined based on the current stop address and specified address.

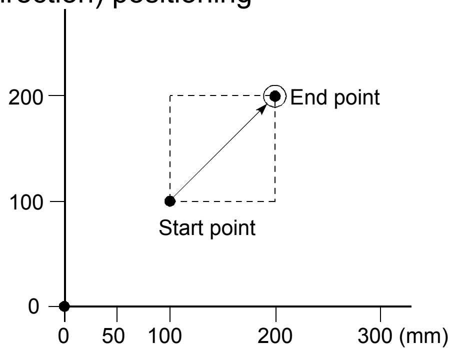




Linear control of 1 to 4 axes with INC-1 to INC-4 (incremental method)

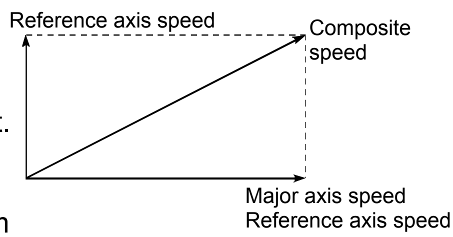
(1) Controls positioning by the specified travel value from the current stop position address.
(2) The movement direction is determined based on the movement symbol (+/-).
  1) When the movement direction is positive:
     Forward direction (address increase direction) positioning
  2) When the movement direction is negative:
     Reverse direction (address decrease direction) positioning





Speed designation (speed type) when performing linear 2 axis, 3 axis, and 4 axis interpolation control

1. Composite speed
   This is the speed designation for moving with interpolation.
2. Major axis speed
   This the speed for the interpolation axis with longest movement.
   (Major axes are judged and processed automatically.)
3. Reference axis speed
   This is the speed setting for the axis to be set as reference from among interpolation axes.



6.1.4 Circular interpolation control for interpolation point designation

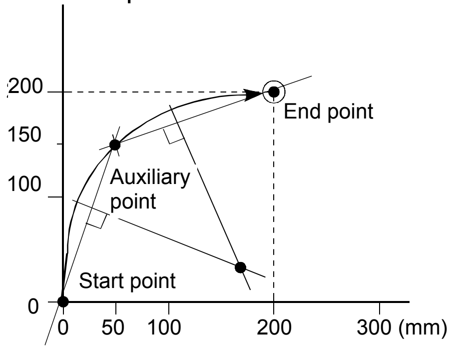
Control of 2 axes with ABS (absolute method)

(1) Performs circular interpolation from the current stop address (address prior to positioning) with home position as reference to the end point address via the specified auxiliary point address.
(2) This is an arc produced with point the start address (current stop address) and auxiliary point address intersects the auxiliary point address and end point address perpendicular bisector as the center point.

Real
<K 52>

ABS

End point	Axis 1	200000.0	(μm)
	Axis 2	200000.0	(μm)
	Speed	5000.00	(mm/min)
	Auxiliary point 1	50000.0	(μm)
	Auxiliary point 2	150000.0	(μm)



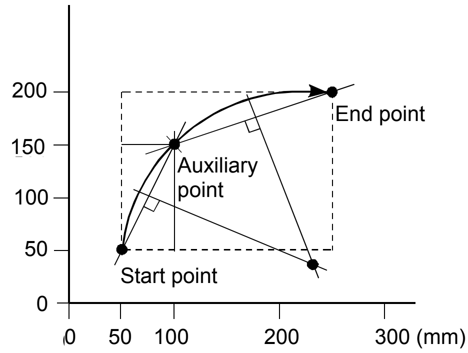
Control of 2 axes with INC (incremental method)

(1) Performs circular interpolation from the current stop address to the end point via the specified auxiliary point.
(2) This is an arc produced with the point the start point (current stop position) and auxiliary point intersect the auxiliary point and end point perpendicular bisector as the center point.

Real
<K 53>

INC

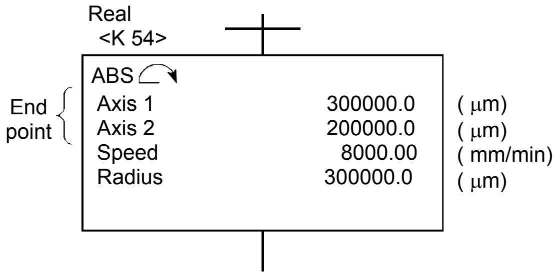
Travel value	Axis 1	200000.0	(μm)
	Axis 2	150000.0	(μm)
	Speed	5000.00	(mm/min)
	Auxiliary point 1	50000.0	(μm)
	Auxiliary point 2	100000.0	(μm)

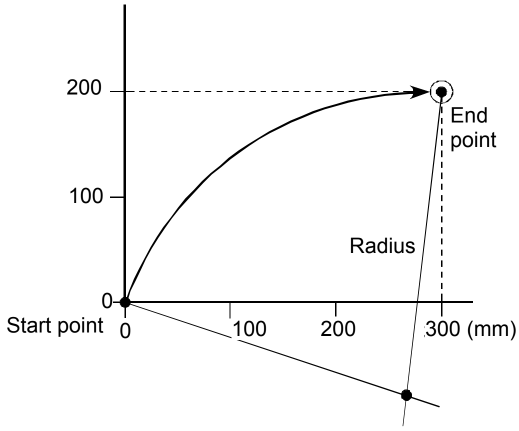


6.1.5 Circular interpolation control for radius designation

Control of 2 axes with ABS, ABS, ABS, and ABS (absolute method)

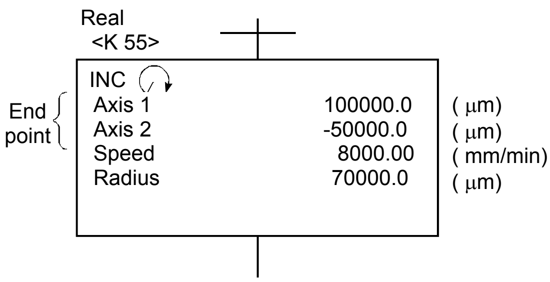
(1) Performs circular interpolation from the current stop address (address prior to positioning) with home position as reference to the specified end point address at the specified radius.
(2) This is an arc produced with the point that the start address (current stop address) and end point address perpendicular bisector intersects the specified radius as the center point.

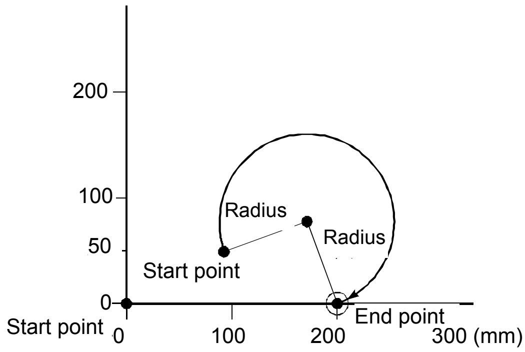




Control of 2 axes with INC, INC, INC, and INC (incremental method)

(1) Performs circular interpolation to the end point specified at the specified radius with the current stop address as the start point (0, 0).
(2) This is an arc produced with the point that the start address (current stop address) and end point address perpendicular bisector intersects the specified radius as the center point.





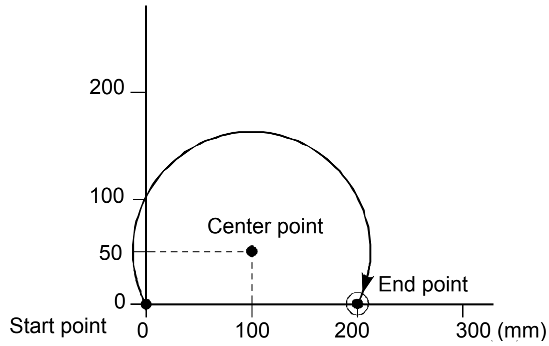
6.1.6 Circular interpolation control for center point designation

Control of 2 axes with ABS , ABS (absolute method)

(1) Performs circular interpolation with the current stop address (address prior to positioning) with home position as reference as the start point address to the end point address with arc with radius of distance to the center point.

Real
<K 56>

	ABS		
End point	Axis 1	200000.0	(μm)
	Axis 2	0.0	(μm)
	Speed	5000.00	(mm/min)
	Center 1	100000.0	(μm)
	Center 2	50000.0	(μm)

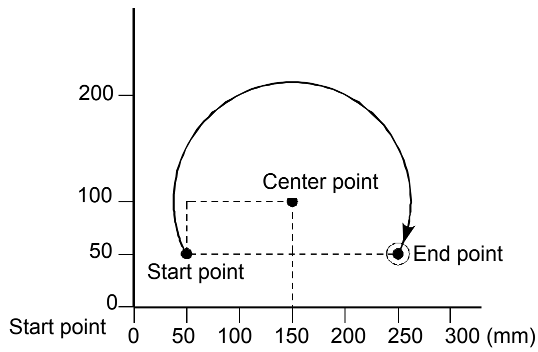


Control of 2 axes with INC , INC (incremental method)

(1) Performs circular interpolation with the current stop address as the start point (0, 0) with travel value to the end point with arc with radius of distance to the center point.

Real
<K 57>

	INC		
Travel value	Axis 1	200000.0	(μm)
	Axis 2	0.0	(μm)
	Speed	5000.00	(mm/min)
	Center 1	100000.0	(μm)
	Center 2	50000.0	(μm)



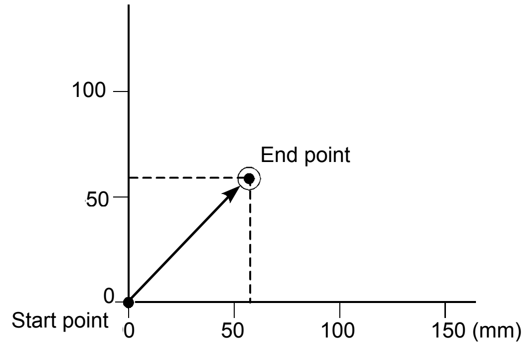
6.1.7 Fixed feeding

Control of 1 to 3 axes with FEED-1, FEED-2, FEED-3 (incremental method)

(1) Controls positioning by the specified travel value with the current stop position as 0.
(2) The movement direction is determined based on the movement symbol.
  (a) When the movement direction is positive:
  Forward direction (address increase direction) positioning
  (b) When the movement direction is negative:
  Reverse direction (address decrease direction) positioning

Real
<K 58>

	FEED-2		
Travel value	Axis 1	60000.0	(μm)
	Axis 2	60000.0	(μm)
	Speed	5000.00	(mm/min)



6.1.8 Speed control

Control of 1 axis with VF, VR, VVF, VVR

(1) Performs control at a specified speed from the moment the servo motor starts until a stop command is input.

  (a) VF: Starts moving in forward direction.
  (b) VR: Starts moving in reverse direction.

Servo amplifier control contains a position loop.

  (c) VVF: Starts moving in forward direction.
  (d) VVR: Starts moving in reverse direction.

Servo amplifier control involves speed control that does not contain a position loop.
Consequently, this can be used for contact positioning control and so on to prevent excessive error.

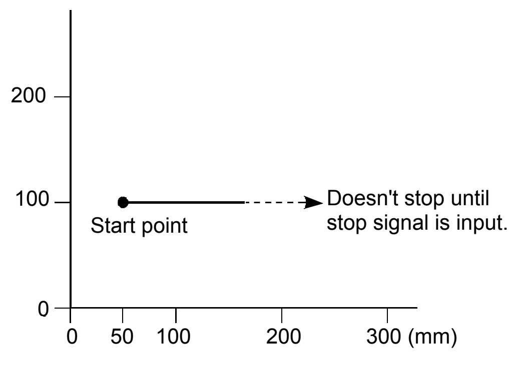
(2) The current value does not change with 0.

Real
<K 30>

	VF		
Forward	Axis 1		
	Speed	2500.00	(mm/min)

Real
<K 31>

	VR		
Reverse	Axis 1		
	Speed	2500.00	(mm/min)

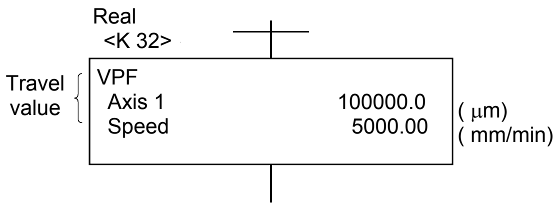


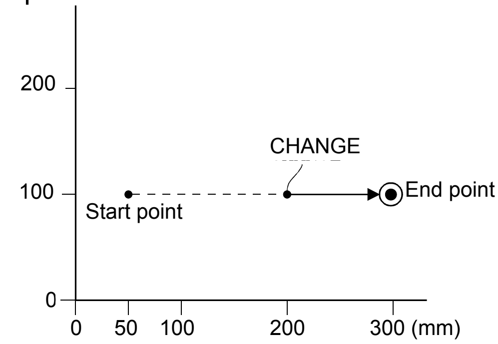
6.1.9 Speed, position switching control

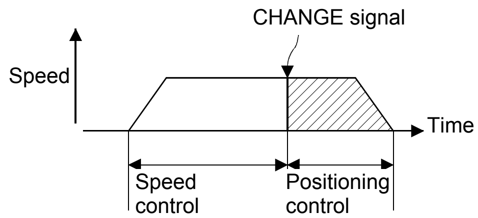
Control of 1 axis with VPF, VPR (incremental method)

(1) Speed control is performed after the servo motor starts, switches to position control with an external CHANGE (speed, position switching) signal when the speed/position switching enable signal (M3205/axis 1) turns ON, and then performs positioning with the specified travel value.
   (a) VPF: Starts moving in forward direction (address increase direction).
   (b) VPR: Starts moving in reverse direction (address decrease direction).

(2) The specified positioning is performed with the incremental method the moment an external CHANGE signal is input.







Remarks

There is no response delay after the external CHANGE signal is input.

6.1.10 Constant speed control

Control of 1 to 4 axes with CPSTART1 to CPSTART4 and CPEND

(1) Performs positioning control at a constant speed to the end point address while relaying the pass point with a single start.

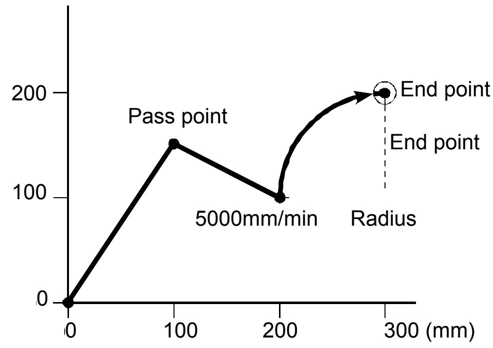
Pass point

ABS-2, ABS-3, ABS-4, ABS, ABS, ABS, ABS, ABS, ABS, ABS, ABH, ABH, ABH, ABH, ABH, ABH, ABH
INC-2, INC-3, INC-4, INC, INC, INC, INC, INC, INC, INC, INH, INH, INH, INH, INH, INH, INH

The absolute or incremental method is determined based on whether the pass point command is ABS or INC, and a mix of both is possible.

```
Real
<K 61>

                   CPSTART2
Speed               Axis 1
designation         Axis 2
                    Speed        5000.00    ( mm/min)
                   ABS-2
                    Axis 1     100000.0     (μm)
                    Axis 2     150000.0     (μm)
Pass point         INC-2
                    Axis 1     100000.0     (μm)
                    Axis 2     -50000.0     (μm)
                   INC
End point is        Axis 1     100000.0     (μm)
before              Axis 2     100000.0     (μm)
CPEND               Radius     100000.0     (μm)
                   CPEND
```



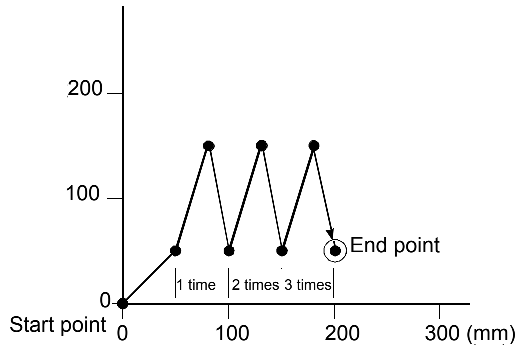
6.1.11 Repeat control (for speed switching control and constant speed control)

Control of 1 to 4 axes with FOR-TIMES, FOR-ON, FOR-OFF/NEXT

(1) Repeats speed switching control speed switching point VABS and VINC commands.
(2) Repeats constant speed control pass point ABS and INC commands.
(3) Repeat count specification method
FOR-TIMES specifies the repeat count with a numerical value from K1 to K32767, or indirectly with D, W, or #.
FOR-ON specifies repeat bit device X, Y, M, L, B, or F until the command turns ON.
FOR-OFF specifies repeat bit device X, Y, M, L, B, or F until the command turns OFF.

Real
<K 62>

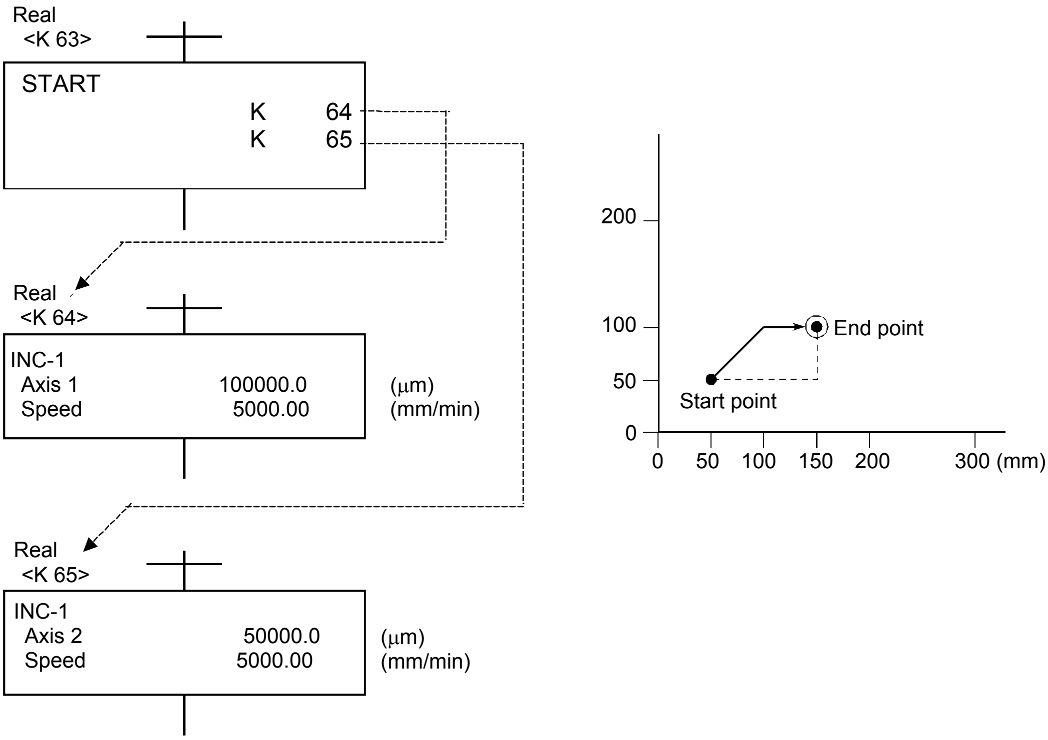
```
CPSTART2
  Axis 1
  Axis 2
  Speed              5000.00    ( mm/min)
ABS-2
  Axis 1            50000.0     ( μm)
  Axis 2            50000.0     ( μm)
FOR-TIMES
                   K      3
INC-2                                     Repeats 3 times.
  Axis 1            30000.0     ( μm)
  Axis 2           100000.0     ( μm)
INC-2
  Axis 1            20000.0     ( μm)
  Axis 2          -100000.0     ( μm)
NEXT
CPEND
```



6.1.12 Simultaneous start

Simultaneous start control with START

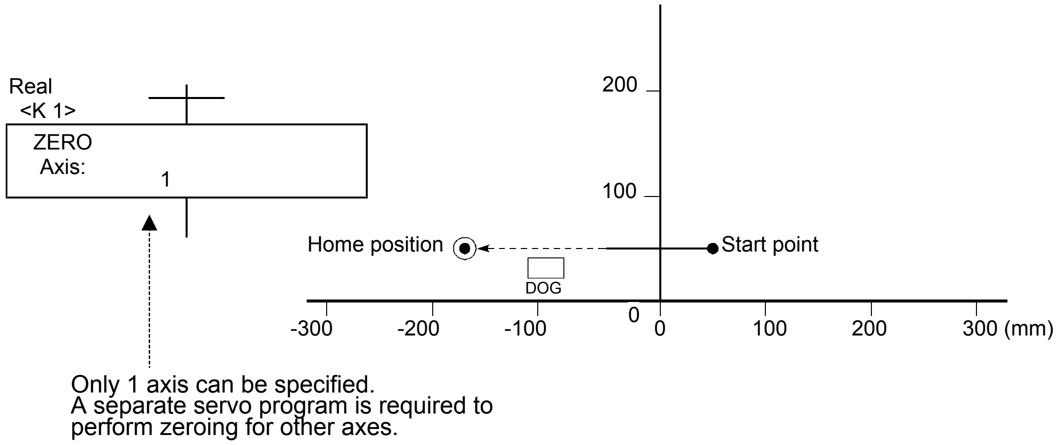
(1) Starts two to three types of servo program (exc. START command) simultaneously.
(2) Up to 12 axes can be started simultaneously if three servo programs are controlling four axes.
(3) Servo program Nos. specified with a START command cannot be specified indirectly.



6.1.13 Zeroing

1 axis zeroing with ZERO

(1) Zeroing is performed from the current stop position based on the zeroing data return method.
(2) If the proximity dog method or count method, the axis advances in the zeroing data return direction.
(3) If the data set method, the stop address is the home position, and the axis does not move.
(4) The axis No. cannot be specified indirectly.



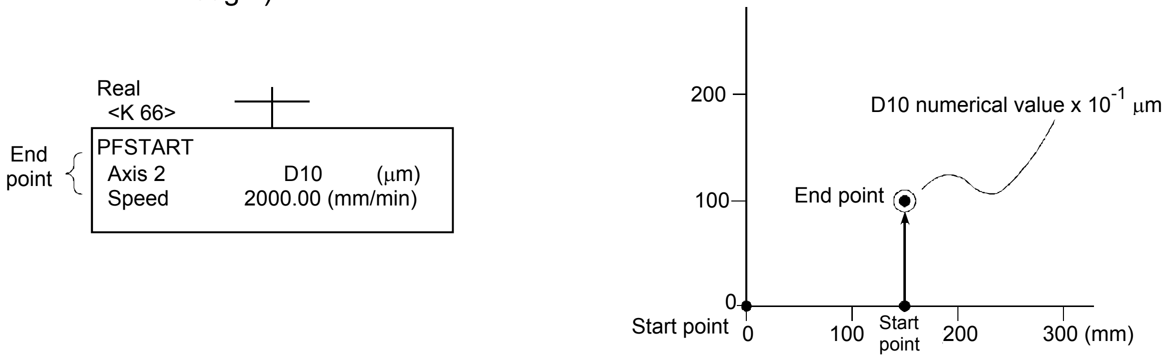
Remarks

The simultaneous starting of zeroing is performed with a START command, and ZERO command servo programs are started simultaneously.

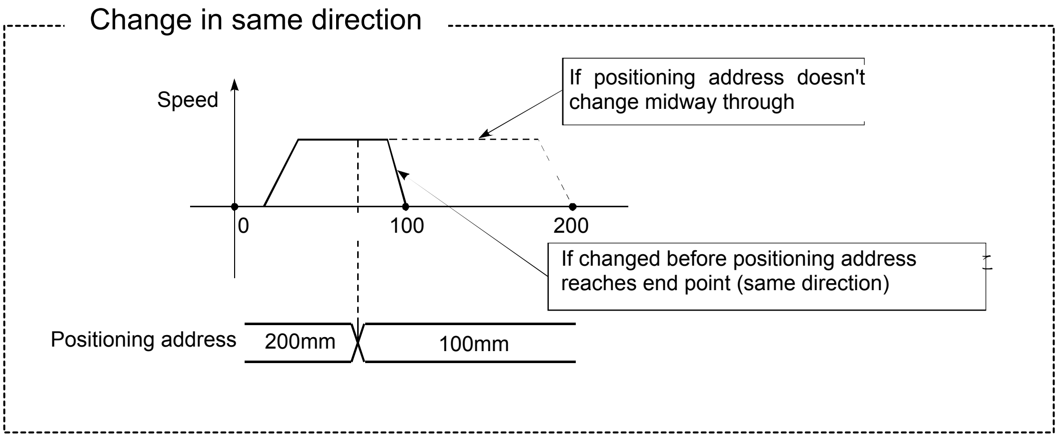
6.1.14 Fixed-pitch feed control

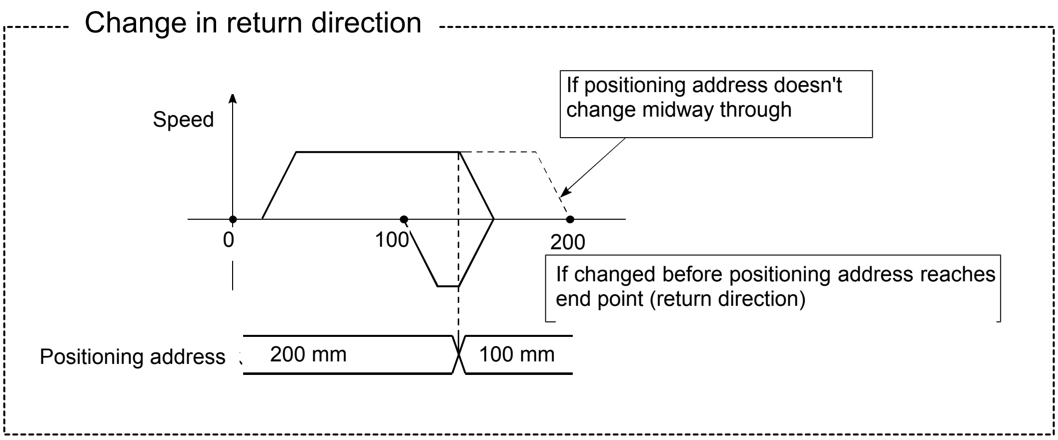
Control of 1 axis with PFSTART (absolute method)

(1) The axis is positioned at the address word device (even number for D, W, #) specified in the servo program with a single start.
(Fixed-pitch feeding is performed if the content of D, W, # changes midway through.)



(2) The movement when the content of the word device changes midway through is as follows.



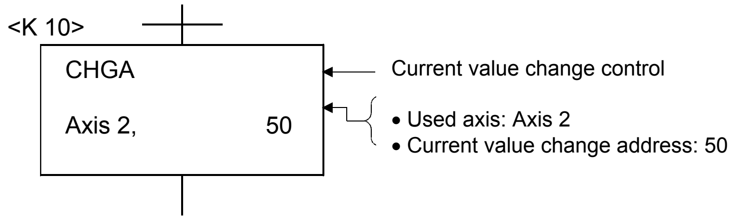


(3) Fixed-pitch feed control continues until a stop command is input.

6.1.15 Current value change

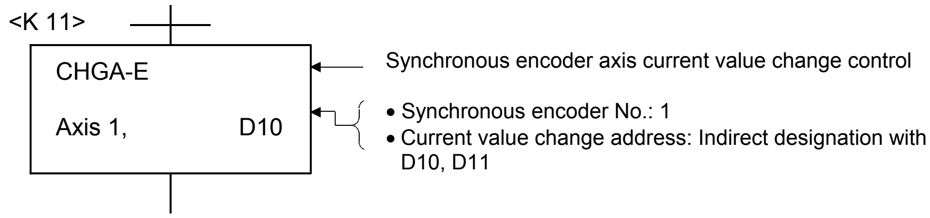
CHGA Servo motor/virtual servo motor axis current value change control

(1) Changes current values for the specified axis when in real mode.
(2) Changes current values for the specified virtual servo motor axis when in virtual mode.



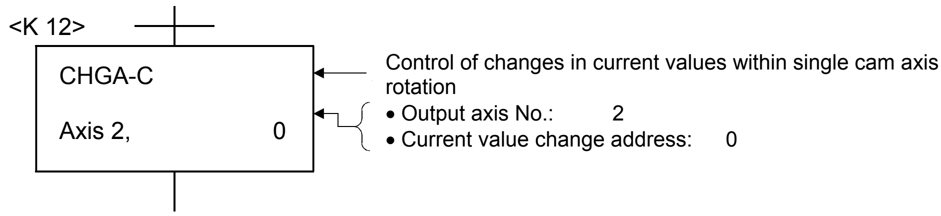
CHGA-E Synchronous encoder axis current value change control

(1) Changes the current value for the specified synchronous encoder axis to the specified address.



CHGA-C Control of changes in current values within single cam axis rotation

(1) Executing a CHGA-C command changes the current values within a single rotation for the specified cam axis to the specified address.
(2) Cam axes may be in the middle of movement.



Chapter 7 Operation Control Programs

Substitute operational expressions, dedicated motion functions, and bit device control commands can be set in operation control programs.
Multiple blocks can be set in a single operation control program, however, only transition programs can be set for transition conditions.
This section describes operation control programs, and operational expressions that can be described in transition programs.

7.1 Operator, function priority order

The priority order for operators and functions is as follows.
By using parentheses, the operation order can be specified freely.

Priority order	Item (operator, function)
High	Calculation inside parentheses ((...))
	Standard function (SIN, COS, etc.), type conversion (USHORT, LONG, etc.)
	Bit inversion (¯), logical negation (!), sign inversion (-)
	Multiplication (*), division (/), remainder (%)
	Addition (+), subtraction (-)
	Bit left shift (<<), bit right shift (>>)
	Comparison operator: less than (<), less than or equal to (<=), greater than (>), greater than or equal to (>=)
	Comparison operator: match (==), mismatch (!=)
	Bit logical product (&)
Low	Bit exclusive logical sum (^)
	Bit logical sum (|)
	Logical product (*)
	Logical sum (+)
	Substitution (=)

7.2 Operational control, transition command list

Refer to Appendix 9 for details on the shaded parts in the following table.

Category	Symbol	Function	Format	No. of basic steps	Usable program		Usable expression			
					F/FS	G	Calculation formula	Bit conditional expression	Comparative conditional expression	Y/N transition conditional expression
Binary operation	=	Substitution	(D)=(S)	4	○	○	○	-	-	-
	+	Addition	(S1)+(S2)	4	○	○	○	-	-	-
	-	Subtraction	(S1)-(S2)	4	○	○	○	-	-	-
	*	Multiplication	(S1)*(S2)	4	○	○	○	-	-	-
	/	Division	(S1)/(S2)	4	○	○	○	-	-	-
	%	Remainder	(S1)%(S2)	4	○	○	○	-	-	-
Bit operation	‾	Bit inversion (complement)	‾ (S)	2	○	○	○	-	-	-
	&	Bit logical product	(S1)&(S2)	4	○	○	○	-	-	-
	|	Bit logical sum	(S1)|(S2)	4	○	○	○	-	-	-
	^	Bit exclusive logical sum	(S1)^(S2)	4	○	○	○	-	-	-
	>>	Bit right shift	(S1)>>(S2)	4	○	○	○	-	-	-
	<<	Bit left shift	(S1)<<(S2)	4	○	○	○	-	-	-
Sign	-	Sign inversion (complement of 2)	-(S)	4	○	○	○	-	-	-
Standard function	SIN	Sine	SIN(S)	2	○	○	○	-	-	-
	COS	Cosine	COS(S)	2	○	○	○	-	-	-
	TAN	Tangent	TAN(S)	2	○	○	○	-	-	-
	ASIN	Arc sine	ASIN(S)	2	○	○	○	-	-	-
	ACOS	Arc cosine	ACOS(S)	2	○	○	○	-	-	-
	ATAN	Arc tangent	ATAN(S)	2	○	○	○	-	-	-
	SQRT	Square root	SQRT(S)	2	○	○	○	-	-	-
	LN	Natural logarithm	LN(S)	2	○	○	○	-	-	-
	EXP	Exponent operation	EXP(S)	2	○	○	○	-	-	-
	ABS	Absolute value	ABS(S)	2	○	○	○	-	-	-
	RND	Round-off	RND(S)	2	○	○	○	-	-	-
	FIX	Omission of fractions	FIX(S)	2	○	○	○	-	-	-
	FUP	Round-up	FUP(S)	2	○	○	○	-	-	-
	BIN	BCD → BIN conversion	BIN(S)	2	○	○	○	-	-	-
	BCD	BIN → BCD conversion	BCD(S)	2	○	○	○	-	-	-
Type conversion	SHORT	Conversion to 16 bit integer type (with sign)	SHORT(S)	2	○	○	○	-	-	-
	USHORT	Conversion to 16 bit integer type (without sign)	USHORT(S)	2	○	○	○	-	-	-
	LONG	Conversion to 32 bit integer type (with sign)	LONG(S)	2	○	○	○	-	-	-
	ULONG	Conversion to 32 bit integer type (without sign)	ULONG(S)	2	○	○	○	-	-	-
	FLOAT	Deem as data with sign, convert to 64 bit floating decimal type	FLOAT(S)	2	○	○	○	-	-	-
	UFLOAT	Deem as data without sign, convert to 64 bit floating decimal type	UFLOAT(S)	2	○	○	○	-	-	-
	DFLT	Floating decimal type 32 → 64 bit conversion	DFLT(S)	2	○	○	○	-	-	-
	SFLT	Floating decimal type 64 → 32 bit conversion	SFLT(S)	2	○	○	○	-	-	-
Bit device status	(None)	ON (A contact)	(bit conditional expression)	2	○	○	-	○	-	○
	!	OFF (B contact)	! (bit conditional expression)	2	○	○	-	○	-	○

Category	Symbol	Function	Format	No. of basic steps	Usable program		Usable expression			
					F/FS	G	Calculation formula	Bit conditional expression	Comparative conditional expression	Y/N transition conditional expression
Bit device control	SET	Device set	SET(D)	3	○	○	-	○	-	-
			SET(D) = (conditional expression)	4	○	○	-	○	○	-
	RST	Device reset	RST(D)	3	○	○	-	○	-	-
			SET(D) = (conditional expression)	4	○	○	-	○	○	-
	DOUT	Device output	DOUT(D),(S)	4	○	○	-	○	-	-
	DIN	Device input	DIN(D),(S)	4	○	○	-	○	-	-
	OUT	Bit device output	OUT(D) = (conditional expression)	4	○	○	-	○	○	-
Logical operation	(None)	Logical affirmation	(Conditional expression)	0	○	○	-	○	○	○
	!	Logical negation	! (conditional expression)	2	○	○	-	○	○	○
	*	Logical product	(Conditional expression) * (conditional expression)	4	○	○	-	○	○	○
	+	Logical sum	(Conditional expression) + (conditional expression)	4	○	○	-	○	○	○
Comparison operation	==	Match	(Calculation formula) == (calculation formula)	4	○	○	-	-	○	○
	!=	Mismatch	(Calculation formula) != (calculation formula)	4	○	○	-	-	○	○
	<	Less than	(Calculation formula) < (calculation formula)	4	○	○	-	-	○	○
	<=	Less than or equal to	(Calculation formula) <= (calculation formula)	4	○	○	-	-	○	○
	>	Greater than	(Calculation formula) > (calculation formula)	4	○	○	-	-	○	○
	>=	Greater than or equal to	(Calculation formula) >= (calculation formula)	4	○	○	-	-	○	○
Dedicated motion function	CHGV	Speed change request	CHGV((S1),(S2))	4	○	○	-	-	-	-
	CHGT	Torque limit value change request	CHGT((S1),(S2))	4	○	○	-	-	-	-
	CHGT2	Torque limit value individual change request	CHGT2((S1),(S2),(S3))	5	○	○	(S1) only not possible	-	-	-
	CHGP	Target pos. change request	CHGP((S1),(S2),(S3))	6	○	○	-	-	-	-
Other	EI	Event task authorized	EI	1	○	○	-	-	-	-
	DI	Event task prohibited	DI	1	○	○	-	-	-	-
	NOP	No processing	NOP	1	○	○	-	-	-	-
	FMOV	Same data block transfer	FMOV(D),(S),(n)	6	○	○	-	-	-	-
	BMOV	Block transfer	BMOV(D),(S),(n)	7	○	○	-	-	-	-
	TIME	Time wait	TIME(S)	7	-	○	-	-	-	-
	MULTW	Data write to self CPU shared memory	MULTW(D),(S),(n),(D1)	8	○	○	-	-	-	-
	MULTR	Data read to other CPU shared memory	MULTR(D),(S1),(S2),(n)	7	○	○	-	-	-	-
	TO	Word data write to intelligent function module/special module	TO(D1),(D2),(S),(n)	7	○	○	-	-	-	-
	FROM	Word data read to intelligent function module/special module	FROM(D),(S1),(S2),(n)	7	○	○	-	-	-	-
	TIME	Time wait	TIME(S)	7	-	○	-	-	-	-

Category	Symbol	Function	Format	No. of basic steps	Usable program		Usable expression			
					F/FS	G	Calculation formula	Bit conditional expression	Comparative conditional expression	Y/N transition conditional expression
Dedicated vision system function	MVOPEN	Line open	MVOPEN(S1),(S2)	4	○	○	-	-	-	-
	MVLOAD	Vision program load	MVLOAD(S1),(S2)	4	○	○	-	-	-	-
	MVTRG	Trigger issue	MVTRG(S1),(S2)	4	○	○	-	-	-	-
	MVPST	Vision program start	MVPST(S1),(S2)	4	○	○	-	-	-	-
	MVIN	Data input	MVIN(S1),(S2),(D),(S3)	8 or higher	○	○	-	-	-	-
	MVOUT	Data output	MVOUT(S1),(S2),(S3),(S4)	8 or higher	○	○	-	-	-	-
	MVFIN	Status storage device reset	MVFIN(S)	2	○	○	-	-	-	-
	MVCLOSE	Line close	MVCLOSE(S)	2	○	○	-	-	-	-
	MVCOM	Random native mode command transmission	MVCOM(S1),(S2),(D),(S3),(S4)	9 or higher	○	○	-	-	-	-
Data control	SCL	16-bit integer type scaling	SCL(S1),(S2),(S3),(D)	8	○	○	-	-	-	-
	DSCL	32-bit integer type scaling	DSCL(S1),(S2),(S3),(D)	8	○	○	-	-	-	-

Program control	IF ~ ELSE ~ IEND	Conditional branch control	IF(S)
			:
			ELSE
			:
			IEND
				IF :4
				ELSE:3
				IEND:1
					○	○	-	-	-	-

	SELECT ~ CASE ~ SEND	32-bit integer type scaling	SELECT CASE(S1) : CEND CASE(Sn) : CEND CLELSE : CEND SEND	SELECT :1 CASE:4 CEND:3 CLELSE :1 SEND:1	○	○	-	-	-	-
	FOR ~ NEXT	No. of times designation repeat control	FOR(D) = (S1) TO (S2) STEP (S3) : NEXT	FOR:9 NEXT:8	○	○	-	-	-	-
	BREAK	Repeat control forced exit	BREAK	3	○	○	-	-	-	-

1 program code for operation control program, transition program Size approximate expression

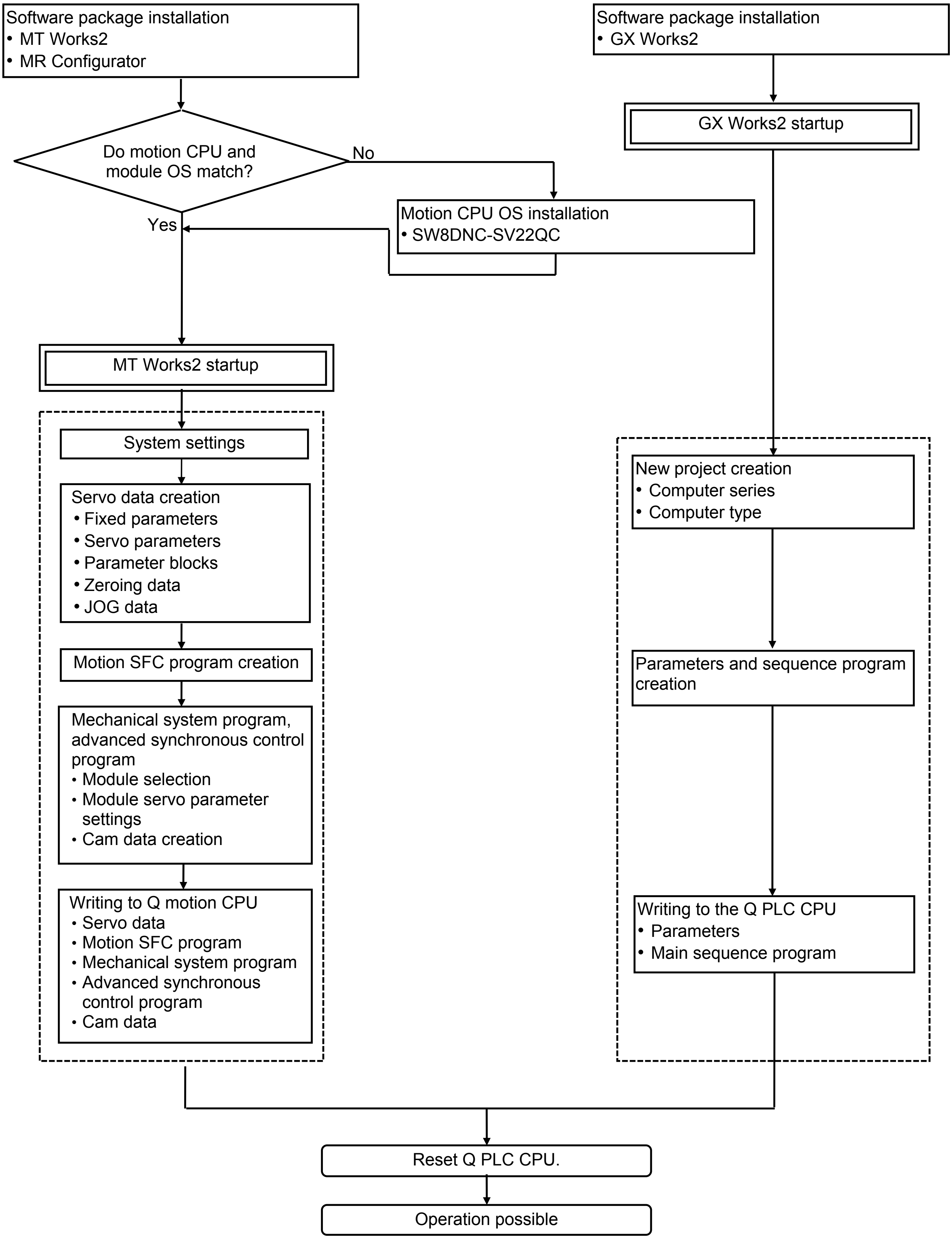
2 + (1 + total no. of basic steps in 1 block)
+ 32 bit constant qty/1 block × 1
+ 64 bit constant qty/1 block × 3) × no. of blocks (steps)
(1 step = 2 bytes)

POINT

- A transition condition must be set in the final block of the transition program.

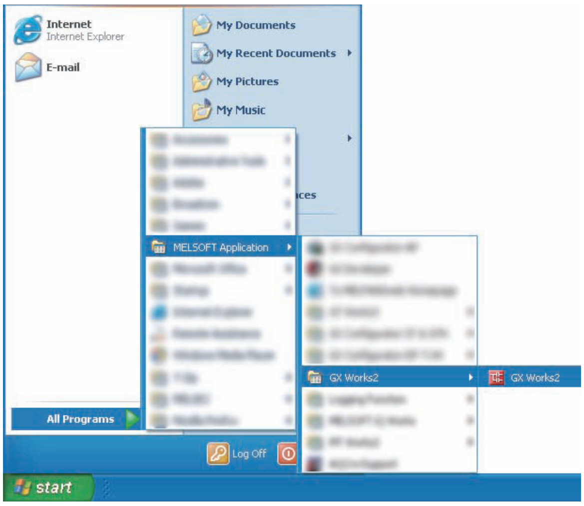
Chapter 8 Windows Computer Operation

8.1 Data Creation Flow for Motion Controller Operation

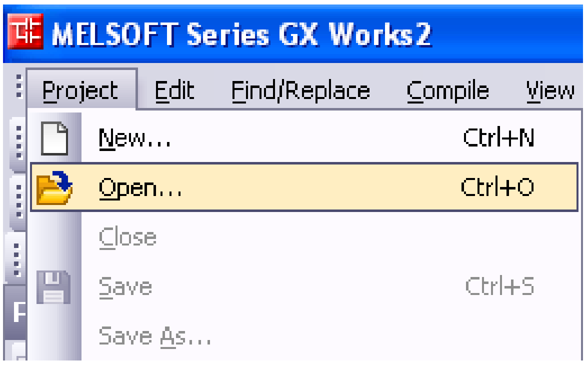


8.2 Q PLC CPU Settings

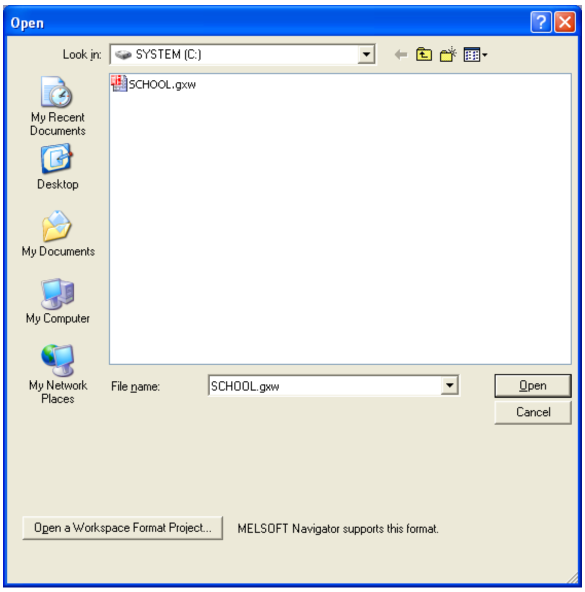
8.2.1 Opening a project



(1) Click the Windows [start] button, and then select [All Programs] → [MELSOFT Application] → [GX Works2] → [GX Works2].



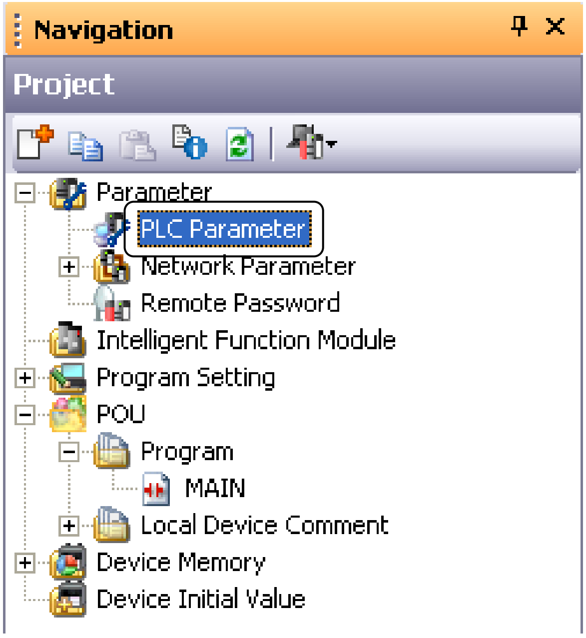
(2) When GX Works2 starts up, click [Open…] on the [Project] menu.



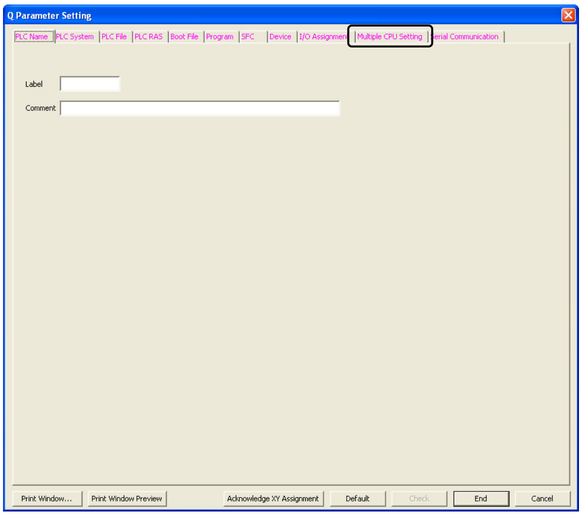
(3) A dialog box prompting the user to open a project appears. Select the project to read.

By clicking the Open button, the sequence program and computer parameters are read.

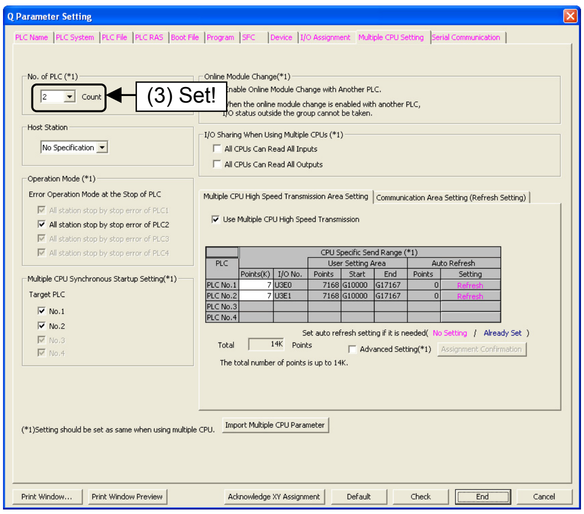
8.2.2 Multiple CPU settings



(1) Select [Parameter] in the project window, and then double-click [PLC Parameter].



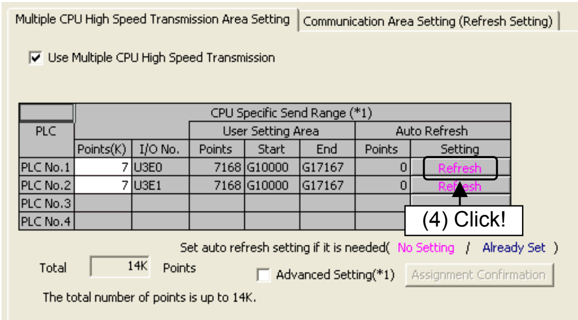
(2) Click the [Multiple CPU setting] tab at the Q Parameter Setting dialog box that appears.



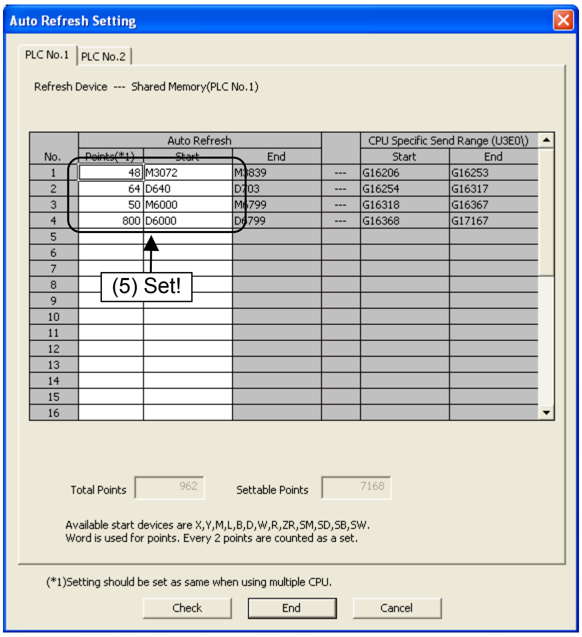
(3) Set "No. of PLC" to "2".

Go to next page

From previous page

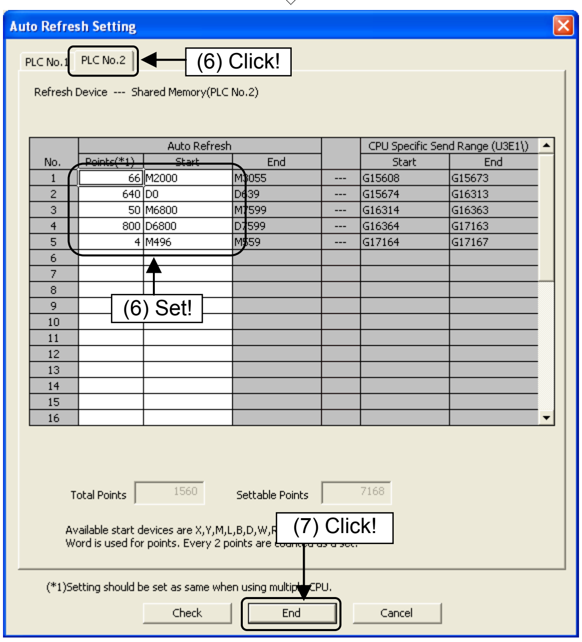


(4) Click the PLC No.1 Refresh button in the "Multiple CPU High Speed Transmission Area Setting "tab.



(5) An Auto Refresh Setting dialog box then appears. Specify the auto refresh settings for the PLC No.1 as follows.

"No. 1 - Points" : "48"
"No. 1 - Start" : "M3072"
"No. 2 - Points" : "64"
"No. 2 - Start" : "D640"
"No. 3 - Points" : "50"
"No. 3 - Start" : "M6000"
"No. 4 - Points" : "800"
"No. 4 - Start" : "D6000"



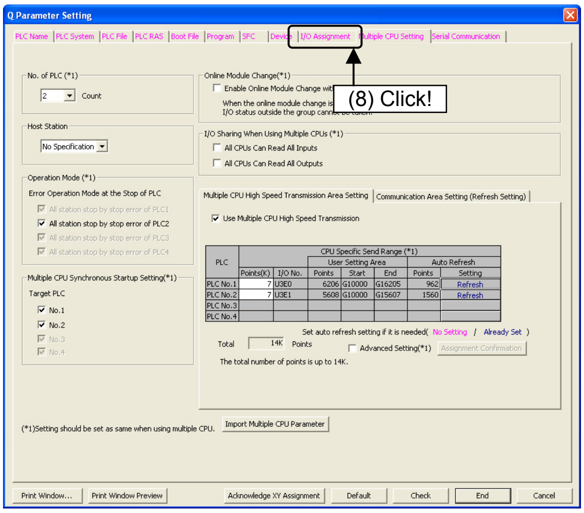
(6) Click the "PLC No.2" tab, and specify the auto refresh settings for the PLC No.2 as follows.

"No. 1 - Points" : "66"
"No. 1 - Start" : "M2000"
"No. 2 - Points" : "640"
"No. 2 - Start" : "D0"
"No. 3 - Points" : "50"
"No. 3 - Start" : "M6800"
"No. 4 - Points" : "800"
"No. 4 - Start" : "D6800"
"No. 5 - Points" : "4"
"No. 5 - Start" : "M496"

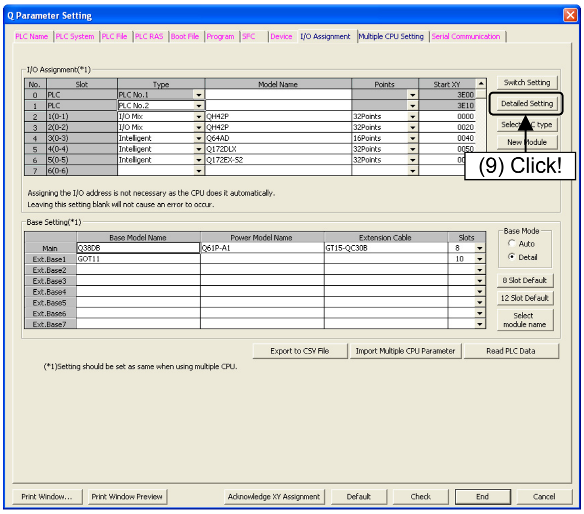
(7) When settings are complete, click the End button.

Go to next page

From previous page

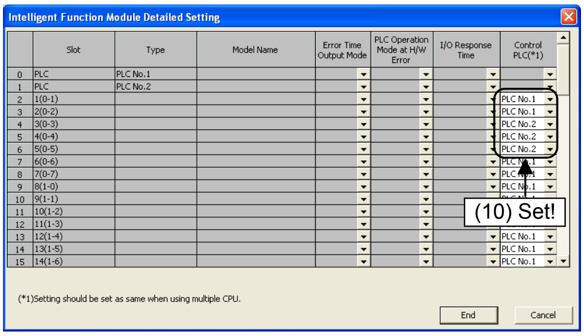


(8) The display then returns to the Q Parameter Setting dialog box. Click the [I/O Assignment setting] tab.



(9) At the I/O Assignment tab, set the "Base Setting - Main" - "Slots" to "8", select "10" for the "Basic Setting – Ext.Base1" - "Slots", and then click the Detailed Setting button.

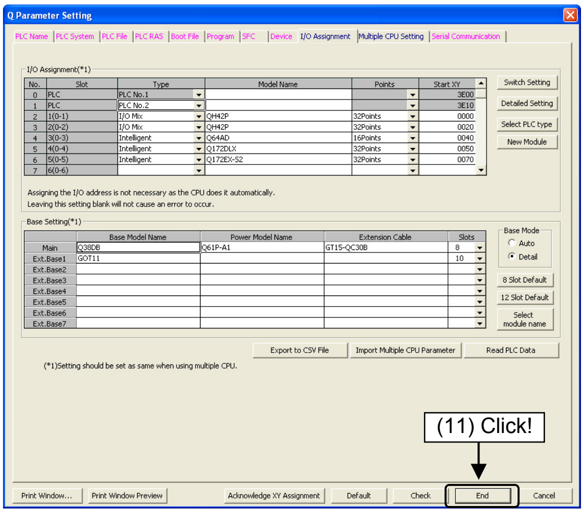
* "Type: Intelligent" and "No. of points: 16" are set for I/O assignment expansion slot "8(1-0)".



(10) At the I/O Module, Intelligent Function Module Detailed Setting dialog box that appears, Set "Slot 1 (*-1)" - "Control PLC" to "PLC No.1", "Slot 2 (*-2)" - "Control PLC" to "PLC No.1", "Slot 3 (*-3)" - "Control PLC" to "PLC No.2", "Slot 4 (*-4)" - "Control PLC" to "PLC No.2", and "Slot 5 (*-5)" - "Control PLC" to "PLC No.2", and then click the End button.

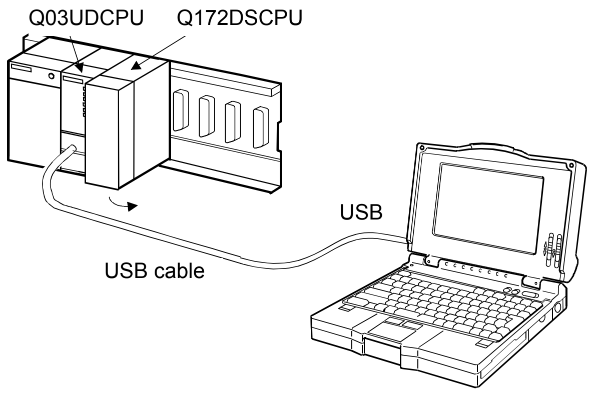
Go to next page

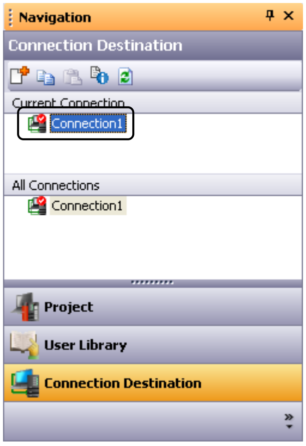
From previous page



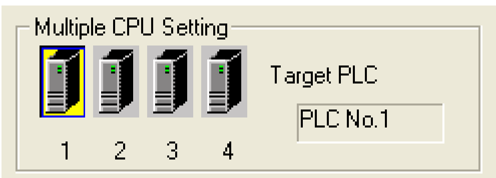
(11) The display then returns to the Q Parameter Settings dialog box. Click the [End] button.

8.2.3 Writing sequence programs

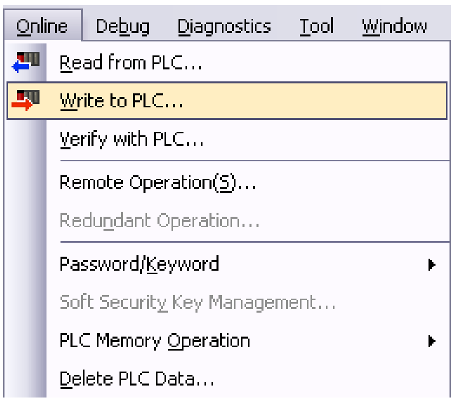




(1) Double-click "Connection1" at "Connection Destination" in the Project window.



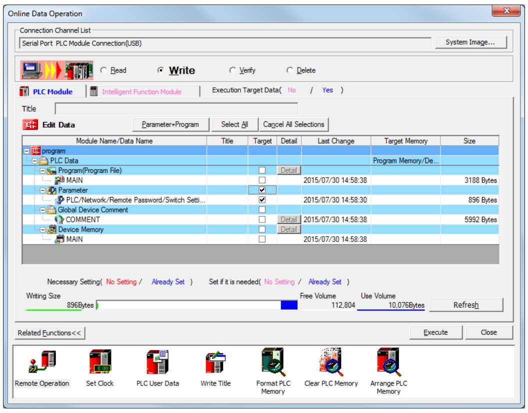
(2) A Connection Destination Setting dialog box appears. Set the "Multiple CPU Setting" - "Target PLC" to "PLC No.1", and then click the OK button.



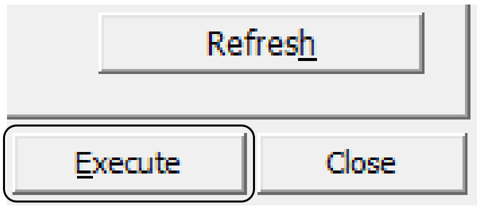
(3) Click [Write to PLC…] on the [Online] menu.

Go to next page

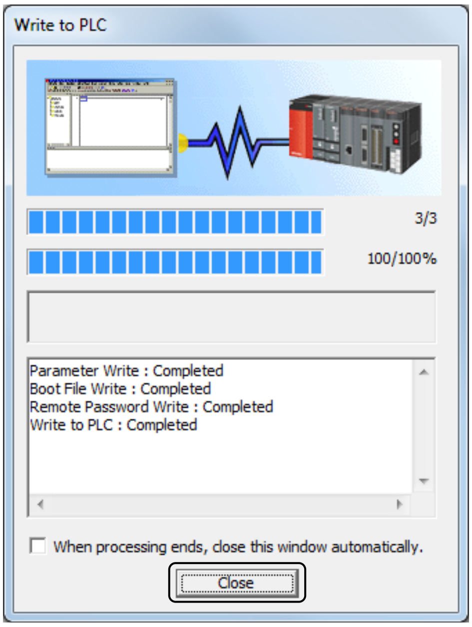
From previous page



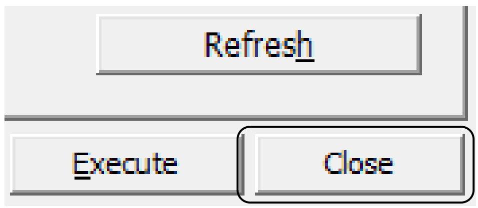
(4) Click the Parameter + Program button at the Online Data Operation dialog box that appears.



(5) Click the Execute button.



(6) A "PC write: Complete" message appears when writing to the computer is complete. Click the Close button.

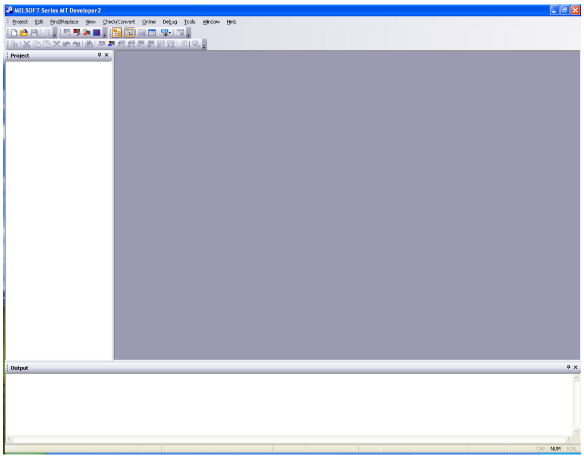


(7) Click the Close button at the Online Data Operation dialog box.

8.3 Starting MT Works2

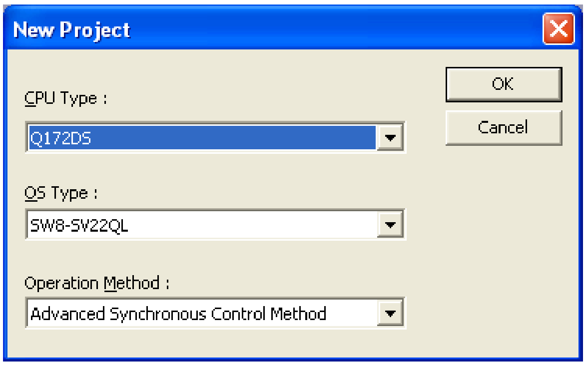
The following is a description of the procedure from MT Works2 startup to new project creation.

(1) Click the Windows [start] button, and then select [All Program] → [MELSOFT Application] → [MT Works2] → [MT Developer2].



(2) MT Works2 starts up.

(3) Click [New...] on the [Project] menu.

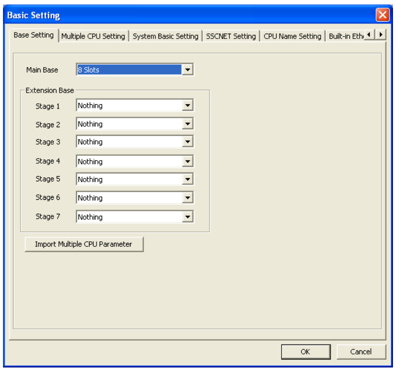


(4) A Create New Project dialog box appears. Select the "CPU Type" and "OS Type", and then click the OK button.

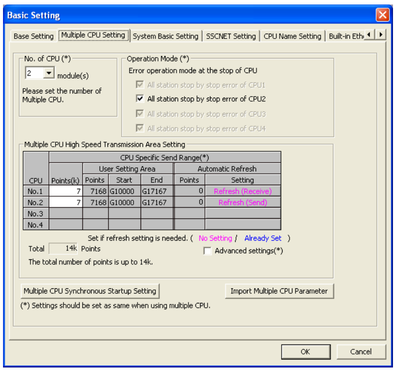
Go to next page

From previous page

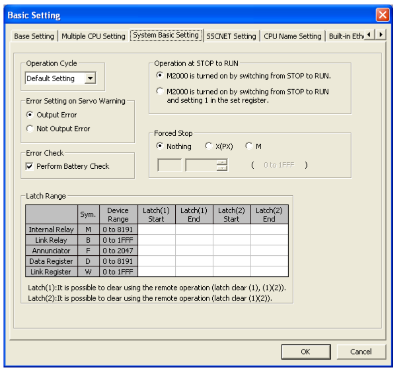
(5) A new project is created, and a Basic Settings dialog box appears.
The content of settings to be specified at each tab screen in the Basic Settings dialog box is as follows.



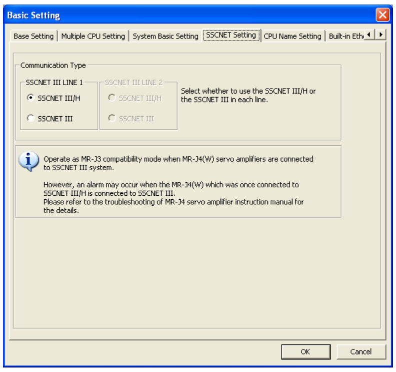
[Base Setting] tab
Set the number of main base slots and number of expansion base levels and slots.



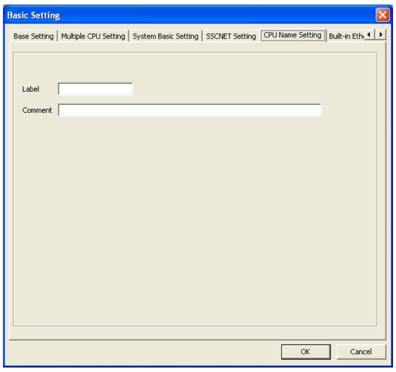
[Multiple CPU Setting] tab
Specify multiple CPU system settings such as the number of CPUs, operation mode when a CPU stop error occurs, devices used with auto refresh, etc.



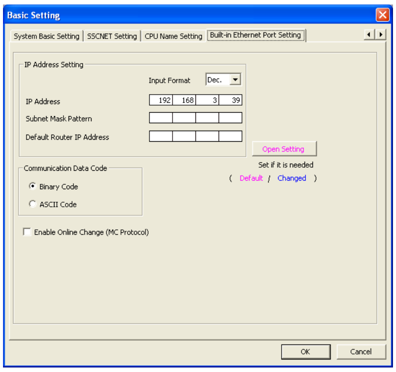
[System Basic Setting] tab
Set the motion CPU operation cycle, latch range, etc.



[SSCNET Setting] tab
Set the communication type and SSCNET system.

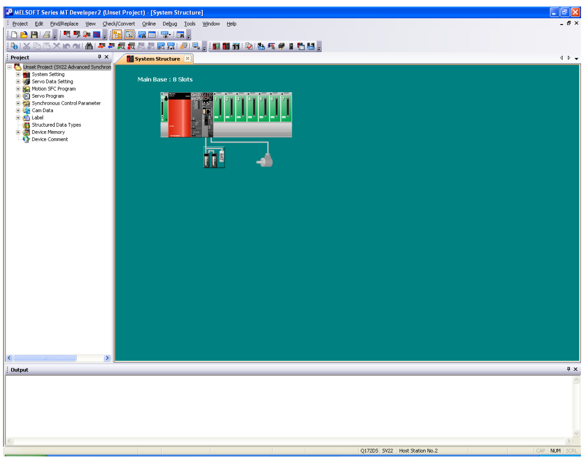


[CPU Name Setting] tab
Set labels and comments.



[Built-in Ethernet Port Setting] tab
Set the IP address, protocol, etc.

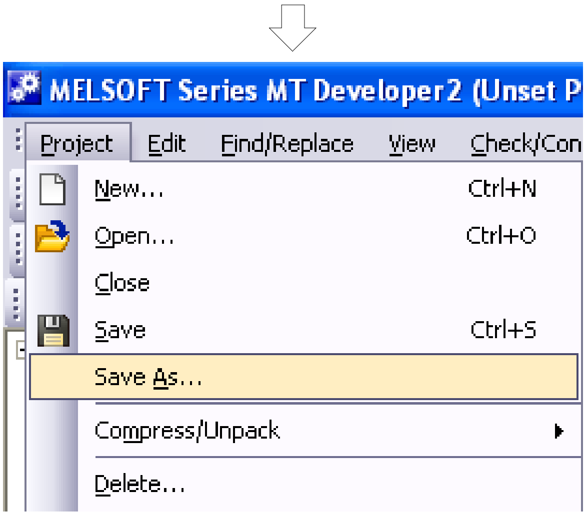
Click the OK button to close the Base Setting dialog box.



(6) Startup and new project creation are now complete.

Go to next page

From previous page

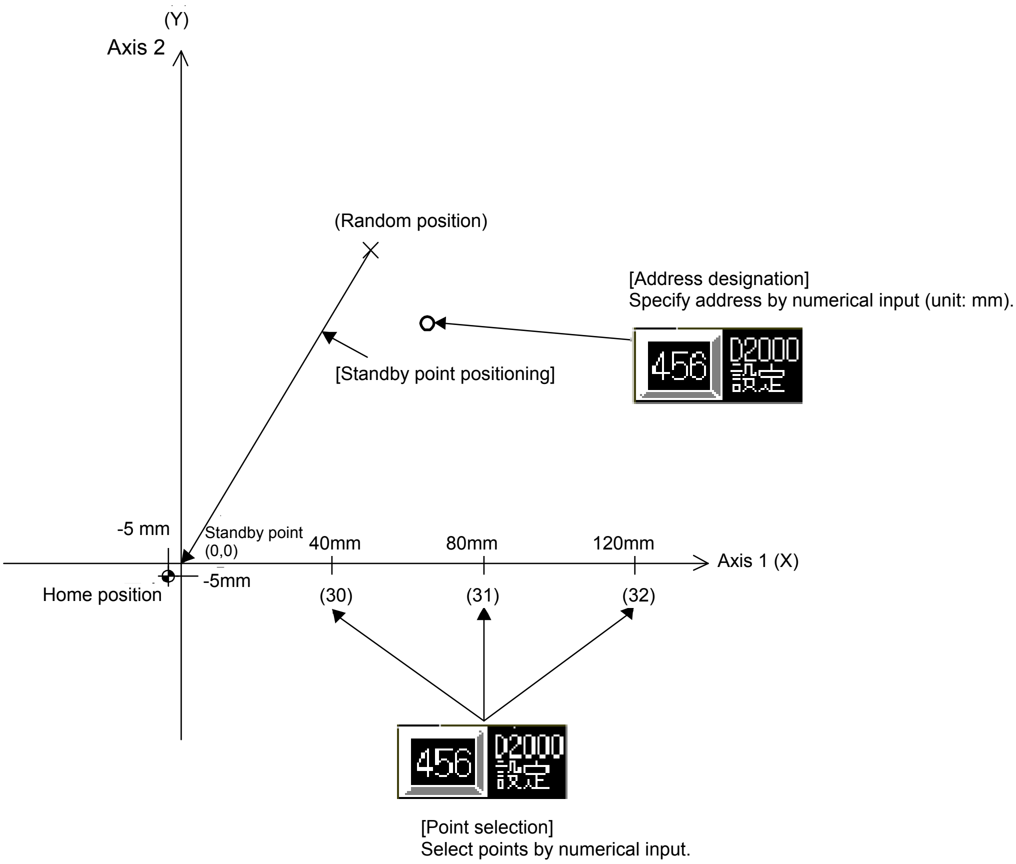


(7) Click [Save As...] on the [Project] menu, and save the project.

Chapter 9 Basic Practice in SV22 Real Mode

9.1 Practice Content

Basic practice involves initial processing, zeroing, and JOG operation. Furthermore, this practice will be based on a basic positioning program example using a motion SFC program.

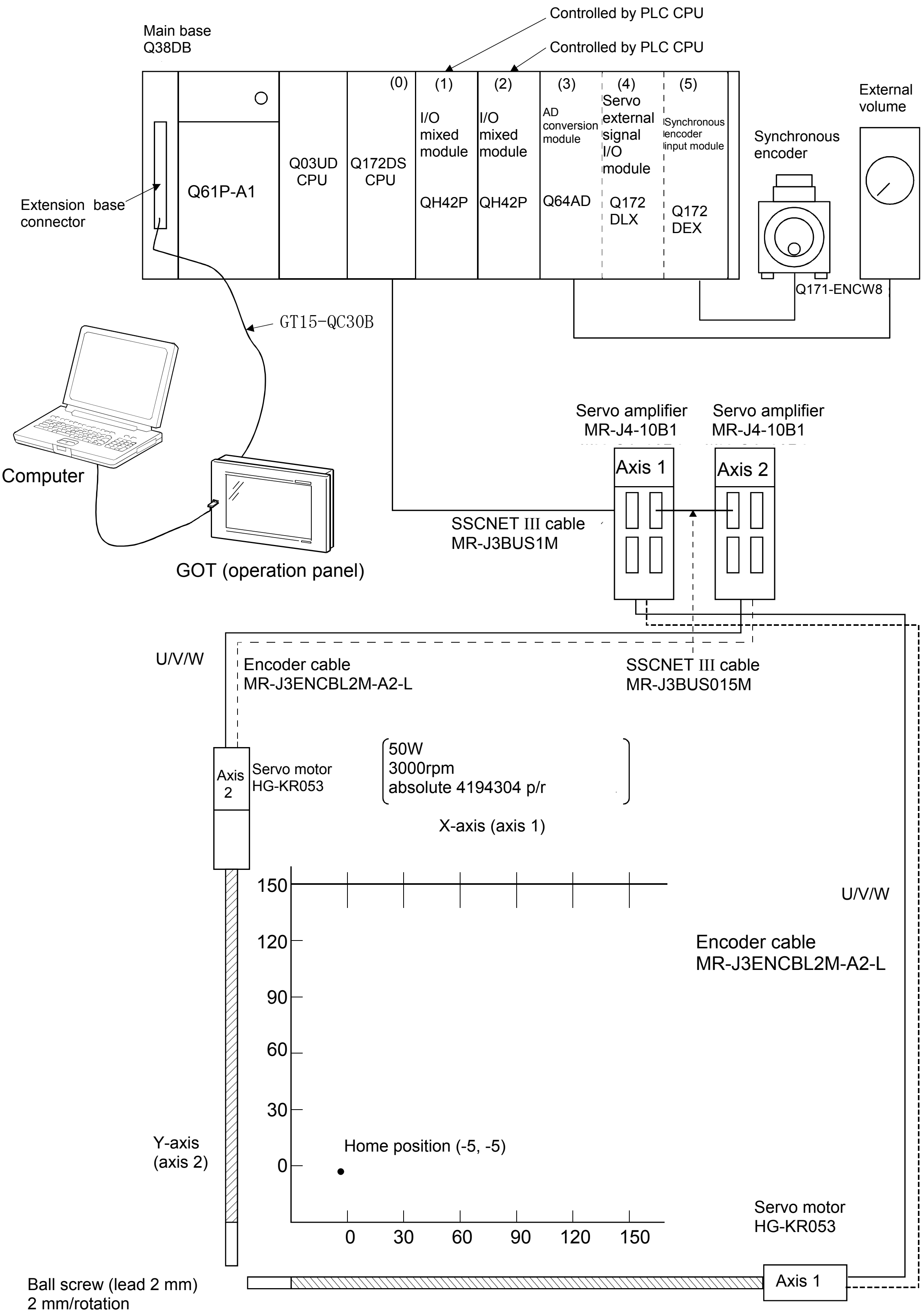


Specify an address by numerical input at the demonstration machine operation panel.

You will practice two positioning methods, one of which involves specifying points, and the other which involves specifying with an X, Y address.

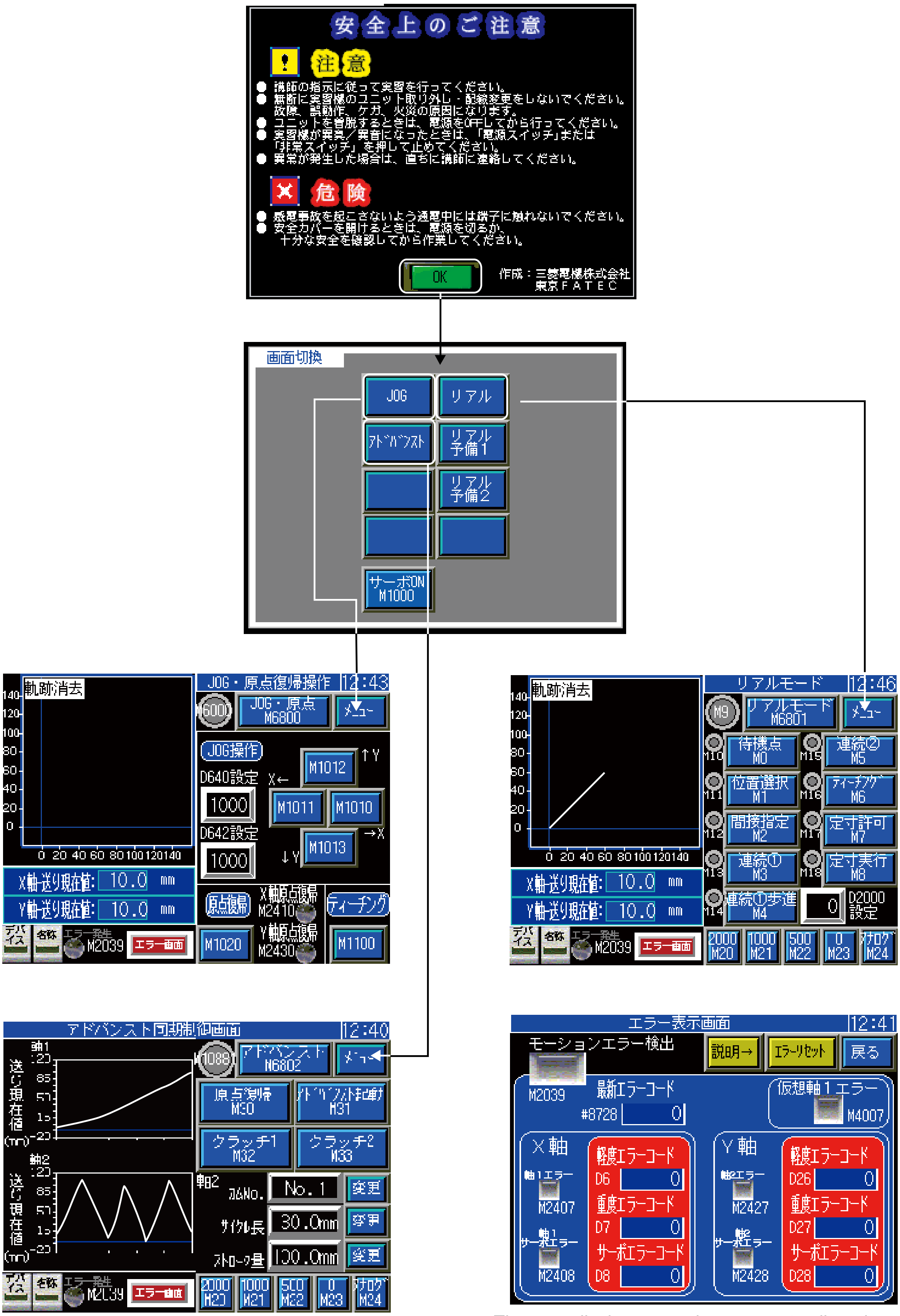
9.2 Q172DSCPU Demonstration Machine System Configuration

In this practice, external signals (boundary limits, DOG) are read using the Q172DLX module.

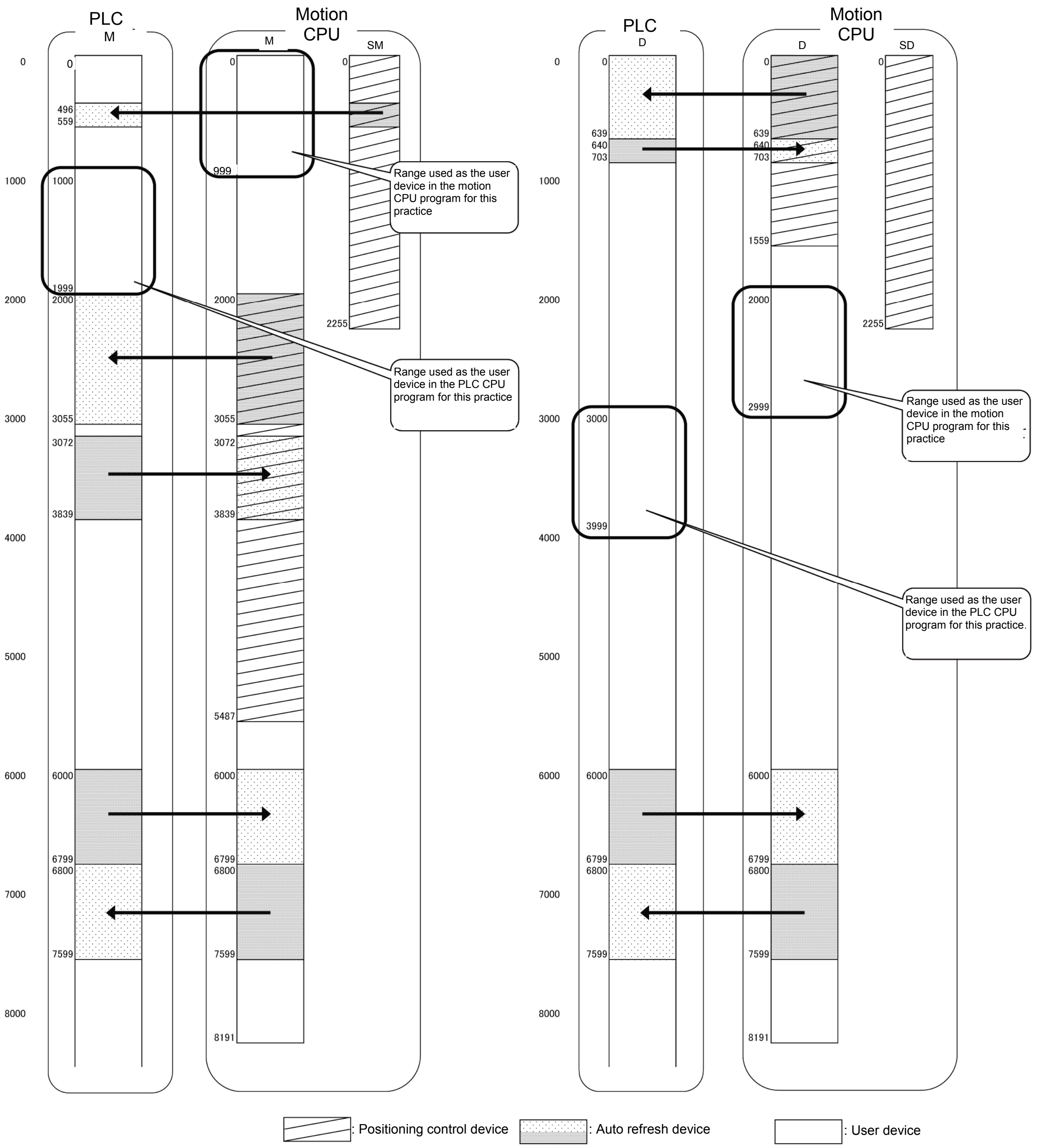


Demonstration machine operation panel

Start screen



The error display screen is common to all modes.

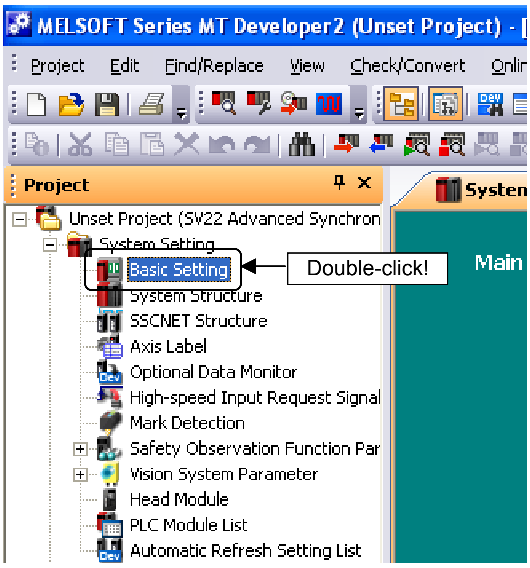


PLC
M
Motion CPU
M
SM
Range used as the user device in the motion CPU program for this practice
Range used as the user device in the PLC CPU program for this practice
PLC
D
Motion CPU
D
SD
Range used as the user device in the motion CPU program for this practice
Range used as the user device in the PLC CPU program for this practice.
: Positioning control device
: Auto refresh device
: User device

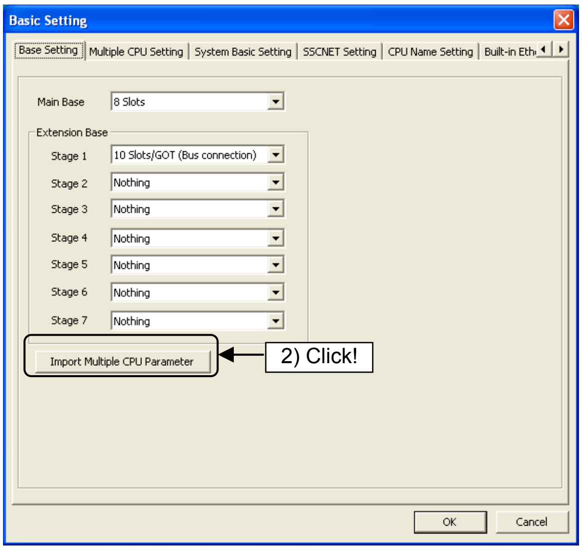
9.3 System Settings

It is first of all necessary to specify system settings at MT Works2.

(1) System settings



1) Double-click the [System Setting] → "Basic Setting" tab in the project window for the new project created at section 8.3.



2) Click the [Import Multiple CPU Parameter] button at the Basic Setting dialog box that appears.

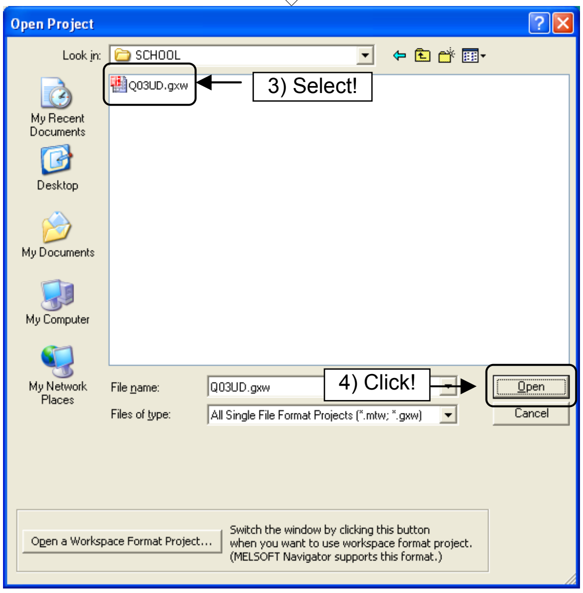
Go to next page

POINT

Import Multiple CPU Parameter
This section introduces the function used to specify settings at the motion CPU side also using the PLC side CPU parameters set at section 8.2.2.
☆ **Setting mistakes will be minimized!**

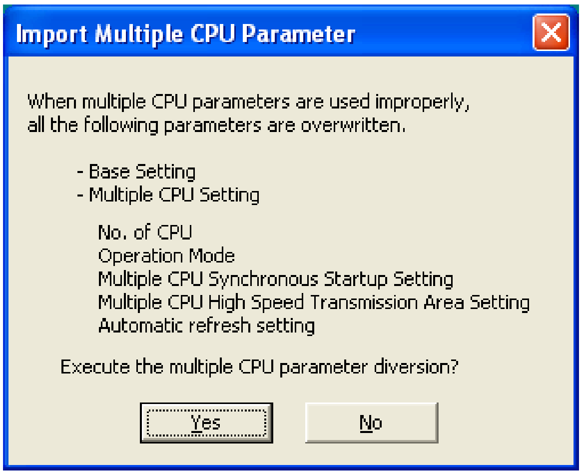
If not using "Import Multiple CPU Parameter"
Click the "Multiple CPU Settings" tab. ⇒ Go to page 9-8.

From previous page

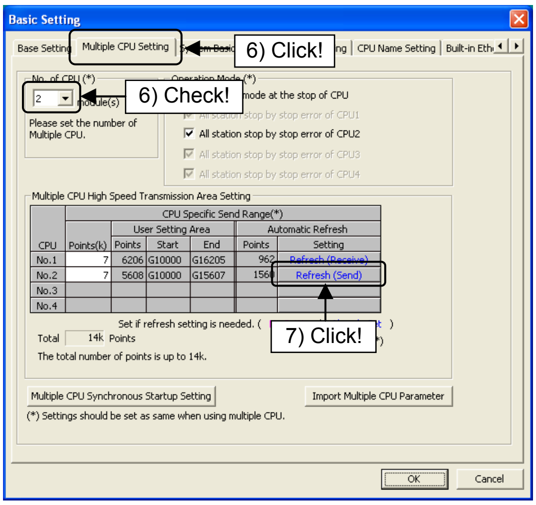


3) Click the Browse button at the dialog box used to open a project, and select a project for which PLC side CPU parameters have been set.

4) Click the Open button.



5) Click the Yes button at the Import Multiple CPU Parameter dialog confirmation message box that appears.

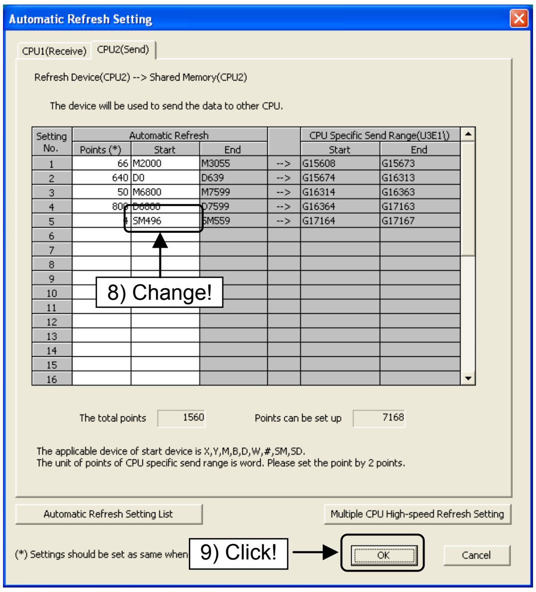


6) Click the "Multiple CPU Setting" tab, and ensure that the "No. of CPU" is "2".

7) Click the CPU No.2 Refresh button in the "Multiple CPU High Speed Transmission Area Setting" tab.

Go to next page

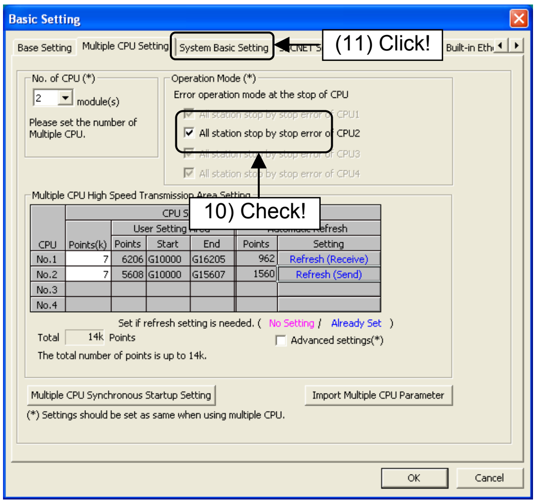
From previous page



8) An Automatic Refresh Setting dialog box for the CPU2 then appears. Change the start of Setting No. 5 as follows.
(A special relay is used at the motion side.)

"M496" → "SM496"

9) Click the OK button.



10) Ensure that "All station stop by stop error of CPU2" is selected at "Operation Mode".

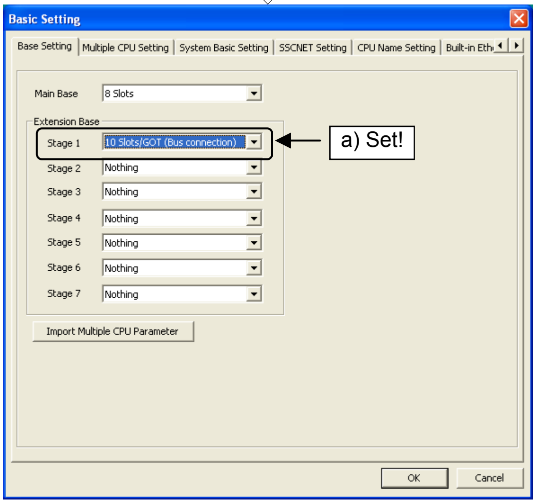
11) Once confirmed, click the "System Basic Setting" tab in the Basic Setting dialog box.

→ Go to 12) on page 9-10.

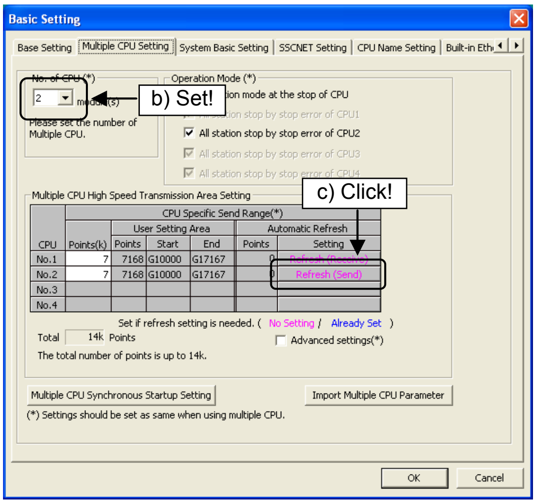
* Pages 9-8 to 9-9 describe the setting method when not reusing multiple CPU parameters.

[If not using "Reuse Multiple CPU Parameters"]

From page 9-5

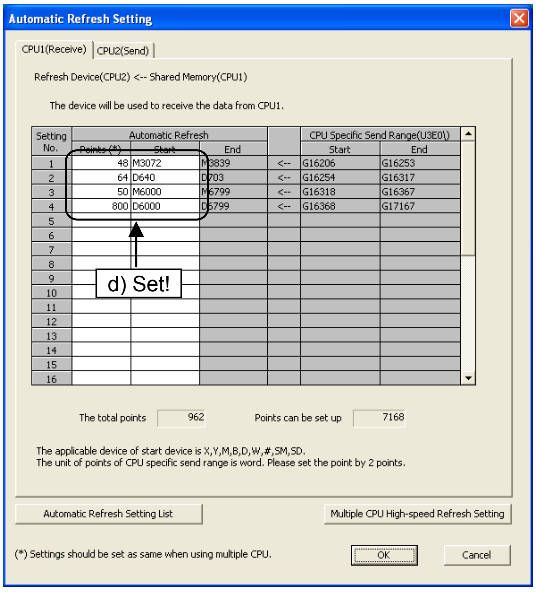


a) Set the "1st row" of the "Extension Base" to "10 Slots/GOT (Bus connection)".



b) Set "No. of CPU" to "2".

c) Click the CPU No.1 Refresh button at "Multiple CPU High Speed Transmission Area Setting".

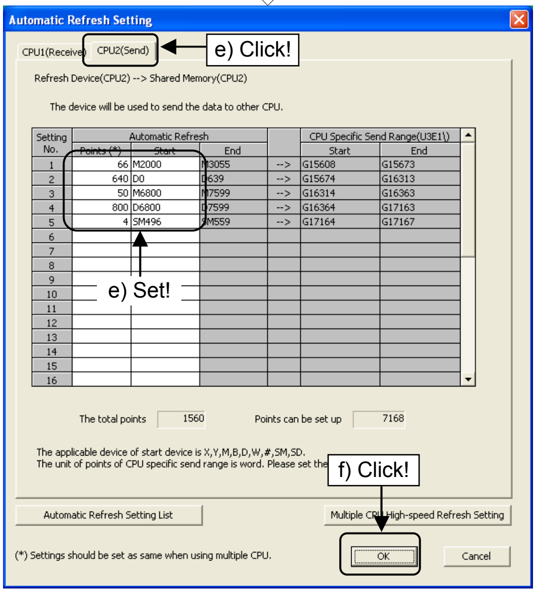


d) An Automatic Refresh Setting dialog box then appears. Specify the automatic refresh settings for the CPU1 as follows.

"Setting No.1 - Points" : "48"
"Setting No.1 - Start" : "M3072"
"Setting No.2 - Points" : "64"
"Setting No.2 - Start" : "D640"
"Setting No.3 - Points" : "50"
"Setting No.3 - Start" : "M6000"
"Setting No.4 - Points" : "800"
"Setting No.4 - Start" : "D6000"

Go to next page

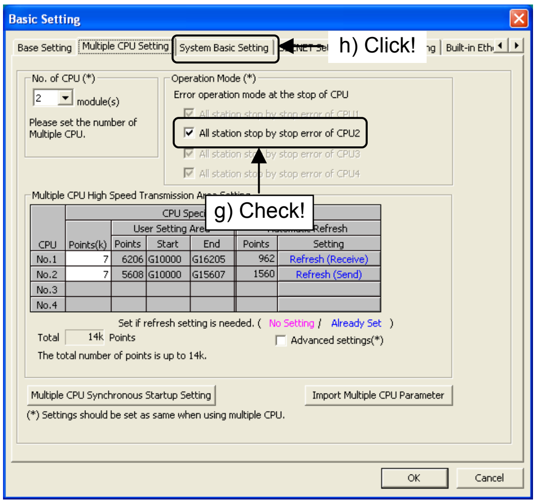
From previous page



e) Click the "CPU2" tab, and specify the automatic refresh settings for the CPU No.2 as follows.

"Setting No.1 - Points" : "66"
"Setting No.1 - Start" : "M2000"
"Setting No.2 - Points" : "640"
"Setting No.2 - Start" : "D0"
"Setting No.3 - Points" : "50"
"Setting No.3 - Start" : "M6800"
"Setting No.4 - Points" : "800"
"Setting No.4 - Start" : "D6800"
"Setting No.5 - Points" : "4"
"Setting No.5 - Start" : "SM496"

f) When settings are complete, click the OK button.

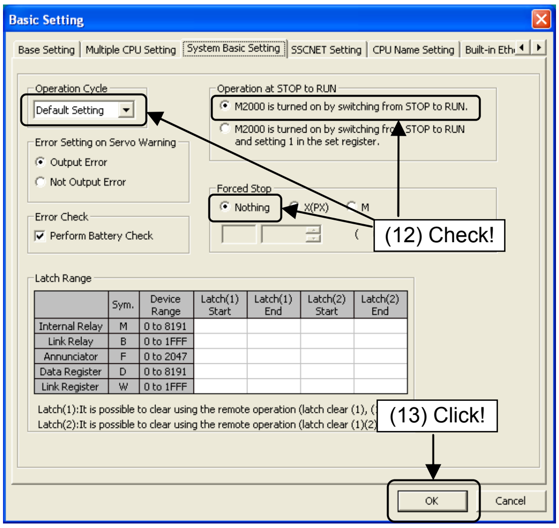


g) The display then returns to the Basic Setting dialog box. Ensure that "All station stop by stop error of CPU2" is selected at "Operation Mode".

h) Once confirmed, click the "System Basic Setting" tab in the Basic Setting dialog box.

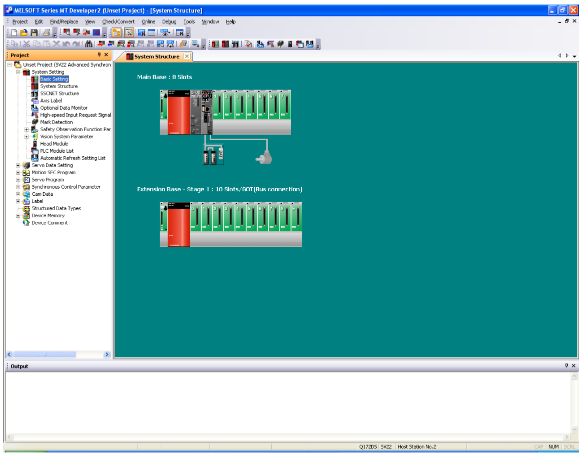
Go to next page

From previous page



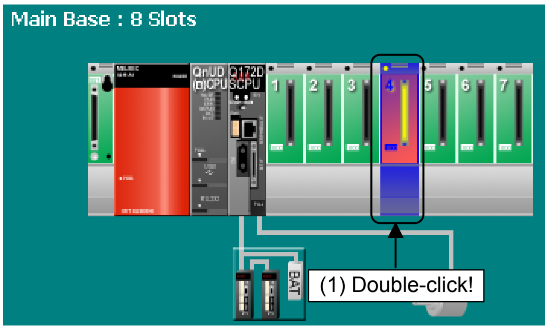
12) Ensure that the following settings are as shown.
"Forced Stop" : Nothing
"Operation Cycle" : Default Setting
"Operation at STOP to RUN"
: M2000 is turned on by switching from STOP to RUN.

13) Once set, click the OK button at the Basic Setting dialog box.

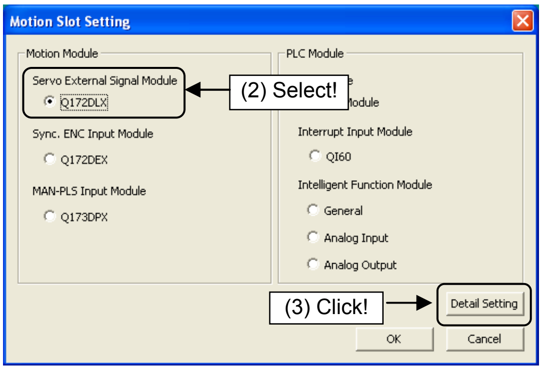


(14) Basic setting is now complete.
System configuration settings are described at (2) from the following page.

(2) Motion slot settings

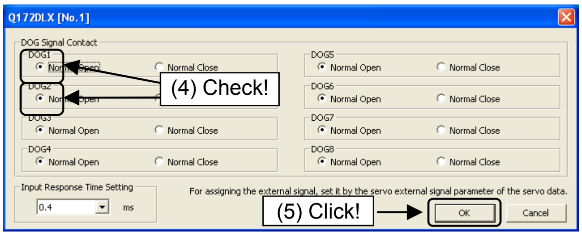


1) To specify settings for the slot 4 servo external signal input module, double-click main base slot 4 in the system configuration window.



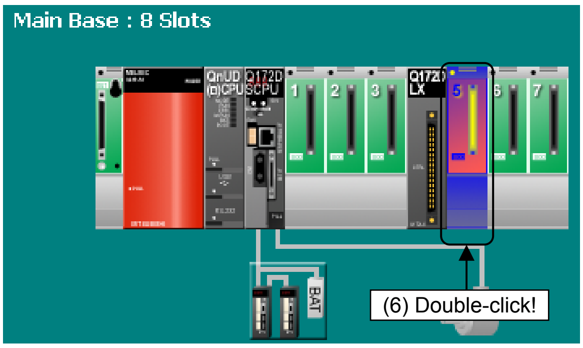
2) A Motion Slot Setting dialog box then appears. Select "Servo External Signal Module" - "Q172DLX" at "Motion Module".

3) When settings are complete, click the Detail Setting button.



4) A Q172DLX Setting dialog box then appears. Ensure that the DOG signal contacts are set as follows.
- DOG1: Normal Open
- DOG2: Normal Open

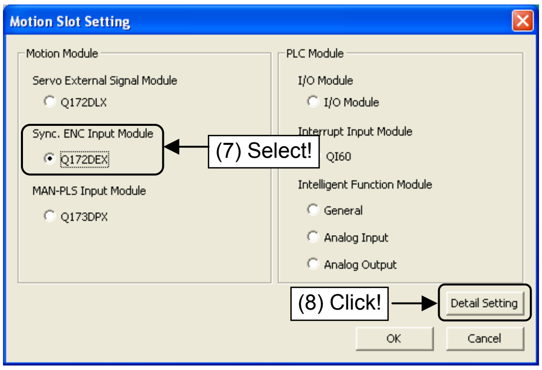
5) When settings are complete, click the OK button.
The display then returns to the Motion Setting dialog box. Click the OK button.



6) To specify settings for the slot 5 synchronous encoder input module, double-click main base slot 5 in the system settings window.

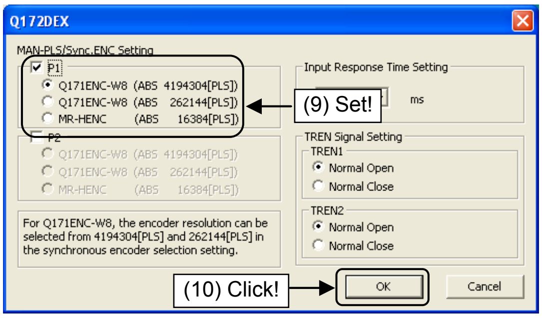
Go to next page

From previous page



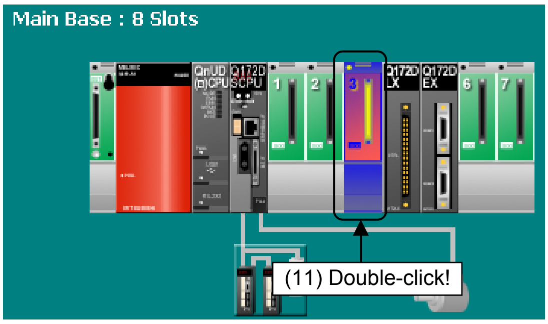
7) A Motion Slot Setting dialog box then appears. Select "Sync. ENC. Input Module" - "Q172DEX" at "Motion Module".

8) When settings are complete, click the Detail Setting button.

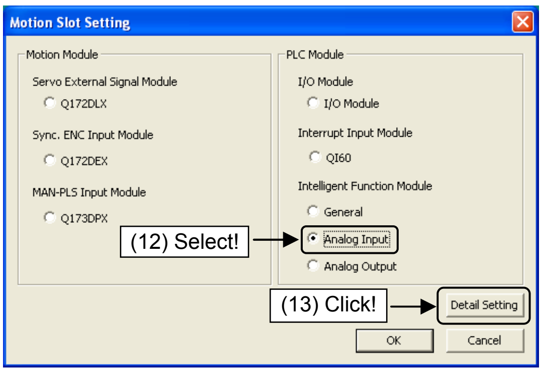


9) A Q172DEX Setting dialog box then appears. Select the "P1" check box at "MAN-PLS/Sync. ENC Setting", and then select "Q171ENC-W8 (ABS 4194304[PLS])" (set "High-speed Read Data Setting" to "Not used", and "Input Response Time (operation mode)" to "0.4".)

10) When settings are complete, click the OK button.
The display then returns to the Motion Setting dialog box. Click the OK button.



11) To specify settings for the slot 3 analog input module, double-click main base slot 3 in the system settings window.

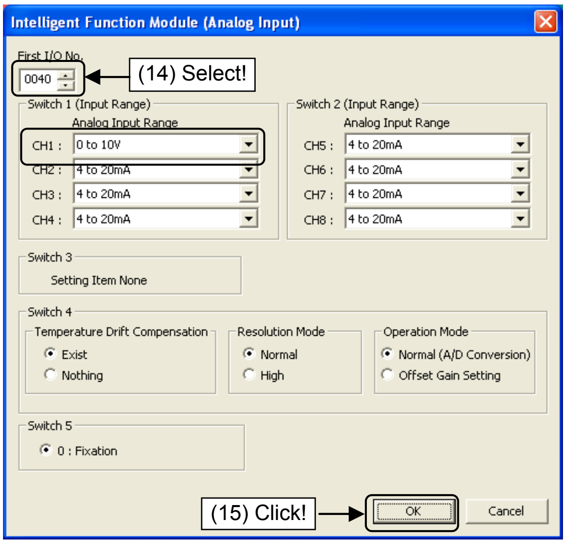


12) A Motion Slot Setting dialog box then appears. Select "Analog Input" at "PLC Module".

13) When settings are complete, click the Detail Setting button.

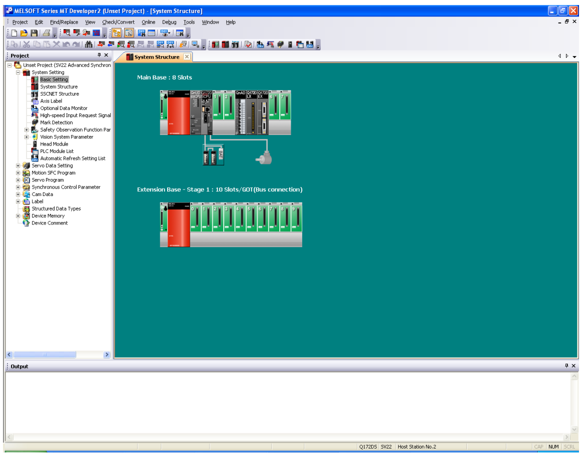
Go to next page

From previous page



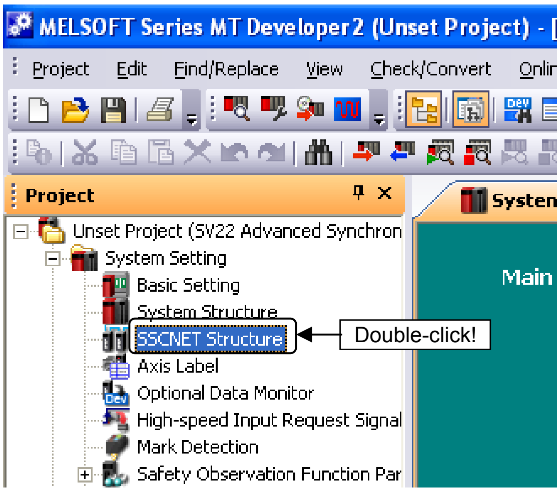
14) An Analog Module Setting dialog box then appears. Select "0040" at "First I/O No.", and then specify the following setting.
    Switch 1
    "CH1": 0 to 10V

15) When settings are complete, click the OK button.
    The display then returns to the Motion Setting dialog box. Click the OK button.

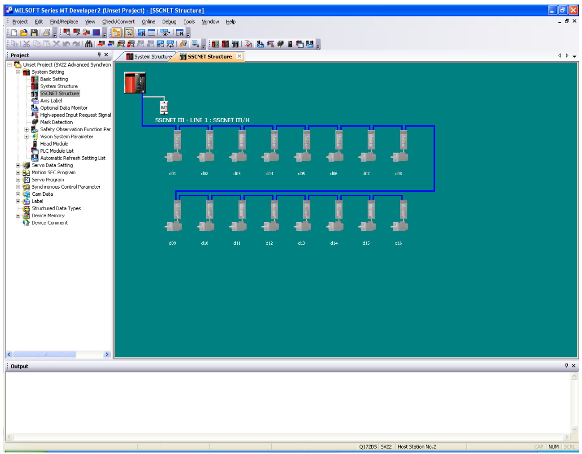


16) System configuration settings are now complete.

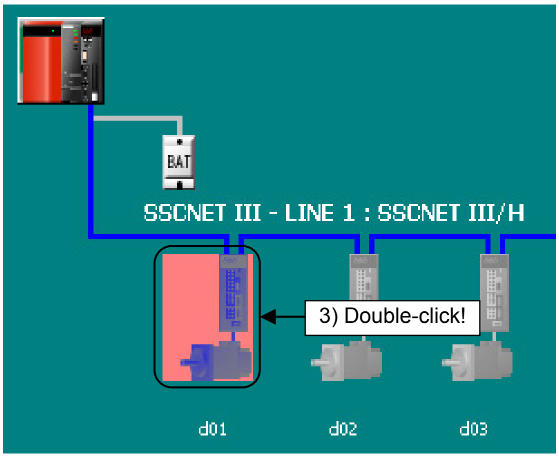
(3) Amplifier settings



1) Double-click [System Setting] → [SSCNET Structure] in the Project window.



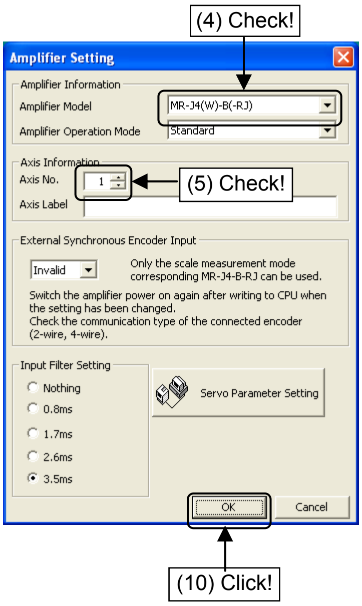
2) An SSCNET Structure window appears.



3) To specify settings for the first servo amplifier and servo motor, double-click the first (d01) servo amplifier from the left in the SSCNET Structure window.

Go to next page

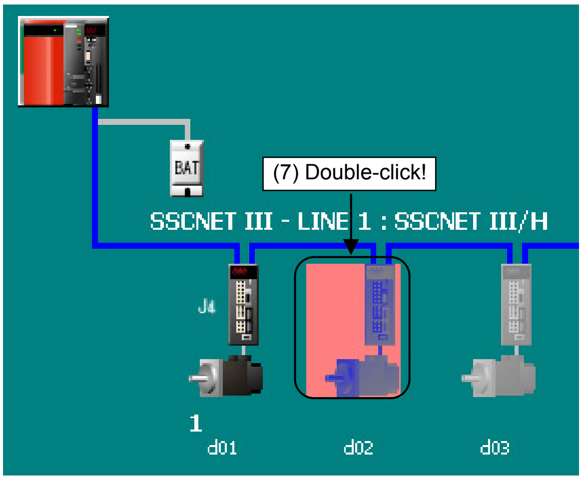
From previous page



4) An Amplifier Setting dialog box then appears. Ensure that the "Amplifier Model" is "MR-J4(W)-B".

5) Ensure that the "Axis No." is "1".

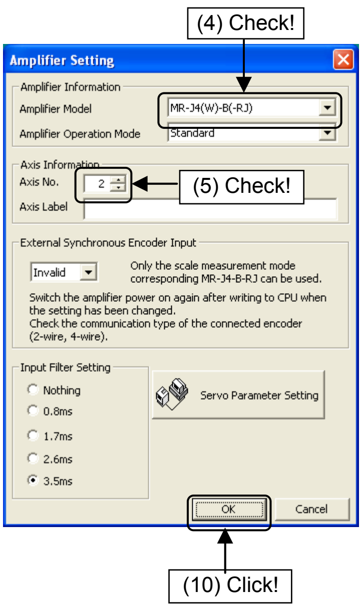
6) Once set, click the OK button at the Amplifier Setting dialog box.



7) To then specify settings for the second servo amplifier and servo motor, double-click the second (d02) servo amplifier from the left in the System Setting window.

Go to next page

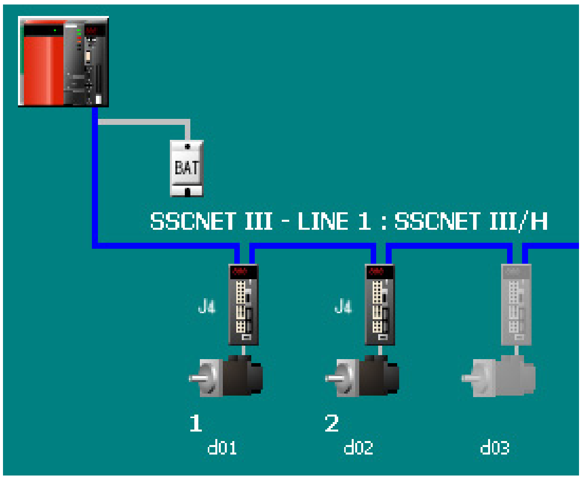
From previous page



(8) An Amplifier Setting dialog box then appears. Ensure that the "Amplifier Model" is "MR-J4(W)-B".

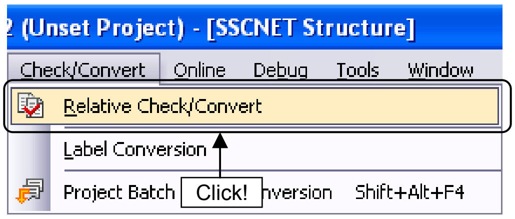
(9) Ensure that the "Axis No." is "2".

(10) Once set, click the OK button at the Amplifier Setting dialog box.

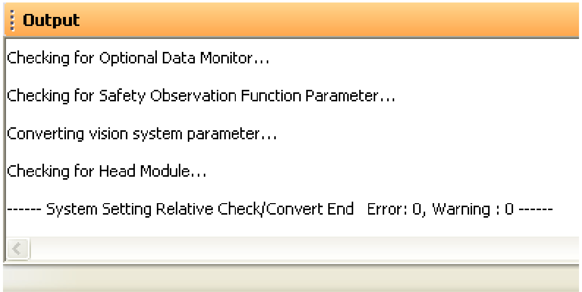


(11) Settings for the first (d01) and second (d02) servo amplifier and servo motor are now complete.

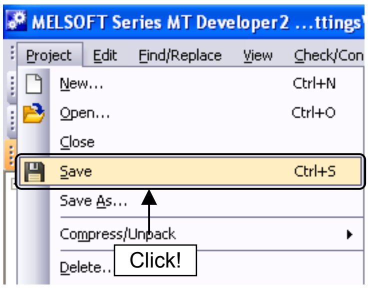
(4) Relativity check, saving



1) When motion slot settings and amplifier settings are complete, click [Relative Check/ Convert] on the [Check/Convert] menu.



2) Ensure that there are no errors at the output window.
If any error items are displayed in the output window, edit the setting(s) and retry the relativity check.

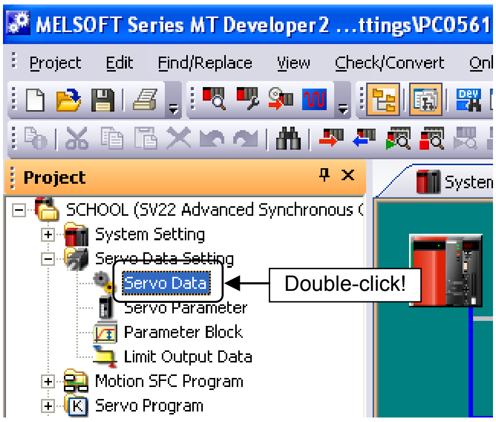


3) Click [Save] on the [Project] menu.

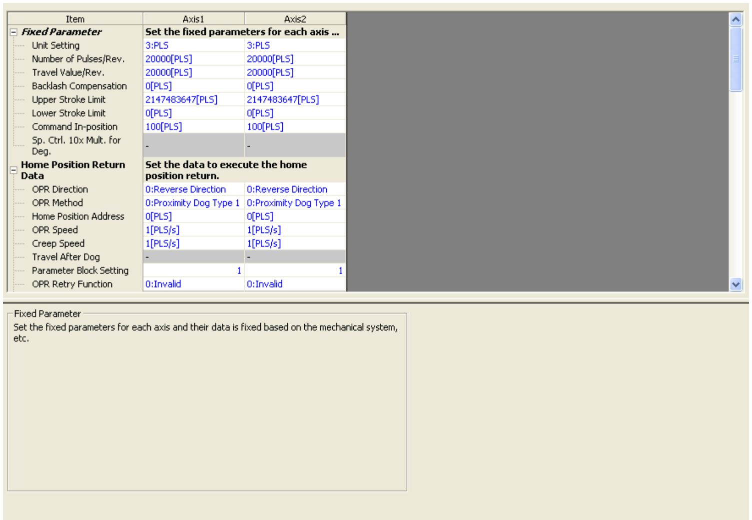
System settings are now complete.

9.4 Servo Data Input Operation

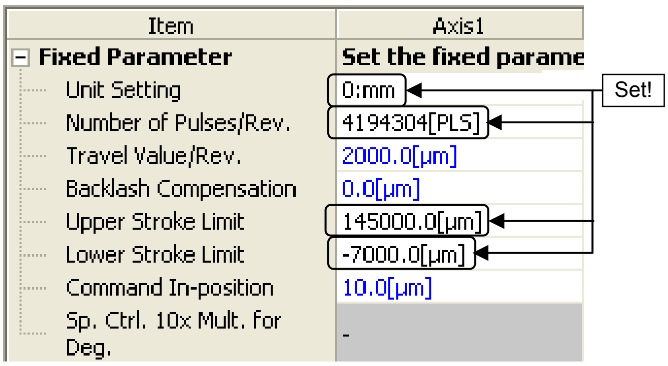
After specifying system settings, specify servo data settings.



(1) Double-click [Servo Data Setting] → [Servo Data] in the project window.



(2) A Servo Data Setting window appears.



(3) Specify the content shown on the left for the Axis 1 Fixed Parameter.

Go to next page

From previous page

Item	Axis1
Home Position Return Data	**Set the data to execu position return.**
OPR Direction	0:Reverse Direction
OPR Method	4:Proximity Dog Type 2
Home Position Address	-5000.0[μm]
OPR Speed	1000.00[mm/min]
Creep Speed	100.00[mm/min]
Travel After Dog	-
Parameter Block Setting	1
OPR Retry Function	1:Valid
Dwell Time at OPR Retry	0[ms]
Home Position Shift Amount	-5000.0[μm]
Speed Set at Home Pos. Shift	0:OPR Speed
Torque Limit Value at Creep Speed	-
Operation for OPR Incompletion	0:Execute Servo Program
OPR Request Setting in Pulse Conversion Unit	-
Standby Time after Clear Signal Output in Pulse C...	-

Set!

(4) Specify the content shown on the left for the Axis 1 Home Position Return Data settings.

Item	Axis1
JOG Operation Data	**Set the data to execu**
JOG Speed Limit Value	10000.00[mm/min]
Parameter Block Setting	1

Set!

(5) Specify the content shown on the left for the Axis 1 JOG Operation Data settings.

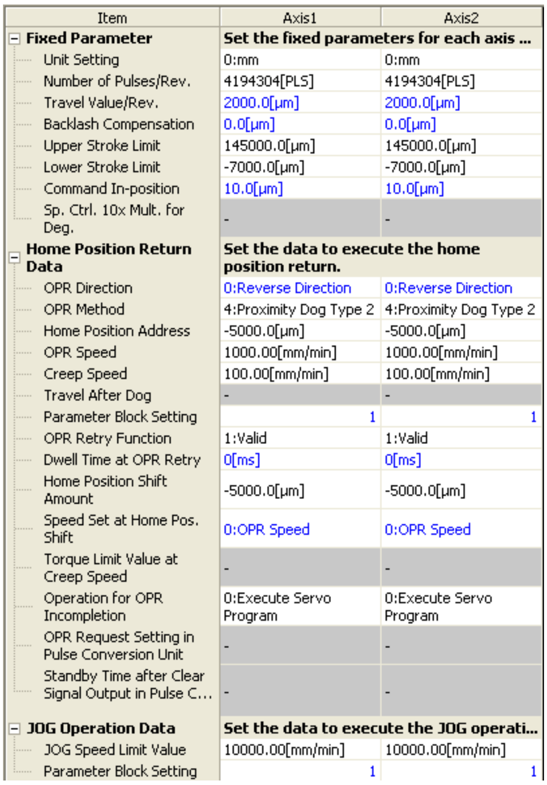
Item	Axis1	Axis2
Servo External Signal Parameter	**It is the parameter of setting servo external signal (FLS/RLS/STOP/DOG) to...**	
FLS Signal	**Set the signal type and the signal/cont...**	
Signal Type	1:Q172DLX Signal	1:Q172DLX Signal
Q172DLX Module No.	1	1
Q172DLX Signal No.	1	2
Device	-	-
Contact	1:Normal Close	1:Normal Close
RLS Signal	**Set the signal type and the signal/cont...**	
Signal Type	1:Q172DLX Signal	1:Q172DLX Signal
Q172DLX Module No.	1	1
Q172DLX Signal No.	1	2
Device	-	-
Contact	1:Normal Close	1:Normal Close
STOP Signal	**Set the signal type and signal contact t...**	
Signal Type	1:Q172DLX Signal	1:Q172DLX Signal
Q172DLX Module No.	1	1
Q172DLX Signal No.	1	2
Device	-	-
Contact	0:Normal Open	0:Normal Open
DOG Signal	**Set the signal type and signal contact t...**	
Signal Type	1:Q172DLX Signal	1:Q172DLX Signal
Q172DLX Module No./DI Signal Name	1	1
Q172DLX Signal No.	1	2
Device	-	-
Contact	0:Normal Open	0:Normal Open

(6) Specify the content shown on the left for the Axis 1 Servo External Signal Parameter.

Note: The values set for Axis 1 differ from those for Axis 2, and therefore care should be taken if copying Axis 1 settings to Axis 2.

Go to next page

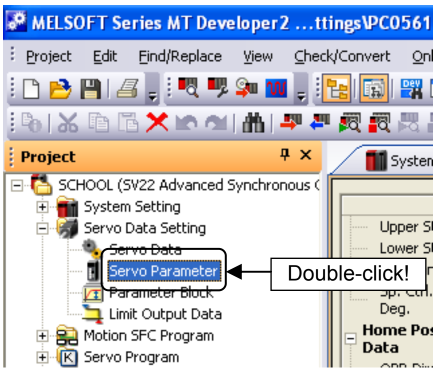
From previous page



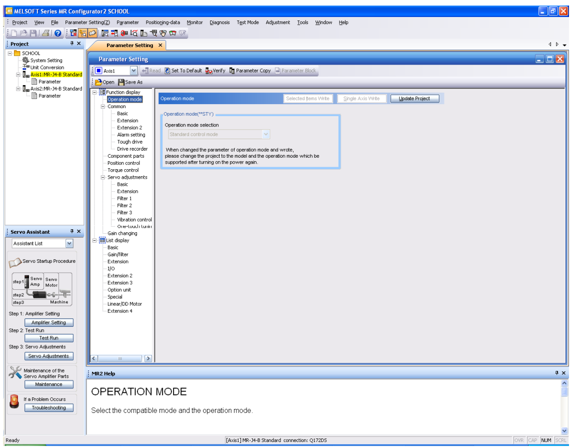
Item	Axis1	Axis2
Fixed Parameter	**Set the fixed parameters for each axis ...**	
Unit Setting	0:mm	0:mm
Number of Pulses/Rev.	4194304[PLS]	4194304[PLS]
Travel Value/Rev.	2000.0[μm]	2000.0[μm]
Backlash Compensation	0.0[μm]	0.0[μm]
Upper Stroke Limit	145000.0[μm]	145000.0[μm]
Lower Stroke Limit	-7000.0[μm]	-7000.0[μm]
Command In-position	10.0[μm]	10.0[μm]
Sp. Ctrl. 10x Mult. for Deg.	-	-
Home Position Return Data	**Set the data to execute the home position return.**	
OPR Direction	0:Reverse Direction	0:Reverse Direction
OPR Method	4:Proximity Dog Type 2	4:Proximity Dog Type 2
Home Position Address	-5000.0[μm]	-5000.0[μm]
OPR Speed	1000.00[mm/min]	1000.00[mm/min]
Creep Speed	100.00[mm/min]	100.00[mm/min]
Travel After Dog	-	-
Parameter Block Setting	1	1
OPR Retry Function	1:Valid	1:Valid
Dwell Time at OPR Retry	0[ms]	0[ms]
Home Position Shift Amount	-5000.0[μm]	-5000.0[μm]
Speed Set at Home Pos. Shift	0:OPR Speed	0:OPR Speed
Torque Limit Value at Creep Speed	-	-
Operation for OPR Incompletion	0:Execute Servo Program	0:Execute Servo Program
OPR Request Setting in Pulse Conversion Unit	-	-
Standby Time after Clear Signal Output in Pulse C...	-	-
JOG Operation Data	**Set the data to execute the JOG operati...**	
JOG Speed Limit Value	10000.00[mm/min]	10000.00[mm/min]
Parameter Block Setting	1	1

(7) Use the same operation to specify the content shown below for the Axis 2 Fixed Parameter, Home Position Return Data, and JOG Operation Data.

POINT
By right-clicking the screen on the left, blocks can be copied and pasted.



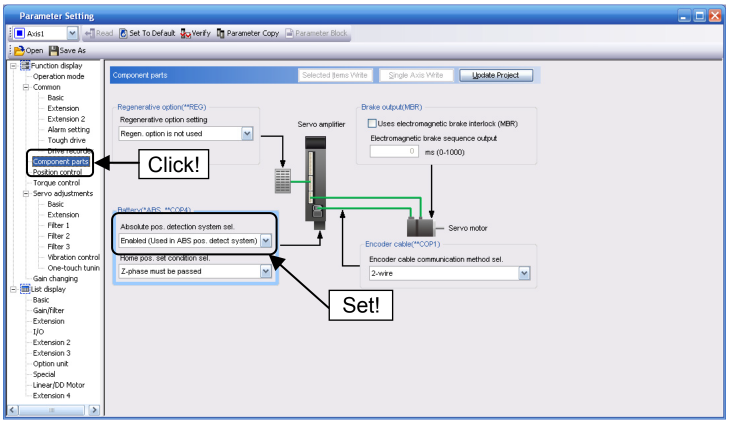
(8) Double-click [Servo Data Setting] → [Servo Parameter] in the Project window.



(9) MR Configurator2 starts up.
MR Configurator2 is software used to set servo amplifier parameters and so on.

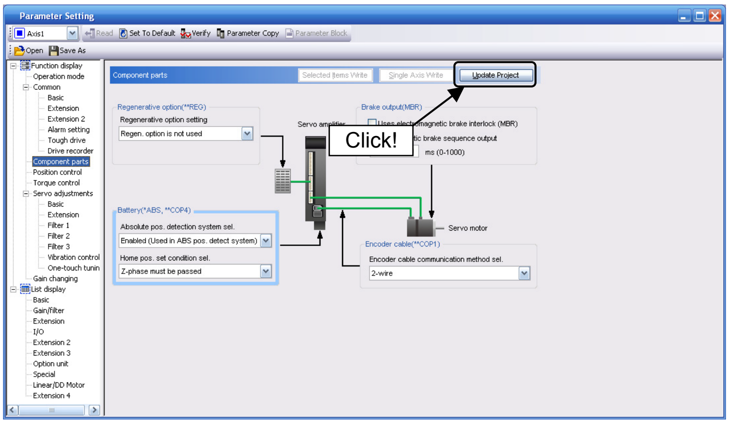
Go to next page

From previous page

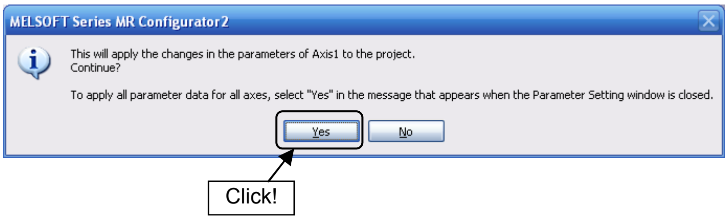


(10) Click [Function display] → [Component parts] in the Parameter Setting screen display selection tree, and then specify the following settings.

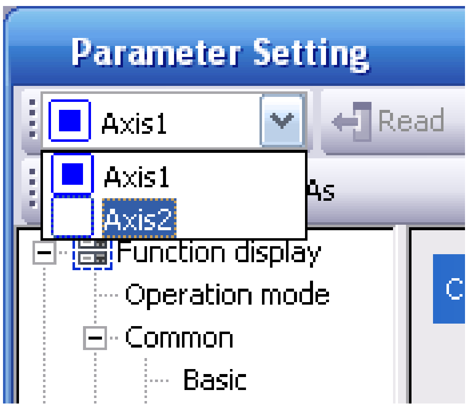
Absolute pos. detection system sel.
: Enabled (Used in ABS pos. detect system)



(11) Click the [Update Project] button.



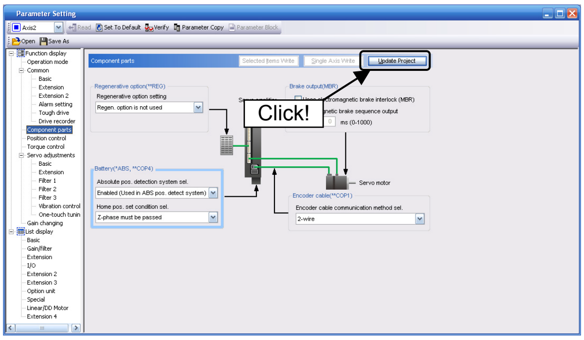
(12) Click the [Yes] button to update the servo parameter changes.



(13) Switch to Axis 2 and set the parameters in the same manner.

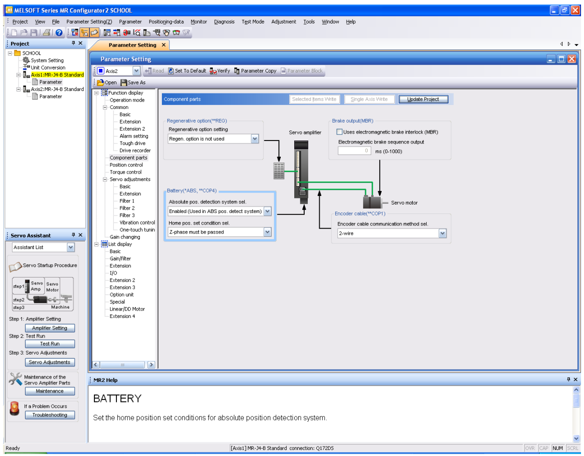
Go to next page

From previous page

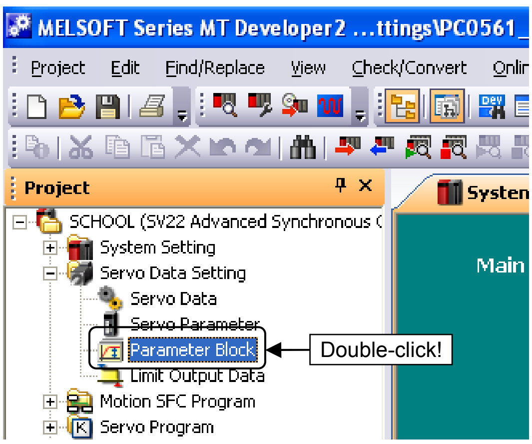


(14) Click the [Update Project] button.

(15) Click the [Yes] button to update the servo parameter changes.



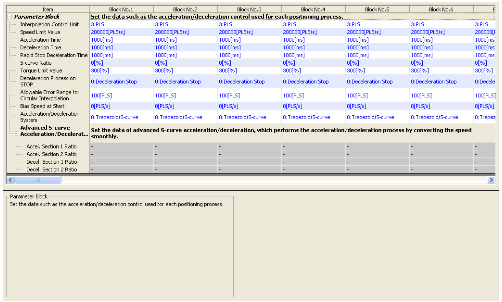
(16) Exit MR Configurator2.



(17) Double-click [Servo Data Setting] → [Parameter Block] in the Project window.

Go to next page

From previous page



(18) The Parameter Block Setting screen appears.

(19) Specify Parameter Block No.1 settings as shown on the left.

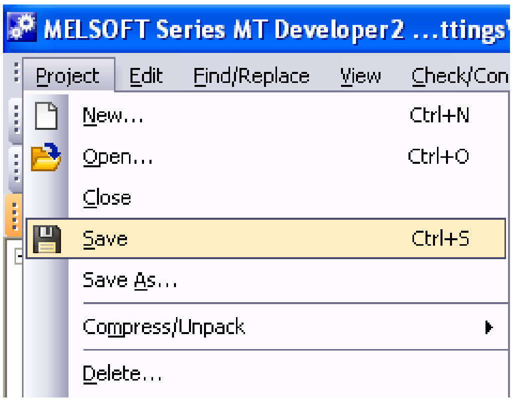
Item	Block No.1
Parameter Block	**Set the data such as th**
Interpolation Control Unit	0:mm
Speed Limit Value	10000.00[mm/min]
Acceleration Time	100[ms]
Deceleration Time	150[ms]
Rapid Stop Deceleration Time	50[ms]
S-curve Ratio	50[%]
Torque Limit Value	300[%]
Deceleration Process on STOP	1:Rapid Stop
Allowable Error Range for Circular Interpolation	10.0[μm]
Bias Speed at Start	0.00[mm/min]
Acceleration/Deceleration System	0:Trapezoid/S-curve
Advanced S-curve Acceleration/Decelerat...	**Set the data of advanc**
Accel. Section 1 Ratio	-
Accel. Section 2 Ratio	-
Decel. Section 1 Ratio	-
Decel. Section 2 Ratio	-

(20) Specify Parameter Block No.2 settings as shown on the left.

Item	Block No.2
Parameter Block	**Set the data such as th**
Interpolation Control Unit	3:PLS
Speed Limit Value	13107200[PLS/s]
Acceleration Time	400[ms]
Deceleration Time	400[ms]
Rapid Stop Deceleration Time	50[ms]
S-curve Ratio	50[%]
Torque Limit Value	300[%]
Deceleration Process on STOP	1:Rapid Stop
Allowable Error Range for Circular Interpolation	100[PLS]
Bias Speed at Start	0[PLS/s]
Acceleration/Deceleration System	0:Trapezoid/S-curve
Advanced S-curve Acceleration/Decelerat...	**Set the data of advanc converting the speed s**
Accel. Section 1 Ratio	-
Accel. Section 2 Ratio	-
Decel. Section 1 Ratio	-
Decel. Section 2 Ratio	-

Go to next page

From previous page



(21) When all servo data settings are complete, click [Save] on the [Project] menu.
Servo data settings are now complete.

9.5 Practice Motion SFC Programs

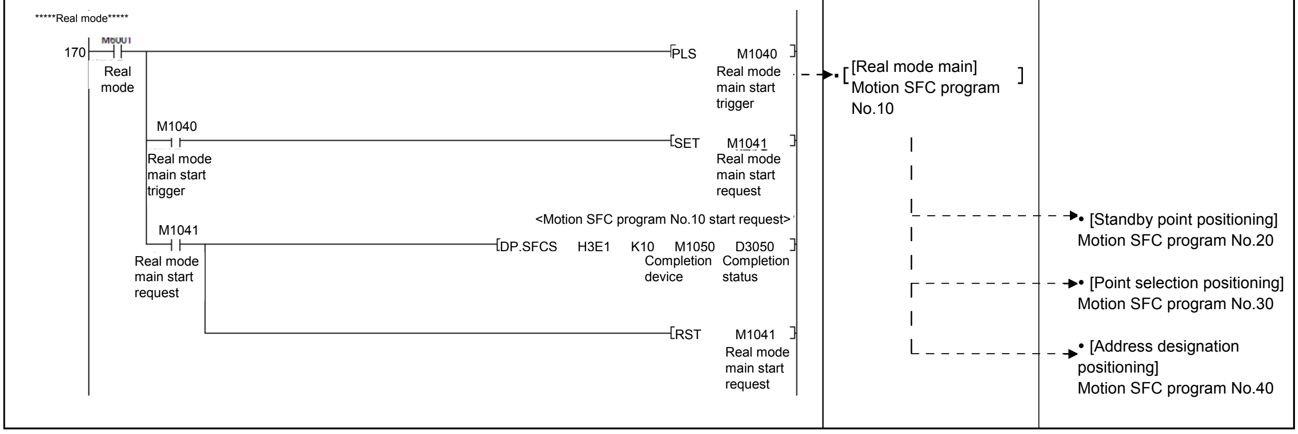
These sequence/motion SFC programs have been created for operation purposes on the assumption that MT Works2 (for Q172DSCPU) be used.
An explanatory drawing of the demonstration machine GOT operation panel is shown in item 9.2.

9.5.1 Program list

The sequence program and motion SFC program used for practice are shown in the following list.
Initial processing, operation type selection, JOG operation, zeroing, and motion SFC program startup are performed from the sequence program. Standby point positioning, positioning by selecting positioning points at the GOT operation panel, and positioning by entering positioning addresses at the GOT operation panel are practiced using the motion SFC program. Refer to the respective descriptions of each program in this manual for details.

Normal execution	Startup with sequence program	Startup with motion SFC program

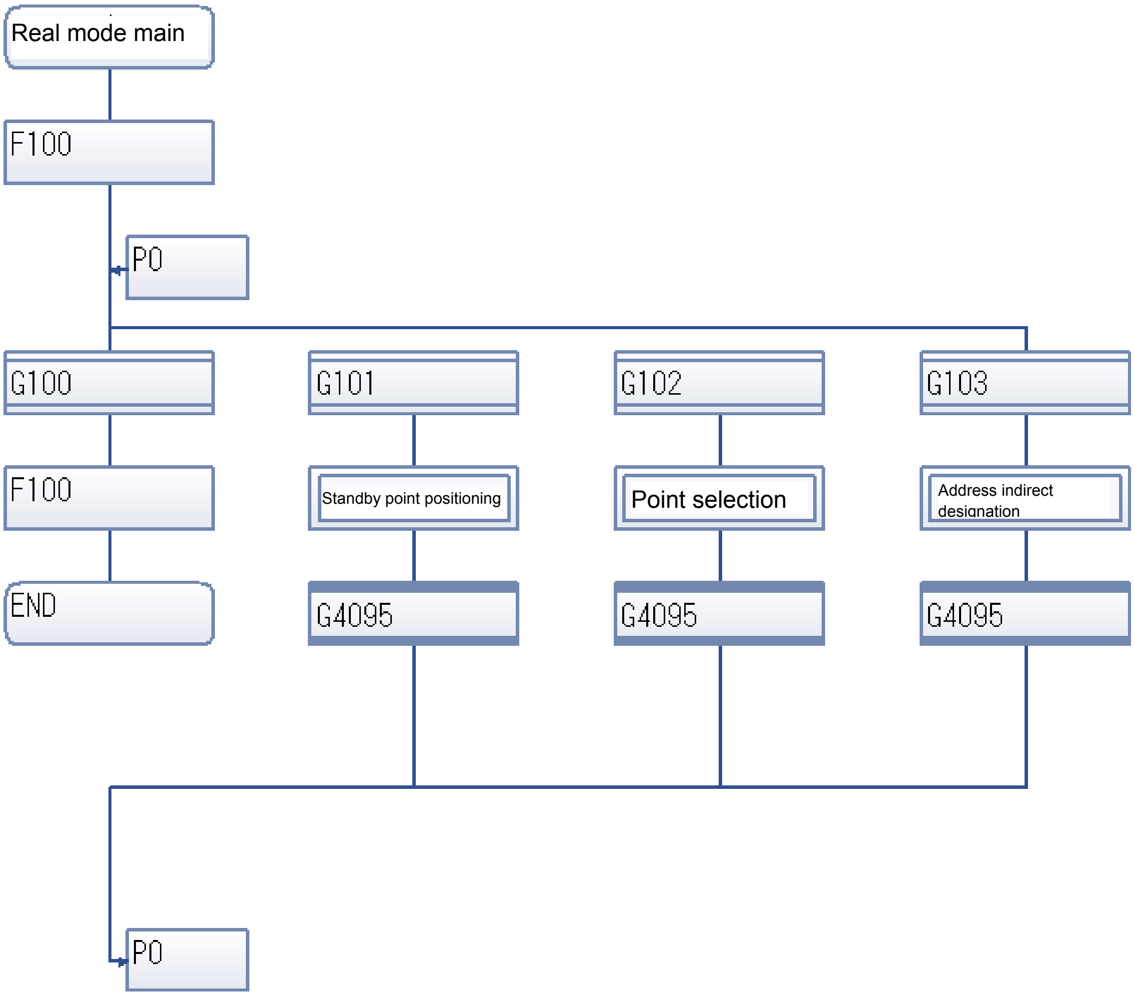
· Sequence program



Motion SFC program parameters

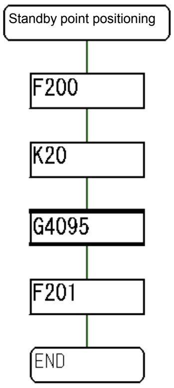
No.	Program name	Automatic start	END operation	No. of transitions	Execution timing
10	Real mode main	No	-	3	Normal
20	Standby point positioning	No	-	3	Normal
30	Point selection	No	-	3	Normal
40	Address indirect designation	No	-	3	Normal

• Start program from sequence program

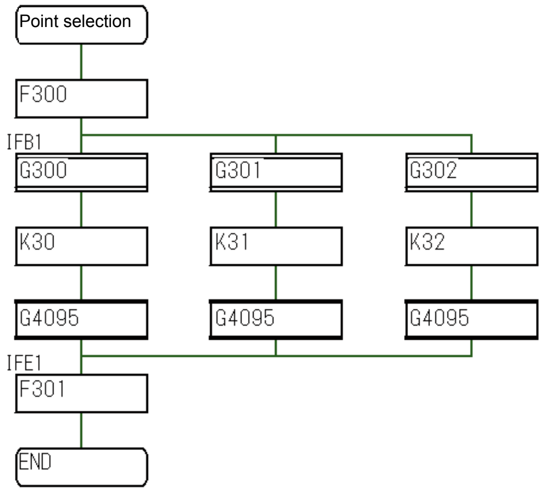


• Start program from motion SFC program [Real mode main] program No.10

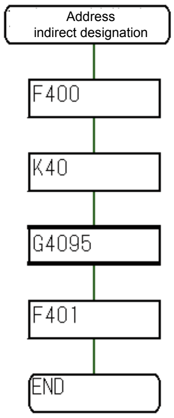
[Standby point positioning] Program No.20



[Point selection positioning] Program No.30



[Address indirect designation] Program No.40

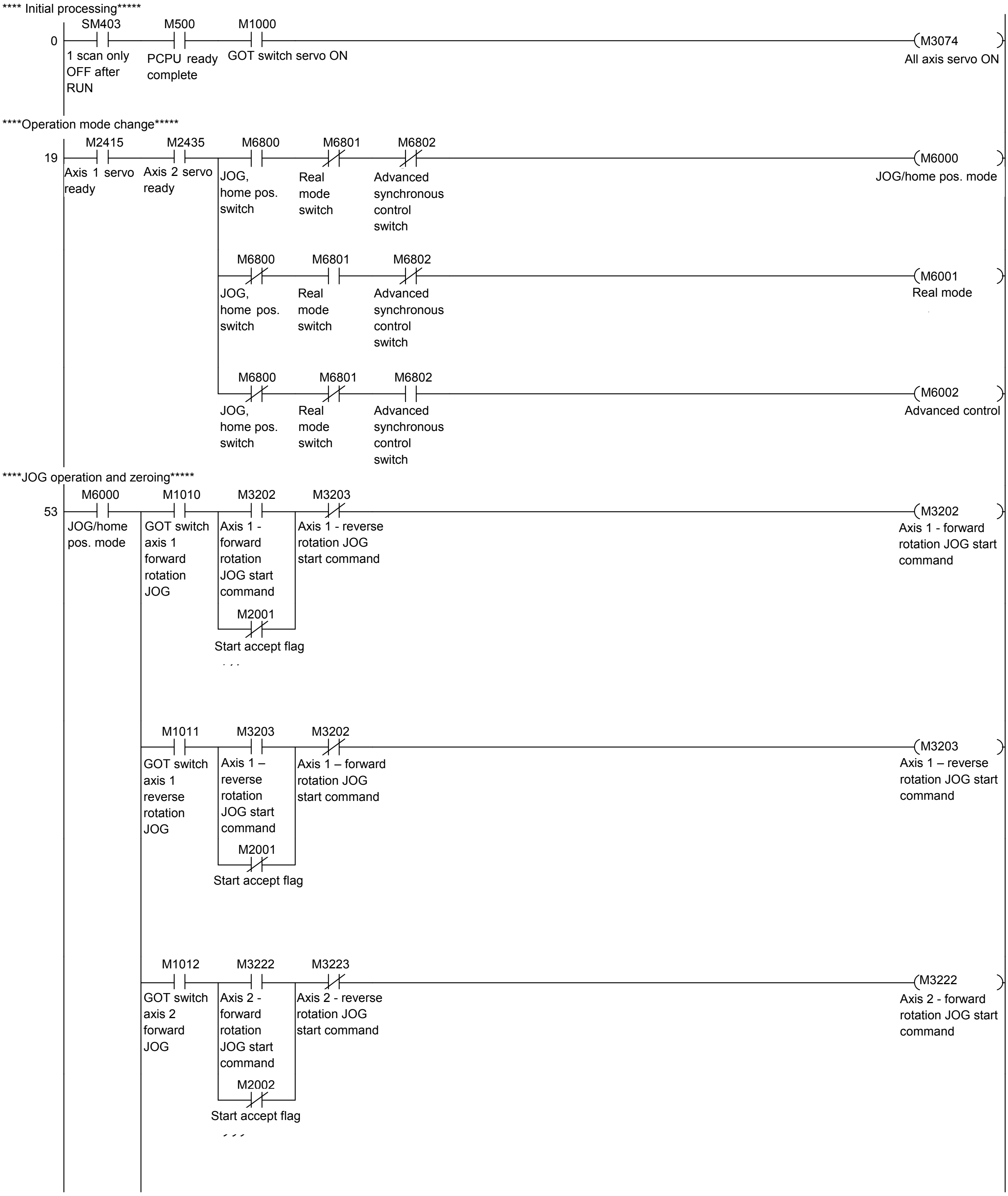


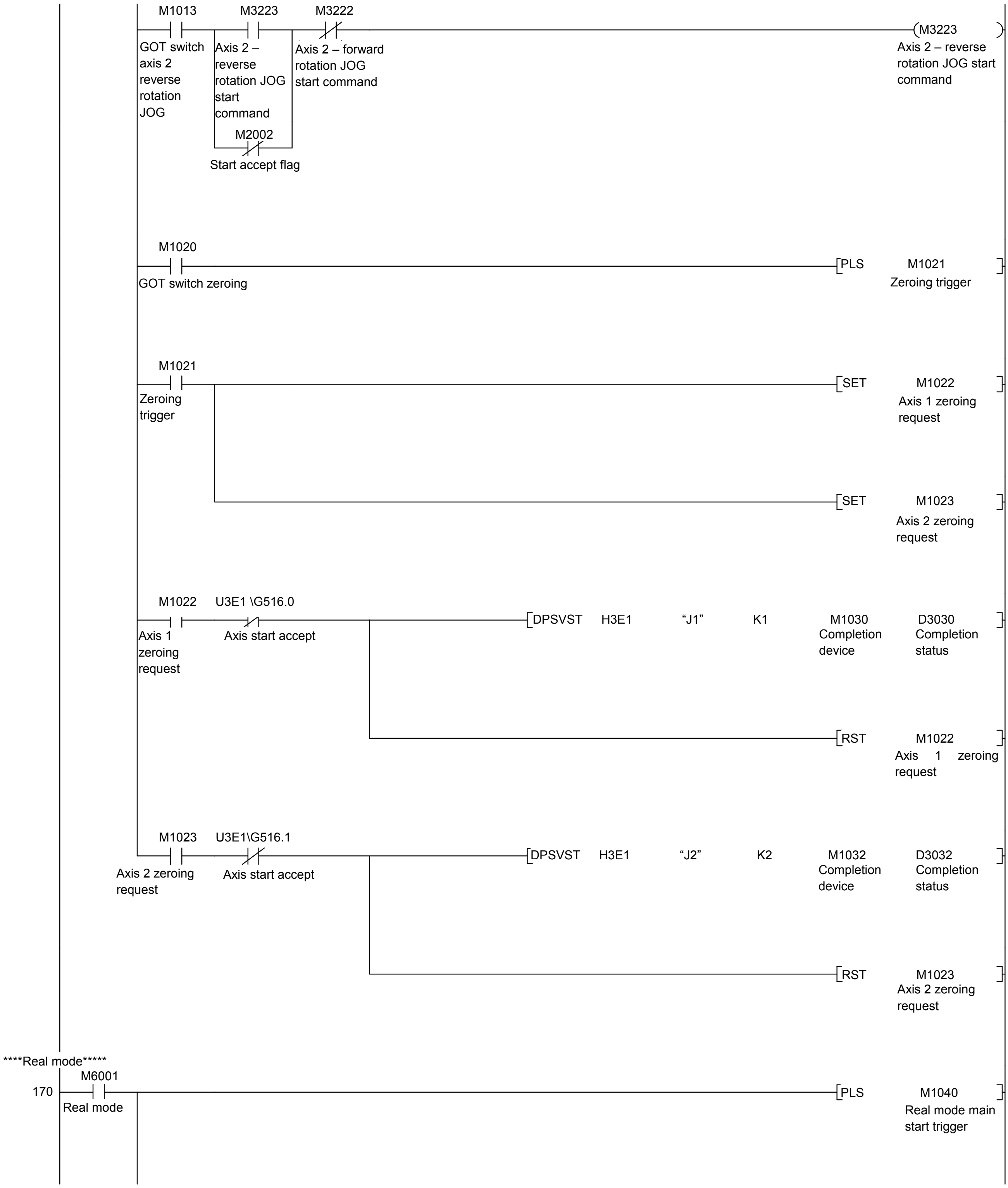
• Start program from sequence program

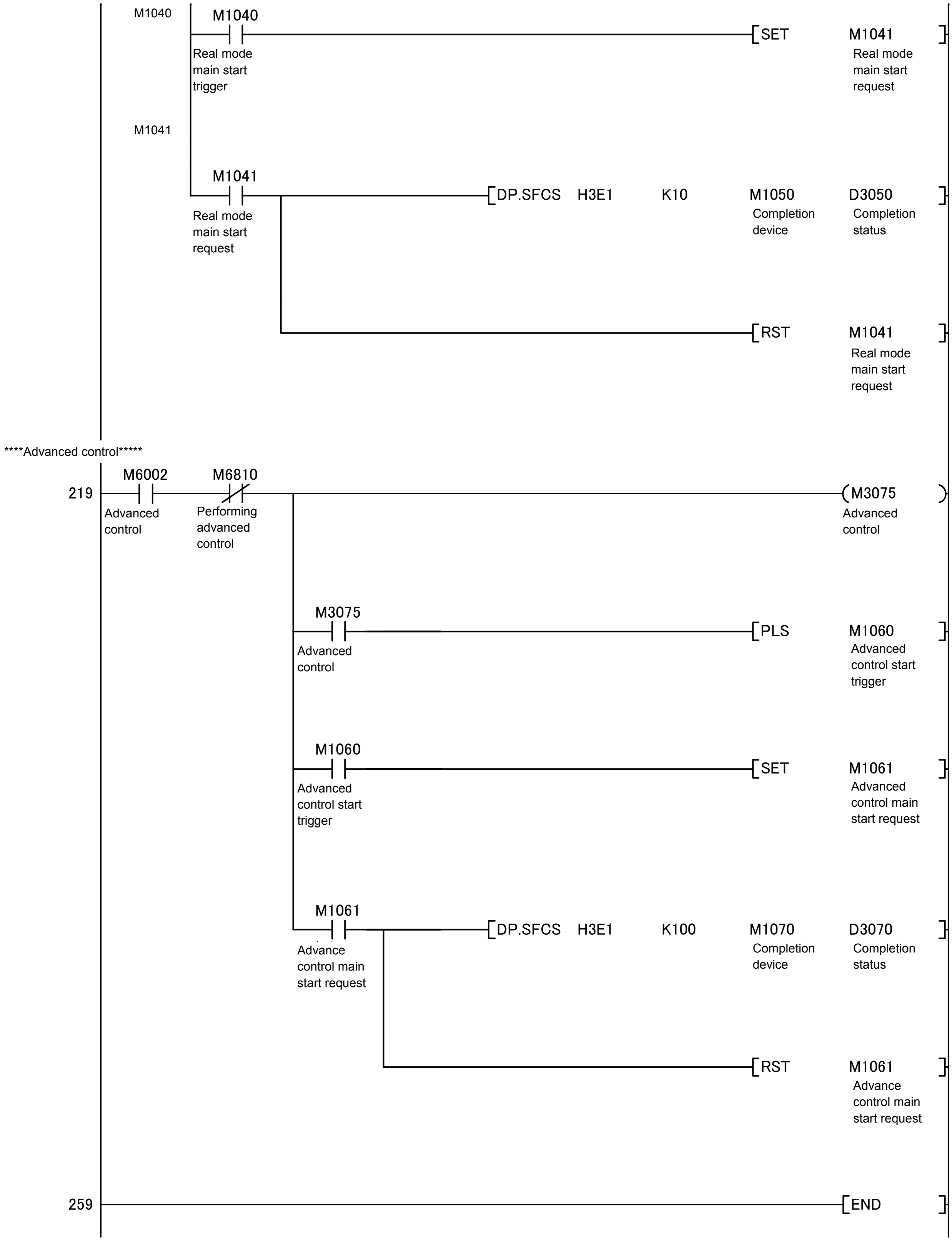
[Zeroing program] Servo program K1 and K2 are started directly with an SVST command from the sequence program.

[Jog operation] JOG start devices M3202, M3203, M3222, and M3223 are started by turning them ON directly from the sequence program.

• Q03UD sequence program




M1013
M3223
M3222
M3223
GOT switch axis 2 reverse rotation JOG
Axis 2 – reverse rotation JOG start command
Axis 2 – forward rotation JOG start command
Axis 2 – reverse rotation JOG start command
M2002
Start accept flag
M1020
PLS
M1021
GOT switch zeroing
Zeroing trigger
M1021
SET
M1022
Zeroing trigger
Axis 1 zeroing request
SET
M1023
Axis 2 zeroing request
M1022
U3E1 \G516.0
DPSVST
H3E1
"J1"
K1
M1030
D3030
Axis 1 zeroing request
Axis start accept
Completion device
Completion status
RST
M1022
Axis 1 zeroing request
M1023
U3E1\G516.1
DPSVST
H3E1
"J2"
K2
M1032
D3032
Axis 2 zeroing request
Axis start accept
Completion device
Completion status
RST
M1023
Axis 2 zeroing request
****Real mode*****
M6001
170
PLS
M1040
Real mode
Real mode main start trigger


M1040
M1040
Real mode main start trigger
SET M1041
Real mode main start request
M1041
M1041
Real mode main start request
DP.SFCS H3E1 K10 M1050 D3050
Completion device
Completion status
RST M1041
Real mode main start request
****Advanced control*****
219
M6002
Advanced control
M6810
Performing advanced control
M3075
Advanced control
M3075
Advanced control
PLS M1060
Advanced control start trigger
M1060
Advanced control start trigger
SET M1061
Advanced control main start request
M1061
Advance control main start request
DP.SFCS H3E1 K100 M1070 D3070
Completion device
Completion status
RST M1061
Advance control main start request
259
END

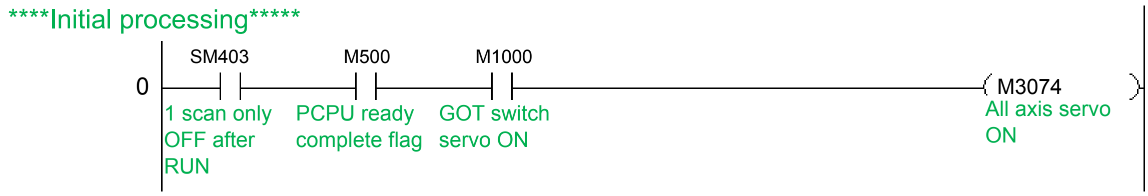
9.5.2 Initial processing

The following is an example of a program used to start all motion CPU servo axes.

Both the PLC CPU and motion CPU are set to the RUN status. With the settings for this practice, a servo data and servo parameter check is performed after the motion CPU status changes from STOP to RUN. If there are no errors, the motion CPU turns the PCPU READY complete flag (SM500) ON.

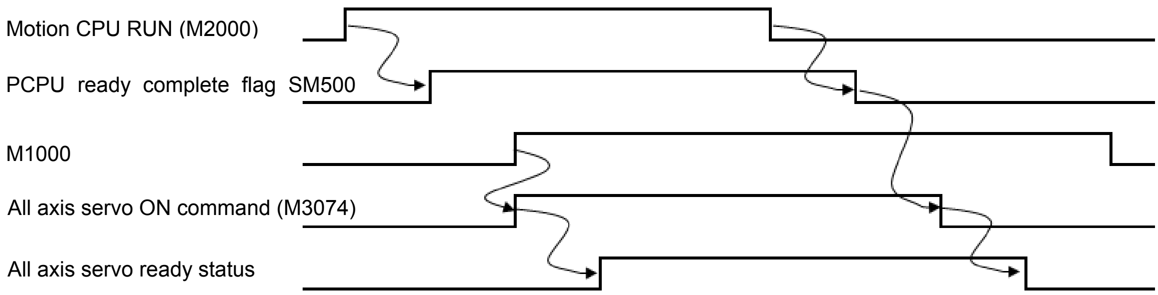
The PLC CPU receives the PCPU READY complete flag (SM500) as M500 through auto refresh. When there are no errors at either the PLC CPU or motion CPU, by turning M1000 ON at the demonstration machine operation panel, an all axis servo ON command is sent from the PLC CPU, and motion CPU startup is completed.

(1) Program example



Demonstration machine operation panel

[Timing chart]



9.5.3 JOG Operation

JOG operation is used to perform operation manually only while buttons are held down.
The devices shown in the table below and content (acceleration/deceleration time) of the parameter blocks set in JOG data are used.
By setting the speed in the JOG speed setting register (table below), and turning ON a forward rotation JOG start signal (M3202/axis 1) or reverse rotation JOG start signal (M3203/axis 1), JOG operation starts.
JOG operation stops when the JOG start signal is turned OFF.

(1) JOG operation speed setting register

Axis No.	JOG operation speed setting register		Speed setting range							
			mm		inch		degree		PULSE	
	Upper	Lower	Setting range	Unit	Setting range	Unit	Setting range	Unit	Setting range	Unit
1	D641	D640	1 to 600000000	× 10^{-2} mm / min	1 to 600000000	× 10^{-3} inch / min	1 to 2147483647	× 10^{-3} / min	1 to 10000000	pulse / s
2	D643	D642								
3	D645	D644								
4	D647	D646								
5	D649	D648								
6	D651	D650								
7	D653	D652								
8	D655	D654								

(2) Forward/reverse rotation JOG start signals

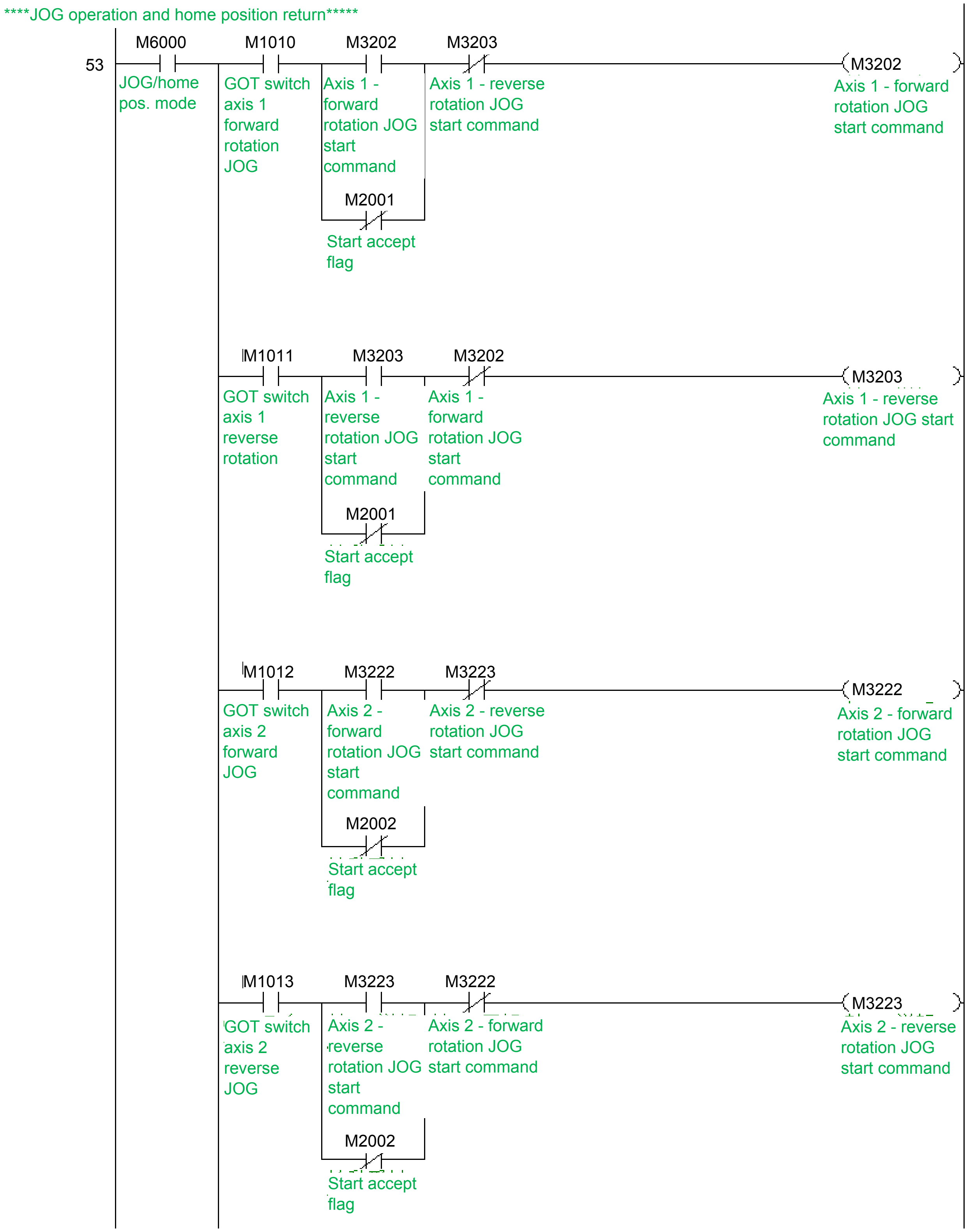
Control axis	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5	Axis 6	Axis 7	Axis 8
Forward rotation	M3202	M3222	M3242	M3262	M3282	M3302	M3322	M3342
Reverse rotation	M3203	M3223	M3243	M3263	M3283	M3303	M3323	M3343

(3) Program example

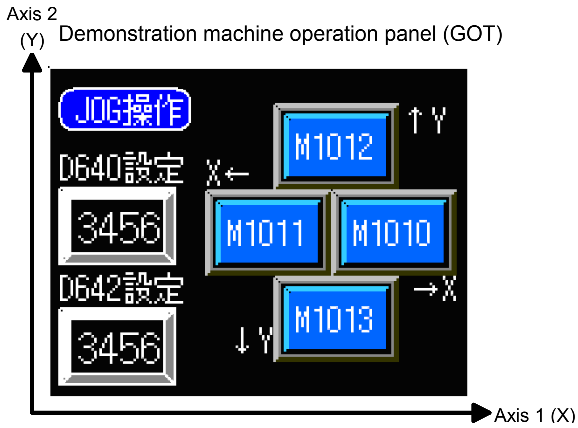
1) JOG operating condition items

Item	Condition	
Control axis	Axis 1	Axis 2
JOG operation command input	Forward rotation (M1010)	Forward rotation (M1012)
	Reverse rotation (M1011)	Reverse rotation (M1013)

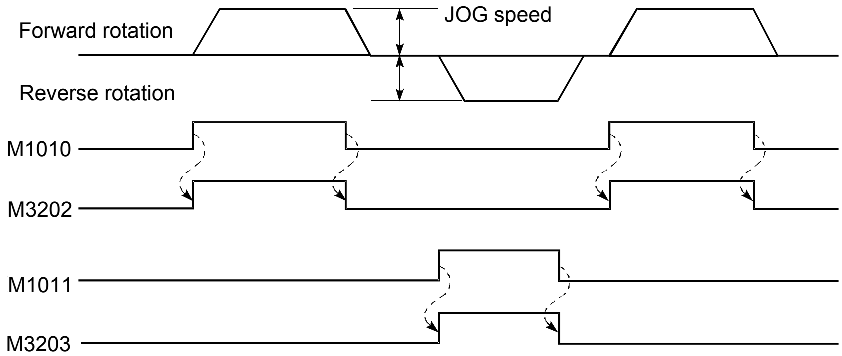
2) Example of program in which JOG operation is performed by starting axes 1 and 2 independently
The JOG speed can be set freely from the demonstration machine operation panel.



M1010 : Axis 1 forward rotation JOG command
M1011 : Axis 1 reverse rotation JOG command
M1012 : Axis 2 forward rotation JOG command
M1013 : Axis 2 reverse rotation JOG command
D641, D640 : Axis 1 JOG speed setting register
D643, D642 : Axis 2 JOG speed setting register



[Timing chart]

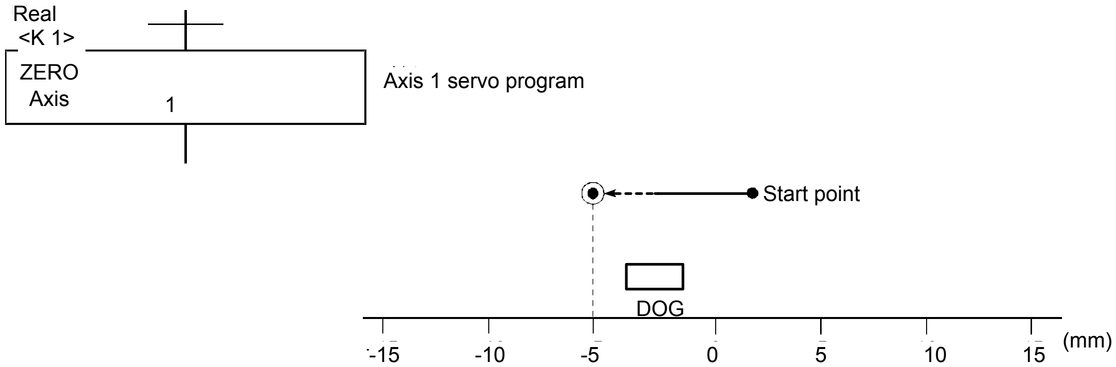


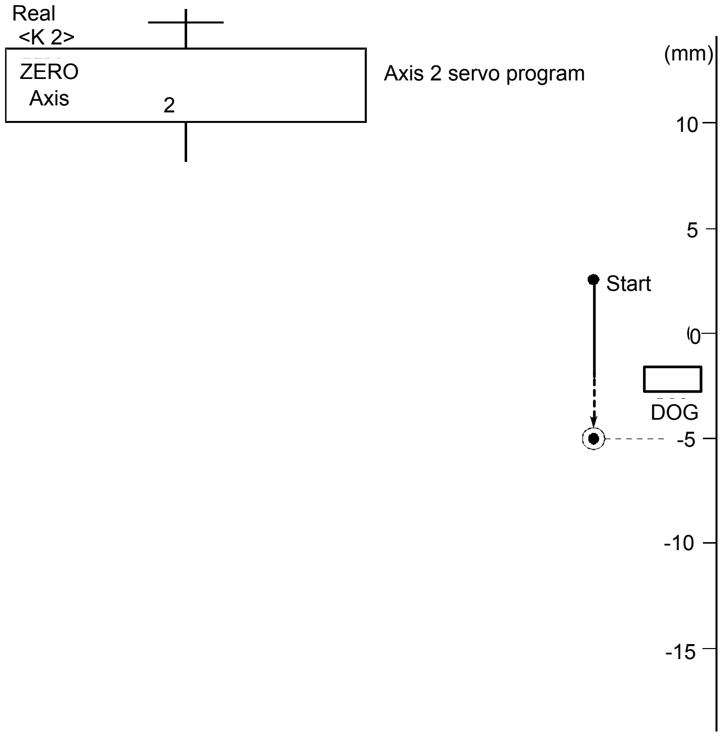
9.5.4 Zeroing

The following is an example of a program in which a servo program is run and zeroing is performed by executing an SVST command from a ladder program.
Actual details of the zeroing operation are determined by the zeroing data at the motion CPU side and the parameter block (acceleration/deceleration time).
The zeroing operation for each axis is as follows.
Zeroing is performed by turning ON the demonstration machine operation panel M1020.

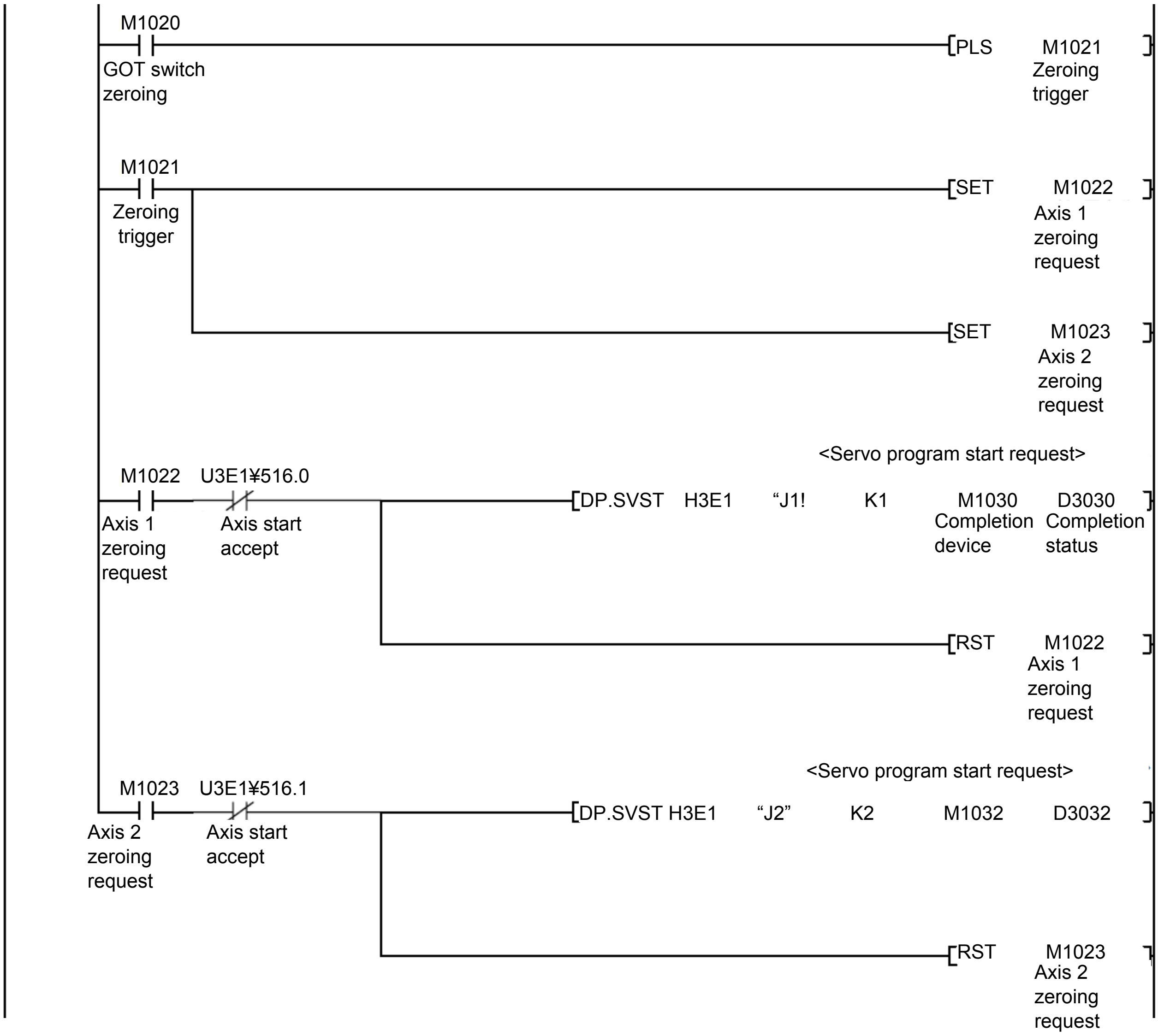
Axis 1/2: Set with proximity dog.
After starting, the motor rotates in the zeroing direction, and the rotation is complete when the home position dog changes from ON to OFF.

[Servo program]





[Sequence program]



MEMO

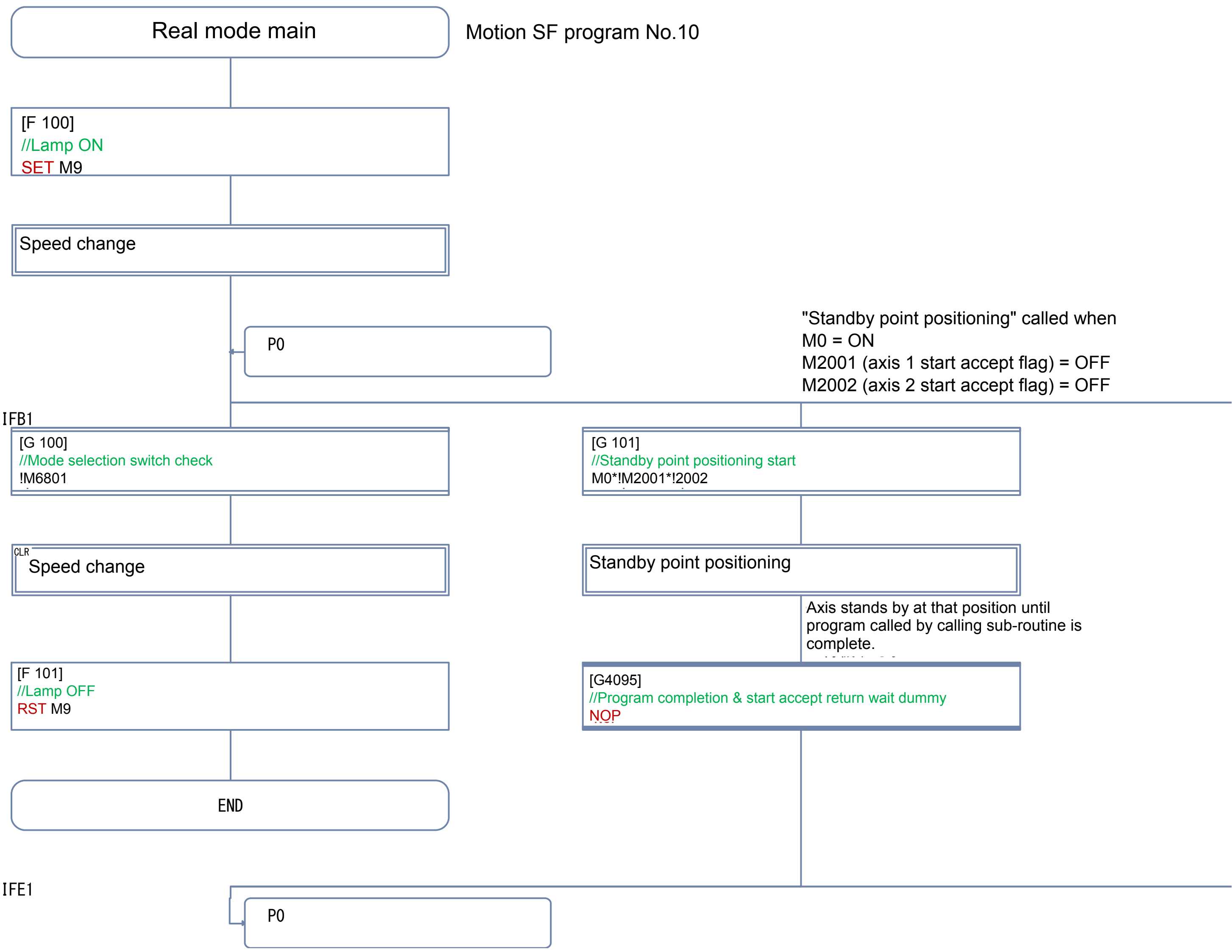
9.5.5 Main routine motion SFC program (real mode operation)

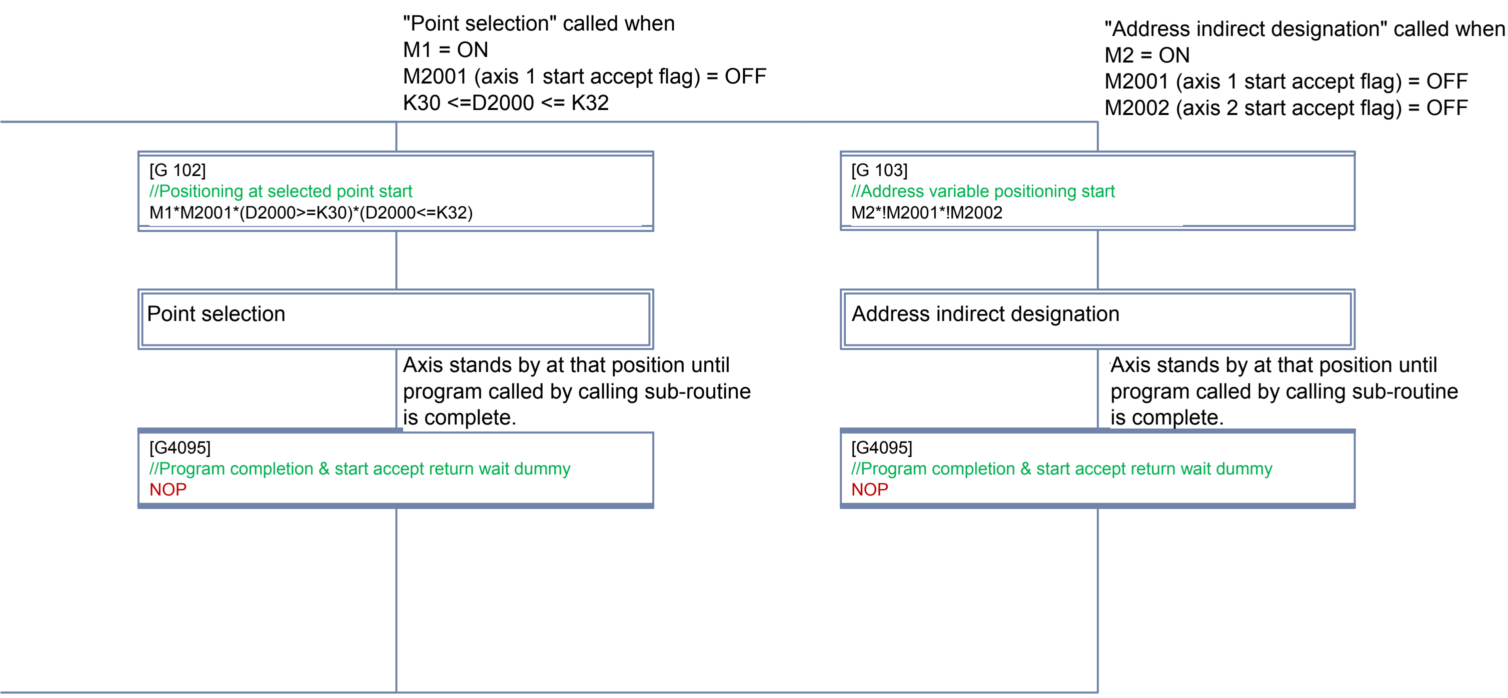
This is a motion SFC program run as the main routine when performing real mode positioning operation (other than manual operation).
Other motion SFC programs used to perform various types of operation when in real mode from this main routine motion SFC program are started as sub-routines.

(1) Motion SFC program started from main routine motion SFC program.

Motion SFC program No.	Program name	Reference section
20	Standby point positioning	9.5.6
30	Point selection	9.5.7
40	Address indirect designation	9.5.8

(2) Program example




"Point selection" called when
M1 = ON
M2001 (axis 1 start accept flag) = OFF
K30 <=D2000 <= K32
"Address indirect designation" called when
M2 = ON
M2001 (axis 1 start accept flag) = OFF
M2002 (axis 2 start accept flag) = OFF
[G 102]
//Positioning at selected point start
M1*M2001*(D2000>=K30)*(D2000<=K32)
[G 103]
//Address variable positioning start
M2*!M2001*!M2002
Point selection
Address indirect designation
Axis stands by at that position until program called by calling sub-routine is complete.
Axis stands by at that position until program called by calling sub-routine is complete.
[G4095]
//Program completion & start accept return wait dummy
NOP
[G4095]
//Program completion & start accept return wait dummy
NOP

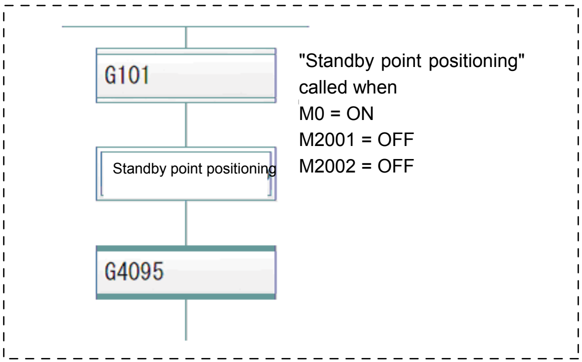
9.5.6 Standby point positioning

Standby point refers to a work standby position at other than the mechanical home position. (There may be times when the position is the same as the home position.)
In this program example, the axis returns to the standby point by specifying the standby point address and performing positioning.

By running the servo program with a motion SFC program motion control step, operation is performed based on the content of the executed servo program data and the parameter block.

[G 102]
//Positioning at selected point start
M1*M2001*(D2000>=K30)*(D2000<=K32)

[Real mode main] program



Standby point positioning — Motion SFC program No.20

[F 200]
//Lamp ON
SET M10

[K20: Real]
ABS-2

Axis 1	0.0 μm
Axis 2	0.0 μm
Composite speed	4000.00mm/min
Dwell	100 msec

[G4095]
//Program completion & start accept return wait dummy
NOP

. . . Ensures transition to next step with "WAIT" type transition after completion of current servo program.

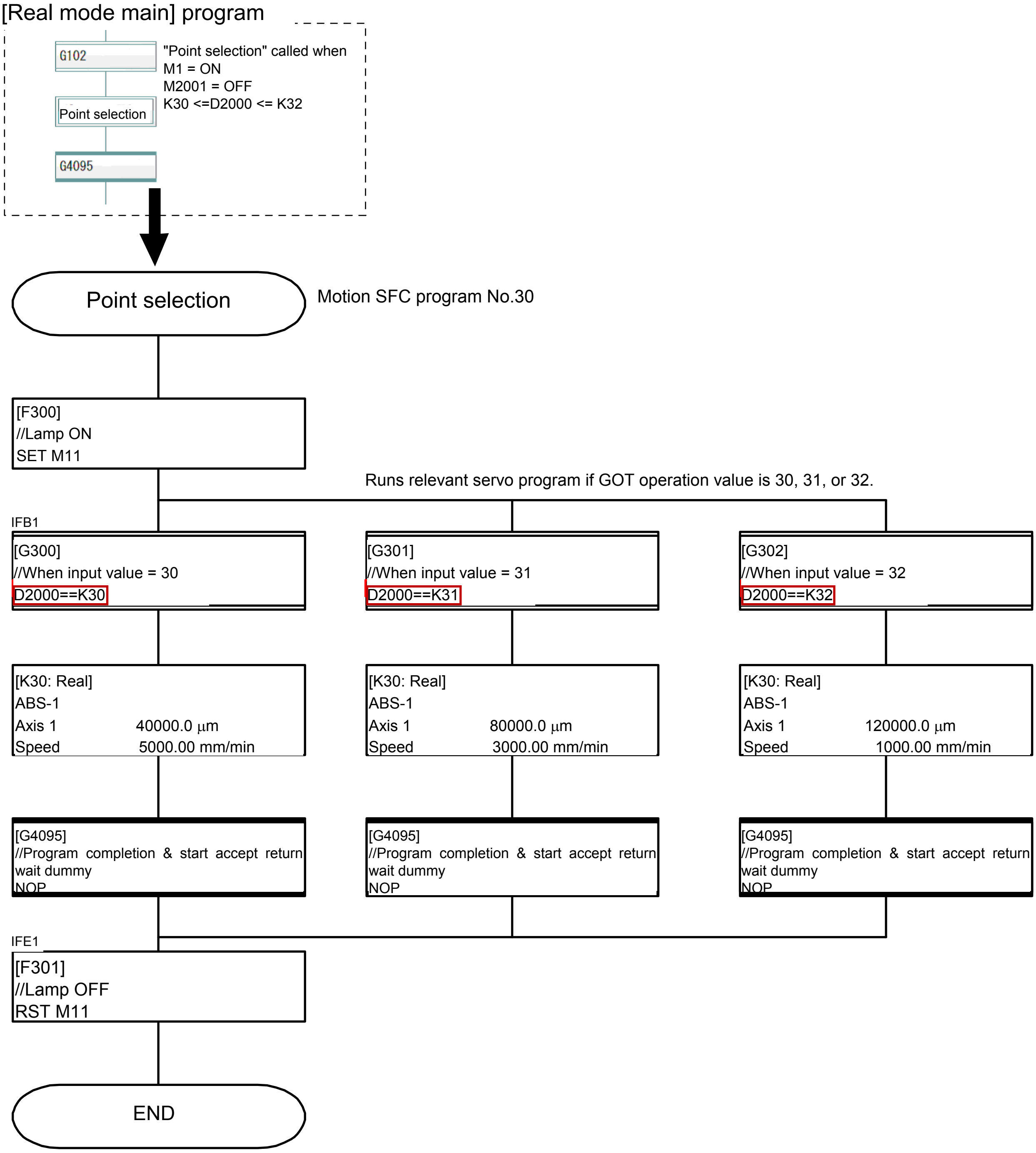
[F201]
//Lamp ON
RST M10

END

9.5.7 Point selection positioning

This is an example of a basic point selection program.
By entering the point No. (servo program No. in this example) at the GOT operation panel and then pressing the START button, the axis is positioned at the address registered beforehand.

[Real mode main] program

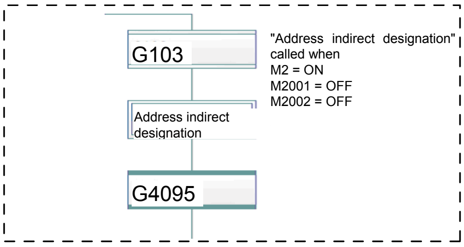


Note: There are two "=" symbols in the "D2000==K30", "D2000==K31", and "D2000==K32" commands in [G300], [G301], and [G302].

9.5.8 Address indirect designation positioning

This is an example of positioning at an address other than the previously registered position.
The axis 1 and axis 2 addresses are computed based on the GOT operation panel values, and then stored in D2020.
Positioning is performed by pressing the START button.
Even number addresses in the unused data register D, link register W, and motion device # can be used for indirect setting.
In addition to addresses, speed, dwell, M-codes, and parameter blocks can also be set indirectly.

[Real mode main] program



Address indirect designation

Motion SFC program No.40

```
[F400]
//Lamp ON
SET M12
//For address designation <32 bit>
D2020L=D2000L*K10000
```

```
[K40: Real]
 1 ABS-2
 Axis 1              D 2020 μm
 Axis 2              D 2020 μm
 Composite speed    1000.00 mm/min
```

Axis 1, 2 addresses set as D2020, D2021 32 bit data.

```
[G4095]
//Program completion & start accept return wait dummy
NOP
```

```
[F401]
//Lamp OFF
RST M12
```

END

9.5.9 Changing the speed (CHGV) [additional practice]

This is an example of a program used to change the speed in three stages at the GOT operation panel and then temporarily stop operation.
Changes to speed are made by executing a speed change command (CHGV command) with a motion SFC program operation control step.
When setting the speed with a CHGV command, operation stops temporarily when setting the speed to "0", and the remainder of the operation is performed when the speed is changed again by setting to a value other than 0.

(1) CHGV speed change request command

Describes the axis No. for which the speed is to be changed, and the changed speed.

CHGV(K1,K30000)

K [speed after change]
D0 to D8191
W0 to 1FFF
#0 to #7999

Axis No. (1 to 8) for which speed changed

(2) Speed change setting range

Speed change setting range							
mm		inch		degree		pulse	
Setting	Unit	Setting	Unit	Setting	Unit	Setting	Unit
-600000000 to 600000000	× 10^{-2} mm/min	-600000000 to 600000000	× 10^{-3} inch/min	-2147483648 to 2147483648	× 10^{-3} degrees/ min	-10000000 to 10000000	pulse/s

POINT

If setting the speed with the CHGV command, set a value 100 times (mm) or 1000 times (inch/degrees) the actual speed.

Example

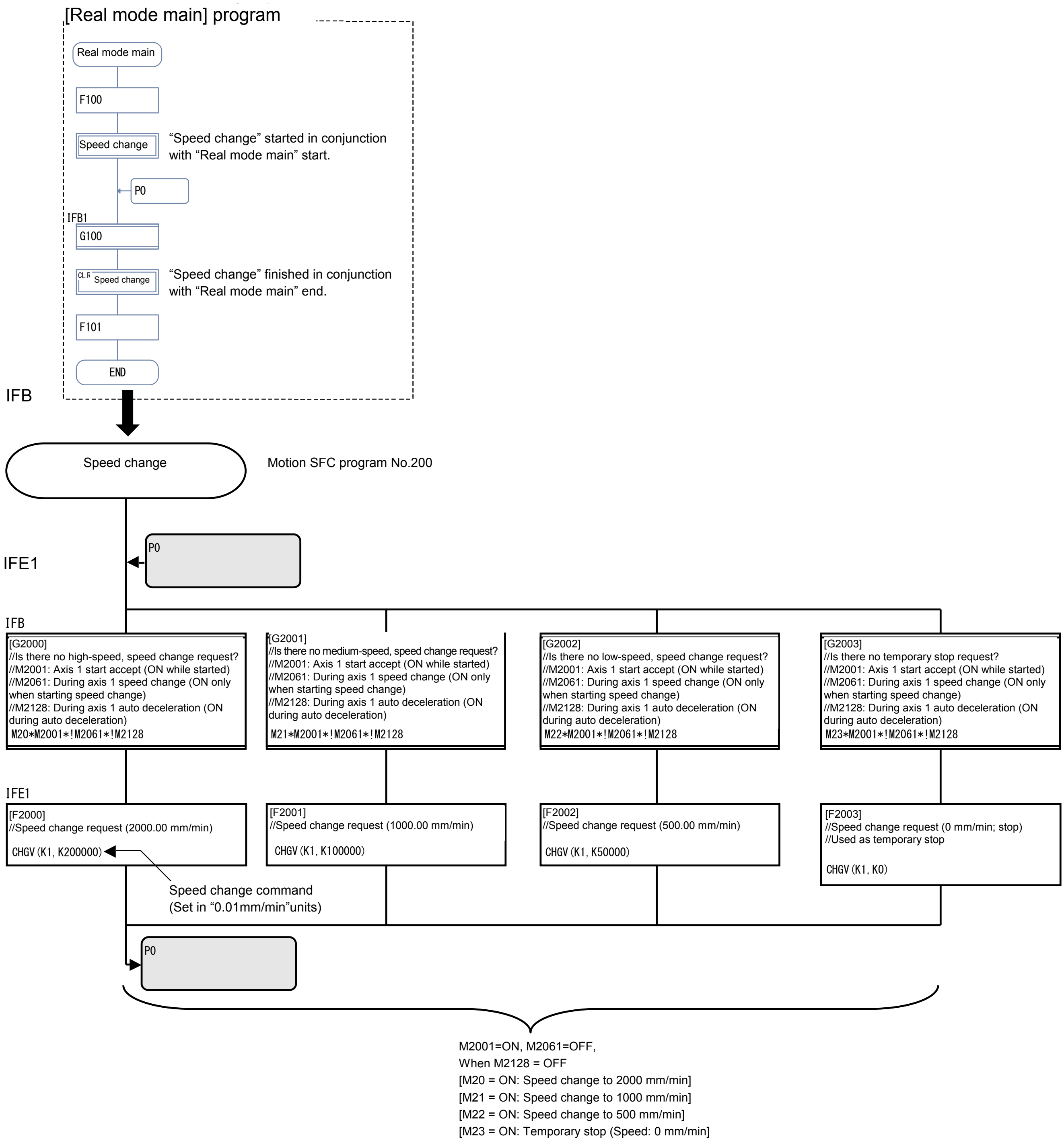
If setting the speed to 10000.00 mm/min, set a value of "1000000".

(3) Program example

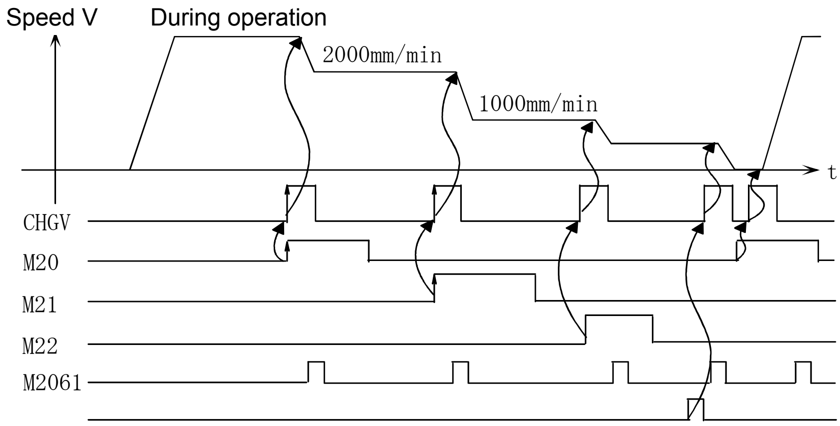
1) Speed change conditions

Item		Condition	
Control axis		Axis 1	Axis 2
Speed change command input	M20	Speed after change: 2000 mm/min	————
	M21	Speed after change: 1000 mm/min	————
	M22	Speed after change: 500 mm/min	————
	M23	Temporary stop (0 mm/min)	————

2) Speed change program example



[Timing chart]



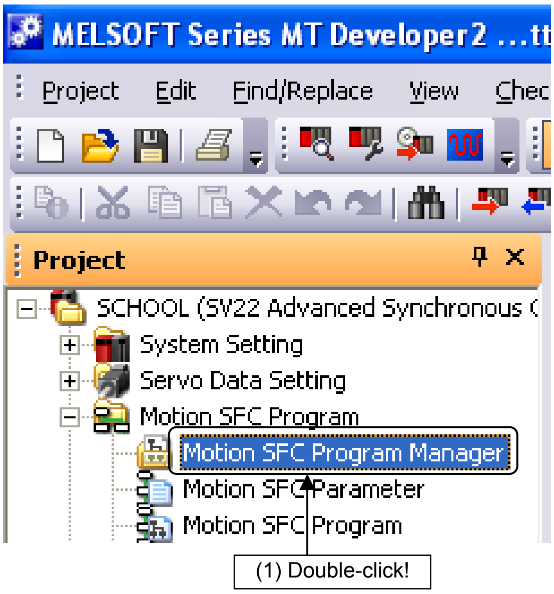
POINT
• The speed cannot be changed while the start accept flag is OFF. • The speed cannot be changed during zeroing, circular interpolation, or while decelerating. • The speed can be changed within the 0 to start speed range.

9.6 Motion SFC Program Creation Procedure

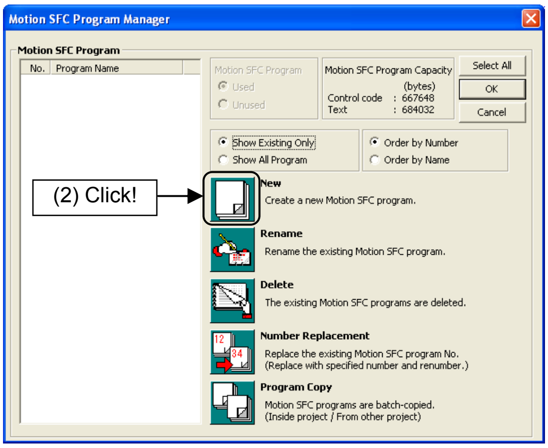
This section describes how to create motion SFC programs used to set motion control operation.

9.6.1 Creating a new motion SFC program

To create a new motion SFC program, begin by specifying the "Program name".

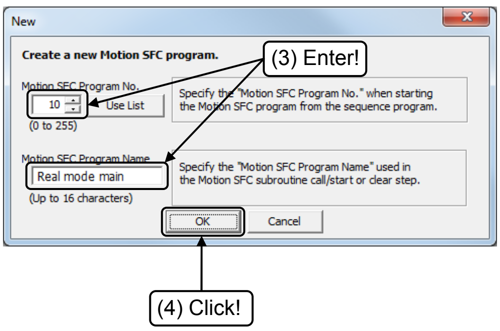


(1) Double-click [Motion SFC Program] → [Motion SFC Program Manager] in the Project window.



(2) A Motion SFC Program Manager dialog box appears.

Click the NEW button.

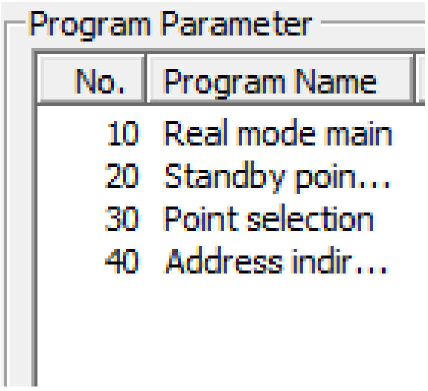


(3) A New dialog box appears. Set the program No. for the motion SFC program being created. Enter "10" for the "Motion SFC program No.", and "Real mode main" for the "Motion SFC program name".

(4) Click the OK button after entering.

Go to next page

From previous page



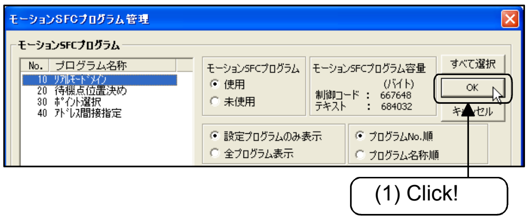
(Motion SFC programs other than No.10 and No.20 created here will not be described in detail.) Refer to the section on motion SFC programs for operation described later to create.

(4) The set motion SFC program appears in a list. Press the NEW button again to create a motion SFC program such as the following.

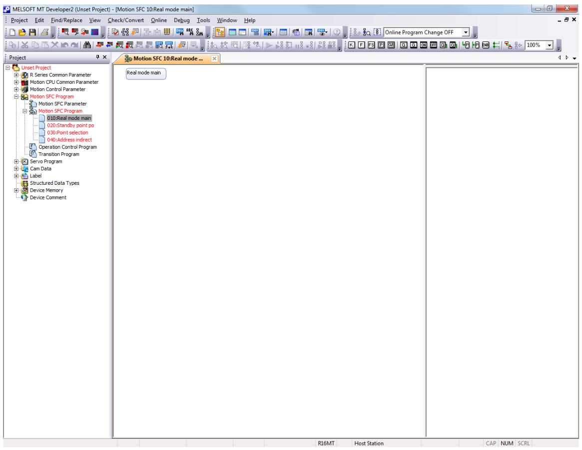
No.	Program name
10	Real mode main
20	Standby point positioning
30	Point selection
40	Address indirect designation

9.6.2 SFC diagram creation procedure

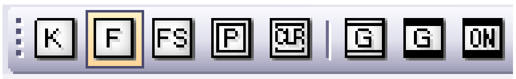
Allocate SFC diagram symbols to create an SFC diagram.



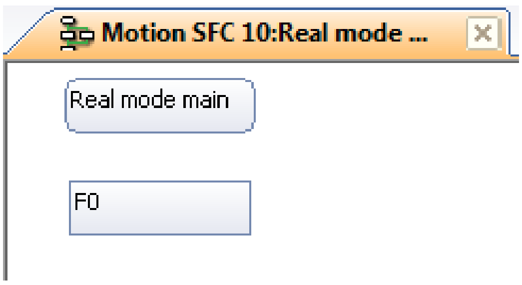
(1) Select "10 Real mode main" from the motion SFC program list in the Motion SFC Program Management dialog box, and then click the OK button.



(2) An Edit Program screen used to create individual motion SFC programs appears.



(3) Click the single execution type operation control step tool button ([F]) on the Program Edit screen.



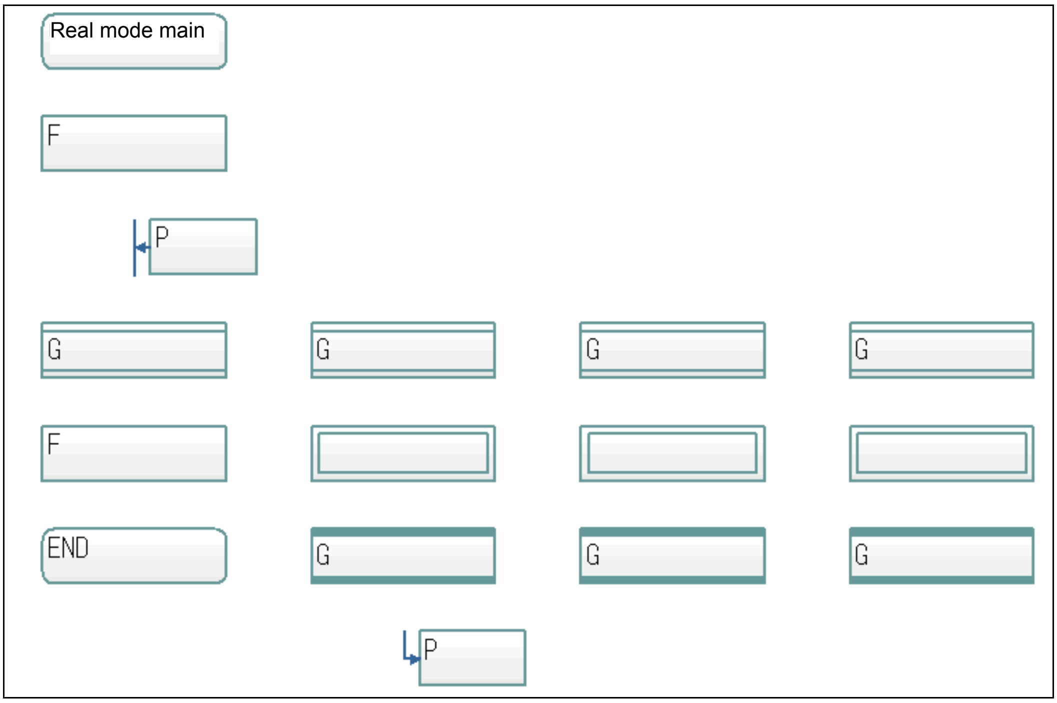
(4) Click on a random position to allocate an SFC diagram symbol pointer.

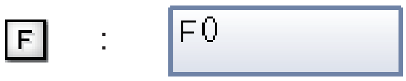
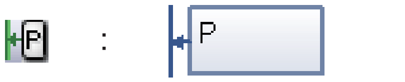
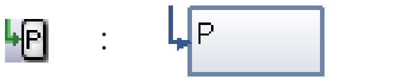
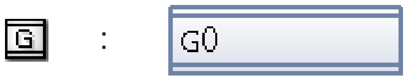
Multiple pointers can be allocated.
When allocation is complete, right-click to clear the SFC diagram symbol.

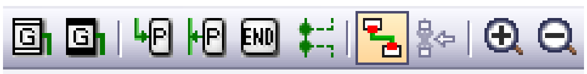
Go to next page

From previous page

(5) Now, click each tool button in the same manner to allocate SFC diagram symbols as shown on the left.



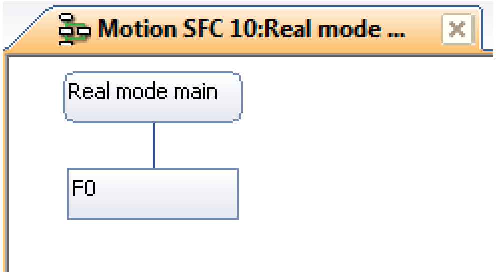
		(Single execution type operation control step)
		(Pointer)
		(Jump)
		(Shift transition)
		(Wait transition)
		(Sub-routine call/start step)
		(END)



(6) Connect the allocated SFC diagram symbols.
Click the connect tool button at the Edit Program screen.

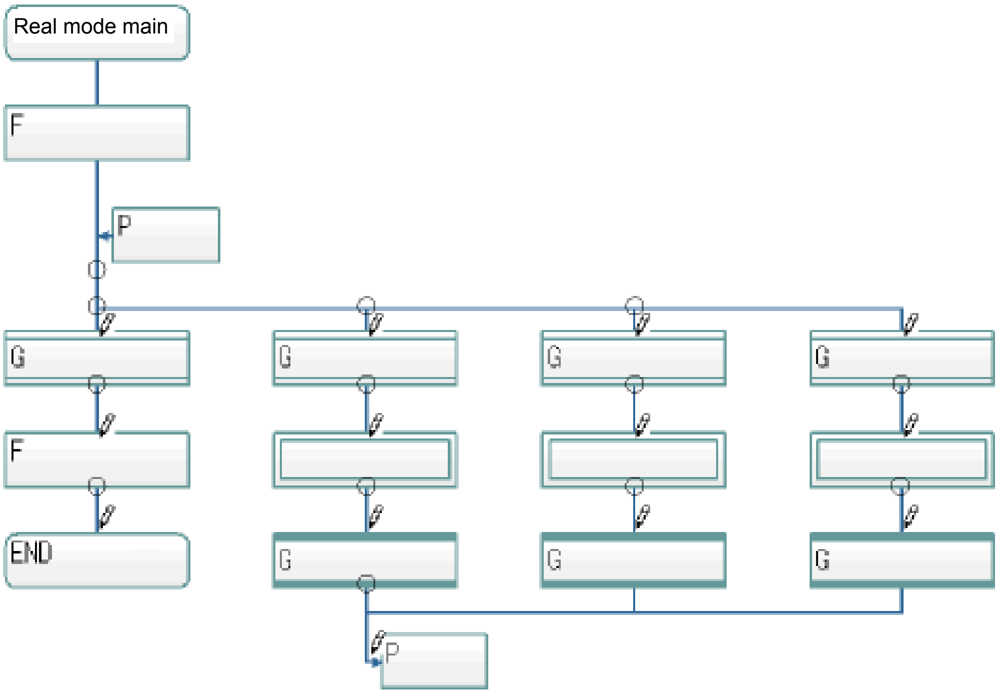
Go to next page

From previous page

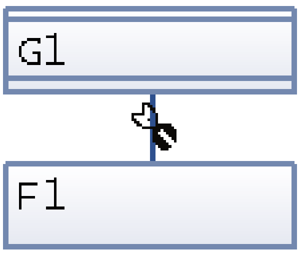


(7) By moving the mouse cursor over an SFC diagram symbol, the shape of the cursor changes.
Drag to connect the start of the motion SFC program and pointer.

(8) Connect other SFC diagram symbols in the same manner.

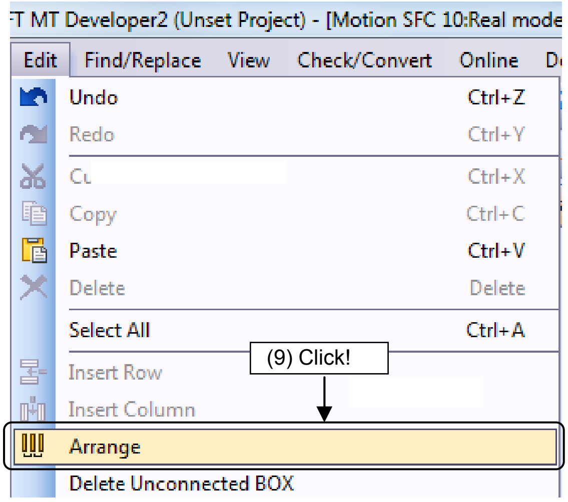


If connections are mistaken, click the Select/Cut tool button at the Edit Program screen ( ), move the mouse cursor over the connecting line, and then click to cut the line.

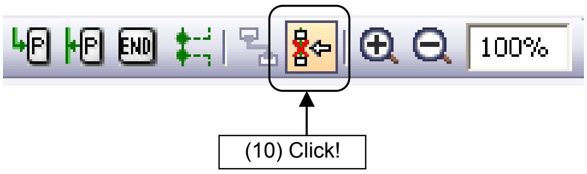


Go to next page

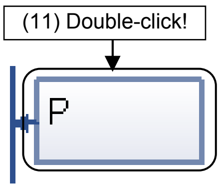
From previous page



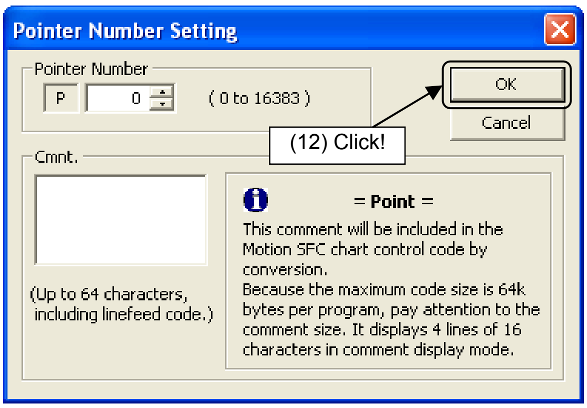
(9) Click [Arrange] on the [Edit] menu at the Edit Program window.
Arrange the allocated SFC diagram symbols.



(10) Set program Nos. and pointer Nos. for the allocated SFC diagram symbols.
Click the Select/Cut tool button at the Edit Program screen.



(11) Double-click a pointer (P).

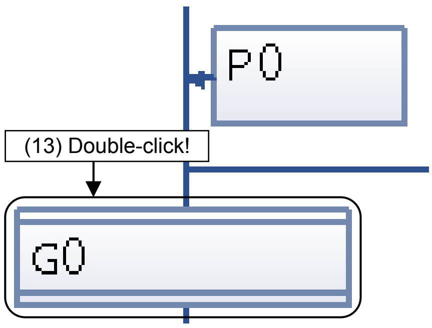


(12) A Pointer Number Setting dialog box appears.
Enter "0" for the "Pointer Number", and then click the OK button.

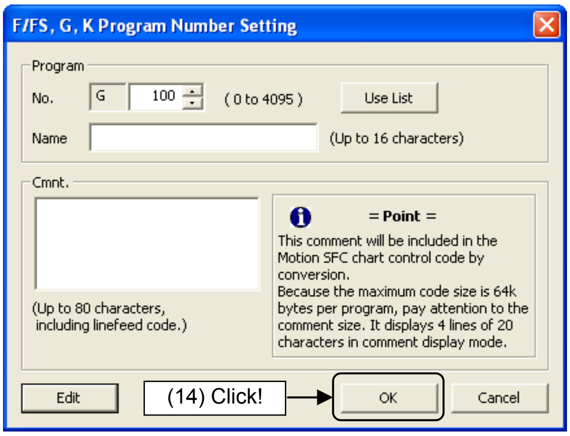
Pointer Nos. can be set fro 0 to 16383 for each motion SFC program.
("P0" for motion SFC program No.0 and motion SFC program No.10 are different.)

Go to next page

From previous page

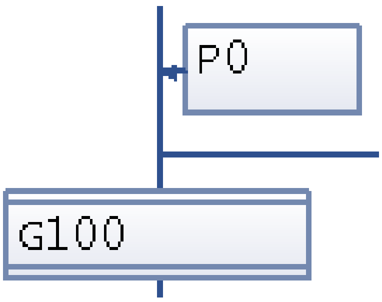


(13) Pointer No. "0" is set.
Next, double-click a transition (G).



(14) A Program Number Setting dialog box appears.
Enter "100" for the "Program No.", and then click the OK button.

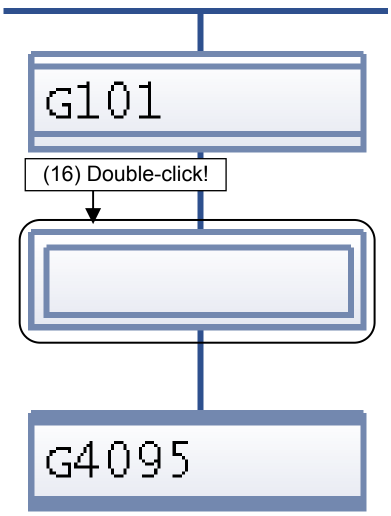
The program No. is a common number in the project.



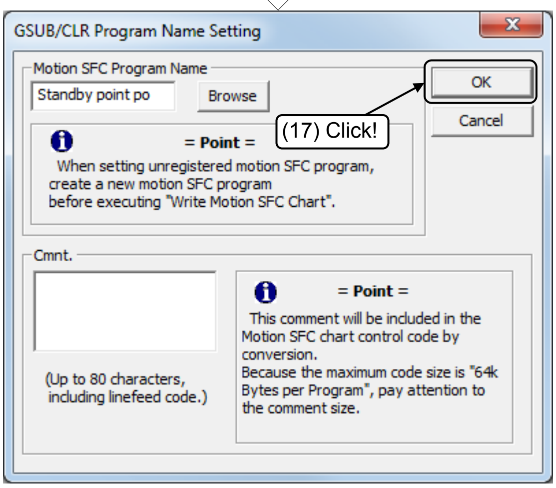
(15) Program No. "G100" is set for the transition.
Set operation control steps (F) and transitions (G) in the same manner.

Go to next page

From previous page



(16) Next, double-click a sub-routine call/start step.

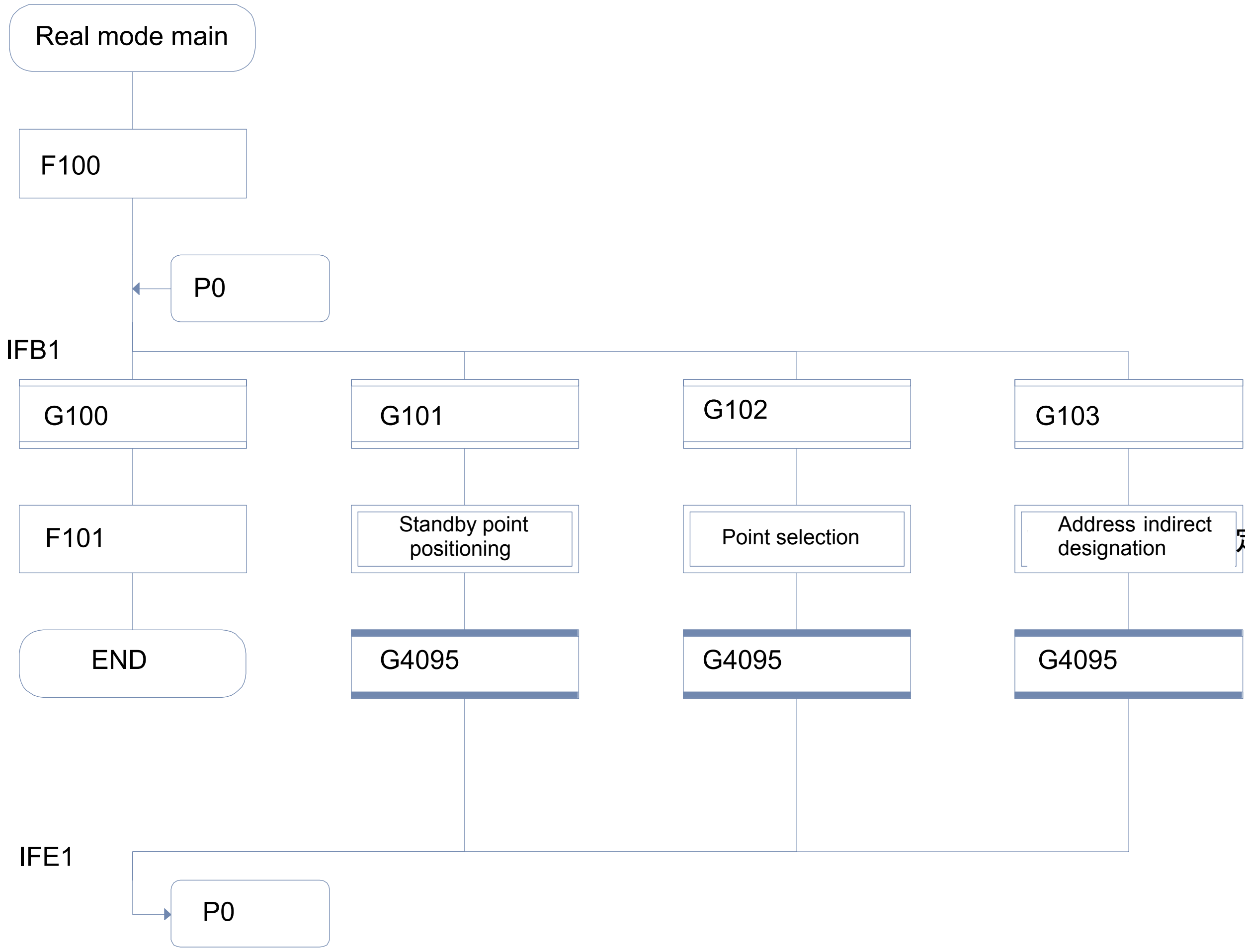


(17) A Program Name Setting dialog box appears. Enter "Standby point positioning" for the "Motion SFC Program Name", and then click the OK button.

Go to next page

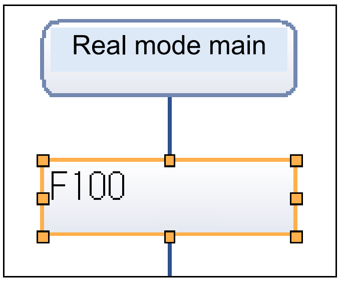
From previous page

(18) Program name "Standby point positioning" is set for the sub-routine call/start step.
Set program Nos. and pointer Nos. for other SFC diagram symbols in the same manner as shown on the left.

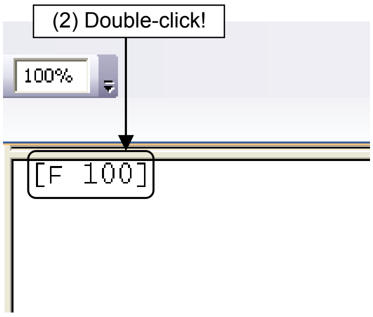


9.6.3 Entering transition and operation control steps

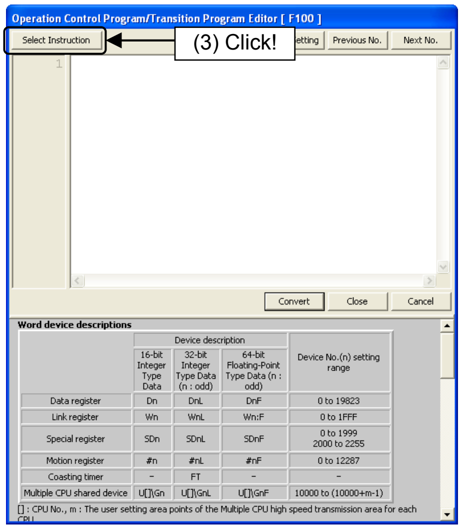
This section describes how to set conditional expressions and operational expressions for transitions and operation control steps allocated to SFC diagrams.



(1) Click operation control step "F100" to select.

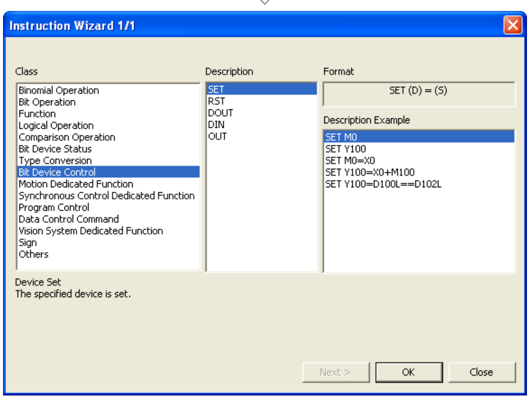


(2) When "[F100]" appears in the area on the right of the screen (step Edit Program screen), double-click.



(3) An Edit Operation Control Program/Transition Program dialog box appears.
Enter a comment and press the Enter key to start a new line.
Click the Select Instruction button.

> Commands can be set by direct entry.
> If entering directly, continue from step (5).

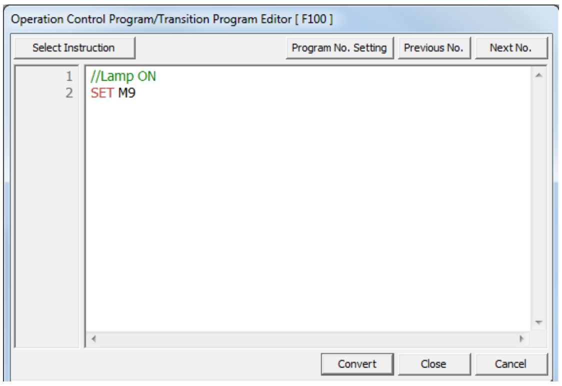


(4) An Instruction Wizard dialog box appears.
Select as follows, and then press the OK button.

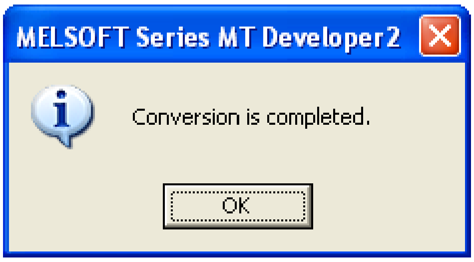
Class : Bit Device Control
Description : SET
Description Example : SET M0

Go to next page

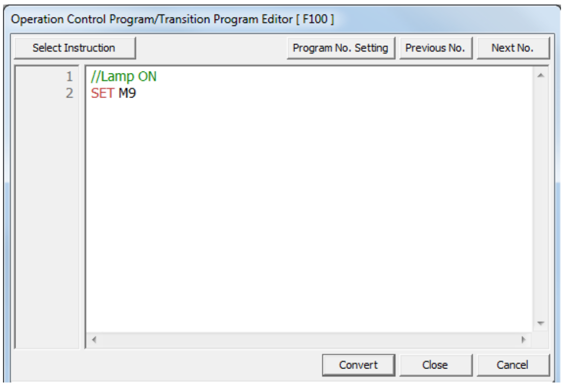
From previous page



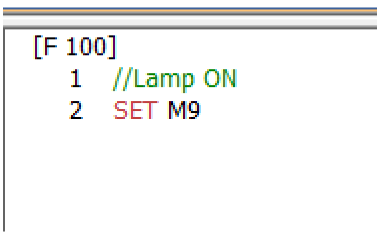
(5) A "SET M0" command is set. Change "M0" to "M9".
Press the Enter key again to start a new line, and then enter a comment and command.
Click the Convert button after entering.



(6) Click the OK button at the conversion complete message that appears.



(7) Click the Close button.

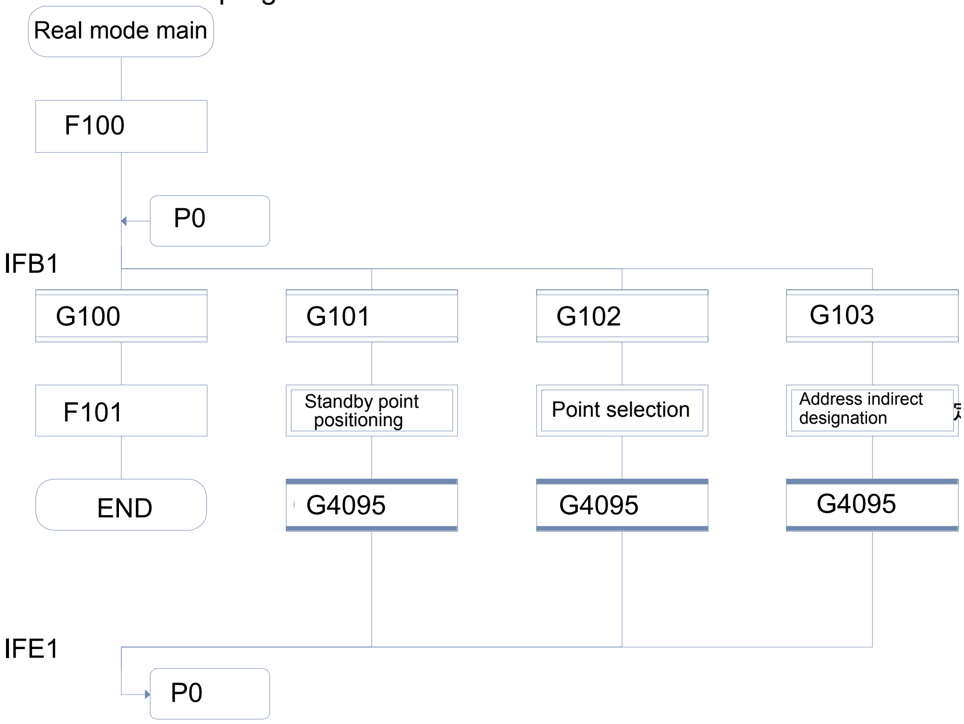


(8) The set command appears on the step Edit Program screen.

Go to next page

From previous page

(9) Set the operational expression and conditional expression for the following operation control programs and transition programs in the same manner.



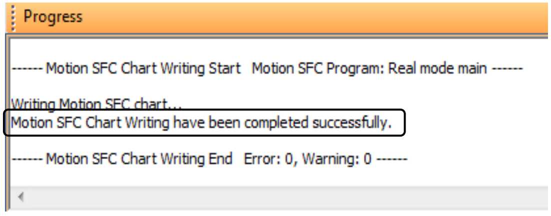
[G100]	//Mode selection switch check !M6801
[G101]	//Standby point positioning start M0*!M2001*!M2002
[G102]	//Positioning at selected point start M1*!M2001*(D2000>=K30)*(D2000<=K32)
[G103]	//Address variable positioning start M2*!M2001*!M2002
[G4095]	//Program completion & start accept return wait dummy NOP
[F100]	//Lamp ON SET M9
[F101]	//Lamp OFF RST M9

*(Logical product)
!(Logical negation)

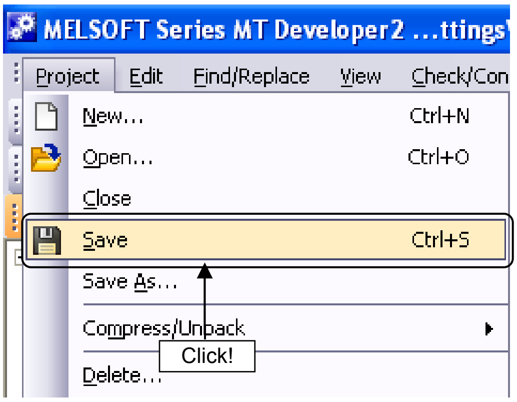
(10) Click the Write Motion SFC Diagram button at the Edit Program screen.

Go to next page

From previous page



(11) When conversion is complete, a "Successful completion" message appears in the output window.

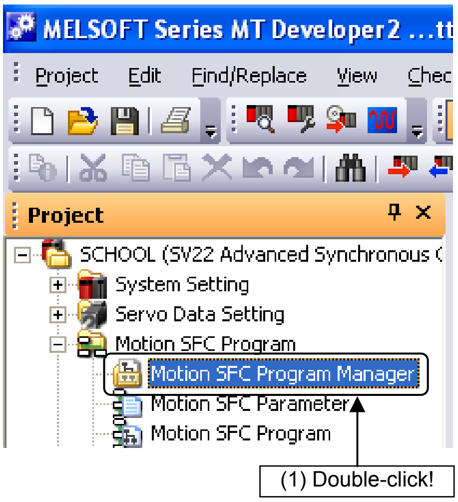


(12) Click [Save] on the [Project] menu at the Edit Program window.
Real mode main creation is now complete.

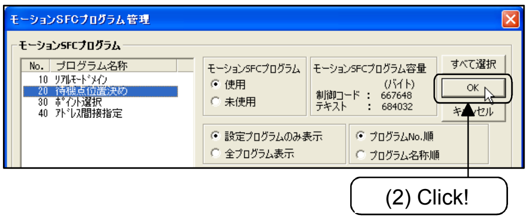
9.6.4 Entering motion control steps

This section describes how to specify motion control steps used to perform positioning control and so on.
Here, a motion SFC program for standby point positioning is created first.

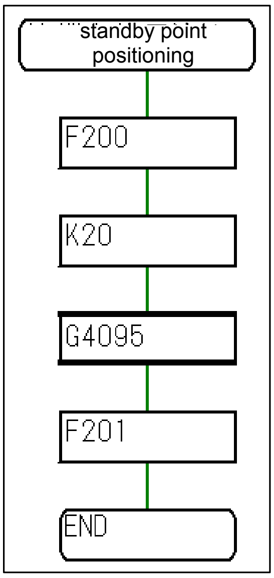
(1) Double-click [Motion SFC Program] → [Motion SFC Program Manager] in the project window.



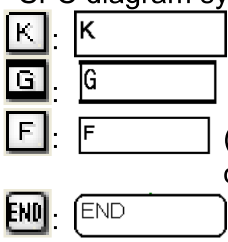
(2) Select "20 Standby point positioning" from the motion SFC program list in the Motion SFC Program Manager dialog box, and then click the OK button.



(3) Create a motion SFC program for standby point positioning as follows.



(a) Allocate SFC diagram symbols.
Use the following tool buttons to allocate SFC diagram symbols.

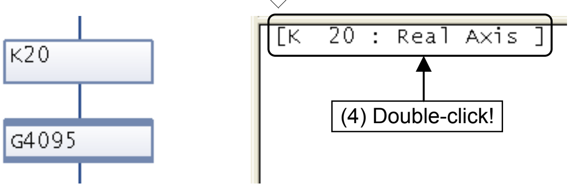


(Motion control step)
(WAIT transition)
(One-time execution type operation control step)
(END)

(b) Connect the SFC diagram symbols with the Connect tool button.

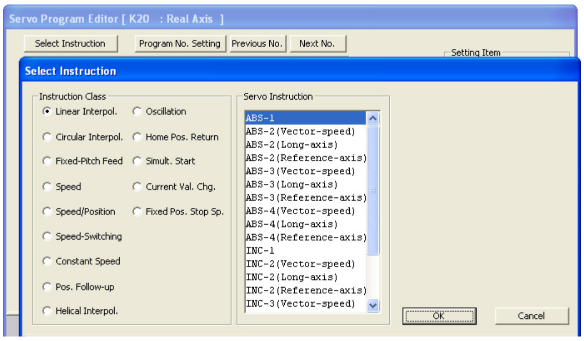
(c) Select the servo program No. with the Select/Cut tool button, and then set.

(4) Click motion control step "K20" to select it, and then double-click the Edit Program screen.

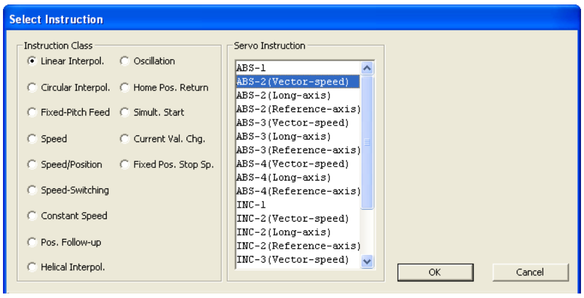


Go to next page

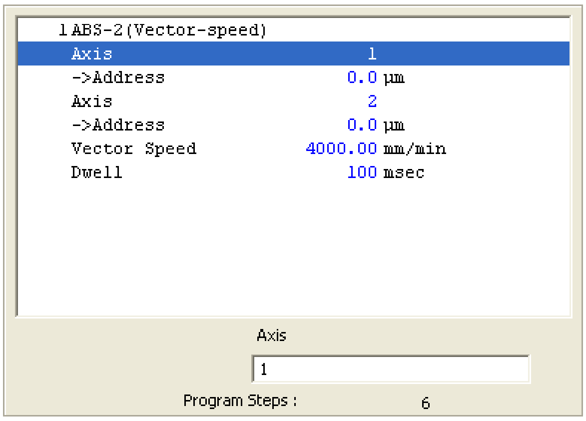
From previous page



(5) A Select Instruction dialog box is displayed at the Servo Program Editor dialog box.



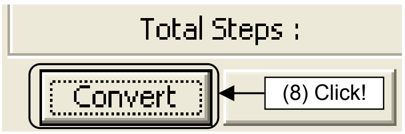
(6) Select "Positioning" for the "Instruction Class", and "ABS-2 (Vector-speed)" for the "Servo Instruction" at the Select Instruction dialog box, and then click the OK button.



(7) Enter "1" and "0.0" in the "Axis" text box. Press the Enter key again to start a new line, and then enter "2" and "0.0".
Enter "4000.00" for "Vector speed".
Add "Dwell" from the setting items, and then enter "100".

Go to next page

From previous page

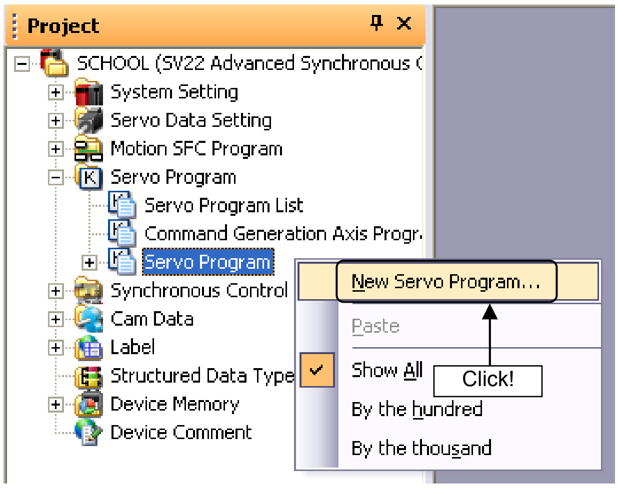


(8) Click the Convert button.
"K20" motion control step settings are now complete.

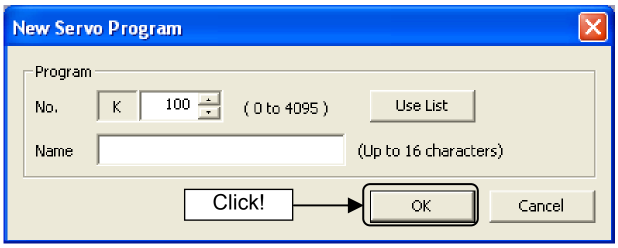
(9) Use the same procedure now to create steps used at other motion SFC programs from the following page.

Motion control step editing schematic procedure

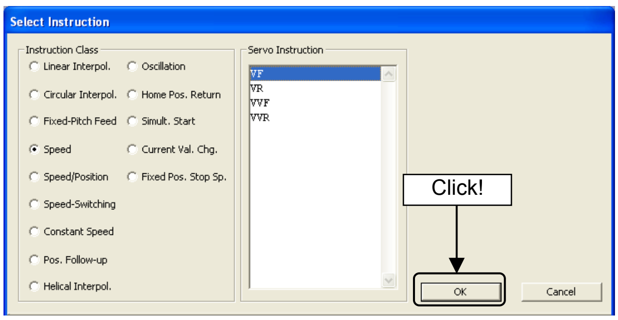
1) Right-click "Servo program" in the Project window, and then click "New Servo Program…".



2) Enter the program No. at the New Servo Program dialog box, and then click the OK button.



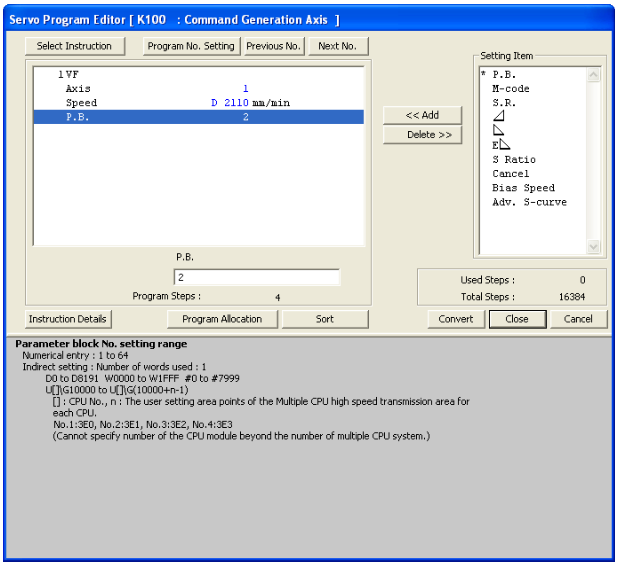
3) Select "Speed" for the "Instruction Class", and "VF" for the "Servo Instruction" at the Select Instruction dialog box, and then click the [OK] button.



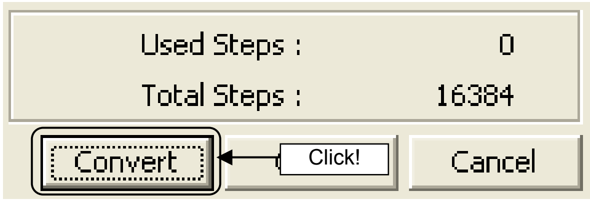
Go to next page

From previous page

4) Enter "1" in the "Axis:" text box, and enter "D 2110" in the "Speed" text box. Add "P.B." from Setting Item, and then enter "2".



5) Click the Convert button.



Zeroing (axis 1)

```
[K    1 : Real Axis ]
   1    ZERO
          Axis                         1
```

Zeroing (axis 2)

```
[K    2 : Real Axis ]
   1    ZERO
          Axis                         2
```

Positioning (1-axis linear)

```
[K  30 : Real Axis ]
   1    ABS-1
          Axis                     1
          ->Address           4000.0   μm
          Speed               5000.00  mm/min
```

Positioning (1-axis linear)

```
[K  31 : Real Axis ]
   1    ABS-1
          Axis                     1
          ->Address          80000.0   μm
          Speed               3000.00  mm/min
```

Positioning (1-axis linear)

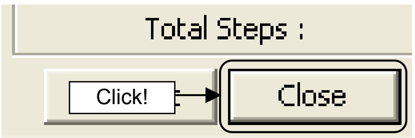
```
[K  32 : Real Axis ]
   1    ABS-1
          Axis                     1
          ->Address         120000.0   μm
          Speed               1000.00  mm/min
```

Go to next page

From previous page

Positioning (2-axis linear interpolation)

```
[K  40 : Real Axis ]
  1   ABS-2(Vector-speed)
       Axis                           1
       ->Address                 D 2020   μm
       Axis                           2
       ->Address                 D 2020   μm
       Vector Speed             1000.00   mm/min
```



(10) After creating motion control steps, click the Close button to close the Edit Servo Program dialog box.

```
[K  20 : Real Axis ]
  1   ABS-2(Vector-speed)
       Axis                       1
       ->Address                0.0   μm
       Axis                       2
       ->Address                0.0   μm
       Vector Speed         4000.00   mm/min
       Dwell                    100   msec
```

(11) The set motion control steps appear in the Program Editor dialog box.

```
[F 200]
   1   //Lamp ON
   2   SET M10
```

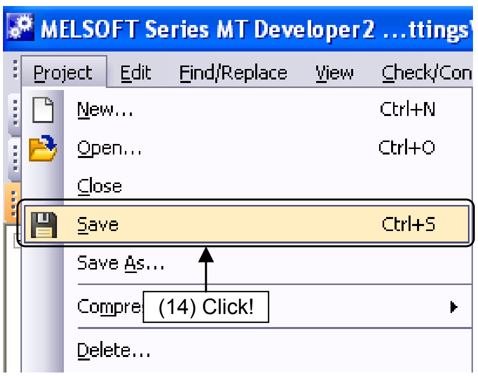
```
[F 201]
   1   //Lamp OFF
   2   RST M10
```

(12) Set the operation control program shown on the left.

(13) Click the Program Editor dialog box Write Motion SFC Diagram button to convert to a motion SFC program.

Refer to section 9.5.7 to 9.5.8 and use the same procedure to create motion SFC programs with the following numbers.

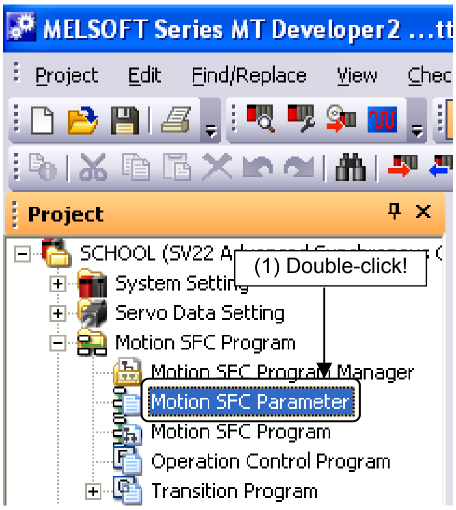
30 40



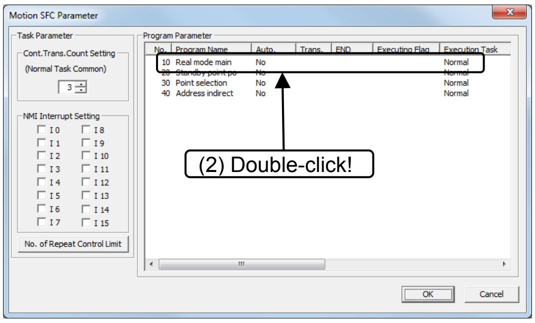
(14) Click [Save] on the [Project] menu at the Program Editor window.
Motion control step entry is now complete.

9.6.5 Motion SFC program parameter settings, batch conversion

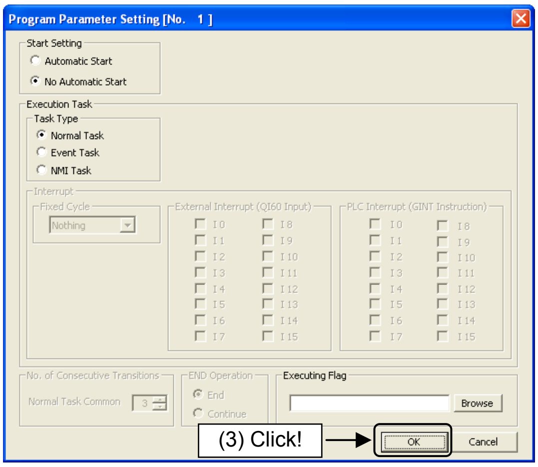
Specify parameter settings and perform batch conversion to motion SFC programs for the created motion SFC programs.



(1) Double-click [Motion SFC Program] → [Motion SFC Parameter] in the Project window.



(2) A Motion SFC Parameter dialog box appears. Created motion SFC programs appear in a list. Double-click "Real mode main".



(3) A Program Parameter Setting dialog box appears.
Ensure that "Start setting" is set to "No automatic start".
Click the OK button after setting.

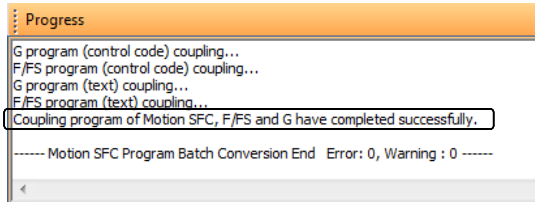
Task (execution timing) settings

1. Normal tasks
   Execution with motion cycle (spare time)
2. Event tasks
   - Execution with fixed cycle (0.22 ms, 0.44 ms, 0.8 ms, 1.7 ms, 3.5 ms, 7.1 ms, 14.2 ms)
   - Execute by entering external interrupts QI60 I0 to I15.
   - Execute with interrupts (I0 to I15) from the PLC (GINT command).
3. NMI tasks (Non-Maskable Interrupt)
   Execute by entering external interrupts QI60 I0 to I15
   Priority is high with event task internal interrupts, even if interrupts are prohibited (DI).

Go to next page

From previous page

(4) Batch convert created SFC diagrams to motion SFC programs.
Click the motion SFC program batch conversion tool button at the Program Editor screen.



(5) When conversion is complete, a "Successful completion" message appears in the output window.
Motion SFC program creation is now complete.

Make corrections to motion SFC programs if a caution message appears.

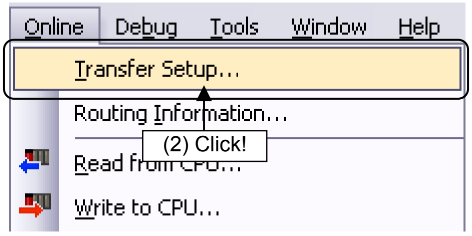
9.7 Writing to the Motion CPU

Write servo settings data and motion SFC programs to the Q172DSCPU.

Point
Select [Change CPU Operation Method] on the [Online] menu, and if the operation method is "Virtual mode switching method", change to "Advanced synchronous control method", and then reboot.

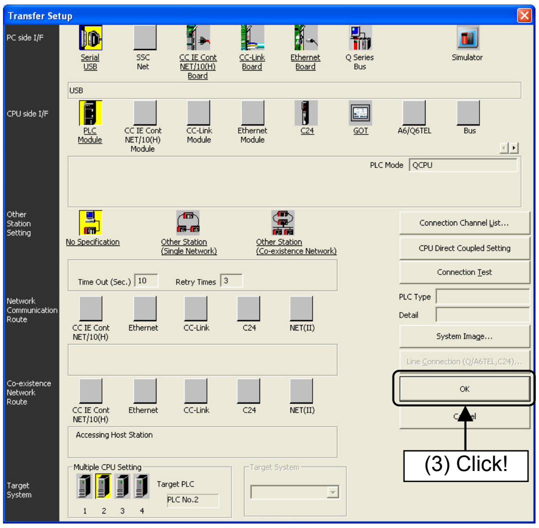
(1) Set the Q motion CPU to "STOP".

(2) Click [Transfer Setup] on the [Online] menu at the Program Editor window.

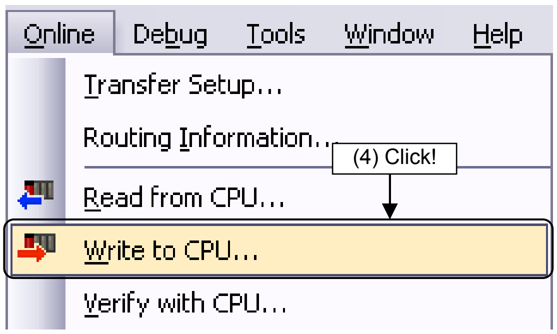


(3) Specify the following settings at the Transfer Setup dialog box that appears, and then click the OK button.

- PC side I/F: Serial USB
- CPU side I/F: PLC Module
- Other Station Setting: No Specification
- Target system: Multiple CPU Setting PLC No.2

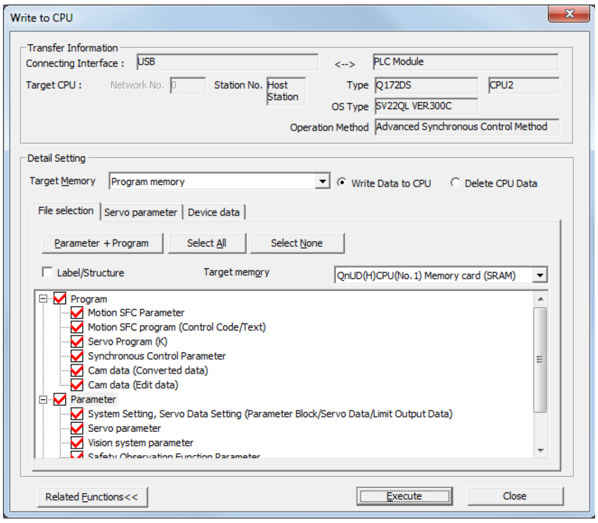


(4) Click [Write to CPU] on the [Online] menu at the Program Editor window.

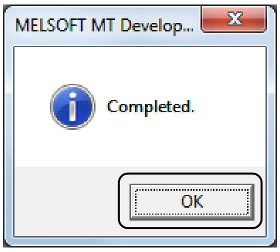


Go to next page

From previous page



(5) Select the "Programs" and "Parameters" check boxes at the CPU Write dialog box that appears, and then click the Execute button.



(6) When a "Complete!" message appears, click the OK button.

(10) Reset the Q PLC CPU.

(11) Run the Q PLC CPU and Q motion CPU.

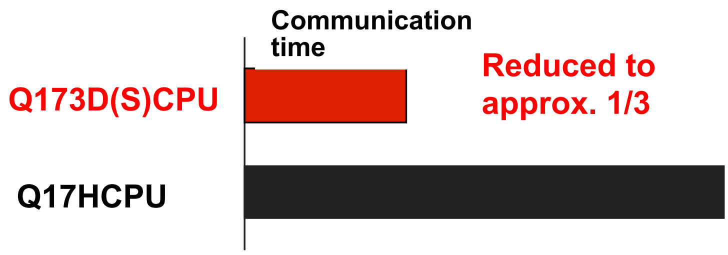
If the Q03UDCPU RUN lamp and Q172DSCPU RUN and M.RUN lamps light up, writing is successful.

POINT

Signification reduction in program data read/write time

With the Q17☐DSCPU, the time required to read/write program data for servo programs and so on is now approximately one third of the time previously taken, facilitating an improvement in debugging efficiency.

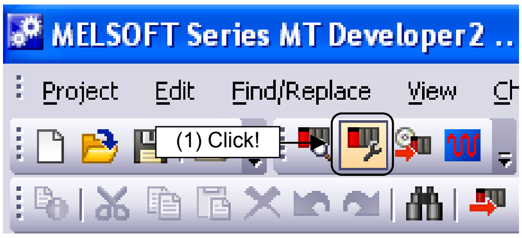
Motion communication time
*** Servo program read time**



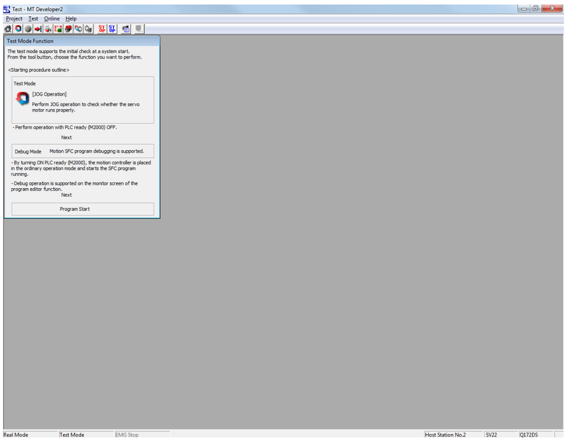
9.8 Test Operation

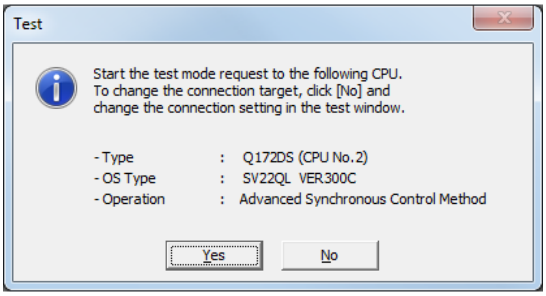
It is necessary to turn OFF PLC ready (M2000) to perform test operation.
Set the Q motion CPU to "STOP", followed by the PLC CPU.

9.8.1 JOG operation



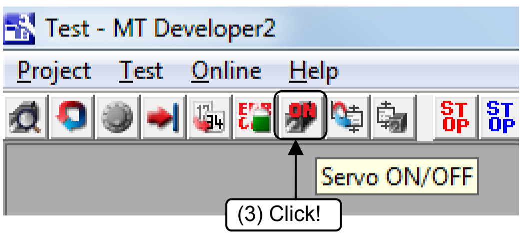
(1) Click the Test tool button at the Program Editor window.





(2) Click Yes at the test mode start request confirmation screen that appears when the Test window appears.

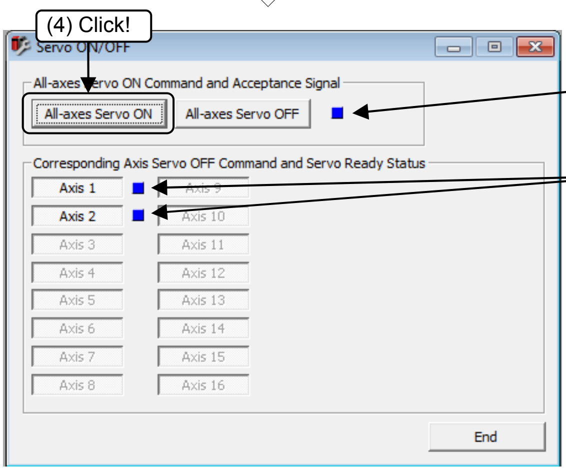
When the motion CPU is in test mode, all test function tool buttons are enabled.
Use the servo ON/OFF tool to turn all servos ON.



(3) Click the Servo ON/OFF tool button.

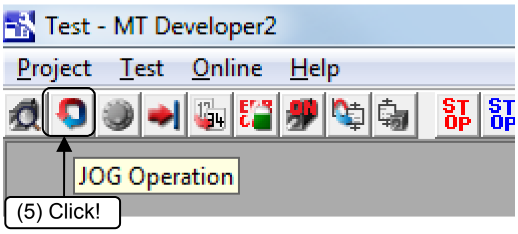
Go to next page

From previous page

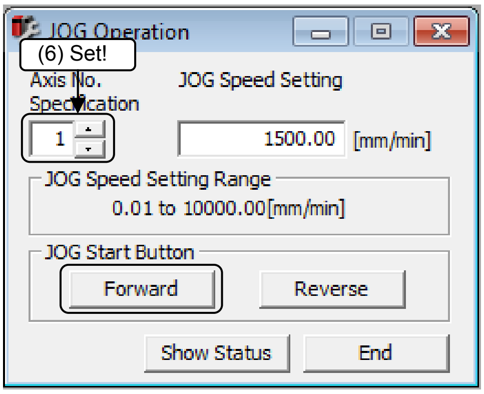


(4) A Servo ON/OFF dialog box appears.
Press the All Axes Servo ON button to turn the servo ON for all axes.

- Black: Servo OFF
  Blue: Servo ON
- Servo ON if turns blue.



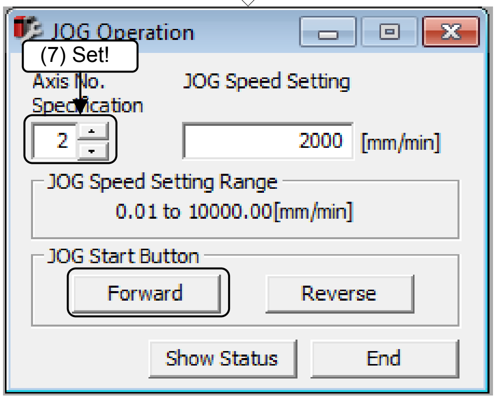
(5) Click the JOG operation tool button.



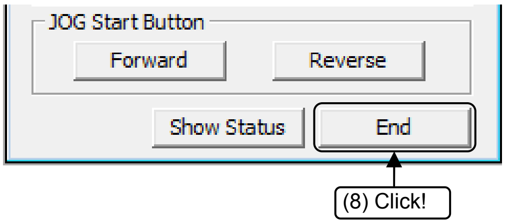
(6) Set the "Axis No." to "1" at the JOG Operation dialog box that appears.
Click the Forward or Reverse button to perform JOG operation.

Go to next page

From previous page



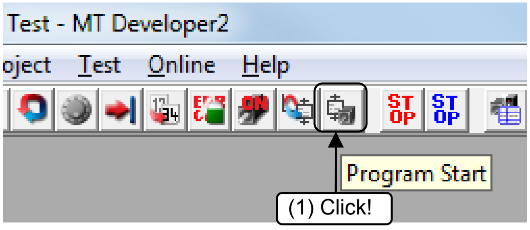
(7) Set the "Axis No." to "2" to perform JOG operation in the same manner as that for axis 1.



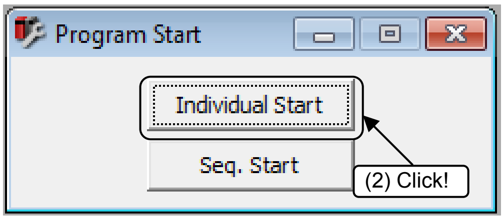
(8) Click the End button to close the JOG Operation dialog box.
JOG operation is now complete.

9.8.2 Servo program execution

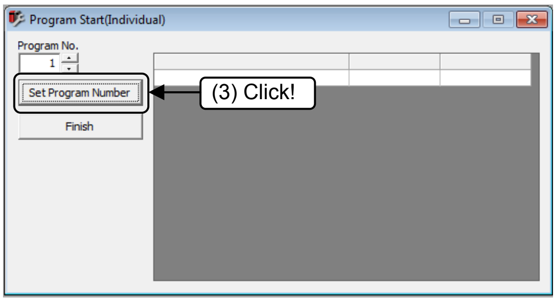
Run the zeroing and positioning servo programs set for program operation in test mode.



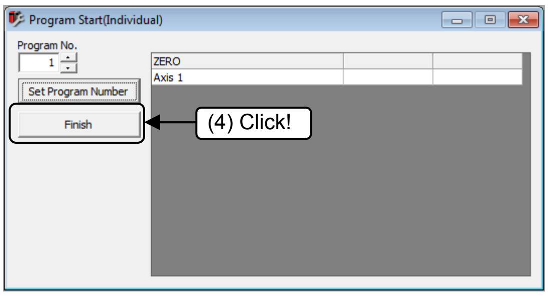
(1) Click the Program operation tool button.



(2) Click the Independent Operation button at the dialog box used to select the program operation type that appears.



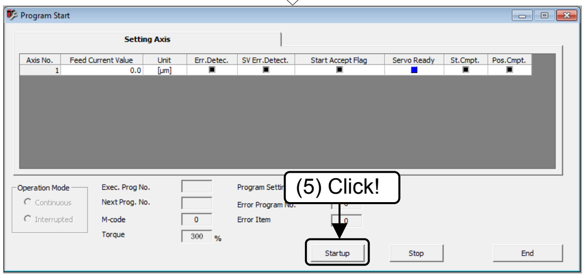
(3) Set "1" at the spin box in the Program Operation (Independent) dialog box that appears, and then click the Program No. Setting button.



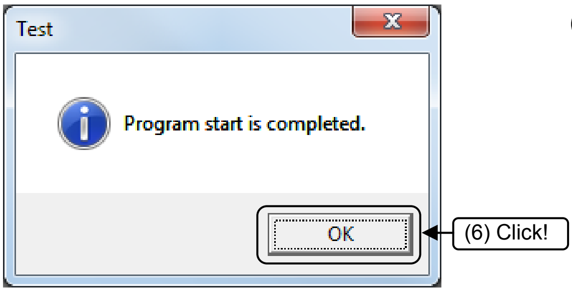
(4) Click the Setting Complete button at the Program Operation (Independent) dialog box.

Go to next page

From previous page

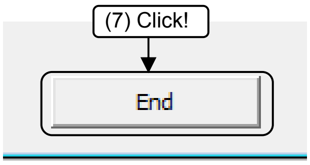


(5) Click the Start button at the Program Operation dialog box that appears.
(Zeroing is performed for axis 1.)



(6) When a "Program operation complete!" message appears, click the OK button.

The feed current value will be "-5000.0 μm".

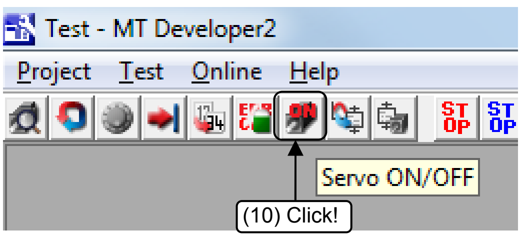


(7) Click the End button at the Program Operation dialog box.

(8) Servo program No.2 is started using the same operations.
(Zeroing is performed for axis 2.)

The feed current value will be "-5000.0 μm".

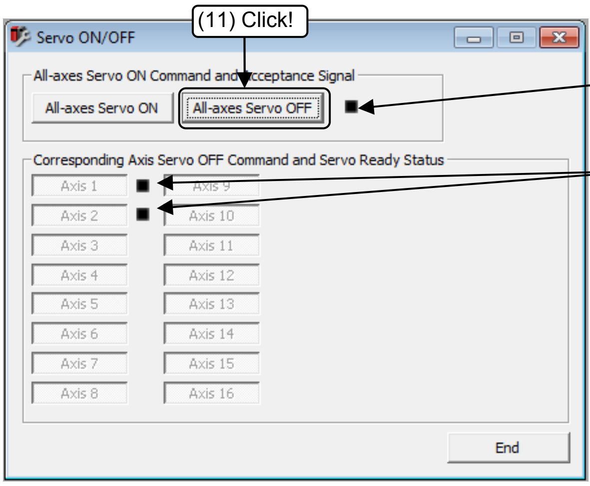
(9) Servo program No.30 is started using the same operations.



(10) Click the Servo ON/OFF tool button.

Go to next page

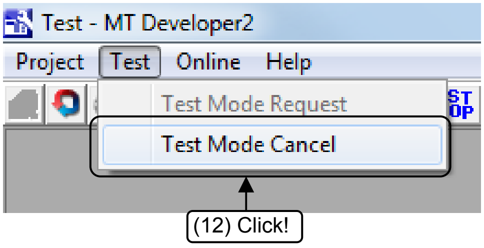
From previous page



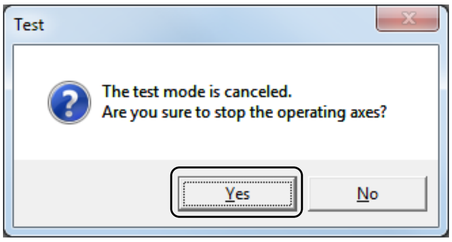
(11) A Servo ON/OFF dialog box appears.
Press the All Axes Servo OFF button to turn the servo OFF for all axes.

■ Black: Servo OFF
Blue: Servo ON

■ Servo OFF if turns black.



(12) Click [Cancel Test Mode] on the [Test] menu at the Test window.



(13) Click the Yes button at the cancel test mode confirmation message box that appears.
Program operation using the test function is now complete.

9.9 Demonstration Machine Operation

9.9.1 Operation

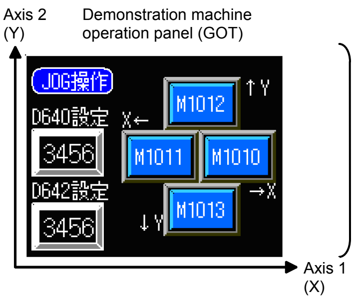
Servo motors are run and servo motor operation is monitored with MT Works2.
Set the PLC CPU and Q motion CPU RUN/STOP switch to "RUN".

[Servo ON]

Press Servo ON at the demonstration machine operation panel.
The servo status for axes 1 and 2 changes to ready.

Demonstration machine operation panel

[JOG operation execution]



Press [JOG] at the demonstration machine operation panel.

Press [JOG/Home Pos.] at the JOG operation panel to turn ON the running lamp.

JOG operation is possible while the JOG operation buttons are ON.

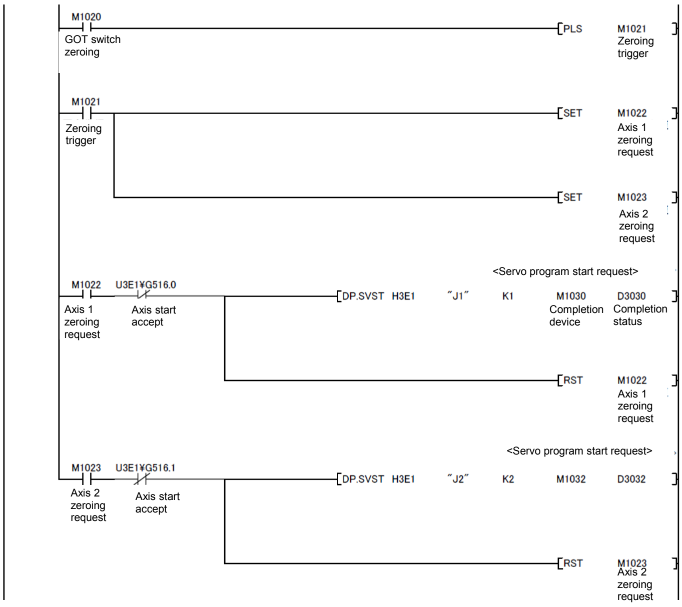
Item	Condition	
Control axis	Axis 1	Axis 2
JOG operation command input	Forward rotation (M1010)	Forward rotation (M1012)
	Reverse rotation (M1011)	Reverse rotation (M1013)

[Zeroing execution]

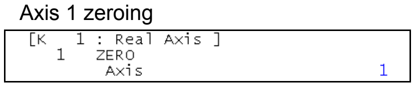
Press [JOG] at the demonstration machine operation panel.

By pressing [JOG/Home Pos.] at the JOG operation panel, the running lamp lights up.
Axes 1 and 2 use proximity dog type 2. Operation is as follows.

• Press zeroing [M1020]: Movement starts in the zeroing direction.

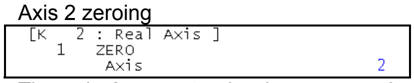


Axis 1 zeroing



The axis 1 current value becomes -5.0 µm.

Axis 2 zeroing



The axis 2 current value becomes -5.0 µm.

POINT

Checks to be performed when there is no movement
- Are the servos ON?
- Are the Q PLC CPU and Q motion CPU switches set to "RUN"?
- Is the computer in test mode? (If in test mode, cancel.)
- Has an alarm occurred? (If so, eliminate the cause.)

Go to next page

From previous page

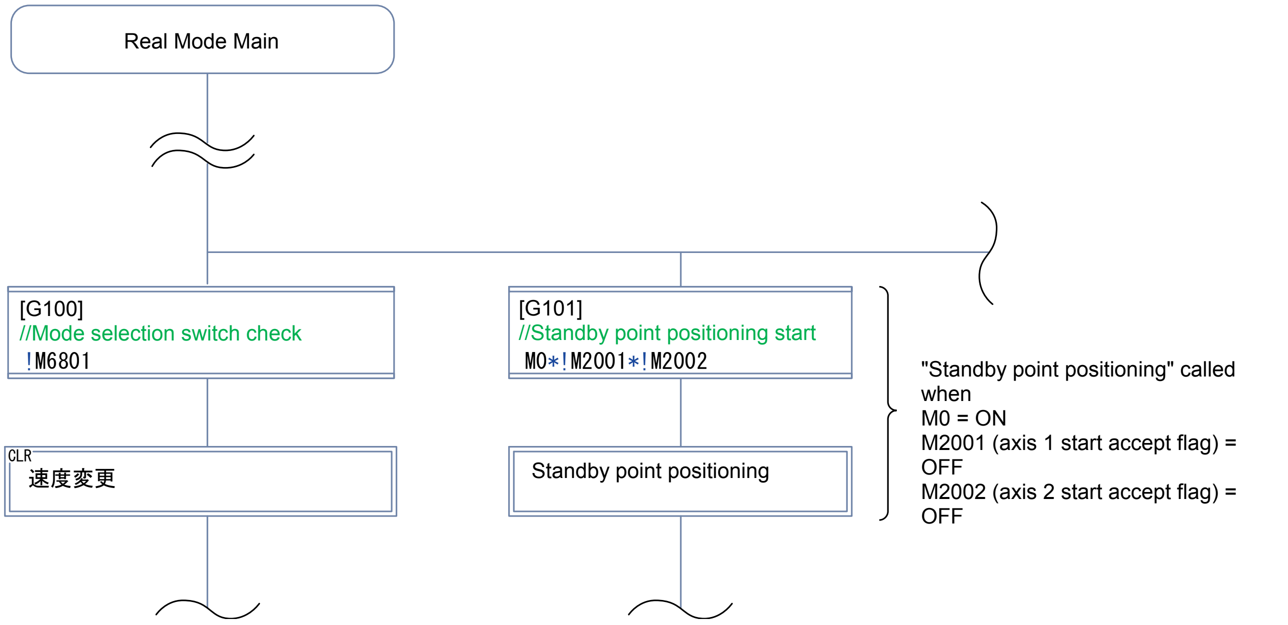
[Standby point positioning]

Press Real at the demonstration machine operation panel.

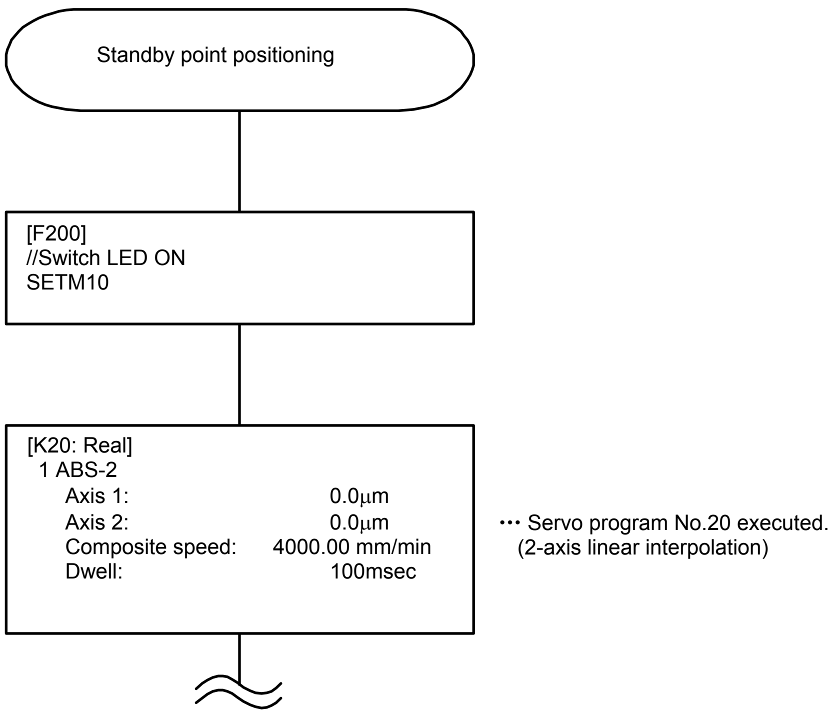
(1) Press Real Mode Main at the Real Mode screen to turn ON the running lamp.

(2) By pressing Standby Point, positioning is performed at the standby point address (0.0).

[Real mode main] program (motion SFC program No.10)



[Standby point positioning] program (motion SFC program No.20)

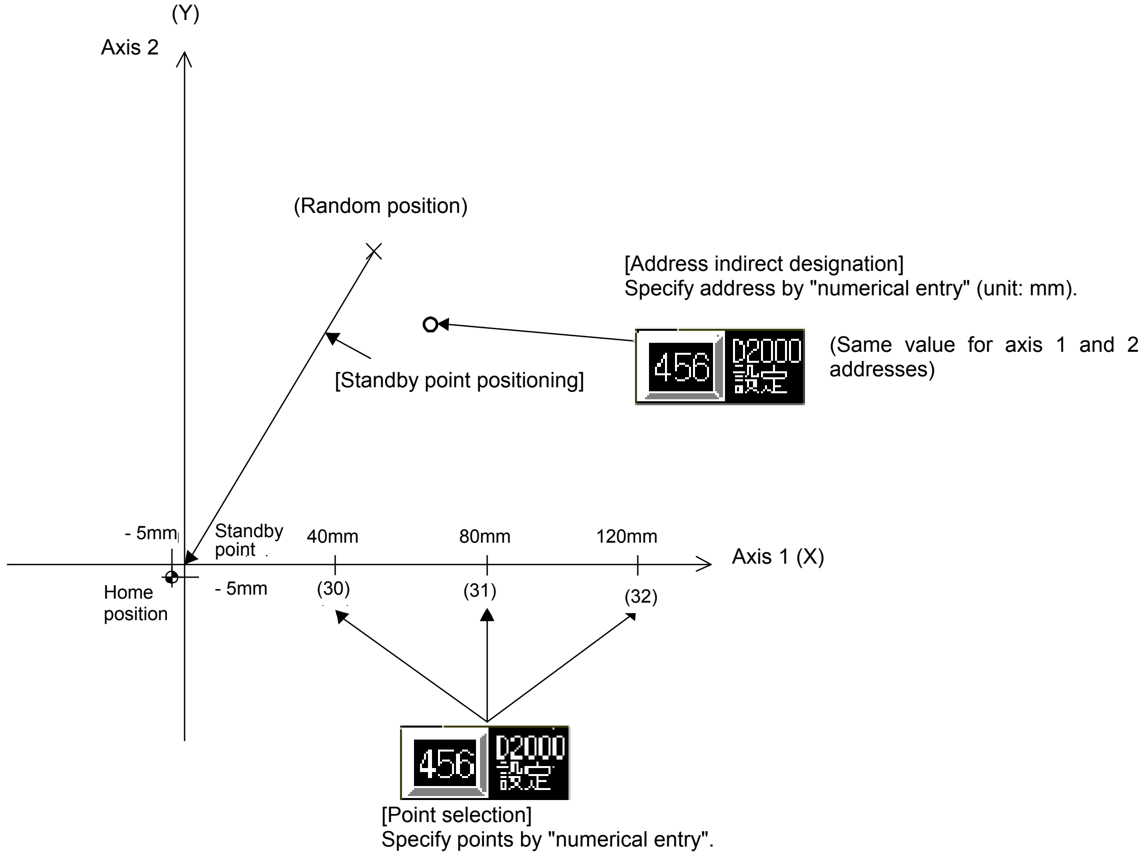


Go to next page

[Real mode main]
[Address indirect designation]
[Point selection]

From previous page

- By turning [Position Select] ON, positioning is performed with the [Point selection] locus.
- By turning [Indirect Designation] ON, positioning is performed with [Address indirect designation].



[Speed change]

Speed change/temporary stop during operation

- By turning [2000] ON, the speed will be 2000 mm/min.
- By turning [1000] ON, the speed will be 1000 mm/min.
- By turning [500] ON, the speed will be 500 mm/min.
- By turning [0] ON, operation will temporarily stop.

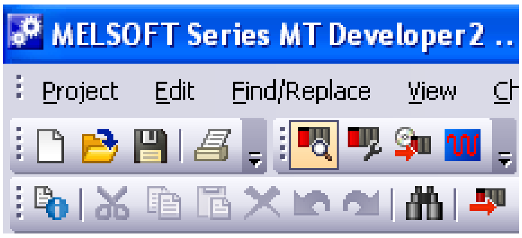
(The speed may be changed multiple times during operation.
However, do not perform operation during zeroing, circular interpolation, or during deceleration. A minor error will occur.)

Operation complete

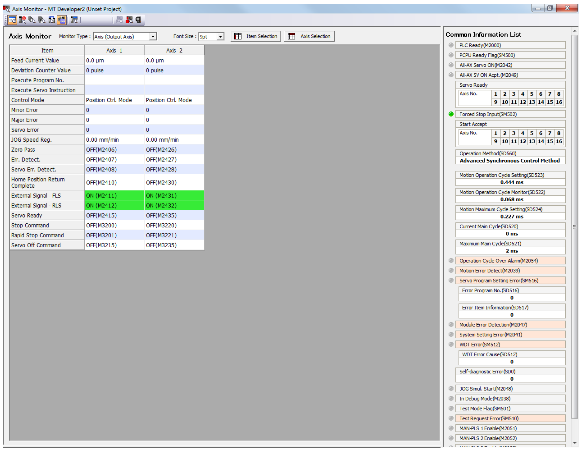
9.9.2 Monitor operation with monitor screen

Current values and error causes and so on can be checked using the Monitor screen.

(1) Monitor startup

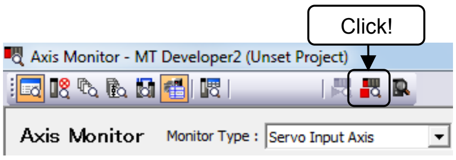


1) Click the monitor icon on the toolbar.

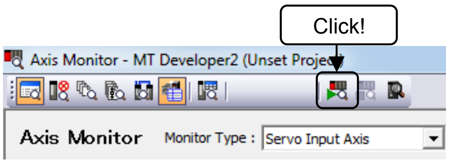


2) The monitor starts up.

(2) Stopping/starting the monitor

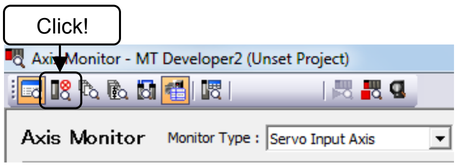


1) To stop the monitor, click the "Stop monitor" button on the Monitor screen toolbar.

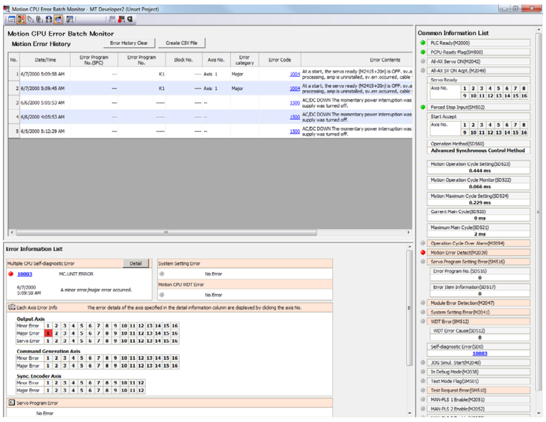


2) To start the monitor again, click the "Start monitor" button on the Monitor screen toolbar.

(3) Motion CPU error batch monitor



1) Click the "Motion CPU error batch monitor" button on the Monitor screen toolbar.



2) The Motion CPU error batch monitor appears.

POINT

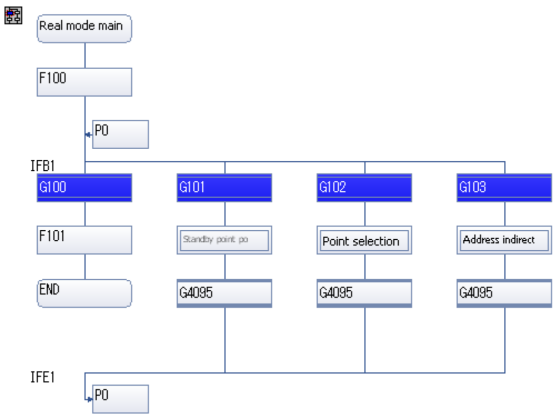
By using the Motion CPU error batch monitor, all motion CPU error information is displayed on the monitor.

9.9.3 Motion SFC program monitor

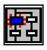
This section describes how to display the motion CPU program monitor. The start and stop status of each program, and current device values can be monitored and so on.

(1) Mode change

1) Click the "Monitor mode" button at the Program Editor screen.



2) The motion SFC program changes to monitor mode.

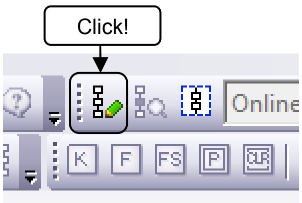
: Executing

: Stopped

: Taking break

(Blue): Active

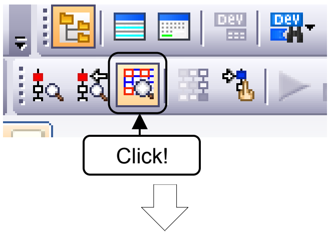
(Red): Awaiting parallel connection



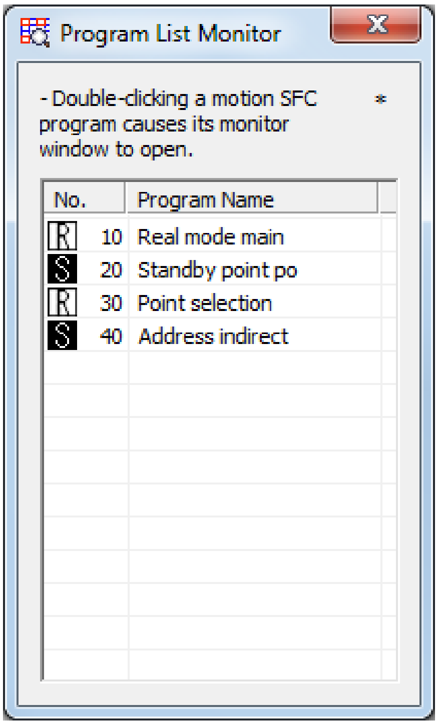
3) By clicking the "Edit mode" button at the Edit Program screen, the mode changes to edit mode.

(2) **Program batch monitor**

Displays the program start and stop statuses in a list.



1) Click the program list monitor button.



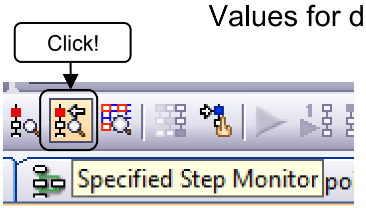
2) The program list monitor appears.

 : Executing

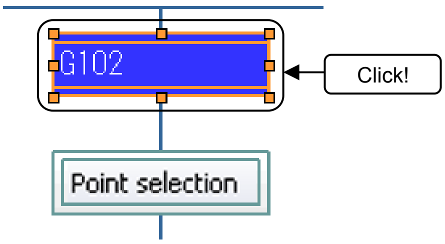
 : Stopped

(3) Specific step monitor

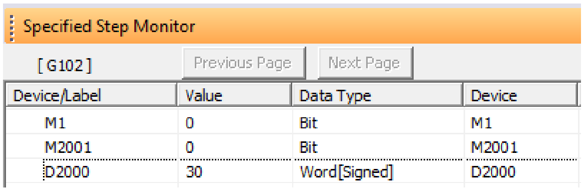
Values for devices used at selected steps can be monitored.



1) Click the specified step monitor button.



2) Click the step to be monitored.


Specified Step Monitor

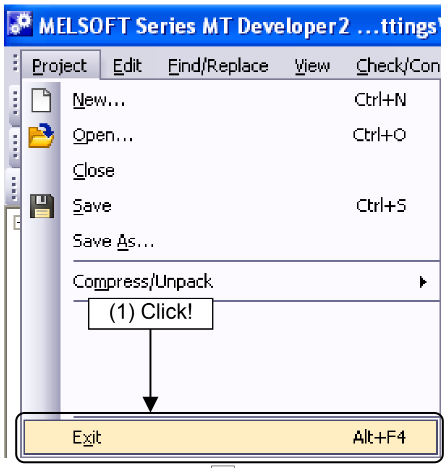
[G102] Previous Page Next Page

Device/Label	Value	Data Type	Device
M1	0	Bit	M1
M2001	0	Bit	M2001
D2000	30	Word[Signed]	D2000

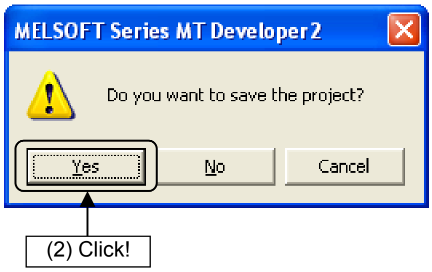
3) Values for devices at specific steps can be monitored.

9.10 Exit Operation

9.10.1 Exiting MT Works2

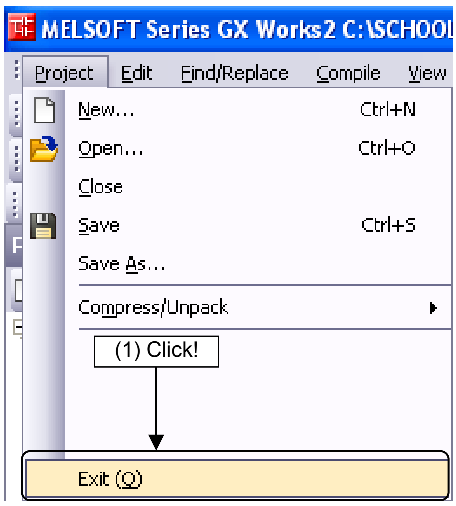


(1) Click [Exit] on the [Project] menu.

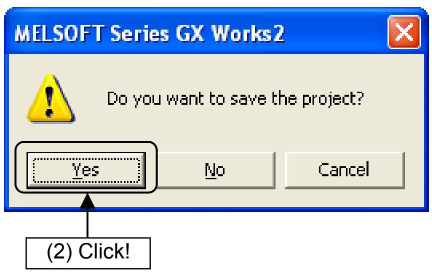


(2) If any changes have been made to setting data, a message appears to confirm whether to save the project.
Click the Yes button.

9.10.2 Exiting GX Works2



(1) Click [Exit] on the GX Works2 [Project] menu.



(2) If the project has not been saved, a message appears to confirm whether to save the project.
Click the Yes button.

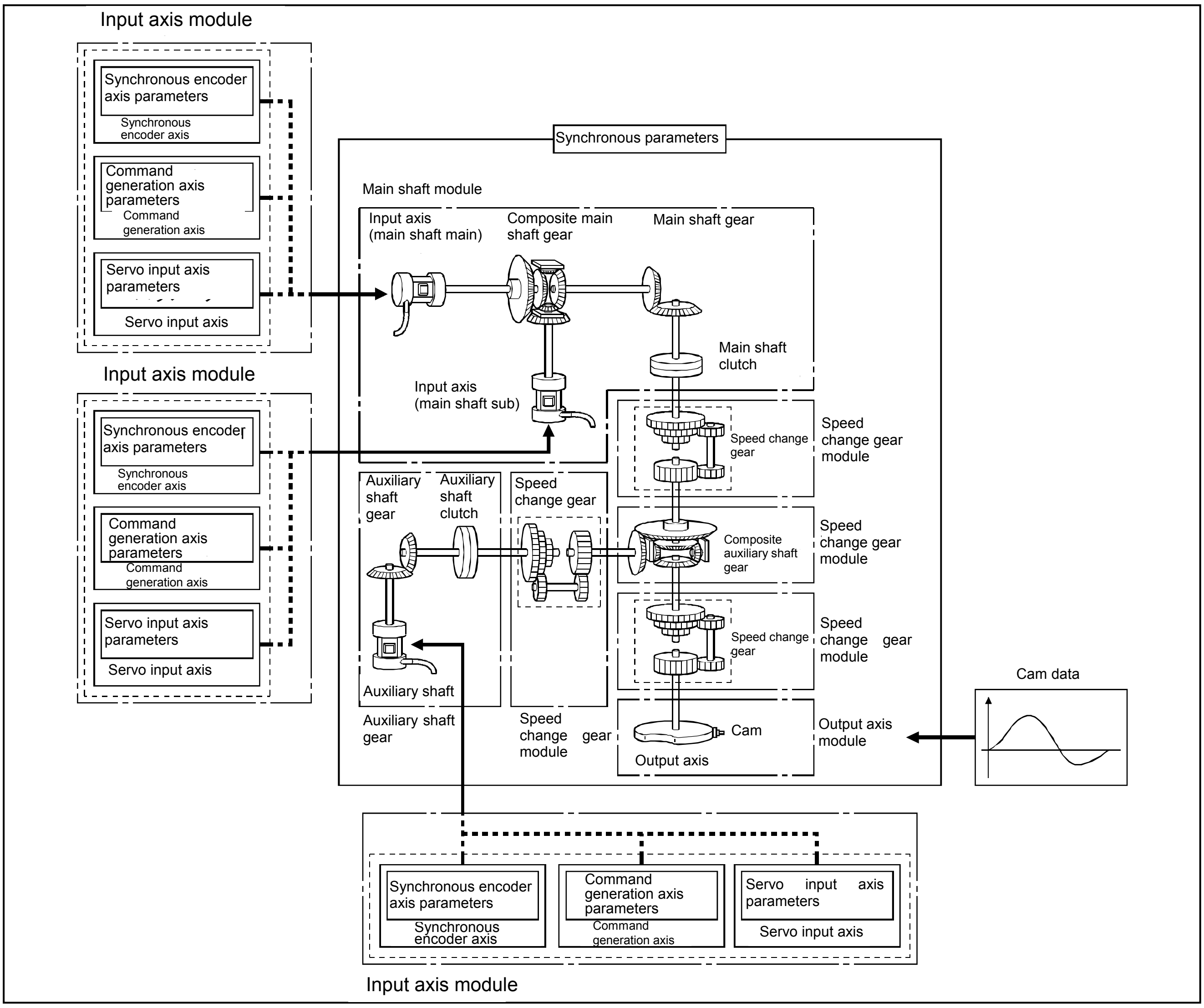
Chapter 10 SV22 Advanced Synchronous Control Practice

10.1 Synchronous Control Parameters

By starting synchronous control for each output axis, control is synchronized for input axes (servo input axis, command generation axis, synchronous encoder axis).

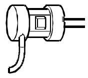
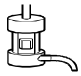
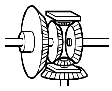
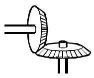
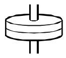
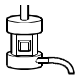
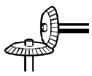
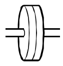
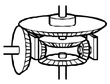
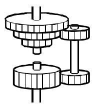
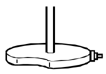
10.1.1 Synchronous control modules

The modules used with synchronous control are shown below.



10.1.2 Synchronous control module list

The number of modules that can be used with synchronous control is shown below. (Indicates the number of modules for Q172DSCPU.)

Category	Name	Part drawing	No. of usable modules	
			Per unit	Per axis
Input axis module	Servo input axis	-	16	-
	Command generation axis	-	16	-
	Synchronous encoder axis	-	12	-
Main shaft module	Main shaft main input axis		16	1
	Main shaft sub input axis		16	1
	Composite main shaft gear		16	1
	Main shaft gear		16	1
	Main shaft clutch		16	1
Auxiliary axis module	Auxiliary axis		16	1
	Auxiliary axis gear		16	1
	Auxiliary shaft clutch		16	1
	Composite auxiliary shaft gear		16	1
Speed change gear module	Speed change gear		32	2
Output axis module	Output axis		16	1
Cam data	Cam data	-	Max. 256	-

10.1.3 Servo input axes

Servo input axes are used to drive input axes based on the position of servo motors controlled with the motion CPU (Q173DSCPU/Q172DSCPU).

Symbol	Setting item	Setting details	Setting value	Load cycle	Default	Device No.
Pr.300	Servo input axis type	Sets the current value type from which the servo input axis input value is generated.	0: Disable 1: Feed current value 2: Real current value 3: Servo command value 4: Feedback value	When power turned ON	0	-
Pr.301	Servo input axis smoothing time constant	Set if performing smoothing processing for input values.	0 to 5000 [ms]		0 [ms]	-
Pr.302	Servo input axis phase compensation advance time	Sets the time to advance or delay the phase.	-2147483648 to 2147483647 [μs]	Operation cycle	0 [μs]	D14600+2n D14601+2n
Pr.303	Servo input axis phase compensation time constant	Sets the time to reflect phase compensation.	0 to 65535 [ms]	When power turned ON	10 [ms]	-
Pr.304	Servo input axis rotation direction restriction	Set if restricting the input travel value to a single direction.	0: No rotation direction restriction 1: Permit only when current value is increase direction 2: Permit only when current value is decrease direction		0	-

10.1.4 Command generation axes

Axes used to perform command generation only can be controlled independently of axes connected to servo amps. Command generation axes are used if driving input axes with servo programs or with JOG operation.

Symbol	Setting item	Setting details	Setting value	Load cycle	Default	Device No.
Pr.340	Command generation axis enable setting	Enables/disables the used command generation axis.	0: Disable 1: Enable	When power turned ON	0	-
Pr.341	Command generation axis unit setting	Sets the command generation axis unit.	0: mm 1: inch 2: degree 3: PLS		3	-
Pr.342	Command generation axis upper stroke limit	Sets the command generation axis upper stroke limit.	-2147483648 to 2147483647 (when degree: 0 to 35999999) [Command generation axis position unit] *1		0	-
Pr.343	Command generation axis lower stroke limit	Sets the command generation axis lower stroke limit.	-2147483648 to 2147483647 (when degree: 0 to 35999999) [Command generation axis position unit] *1		0	-
Pr.344	Command generation axis command in-position range	Sets the command generation axis command in-position range.	1 to 2147483647 [Command generation axis position unit] *1		100	-
Pr.345	Command generation axis degree axis speed 10 times designation	Sets whether to perform positioning control at a speed 10 times the command speed setting value when the command generation axis unit is degree.	0: Disable 1: Enable		0	-

Symbol	Setting item	Setting details	Setting value	Load cycle	Default	Device No.
Pr.346	Command generation axis 1 cycle length	Sets the command generation axis 1 cycle length.	0: Disable 1 to 2147483647 [Command generation axis position unit] *1	When power turned ON	0	-
Pr.347	Command generation axis JOG speed limit value	Sets the speed limit value when performing JOG operation for a command generation axis.	1 to 2147483647 [Command generation axis speed unit] *2		20000	-
Pr.348	Command generation axis JOG operation parameter block designation	Sets the No. of the parameter block used when performing JOG operation for a command generation axis.	1 to 64	When starting JOG operation	1	D14682+4n
Pr.349	Command generation axis acceleration/deceleration time change enable device *3	Sets the bit device used to permit acceleration/deceleration time changes when requesting a speed change.	Bit device (X, Y, M, B, F, U□\G)	When power turned ON	-	Optional device
Pr.350	Command generation axis acceleration time change value device *3	Sets the word device used to set the acceleration time change value.	Word device (D, W, #, U□\G)		-	Optional device
Pr.351	Command generation axis deceleration time change value device *3	Sets the word device used to set the deceleration time change value.	Word device (D, W, #, U□\G)		-	Optional device

*1: Command generation axis position unit
*2: Command generation axis speed unit
*3: This setting can be omitted.

10.1.5 Synchronous encoder axes

Use if driving input axes with input pulses from externally connected synchronous encoders.

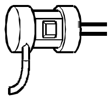
Symbol	Setting item	Setting details	Setting value	Load cycle	Default	Device No.
Pr.320	Synchronous encoder axis type	• Sets the type of synchronous encoder axis used. • Sets the master CPU input axis if using as a slave CPU with multiple CPU high speed synchronous control.	0: Disable 1: Synchronous encoder Pn (synchronous encoder axis No.: 1 to 12) 201: Via device 301: Master CPU servo input axis (Axis No.: 1 to 32) 401: Master CPU command generation axis (Axis No.: 1 to 32) 501: Master CPU synchronous encoder axis (Axis No.: 1 to 12)	When power turned ON	0	-
Pr.321	Synchronous encoder axis unit setting	• Sets the synchronous encoder axis unit. • The position unit is set in the "×1 to 10^{-9} [control unit]" range. • The speed unit is set in the "×1 to 10^{-9} [control unit/s, or control unit/min]" range.	Control unit 0: mm 1: inch 2: degree 3: PLS		3	-
			No. of position decimal point digits 0 to 9		0	
			Speed time unit 0: sec 1: mm		0	
			No. of speed decimal point digits 0 to 9		0	

Symbol	Setting item	Setting details	Setting value	Load cycle	Default	Device No.
Pr.322	Synchronous encoder axis unit conversion numerator	Sets the numerator for converting synchronous encoder axis encoder pulses to synchronous encoder axis units.	-2147483648 to 2147483647 [Synchronous encoder axis position unit] *1	When power turned ON	1	-
Pr.323	Synchronous encoder axis unit conversion denominator	Sets the denominator for converting synchronous encoder axis encoder pulses to synchronous encoder axis units.	1 to 2147483647 [PLS]		1 [PLS]	-
Pr.324	Synchronous encoder axis 1 cycle length	Sets the synchronous encoder axis 1 cycle length.	1 to 2147483647 [Synchronous encoder axis position unit] *1		4000	-
Pr.325	Synchronous encoder axis smoothing time constant	Set if performing smoothing processing for input values.	0 to 5000 [ms]		0 [ms]	-
Pr.326	Synchronous encoder axis phase compensation advance time	Sets the time to advance or delay the phase.	-2147483648 to 2147483647 [μs]	Operation cycle	0 [μs]	D14820+10n D14821+10n
Pr.327	Synchronous encoder axis phase compensation time constant	Sets the time to reflect phase compensation.	0 to 65535 [ms]	When power turned ON	10 [ms]	-
Pr.328	Synchronous encoder axis rotation direction restriction	Set if restricting the input travel value to a single direction.	0: No rotation direction restriction 1: Permit only when current value is increase direction 2: Permit only when current value is decrease direction		0	-
Pr.329	Synchronous encoder via device resolution	• Sets the type of synchronous encoder axis using synchronous encoder resolution when the synchronous encoder axis type is synchronous encoder via device. • If 0 is set, processing is performed with the synchronous encoder via device input value as a 32 bit counter.	0 to 2147483647 [PLS]		0 [PLS]	-

*1: Synchronous encoder axis position unit

10.1.6 Main shaft main input axis

This is the input axis at the main shaft module main side. This is the reference for the main shaft position.

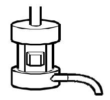


Symbol	Setting item	Setting details	Setting value	Load cycle	Default	Device No.
Pr.400	Main input axis No.	Sets the input axis No. at the main shaft input main side.	0 : Disable 1 to 32 : Servo input axis *1 201 to 232 : Command generation axis *2 801 to 812 : Synchronous encoder axis	When starting Synchro-nous control	0	D15000+150n

*1: With the Q172DSCPU, the 1 to 16 range is valid.
*2: With the Q172DSCPU, the 201 to 216 range is valid.

10.1.7 Main shaft sub input axis

This is the input axis at the main shaft module sub side. This is used if entering a compensation amount for the main shaft main input axis position.

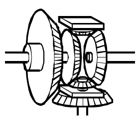


Symbol	Setting item	Setting details	Setting value	Load cycle	Default	Device No.
Pr.401	Sub input axis No.	Sets the input axis No. at the main shaft input sub side.	0 : Disable 1 to 32 : Servo input axis *1 201 to 232 : Command generation axis *2 801 to 812 : Synchronous encoder axis	When starting Synchro-nous control	0	D15001+150n

*1: With the Q172DSCPU, the 1 to 16 range is valid.
*2: With the Q172DSCPU, the 201 to 216 range is valid.

10.1.8 Composite main shaft gear

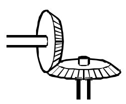
The main shaft main input axis and main shaft sub input axis travel values are compounded and transferred to the main shaft gear.



Symbol	Setting item	Setting details	Setting value	Load cycle	Default	Device No.
Pr.402	Composite main shaft gear	Selects the input value composition method from main input axis and sub input axis.	• Set in hexadecimal notation. H□□□□ Main input method 0: No input 1: Input + 2: Input - Sub input method 0: No input 1: Input + 2: Input -	Operation cycle	0001h	D15002+150n

10.1.9 Main shaft gear

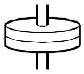
The gear ratio for which the travel value after the composite main shaft gear is set is converted and transferred.



Symbol	Setting item	Setting details	Setting value	Load cycle	Default	Device No.
Pr.403	Main shaft gear numerator	Sets the main shaft gear numerator.	-2147483648 to 2147483647	When starting synchronous control	1	D15004+150n D15005+150n
Pr.404	Main shaft gear denominator	Sets the main shaft gear denominator.	1 to 2147483647		1	D15006+150n D15007+150n

10.1.10 Main shaft clutch

The main shaft travel value is turned ON and OFF with the clutch and transferred. This is used if conveying/isolating command pulses from main shaft input to the output axis module side, and controlling servo motor operation/stoppage.



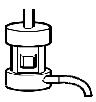
Symbol	Setting item	Setting details	Setting value	Load cycle	Default	Device No.
Pr.405	Main shaft clutch control setting	Sets the clutch control method.	• Set in hexadecimal notation. H□□□□ ON control mode 0: No clutch 1: Clutch command ON/OFF 2: Clutch command Rising 3: Clutch command Falling 4: Address mode 5: High-speed input request OFF control mode 0: OFF control disabled 1: One shot OFF 2: Clutch command Rising 3: Clutch command Falling 4: Address mode 5: High-speed input request High-speed input request signal 00 to 1F: Signal 1 to 32 high-speed input request signal	Operation cycle	0000h	D15008+150n
Pr.406	Main shaft clutch reference address setting	Sets the clutch reference address.	0: Current value after composite main shaft gear 1: Current value per cycle after main shaft gear	When starting Synchro-nous control	0	D15009+150n
Pr.407	Main shaft clutch ON address	• Sets the address for turning ON the clutch when in address mode. (The setting is invalid when in other than address mode.) • If other than "0 to (cam axis 1 cycle length -1)", the clutch is controlled after converting to the "0 to (cam axis 1 cycle length -1)" range.	-2147483648 to 2147483647 [Main input axis position unit *1, or cam axis cycle unit *2]	Operation cycle	0	D15010+150n D15011+150n
Pr.408	Travel value before main shaft clutch ON	• Sets the travel value until the clutch is actually turned ON after the clutch ON conditions are established. • Set a positive value for movements in the increase direction, and negative value for movements in the decrease direction.	-2147483648 to 2147483647 [Main input axis position unit *1, or cam axis cycle unit *2]	When clutch ON conditions established	0	D15012+150n D15013+150n

Symbol	Setting item	Setting details	Setting value	Load cycle	Default	Device No.
Pr.409	Main shaft clutch OFF address	• Sets the address for turning OFF the clutch when in address mode. (The setting is invalid when in other than address mode.) • If other than "0 to (cam axis 1 cycle length -1)", the clutch is controlled after converting to the "0 to (cam axis 1 cycle length -1)" range.	-2147483648 to 2147483647 [Main input axis position unit *1, or cam axis cycle unit *2]	Operation cycle	0	D15014+150n D15015+150n
Pr.410	Travel value before main shaft clutch OFF	• Sets the travel value until the clutch is actually turned OFF after the clutch OFF conditions are established. • Set a positive value for movements in the increase direction, and negative value for movements in the decrease direction.	-2147483648 to 2147483647 [Main input axis position unit *1, or cam axis cycle unit *2]	When clutch OFF conditions established	0	D15016+150n D15017+150n
Pr.411	Main shaft clutch smoothing method	Sets the clutch smoothing method.	0: Direct 1: Time constant method (index) 2: Time constant method (linear) 3: Slippage amount method (index) 4: Slippage amount method (linear)	When starting Synchro-nous control	0	D15018+150n
Pr.412	Main shaft clutch Smoothing time constant	Sets the smoothing time constant if time constant method smoothing.	0 to 5000 [ms]		0 [ms]	D15019+150n
Pr.413	Slippage amount at main shaft clutch ON	Sets the slippage amount when the clutch is ON if slippage amount method smoothing.	0 to 2147483647 [Main input axis position unit *1, or cam axis cycle unit *2]	When clutch ON starts	0	D15020+150n D15021+150n
Pr.414	Slippage amount at main shaft clutch OFF	Sets the slippage amount when the clutch is OFF if slippage amount method smoothing.	0 to 2147483647 [Main input axis position unit *1, or cam axis cycle unit *2]	When clutch OFF starts	0	D15022+150n D15023+150n

*1: Main input axis position unit
*2: Cam axis cycle unit

10.1.11 Auxiliary shafts

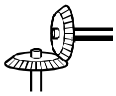
These are input axes for auxiliary shaft modules. Input values are generated from auxiliary shafts. Furthermore, input values can be converted to values taking the mechanical reduction ratio and rotation direction into consideration with an auxiliary shaft gear.



Symbol	Setting item	Setting details	Setting value	Load cycle	Default	Device No.
Pr.418	Auxiliary shaft No.	Sets the auxiliary shaft input axis No.	0 : Disable 1 to 32 : Servo input axis *1 201 to 232 : Command generation axis *2 801 to 812 : Synchronous encoder axis	When starting synchro-nous control	0	D15024+150n

10.1.12 Auxiliary shaft gear

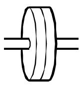
The auxiliary shaft travel value is converted with the set gear ratio and transferred.



Symbol	Setting item	Setting details	Setting value	Load cycle	Default	Device No.
Pr.420	Auxiliary shaft gear numerator	Sets the auxiliary shaft gear numerator.	-2147483648 to 2147483647	When starting synchro-nous control	1	D15026+150n D15027+150n
Pr.421	Auxiliary shaft gear denominator	Sets the auxiliary shaft gear denominator.	1 to 2147483647		1	D15028+150n D15029+150n

10.1.13 Auxiliary shaft clutch

The auxiliary shaft travel value is turned ON and OFF with the clutch and transferred.
This is used if conveying/isolating command pulses from auxiliary shaft input to the output axis module side, and controlling servo motor operation/stoppage.



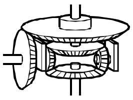
Symbol	Setting item	Setting details	Setting value	Load cycle	Default	Device No.
Pr.422	Auxiliary shaft clutch control setting	Sets the clutch control method.	• Set in hexadecimal notation. H□□□□ ON control mode 0: No clutch 1: Clutch command ON/OFF 2: Clutch command Rising 3: Clutch command Falling 4: Address mode 5: High-speed input request OFF control mode 0: OFF control disabled 1: One shot OFF 2: Clutch command Rising 3: Clutch command Falling 4: Address mode 5: High-speed input request High-speed input request signal 00 to 1F: Signal 1 to 32 high-speed input request signal	Operation cycle	0000h	D15030+150n
Pr.423	Auxiliary shaft clutch reference address setting	Sets the clutch reference address.	0: Auxiliary shaft current value 1: Current value per cycle after auxiliary shaft gear	When starting synchro-nous control	0	D15031+150n
Pr.424	Auxiliary shaft clutch ON address	• Sets the address for turning ON the clutch when in address mode. (The setting is invalid when in other than address mode.) • If other than "0 to (cam axis 1 cycle length -1)", the clutch is controlled after converting to the "0 to (cam axis 1 cycle length -1)" range.	-2147483648 to 2147483647 [Auxiliary input axis position unit *1, or cam axis cycle unit *2]	Operation cycle	0	D15032+150n D15033+150n

Symbol	Setting item	Setting details	Setting value	Load cycle	Default	Device No.
Pr.425	Travel value before auxiliary shaft clutch ON	• Sets the travel value until the clutch is actually turned ON after the clutch ON conditions are established. • Set a positive value for movements in the increase direction, and negative value for movements in the decrease direction.	-2147483648 to 2147483647 [Auxiliary input axis position unit *1, or cam axis cycle unit *2]	When clutch ON conditions established	0	D15034+150n D15035+150n
Pr.426	Auxiliary shaft clutch OFF address	• Sets the address for turning OFF the clutch when in address mode. (The setting is invalid when in other than address mode.) • If other than "0 to (cam axis 1 cycle length -1)", the clutch is controlled after converting to the "0 to (cam axis 1 cycle length -1)" range.	-2147483648 to 2147483647 [Auxiliary input axis position unit *1, or cam axis cycle unit *2]	Operation cycle	0	D15036+150n D15037+150n
Pr.427	Travel value before auxiliary shaft clutch OFF	• Sets the travel value until the clutch is actually turned OFF after the clutch OFF conditions are established. • Set a positive value for movements in the increase direction, and negative value for movements in the decrease direction.	-2147483648 to 2147483647 [Auxiliary input axis position unit *1, or cam axis cycle unit *2]	When clutch OFF conditions established	0	D15038+150n D15039+150n
Pr.428	Auxiliary shaft clutch smoothing method	Sets the clutch smoothing method.	0: Direct 1: Time constant method (index) 2: Time constant method (linear) 3: Slippage amount method (index) 4: Slippage amount method (linear)	When starting Synchro-nous control	0	D15040+150n
Pr.429	Auxiliary shaft clutch smoothing time constant	Sets the smoothing time constant if time constant method smoothing.	0 to 5000 [ms]		0 [ms]	D15041+150n
Pr.430	Slippage amount at auxiliary shaft clutch ON	Sets the slippage amount when the clutch is OFF if slippage amount method smoothing.	0 to 2147483647 [Auxiliary input axis position unit *1, or cam axis cycle unit *2]	When clutch ON starts	0	D15042+150n D15043+150n
Pr.431	Slippage amount at auxiliary shaft clutch OFF	Sets the slippage amount when the clutch is OFF if slippage amount method smoothing.	0 to 2147483647 [Auxiliary input axis position unit *1, or cam axis cycle unit *2]	When clutch OFF starts	0	D15044+150n D15045+150n

*1: Auxiliary shaft position unit
*2: Cam axis cycle unit

10.1.14 Auxiliary shaft clutch

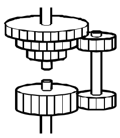
Main shaft and auxiliary shaft travel values are compounded and transferred.



Symbol	Setting item	Setting details	Setting value	Load cycle	Default	Device No.
Pr.419	Auxiliary shaft composite gear	Selects the input value composition method from the main shaft and auxiliary shaft.	• Set in hexadecimal notation. H□□□□ Main shaft input method 0: No input 1: Input + 2: Input - Auxiliary shaft input method 0: No input 1: Input + 2: Input -	Operation cycle	0001h	D15025+150n

10.1.15 Speed change gear

The speed change gear is used if changing the input speed from the main shaft, auxiliary shaft, or composite auxiliary shaft gear during operation. If not used, set "0: No speed change gear" for [Pr.434] speed change gear 1 allocation (D15046+150n) and [Pr.490] speed change gear 2 allocation (D15052+150n).



Symbol	Setting item	Setting details	Setting value	Load cycle	Default	Device No.
Pr.434	Speed change gear 1	Sets the speed change gear 1 allocation.	0: No speed change gear 1: Main shaft side 2: Auxiliary shaft side 3: After composite auxiliary shaft gear	When starting synchro-nous control	0	D15046+150n
Pr.435	Speed change gear 1 smoothing time constant	Sets the speed change gear 1 smoothing time constant.	0 to 5000 [ms]		0 [ms]	D15047+150n
Pr.436	Speed change ratio 1 numerator	Sets the speed change ratio 1 numerator.	-2147483648 to 2147483647	Operation cycle	1	D15048+150n D15049+150n
Pr.437	Speed change ratio 1 denominator	Sets the speed change ratio 1 denominator.	1 to 2147483647		1	D15050+150n D15051+150n
Pr.490	Speed change gear 2	Sets the speed change gear 2 allocation.	0: No speed change gear 1: Main shaft side 2: Auxiliary shaft side 3: After composite auxiliary shaft gear	When starting synchro-nous control	0	D15052+150n
Pr.491	Speed change gear 2 smoothing time constant	Sets the speed change gear 2 smoothing time constant.	0 to 5000 [ms]		0 [ms]	D15053+150n
Pr.492	Speed change ratio 2 numerator	Sets the speed change ratio 2 numerator.	-2147483648 to 2147483647	Operation cycle	1	D15054+150n D15055+150n
Pr.493	Speed change ratio 2 denominator	Sets the speed change ratio 2 denominator.	1 to 2147483647		1	D15056+150n D15057+150n

10.1.16 Output axes

Output axes perform cam conversion processing based on the input travel value and set cam data, and outputs the feed current values that serve as commands to the servo amp.

Symbol	Setting item	Setting details	Setting value	Load cycle	Default	Device No.
Pr.438	Cam axis cycle unit setting	• Sets the cam axis 1 cycle length unit. • This is a parameter for monitor display, and does not affect control.	• Set in hexadecimal notation. H□□□□ Control unit 0: mm 1: inch 2: degree 3: PLS No. of decimal point digits 0 to 9 b0: Unit setting selection 0: Use main shaft main input axis unit. 1: Use this setting unit. b1 to 3: Not used	When starting Synchro-nous control	0000h	D15058+150n
Pr.439	Cam axis 1 cycle length	Sets the input amount required for 1 cam cycle.	1 to 2147483647 [Cam axis cycle unit]*1	When starting synchro-nous control, when passing cam data 0 point	4194304	D15060+150n D15061+150n
Pr.440	Cam No.	Sets the cam No.	0 : Linear cam (preset) 1 to 256 : User created cams		0	D15062+150n
Pr.441	Cam stroke amount	• Sets the cam stroke amount relative to a stroke ratio of 100% for stroke ratio data format cams. • Ignored for coordinate data format cams.	-2147483648 to 2147483647 [Output axis position unit]*2		4194304	D15064+150n D15065+150n
Pr.442	Cam axis 1 cycle length Change setting	Set if changing the [Pr.439] cam axis 1 cycle length (D15060+150n, D15061+150n) during synchronous control.	0: Disable 1: Enable	When starting Synchro-nous control	0	D15059+150n
Pr.444	Cam axis phase compensation advance time	Sets the time to advance or delay the cam axis phase.	-2147483648 to 2147483647 [μs]	Operation cycle	0 [μs]	D15066+150n D15067+150n
Pr.445	Cam axis phase compensation time constant	Sets the time to reflect cam axis phase compensation.	0 to 65535 [ms]	When starting Synchro-nous control	10 [ms]	D15068+150n
Pr.448	Synchronous control parameter block No.	Sets the synchronous control parameter block No.	1 to 64		1	D15069+150n
Pr.447	Output axis smoothing time constant	Set if performing smoothing processing for output axes.	0 to 5000 [ms]		0 [ms]	D15070+150n

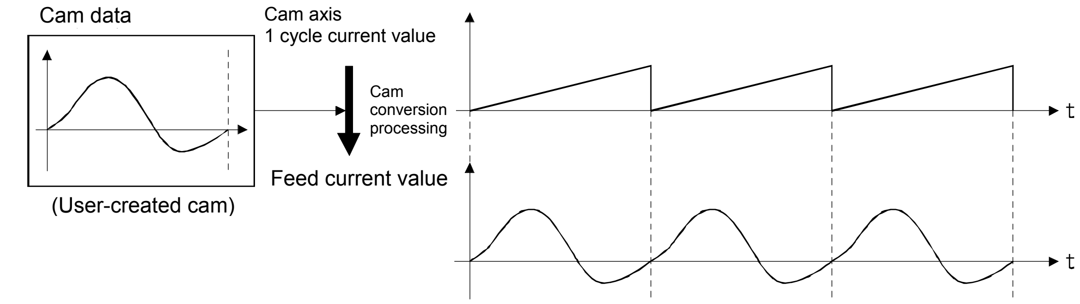
*1: Cam axis cycle unit
*2: Output axis position unit

[Cam data]

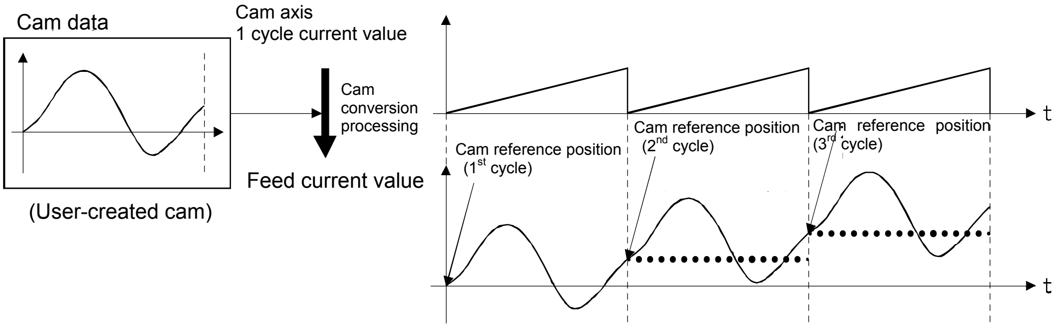
Synchronous control output axes are moved with cams. Output axis movement patterns (return movements, feed movements) relative to output axis module input travel values are registered in the cam data.

The movement patters are as follows.

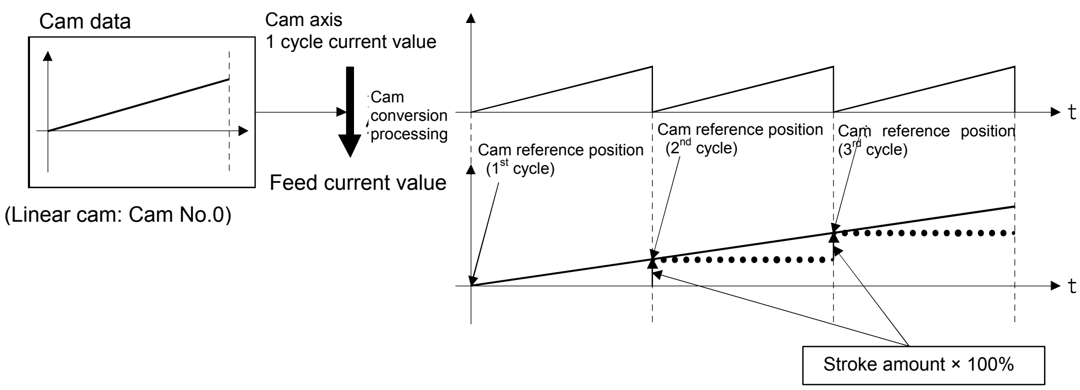
- Return movement: Return movement within fixed cam stroke range



- Feed movement: Movement that involves updating the cam reference position every 1 cycle

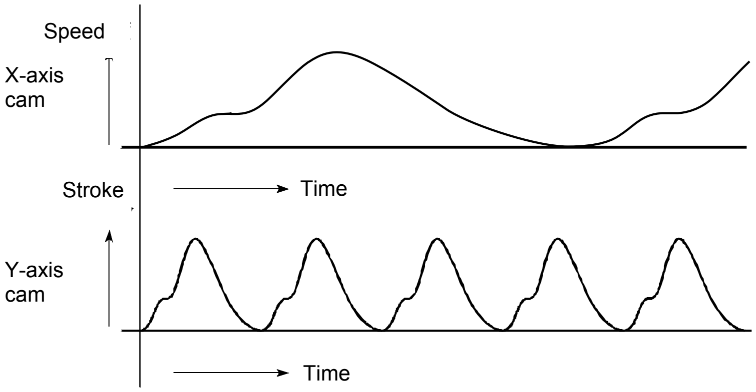


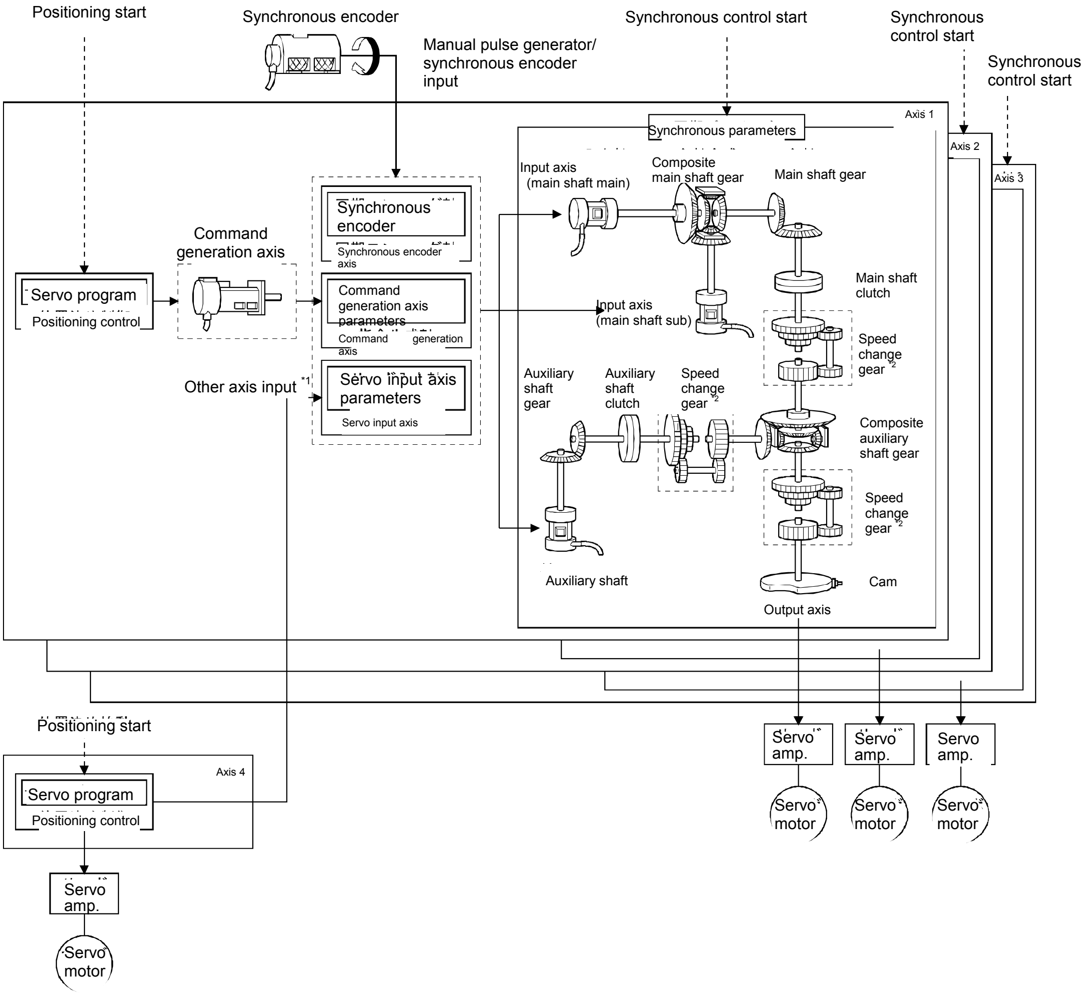
- Linear movement: Linear movement in which 1 cycle has a stroke ratio of 100% (Cam No. 0)



10.2 Practice Content

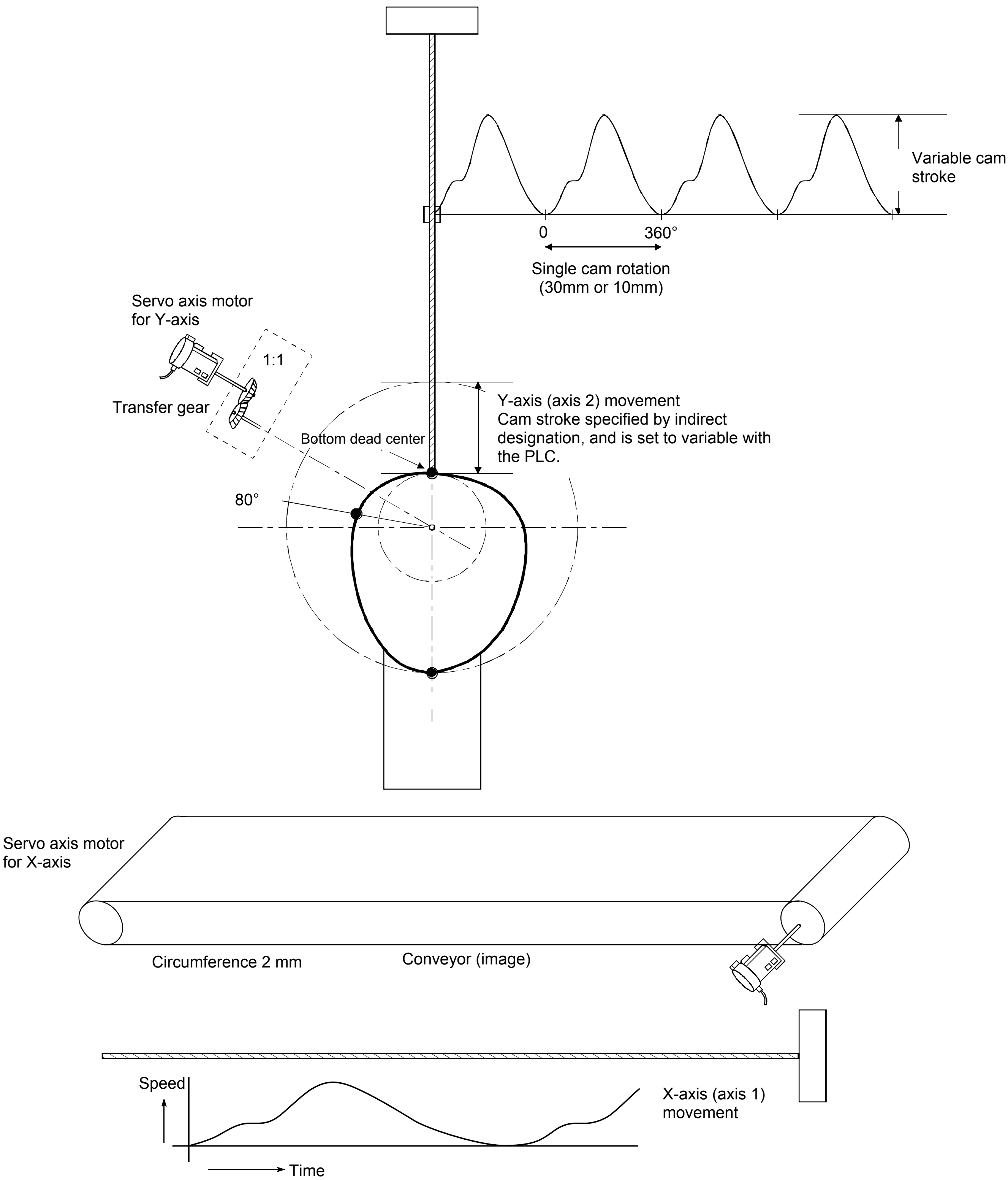
By setting "synchronous control parameters" and starting synchronous control for each output axis, control is synchronized for input axes (servo input axis, command generation axis, synchronous encoder axis).



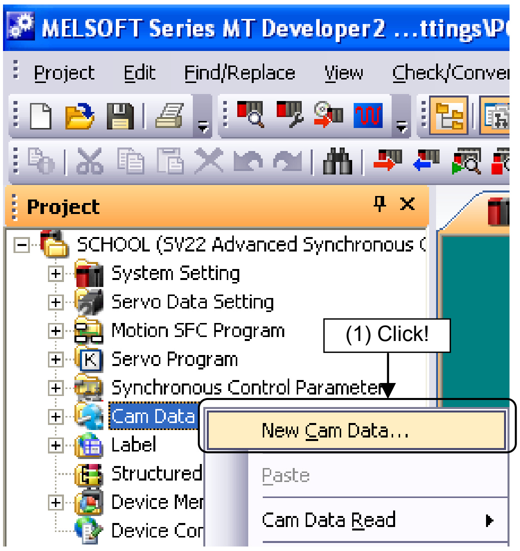


Locus movement concept

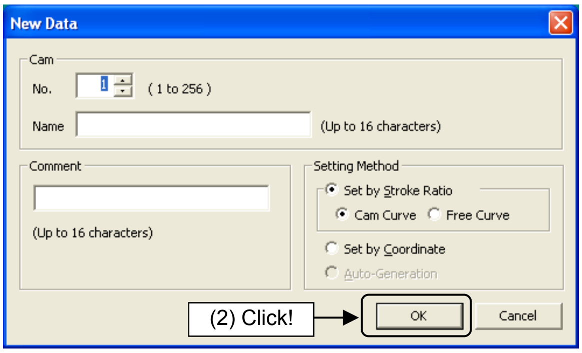
- The X-axis is set to 2 mm per rotation in the basic parameters, and is treated as the Y-axis main shaft.
- The Y-axis (axis 2) ballscrew moves 2 mm per rotation (4194304 pulses/rotation), and therefore the 1 cycle length in the output axis parameters is set to 4194304 pulses (actually 30 mm or 10 mm) in order to make it easier to confirm movements.



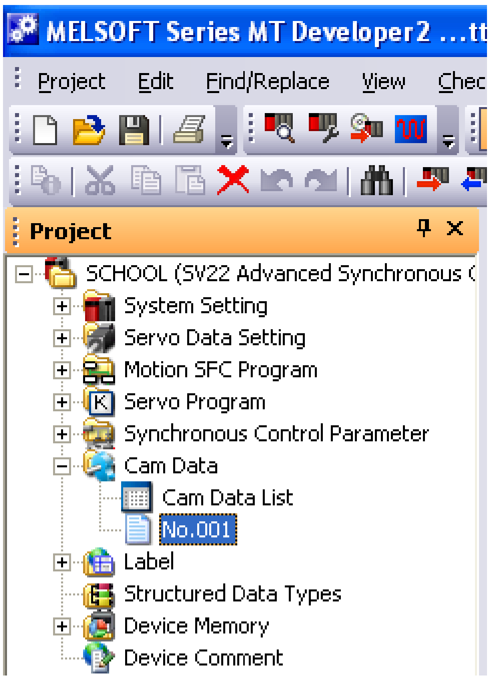
10.3 Cam Data Creation



(1) Right-click "Cam Data" in the Project window, and then click "New Cam Data…".



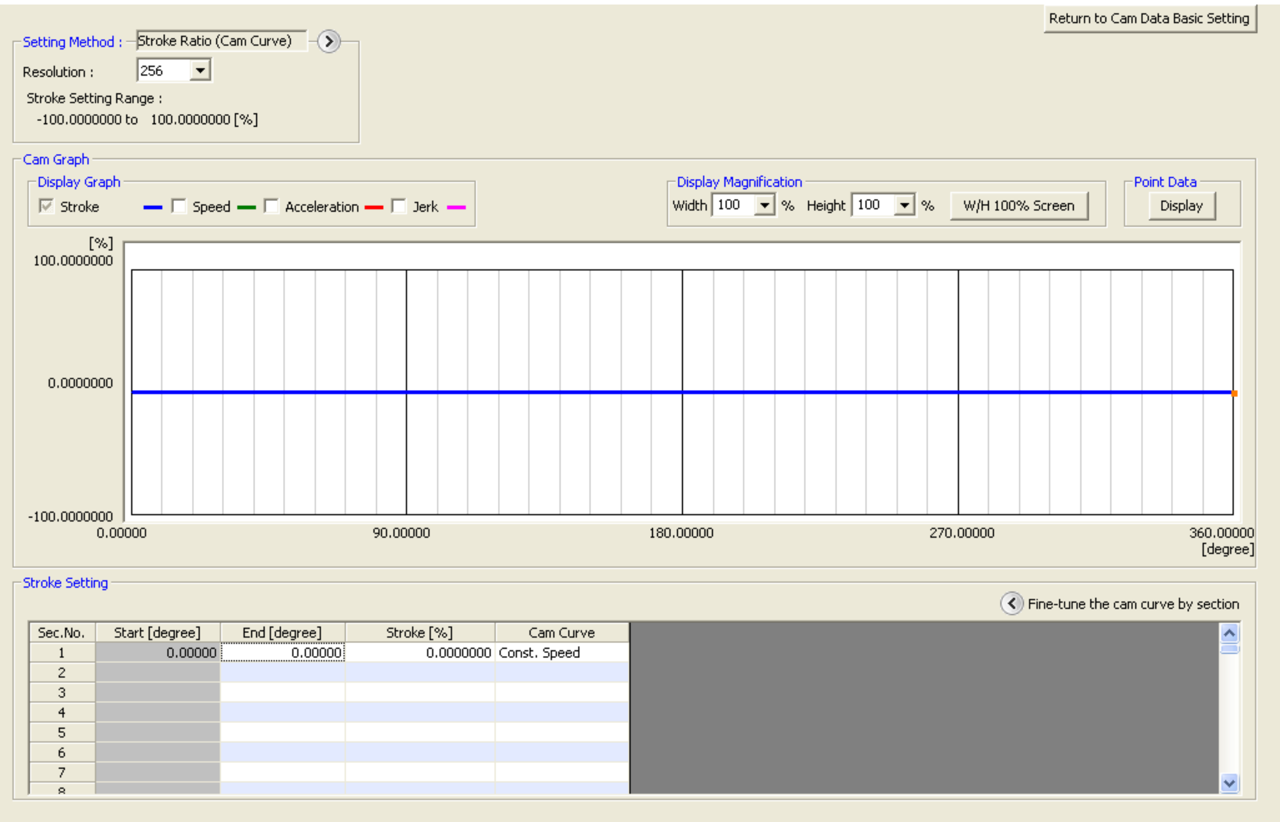
(2) A New Data screen appears. Set the Cam No., and then click the OK button.

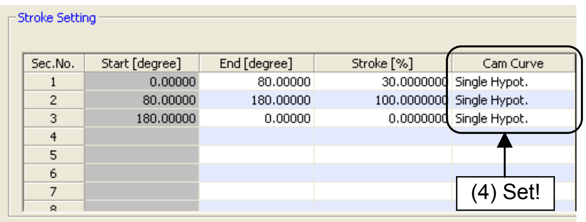


(3) Cam data is created, and a setting screen appears.

Go to next page

From previous page





(4) Specify the setting screen stroke settings as follows.

Div. No.	Start point	End point	Stroke
1	0.00000	80.00000	30.0000000
2		180.00000	100.0000000
3		0.00000	0.0000000

Stroke setting range
"Min. value": 0.00000, "Max. value": 100.0000000

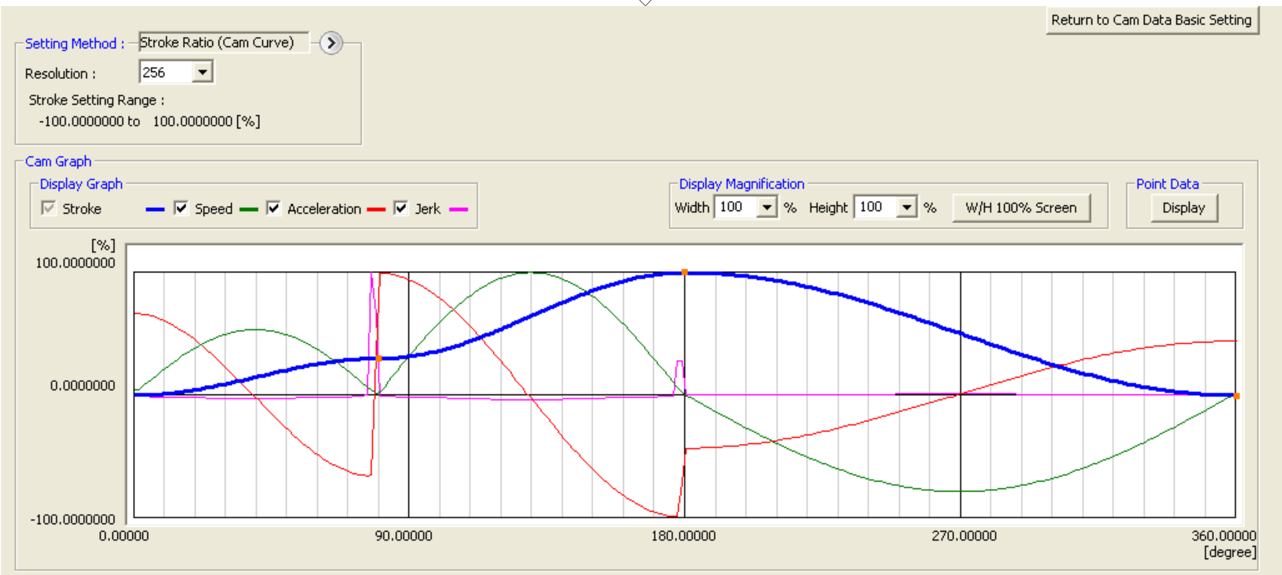
Set all strokes to "Single Hypot." at the "Cam Curve" selection.

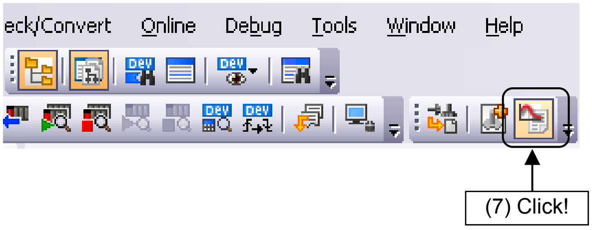
(5) Click the cam data conversion button.

(6) Change the "Display graph" check box selections to change the graph display in order to view the stroke, speed, acceleration, and jerk relative to the movement angle in a chart.

Go to next page

From previous page





(7) To view the stroke ratio, speed, acceleration, and jerk relative to the movement angle in numerical values, click the point data display tool button.

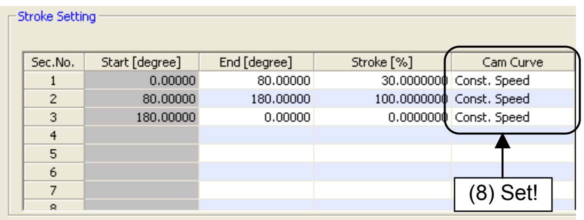
Point Data Display - Cam No.001

Table No.	Len. per Cycle[degree]	Stroke [%]	Speed	Accel.	Jerk	Cam Curve	Cam Data
1	1.40625	0.0228663	0.07	3.29	-0.1	Single Hypot.	228662
2	2.81250	0.0913954	0.12	3.26	-0.1	Single Hypot.	913954
3	4.21875	0.2053785	0.16	3.23	-0.1	Single Hypot.	2053785
4	5.62500	0.3644680	0.21	3.18	-0.2	Single Hypot.	3644680
5	7.03125	0.5681789	0.25	3.13	-0.2	Single Hypot.	5681789
6	8.43750	0.8158901	0.30	3.06	-0.2	Single Hypot.	8158901
7	9.84375	1.1068464	0.34	2.99	-0.3	Single Hypot.	11068463
8	11.25000	1.4401606	0.38	2.91	-0.3	Single Hypot.	14401606
9	12.65625	1.8148166	0.42	2.82	-0.3	Single Hypot.	18148166
10	14.06250	2.2296721	0.46	2.72	-0.3	Single Hypot.	22296721
11	15.46875	2.6834623	0.50	2.61	-0.4	Single Hypot.	26834622
12	16.87500	3.1748036	0.54	2.49	-0.4	Single Hypot.	31748035
13	18.28125	3.7021980	0.57	2.37	-0.4	Single Hypot.	37021980
14	19.68750	4.2640376	0.60	2.24	-0.4	Single Hypot.	42640376
15	21.09375	4.8586094	0.64	2.10	-0.5	Single Hypot.	48586094
16	22.50000	5.4841007	0.66	1.95	-0.5	Single Hypot.	54841007
17	23.90625	6.1386045	0.69	1.80	-0.5	Single Hypot.	61386044
18	25.31250	6.8201252	0.72	1.65	-0.5	Single Hypot.	68201251

Close

After checking, click the Close button.

There are tables from No. 1 to 256. Scroll to view all tables.



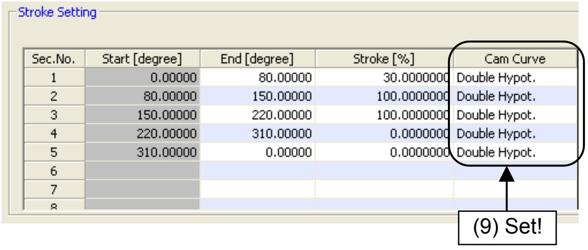
Stroke Setting

Sec.No.	Start [degree]	End [degree]	Stroke [%]	Cam Curve
1	0.00000	80.00000	30.0000000	Const. Speed
2	80.00000	180.00000	100.0000000	Const. Speed
3	180.00000	0.00000	0.0000000	Const. Speed
4				
5				
6				
7				

(8) Create cam data for cam No. 002 using the same procedure as that for cam No. 001.
For cam No. 002, set all strokes to "Const. Speed" at the "Cam Curve" selection.
(All other selections are the same as those for cam No. 001.)

Go to next page

From previous page

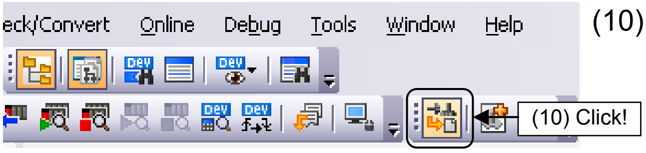


(9) Use the same procedure to create data for cam No. 003.
Specify the stroke settings as follows.

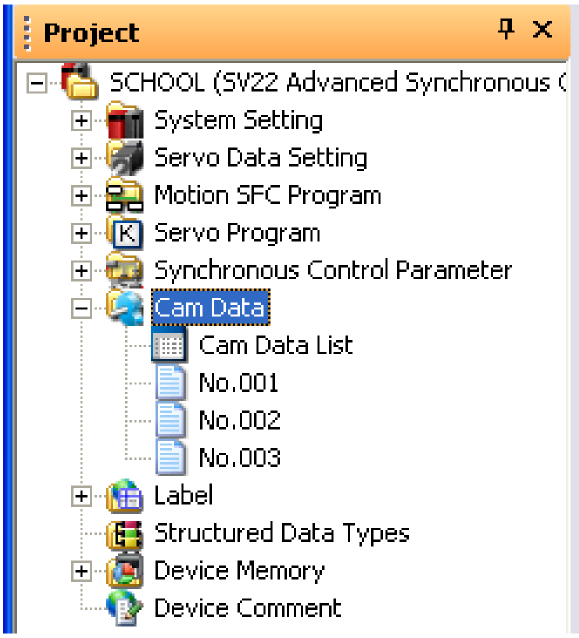
Div. No.	Start point	End point	Stroke
1	0.00000	80.00000	30.0000000
2		150.00000	100.0000000
3		220.00000	100.0000000
4		310.00000	0.0000000
5		0.00000	0.0000000

Stroke setting range
"Min. value": 0.00000, "Max. value": 100.0000000

Set all strokes to "Double Hypot." at the "Cam Curve" selection.



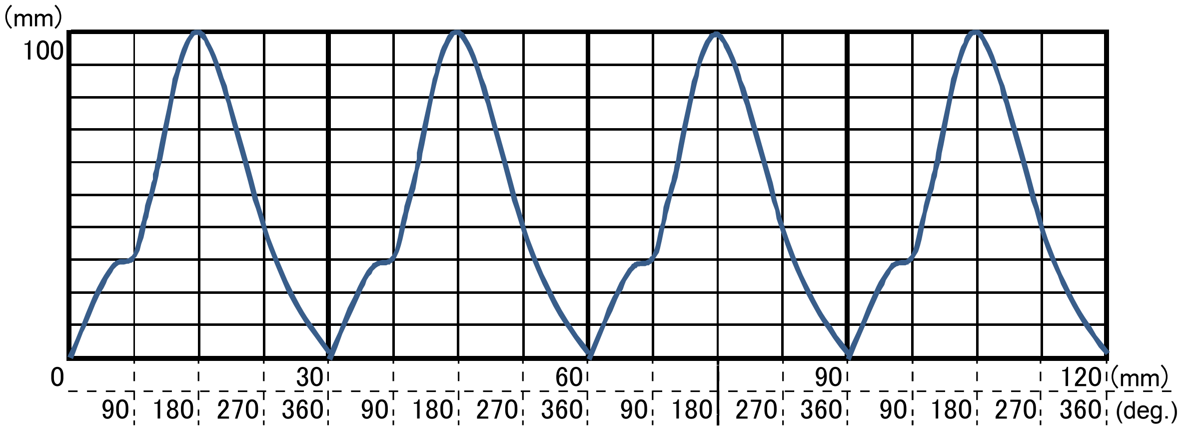
(10) Click the cam data conversion button.



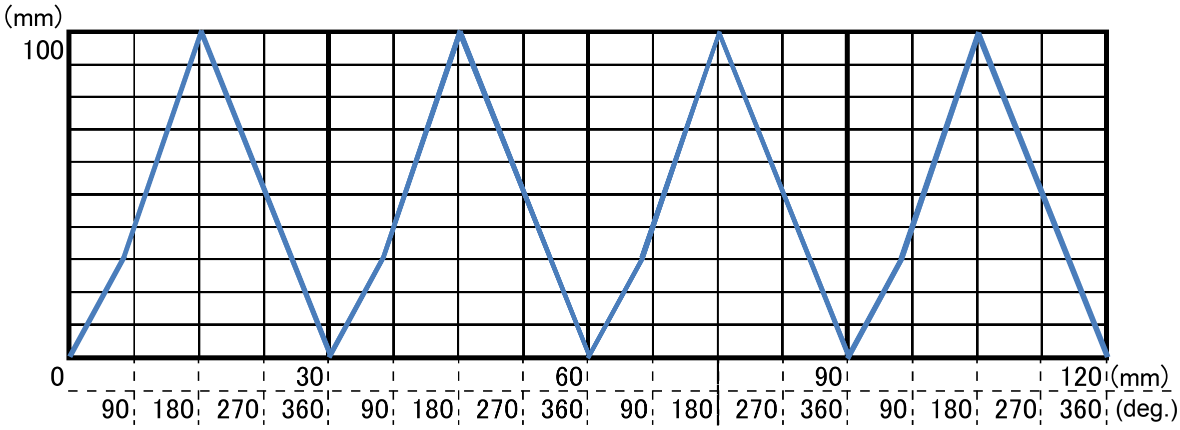
(11) Cam data creation is now complete.

[Locus of each created cam]

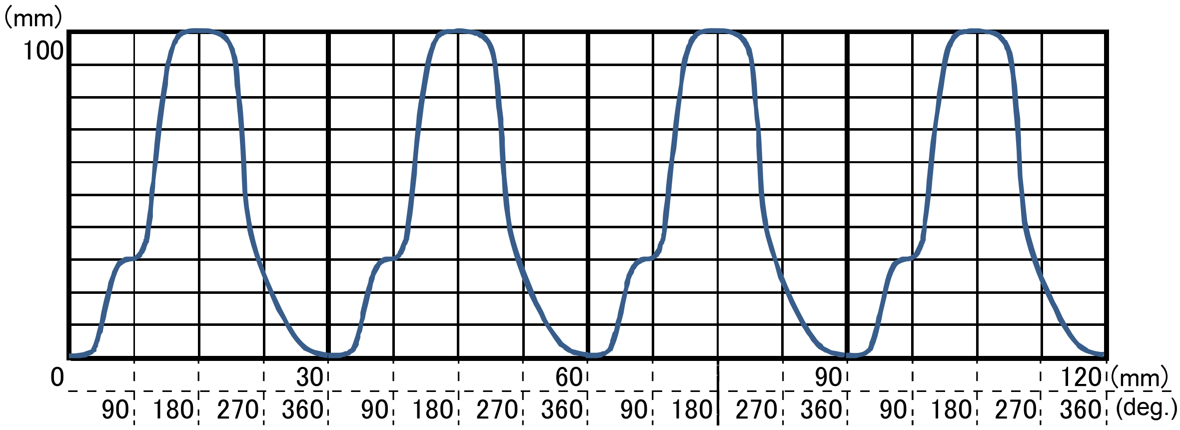
<Cam No.001 waveform>



<Cam No.002 waveform>



<Cam No.003 waveform>



10.4 Advanced Synchronous Control Programs

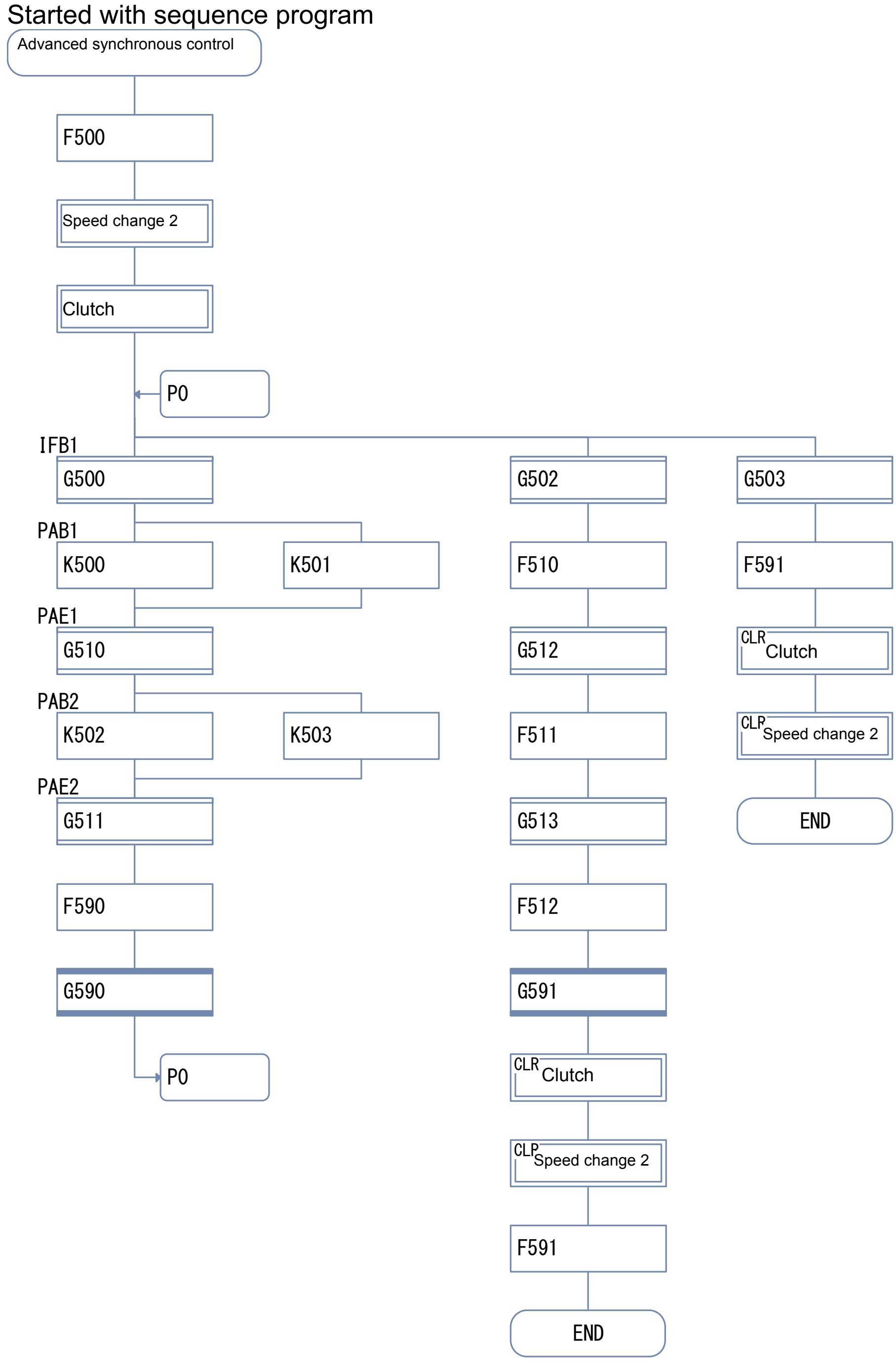
Motion SFC programs used with advanced synchronous control are shown in the following table.

No.	Program name	Automatic start	END operation	No. of transitions	Execution timing
100	Advanced synchronous control	No			Normal
210	Speed change 2	No			Normal
230	Analog speed change	No			Normal

• Start program from sequence program

[Advanced synchronous control] program No.100

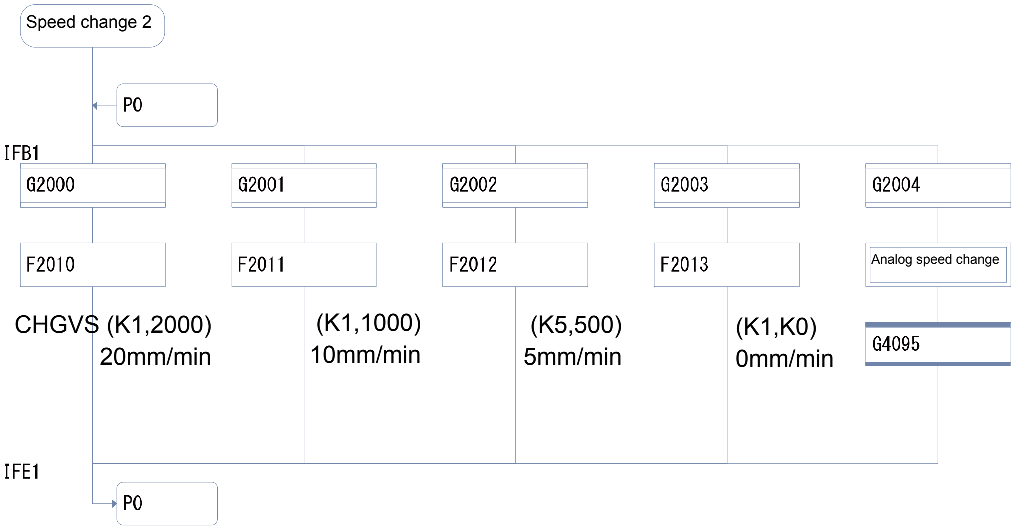
Started with sequence program



• Start program from motion SFC program

[Speed change 2] program No.210 (Program used to change the speed of command generation axes)

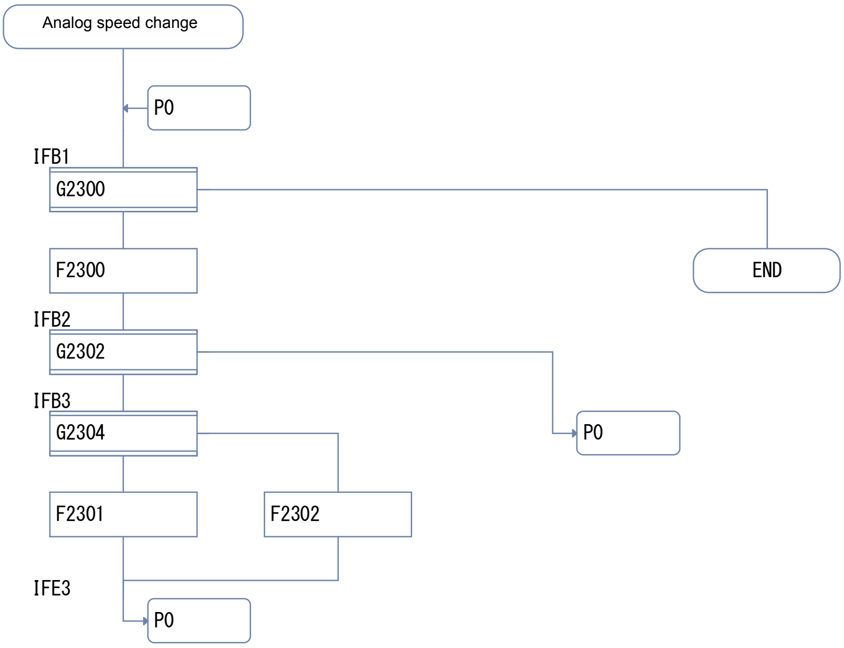
Started with No.100



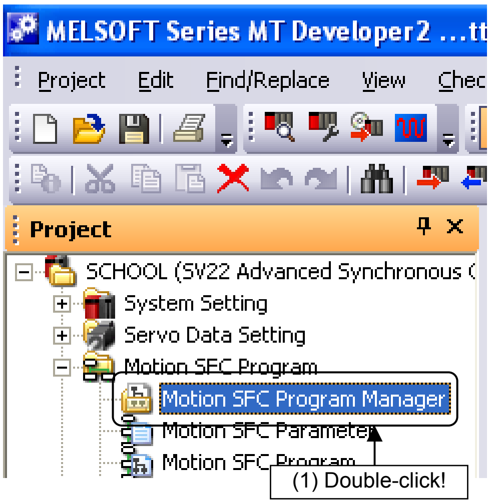
• Start program from motion SFC program

[Analog speed change] program No.230 (Program used to change the speed of command generation axes)

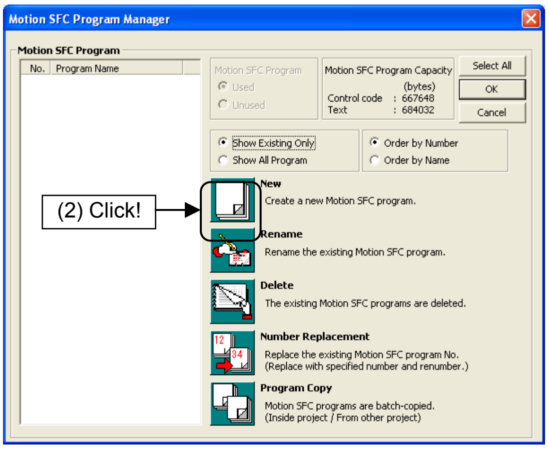
Started with No.100



10.4.1 Creating new advanced synchronous control motion SFC programs

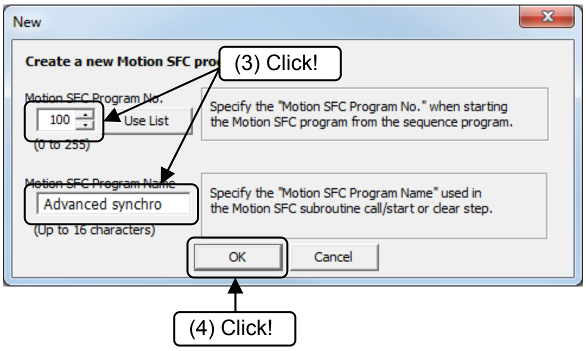


(1) Double-click [Motion SFC Program] → [Motion SFC Program Manager] in the Project window.



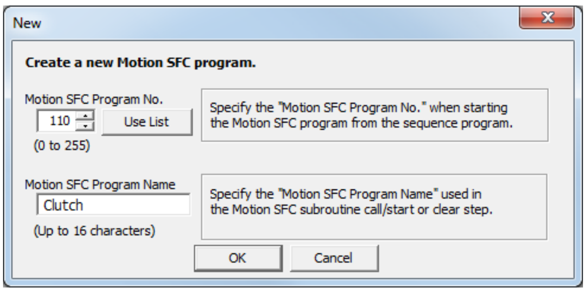
(2) A Motion SFC Program Manager dialog box appears.

Click the New button.



(3) A New dialog box appears. Set the program No. for the motion SFC program being created. Enter "100" for the "Motion SFC program No.", and "Advanced synchronous control" for the "Motion SFC program name".

(4) Click the OK button after entering.

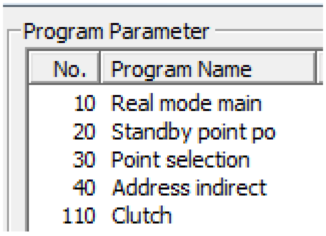


(5) Now create the following new motion SFC program.

"Motion SFC program No.": 110
"Motion SFC program name": Clutch

Go to next page

From previous page



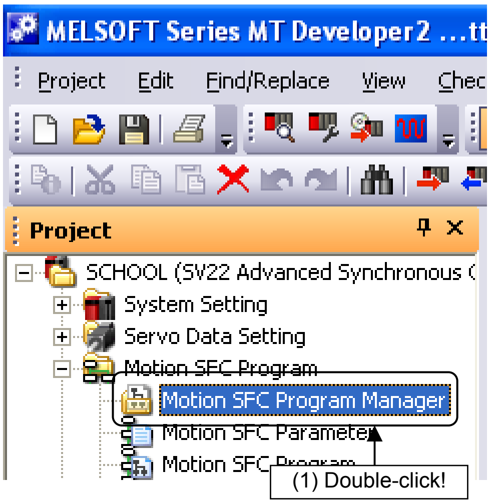
(6) The set motion SFC program appears in a list. Select motion SFC program No.100, and then click the OK button.

No.	Program name
100	Advanced synchronous control
110	Clutch

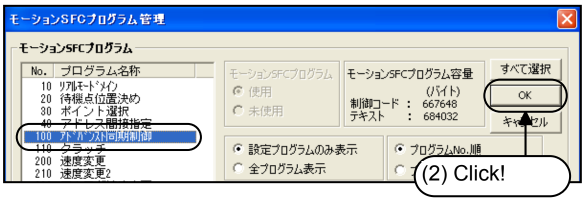
(The creation procedure for the motion SFC program created here will not be described in detail. Refer to the section on motion SFC programs for operation described later to create.)

10.4.2 Entering motion control steps for advanced synchronous control

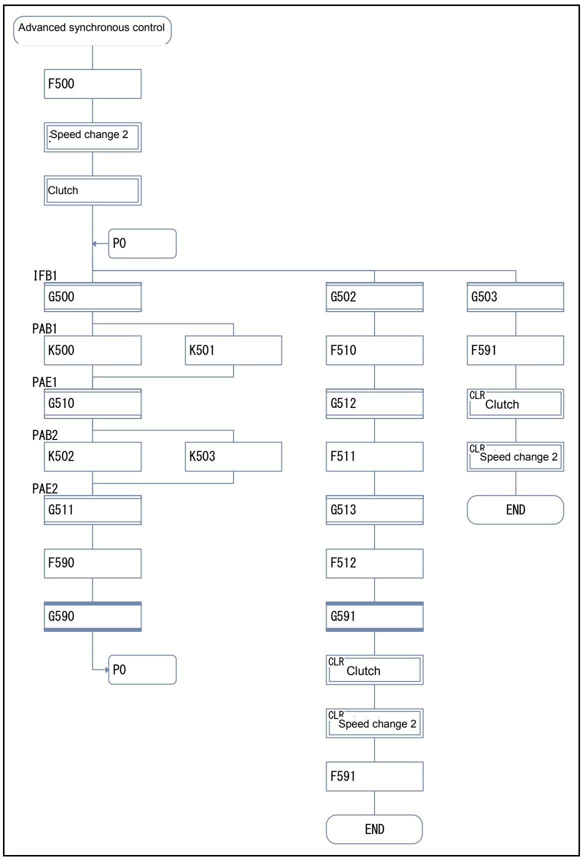
Sets motion control steps for advanced synchronous control.



(1) Create a servo program motion SFC program.
Double-click [Motion SFC Program] → [Motion SFC Program Manager] in the Project window.



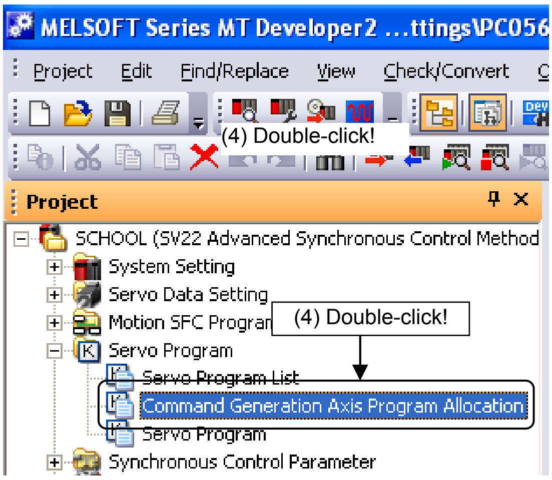
(2) Select "100 Advanced synchronous control" from the motion SFC program list in the Motion SFC Program Management dialog box, and then click the OK button.



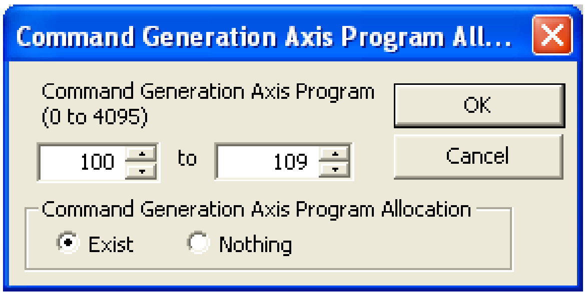
(3) Create the SFC diagram shown on the left.

Go to next page

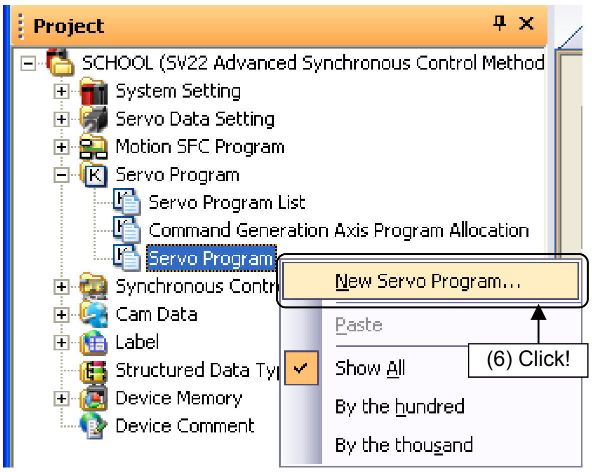
From previous page



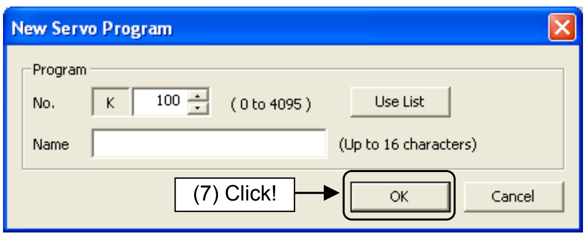
(4) Double-click [Command Generation Axis Program Allocation] in the Project window.



(5) Set "Command Generation Axis Program Allocation" to "Exist" and set the "Command Generation Axis Program" to "100" to "109" at the Command Generation Axis Program Allocation Setting dialog box, and then click the OK button.



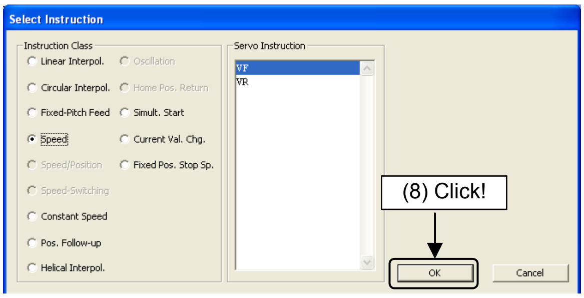
(6) Right-click "Servo program" in the Project window, and then click "Create New Servo Program".



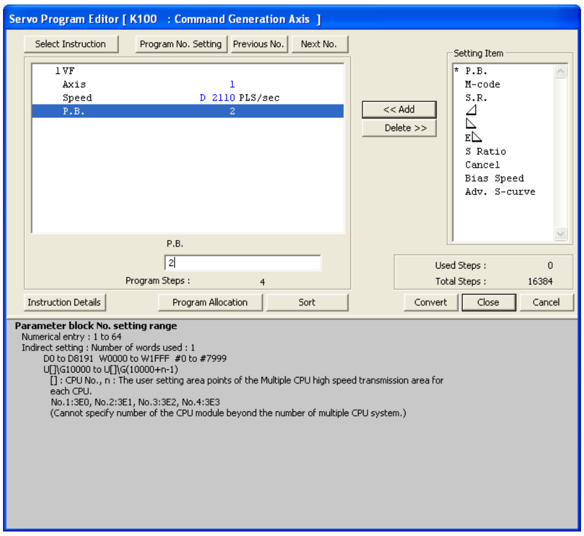
(7) Enter the Program No. at the New Servo Program dialog box, and then click the OK button.

Go to next page

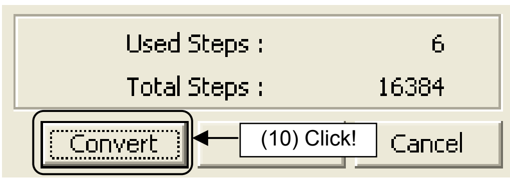
From previous page



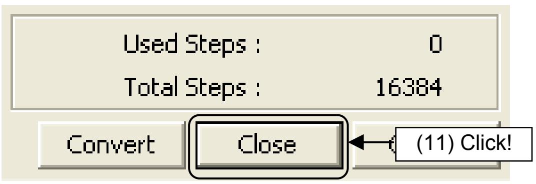
(8) Select "Speed" for the "Instruction Class", and "VF" for the "Servo Instruction" at the Select Instruction dialog box, and then click the OK button.



(9) Enter "1" in the "Axis" text box, and enter "D 2110" in the "Speed" text box.
Add "P.B." from the setting items, and then enter "2".



(10) Click the Convert button.

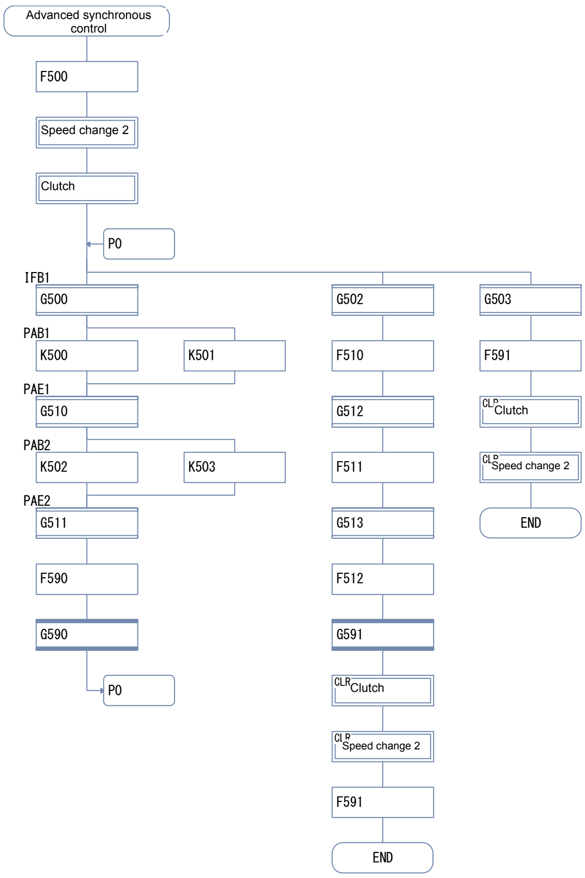


(11) After creating motion control steps, click the Close button to close the Servo Program Editor dialog box.

Go to next page

From previous page

100: Advanced synchronous control



Go to next page

(12) Set the following transition programs.

[G500]	M30*M2402*M2415
[G502]	M31*M2402*M2415
[G510]	M2410*M2430
[G511]	!M30
[G512]	//Axis 1&2_Performing synchronous control M10880*M10881
[G513]	!M31
[G514]	//Axis 1&2_Performing synchronous control !M10880*!M10881
[G590]	!M2001
[F500]	SET M6810 SET M2042

[F510]	//Axis 1_Synchronous control execution
	SET M12000
	//Axis 2_Synchronous control execution
	SET M12001

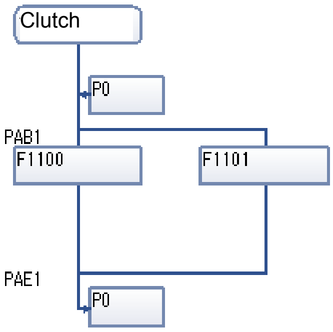
[F511]	//Command generation axis 1_JOG speed
	//D14600L=1000
	//Servo input
	D14680L=1000
	//Command generation axis 1_Forward rotation JOG ON
	SET M10962

[F512]	//Command generation axis 1_Forward rotation JOG ON
	RST M10962
	//Axis 1_Synchronous control stop
	RST M12000
	//Axis 2_Synchronous control stop
	RST M12001

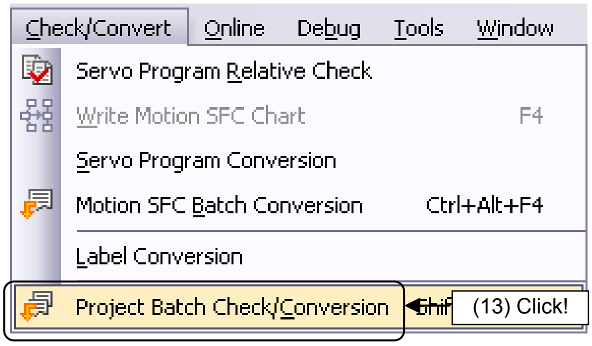
[F590]	RST M12000
[F591]	RST M6810

From previous page

110: Clutch



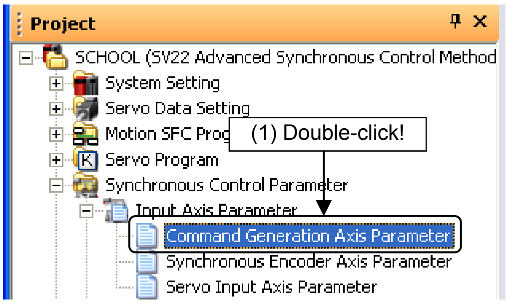
[F1100]	//Clutch 1 control OUT M11680 = !M32
[F1101]	//Clutch 2 control OUT M11690 = !M33



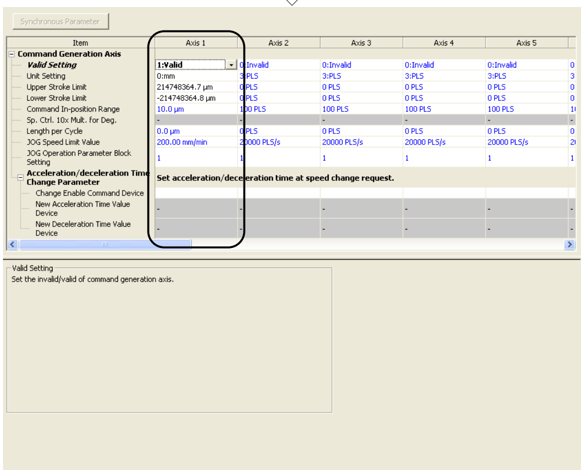
(13) Batch convert created SFC diagrams to motion SFC programs.
Click [Project Batch Check/Conversion] on the [Check/Convert] menu.

(14) Editing of servo program No.100 for advanced synchronous control is now complete.

10.5 Editing Command Generation Axis Parameters



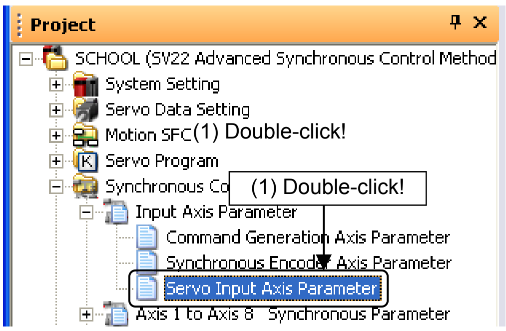
(1) Select [Synchronous Control Parameter] → [Input Axis Parameter] in the Project window, and then double-click [Command Generation Axis parameter].



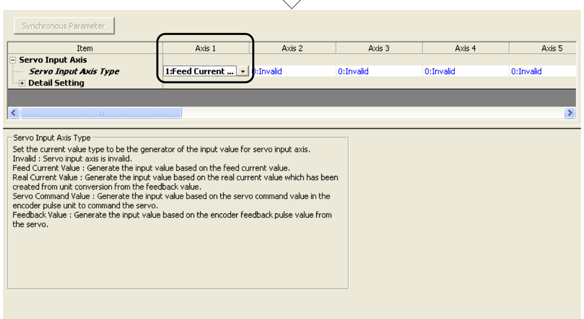
(2) A Command Generation Axis Parameter dialog box appears. Specify the following settings for axis 1 only.

Valid Setting	1: Valid
Unit Setting	0: mm
Upper Stroke Limit	214748364.7 (μm)
Lower Stroke Limit	-214748364.8 (μm)

10.6 Editing Servo Input Axis Parameters



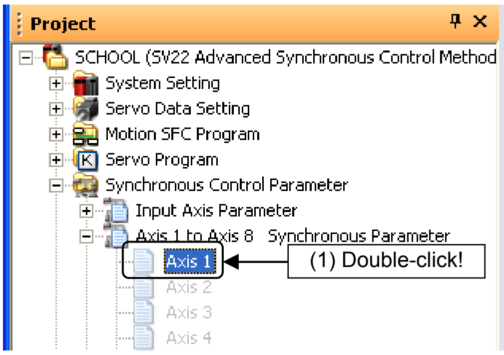
(1) Select [Synchronous Control Parameter] → [Input Axis Parameter] in the Project window, and then double-click [Servo Input Axis Parameter].



(2) A Command Generation Axis Parameter dialog box appears. Specify the following settings for axis 1 only.

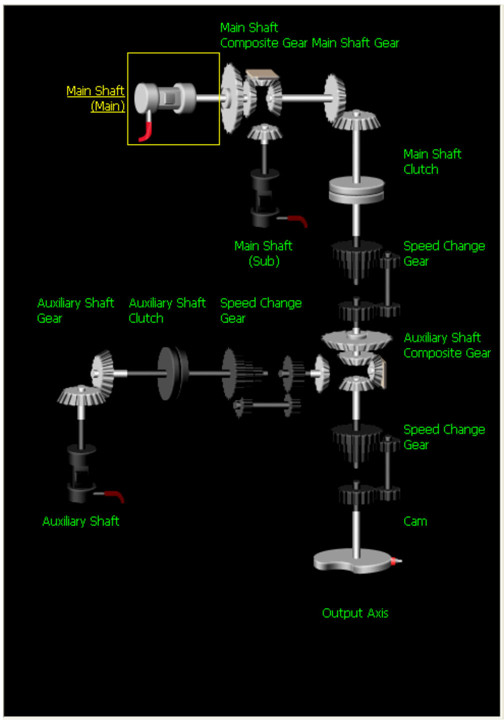
Servo Input Axis Type	1: Feed Current Value

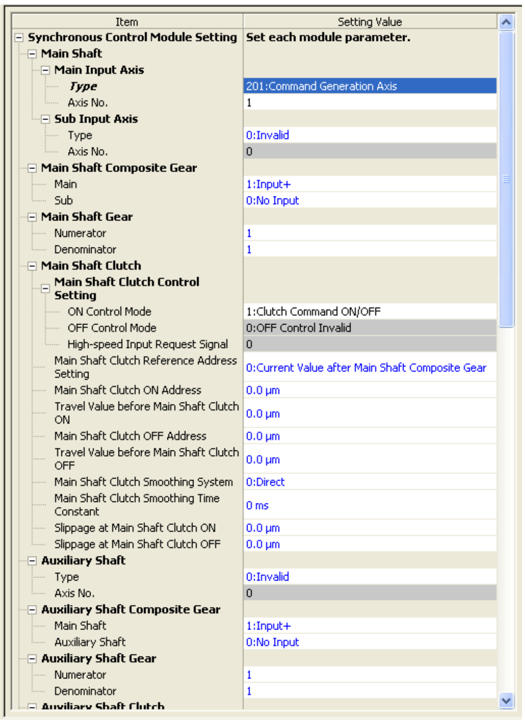
10.7 Editing Synchronous Control Parameters



(1) Select [Synchronous Control Parameter] → [Axis 1 to Axis 8 Synchronous Parameter] in the Project window, and then double-click [Axis 1].

(2) An Axis 1 Synchronous Parameter dialog box appears. Set the "Main Input Axis" "Type" and "Axis No". as follows.



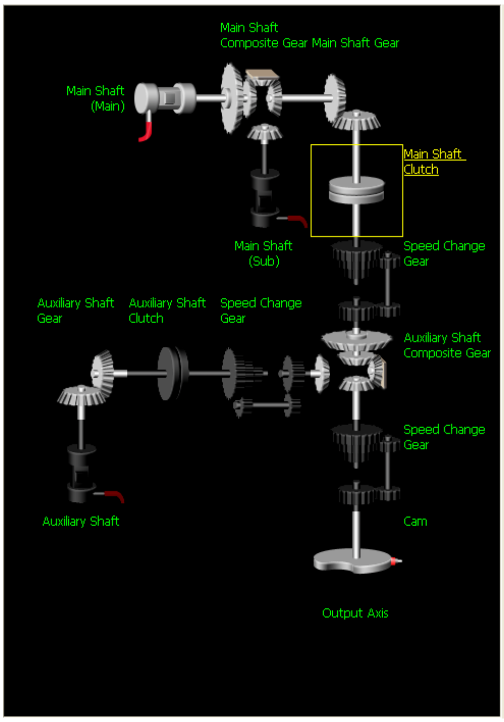


Item	Setting Value
Synchronous Control Module Setting	Set each module parameter.
Main Shaft	
Main Input Axis	
Type	201:Command Generation Axis
Axis No.	1
Sub Input Axis	
Type	0:Invalid
Axis No.	0
Main Shaft Composite Gear	
Main	1:Input+
Sub	0:No Input
Main Shaft Gear	
Numerator	1
Denominator	1
Main Shaft Clutch	
Main Shaft Clutch Control Setting	
ON Control Mode	1:Clutch Command ON/OFF
OFF Control Mode	0:OFF Control Invalid
High-speed Input Request Signal	0
Main Shaft Clutch Reference Address Setting	0:Current Value after Main Shaft Composite Gear
Main Shaft Clutch ON Address	0.0 μm
Travel Value before Main Shaft Clutch ON	0.0 μm
Main Shaft Clutch OFF Address	0.0 μm
Travel Value before Main Shaft Clutch OFF	0.0 μm
Main Shaft Clutch Smoothing System	0:Direct
Main Shaft Clutch Smoothing Time Constant	0 ms
Slippage at Main Shaft Clutch ON	0.0 μm
Slippage at Main Shaft Clutch OFF	0.0 μm
Auxiliary Shaft	
Type	0:Invalid
Axis No.	0
Auxiliary Shaft Composite Gear	
Main Shaft	1:Input+
Auxiliary Shaft	0:No Input
Auxiliary Shaft Gear	
Numerator	1
Denominator	1
Auxiliary Shaft Clutch	

Go to next page

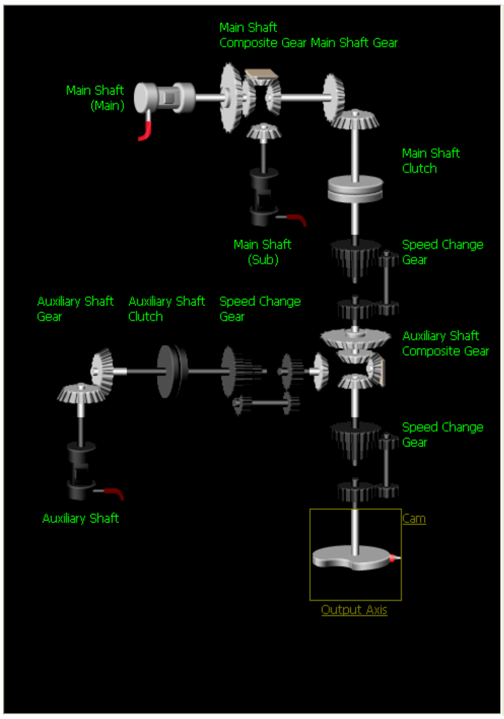
From previous page

(3) Set the "Main Shaft Clutch Control Setting" "ON Control Mode" as follows.



Item	Setting Value
Synchronous Control Module Setting	Set each module parameter.
Main Shaft	
Main Input Axis	
Type	201:Command Generation Axis
Axis No.	1
Sub Input Axis	
Type	0:Invalid
Axis No.	0
Main Shaft Composite Gear	
Main	1:Input+
Sub	0:No Input
Main Shaft Gear	
Numerator	1
Denominator	1
Main Shaft Clutch	
Main Shaft Clutch Control Setting	
ON Control Mode	1:Clutch Command ON/OFF
OFF Control Mode	0:OFF Control Invalid
High-speed Input Request Signal	0
Main Shaft Clutch Reference Address Setting	0:Current Value after Main Shaft Composite Gear
Main Shaft Clutch ON Address	0.0 μm
Travel Value before Main Shaft Clutch ON	0.0 μm
Main Shaft Clutch OFF Address	0.0 μm
Travel Value before Main Shaft Clutch OFF	0.0 μm
Main Shaft Clutch Smoothing System	0:Direct
Main Shaft Clutch Smoothing Time Constant	0 ms
Slippage at Main Shaft Clutch ON	0.0 μm
Slippage at Main Shaft Clutch OFF	0.0 μm
Auxiliary Shaft	
Type	0:Invalid
Axis No.	0
Auxiliary Shaft Composite Gear	
Main Shaft	1:Input+
Auxiliary Shaft	0:No Input
Auxiliary Shaft Gear	
Numerator	1
Denominator	1
Auxiliary Shaft Clutch	

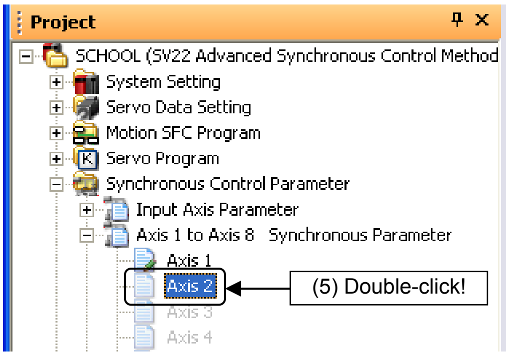
(4) Set the "Cam Axis Length per Cycle", "Cam Stroke Amount", and "Cam No." as follows. Setting of axis 1 synchronous parameter is now complete.



Item	Setting Value
Auxiliary Shaft Clutch OFF Address	0 PLS
Travel Value before Auxiliary Shaft Clutch OFF	0 PLS
Auxiliary Shaft Clutch Smoothing System	0:Direct
Auxiliary Shaft Clutch Smoothing Time Constant	0 ms
Slippage at Auxiliary Shaft Clutch ON	0 PLS
Slippage at Auxiliary Shaft Clutch OFF	0 PLS
Speed Change Gear 1	
Speed Change Gear Arrangement	0:No Speed Change Gear
Speed Change Gear Smoothing Time Constant	0 ms
Speed Change Ratio	
Numerator	1
Denominator	1
Speed Change Gear 2	
Speed Change Gear Arrangement	0:No Speed Change Gear
Speed Change Gear Smoothing Time Constant	0 ms
Speed Change Ratio	
Numerator	1
Denominator	1
Output Axis	
Cam Axis Cycle Unit	
Unit Setting Selection	0:Use Units of Main Input Axis
Unit	0:mm
Number of Decimal Places	0
Cam Axis Length per Cycle Change Setting	0:Invalid
Cam Axis Length per Cycle	10000.0 μm
Cam Stroke Amount	10000.0 μm
Cam No.	1
Cam Axis Phase Correction Advance Time	0 μs
Cam Axis Phase Correction Time Constant	10 ms
Synchronous Control Parameter Block No.	1
Output Axis Smoothing Time Constant	0 ms
Synchronous Control Initial Position Parameter	Set the parameter for the initial alignment when starting the synchronous control.

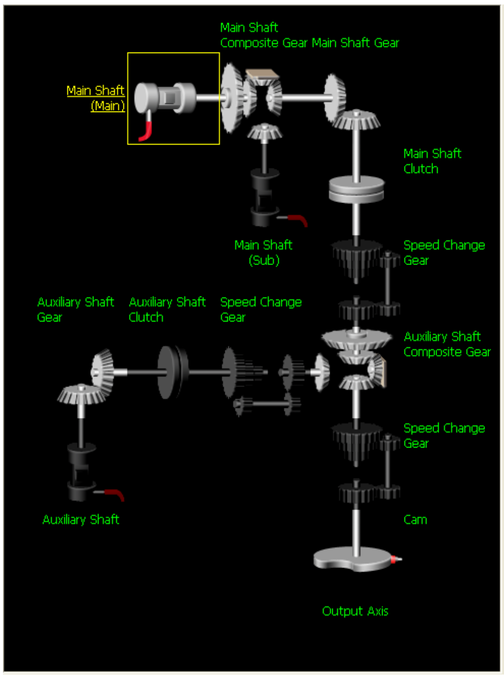
Go to next page

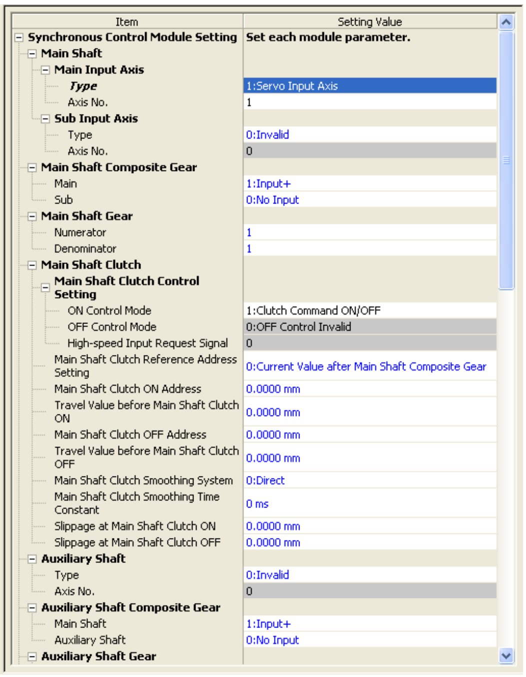
From previous page



(5) Select [Synchronous Control Parameter] → [Axis 1 to Axis 8 Synchronous Parameter] in the Project window, and then double-click [Axis 2].

(6) An Axis 2 Synchronous Parameter dialog box appears. Set the "Main Input Axis" "Type" and "Axis No". as follows.



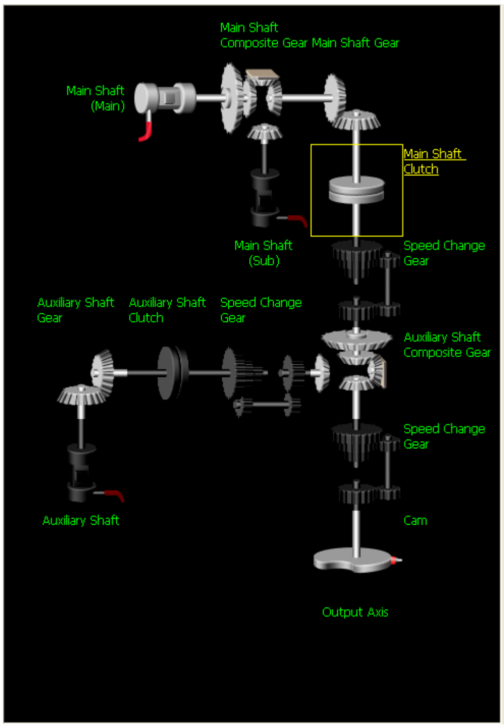


Item	Setting Value
Synchronous Control Module Setting	Set each module parameter.
Main Shaft	
Main Input Axis	
Type	1:Servo Input Axis
Axis No.	1
Sub Input Axis	
Type	0:Invalid
Axis No.	0
Main Shaft Composite Gear	
Main	1:Input+
Sub	0:No Input
Main Shaft Gear	
Numerator	1
Denominator	1
Main Shaft Clutch	
Main Shaft Clutch Control Setting	
ON Control Mode	1:Clutch Command ON/OFF
OFF Control Mode	0:OFF Control Invalid
High-speed Input Request Signal	0
Main Shaft Clutch Reference Address Setting	0:Current Value after Main Shaft Composite Gear
Main Shaft Clutch ON Address	0.0000 mm
Travel Value before Main Shaft Clutch ON	0.0000 mm
Main Shaft Clutch OFF Address	0.0000 mm
Travel Value before Main Shaft Clutch OFF	0.0000 mm
Main Shaft Clutch Smoothing System	0:Direct
Main Shaft Clutch Smoothing Time Constant	0 ms
Slippage at Main Shaft Clutch ON	0.0000 mm
Slippage at Main Shaft Clutch OFF	0.0000 mm
Auxiliary Shaft	
Type	0:Invalid
Axis No.	0
Auxiliary Shaft Composite Gear	
Main Shaft	1:Input+
Auxiliary Shaft	0:No Input
Auxiliary Shaft Gear	

Go to next page

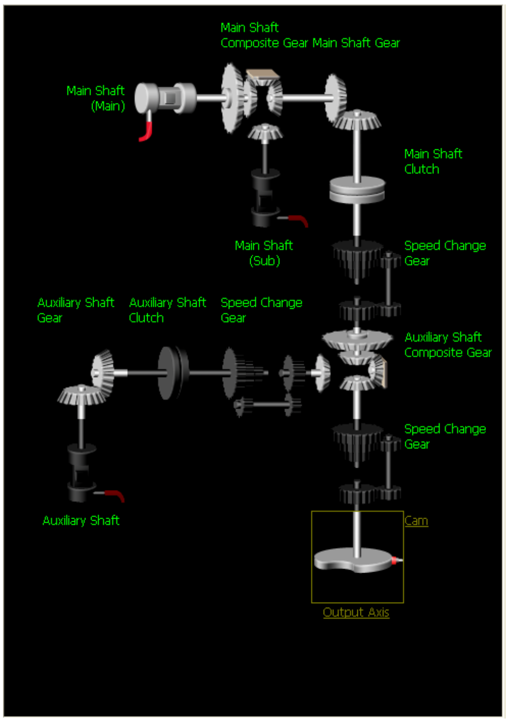
From previous page

(7) Set the "Main Shaft Clutch Control Setting" "ON Control Mode" as follows.



Item	Setting Value
Synchronous Control Module Setting	**Set each module parameter.**
Main Shaft	
Main Input Axis	
Type	1:Servo Input Axis
Axis No.	1
Sub Input Axis	
Type	0:Invalid
Axis No.	0
Main Shaft Composite Gear	
Main	1:Input+
Sub	0:No Input
Main Shaft Gear	
Numerator	1
Denominator	1
Main Shaft Clutch	
Main Shaft Clutch Control Setting	
ON Control Mode	1:Clutch Command ON/OFF
OFF Control Mode	0:OFF Control Invalid
High-speed Input Request Signal	0
Main Shaft Clutch Reference Address Setting	0:Current Value after Main Shaft Composite Gear
Main Shaft Clutch ON Address	0.0000 mm
Travel Value before Main Shaft Clutch ON	0.0000 mm
Main Shaft Clutch OFF Address	0.0000 mm
Travel Value before Main Shaft Clutch OFF	0.0000 mm
Main Shaft Clutch Smoothing System	0:Direct
Main Shaft Clutch Smoothing Time Constant	0 ms
Slippage at Main Shaft Clutch ON	0.0000 mm
Slippage at Main Shaft Clutch OFF	0.0000 mm
Auxiliary Shaft	
Type	0:Invalid
Axis No.	0
Auxiliary Shaft Composite Gear	
Main Shaft	1:Input+
Auxiliary Shaft	0:No Input
Auxiliary Shaft Gear	
Numerator	1
Denominator	1
Auxiliary Shaft Clutch	

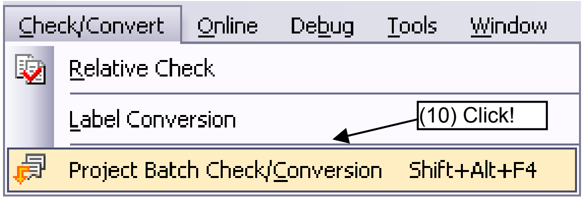
(8) Set the "Cam Axis Length per Cycle", "Cam Stroke Amount", and "Cam No." as follows. Setting of axis 2 synchronous parameters is now complete.



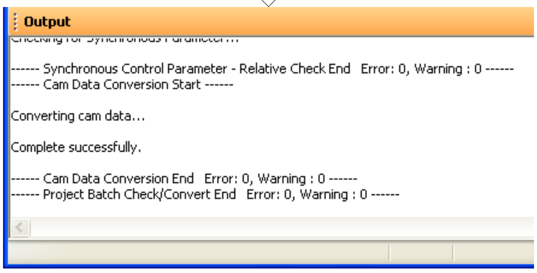
Item	Setting Value
Auxiliary Shaft Clutch OFF Address	0 PLS
Travel Value before Auxiliary Shaft Clutch OFF	0 PLS
Auxiliary Shaft Clutch Smoothing System	0:Direct
Auxiliary Shaft Clutch Smoothing Time Constant	0 ms
Slippage at Auxiliary Shaft Clutch ON	0 PLS
Slippage at Auxiliary Shaft Clutch OFF	0 PLS
Speed Change Gear 1	
Speed Change Gear Arrangement	0:No Speed Change Gear
Speed Change Gear Smoothing Time Constant	0 ms
Speed Change Ratio	
Numerator	1
Denominator	1
Speed Change Gear 2	
Speed Change Gear Arrangement	0:No Speed Change Gear
Speed Change Gear Smoothing Time Constant	0 ms
Speed Change Ratio	
Numerator	1
Denominator	1
Output Axis	
Cam Axis Cycle Unit	
Unit Setting Selection	0:Use Units of Main Input Axis
Unit	0:mm
Number of Decimal Places	0
Cam Axis Length per Cycle Change Setting	0:Invalid
Cam Axis Length per Cycle	30.0000 mm
Cam Stroke Amount	100000.0 μm
Cam No.	2
Cam Axis Phase Correction Advance Time	0 μs
Cam Axis Phase Correction Time Constant	10 ms
Synchronous Control Parameter Block No.	1
Output Axis Smoothing Time Constant	0 ms
Synchronous Control Initial Position Parameter	**Set the parameter for the initial alignment when starting the synchronous control.**

Go to next page

From previous page



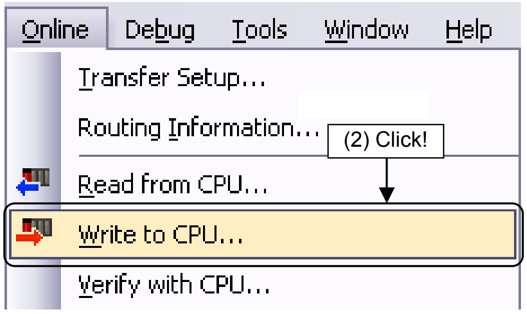
(10) Convert data for advanced synchronous control program editing to an internal code that allows the motion CPU to function.
Click [Project Batch Check/Conversion] on the [Check/Convert] menu.



(11) A cam data conversion complete message appears in the output window.

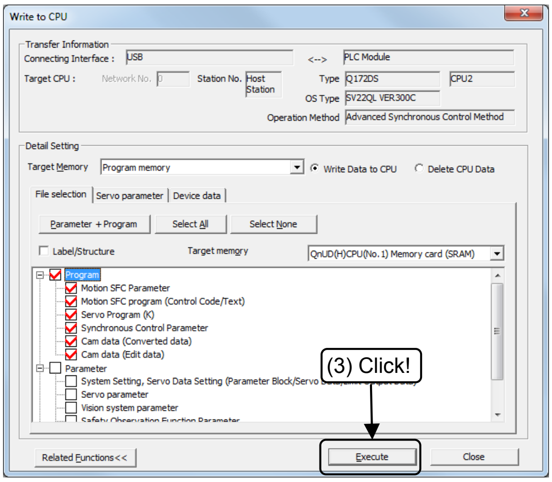
10.8 Writing to the Q Motion CPU

This section describes writing created data (motion SFC programs/synchronous control parameters/cam data) to the Q motion CPU.

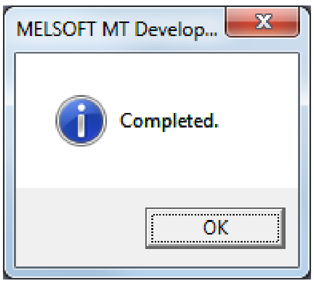


(1) Set the Q motion CPU to "STOP".

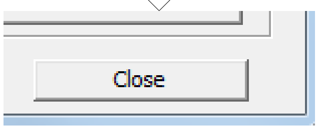
(2) Click [Write to CPU…] on the [Online] menu.



(3) A CPU Write dialog box appears. Select "Program memory" for the applicable memory, select the "Programs" check box, and then click the Execute button.
(The settings for servo settings data (system settings, servo settings data) are the same as those for real mode described in Chapter 6, and therefore writing is not necessary.



(4) Click the OK button when the "Complete" message appears.

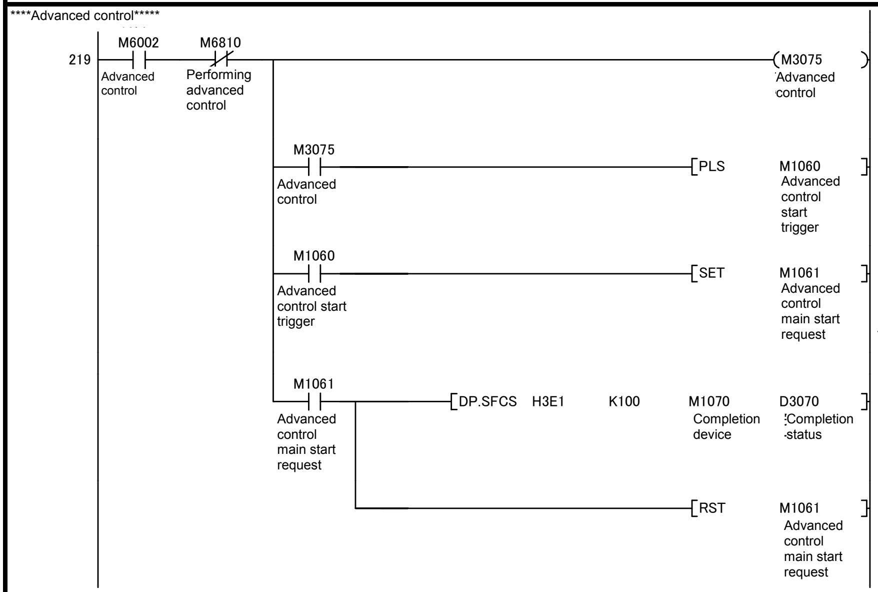


(5) Click the Close button at the CPU Write dialog box to close.

(6) Reset the Q PLC CPU, and then set the Q motion CPU to "RUN".
Data writing to the Q motion CPU is now complete.

10.9 Practice Programs

The sequence program and motion SFC program used for practice are shown in the following list.
Refer to the respective descriptions of each program in this manual for details.

<table>
<tr><th>Constant execution</th><th>Started with sequence program</th></tr>
<tr><td>
</td><td>• [Advanced synchronous control]
Motion SFC program No.100</td></tr>
</table>

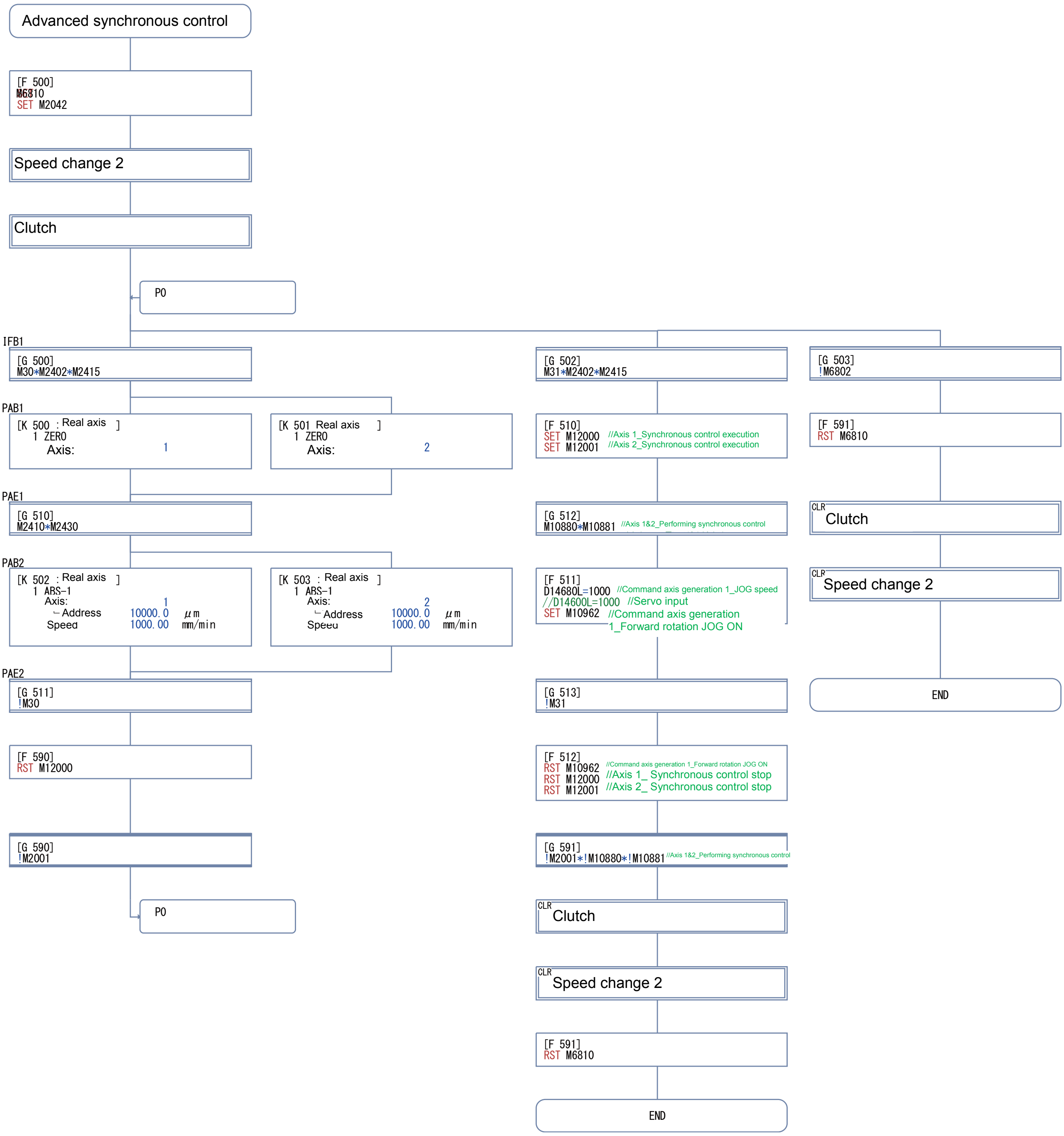
Motion SFC program parameters

No.	Program name	Automatic start	END operation	No. of transitions	Execution timing
100	Advanced synchronous control	No			Normal

[Advanced synchronous control] program No.100

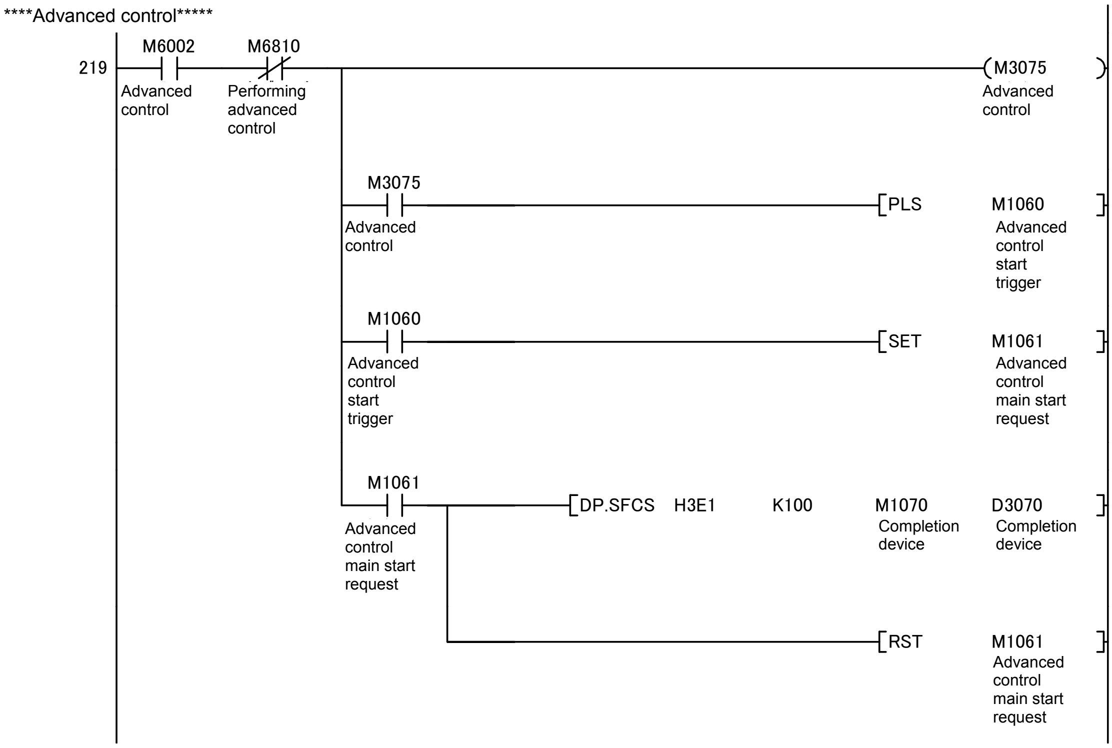
This is an example of a program used to perform positioning after switching to advanced synchronous control.

Output modules operate by starting and stopping command generation axes.



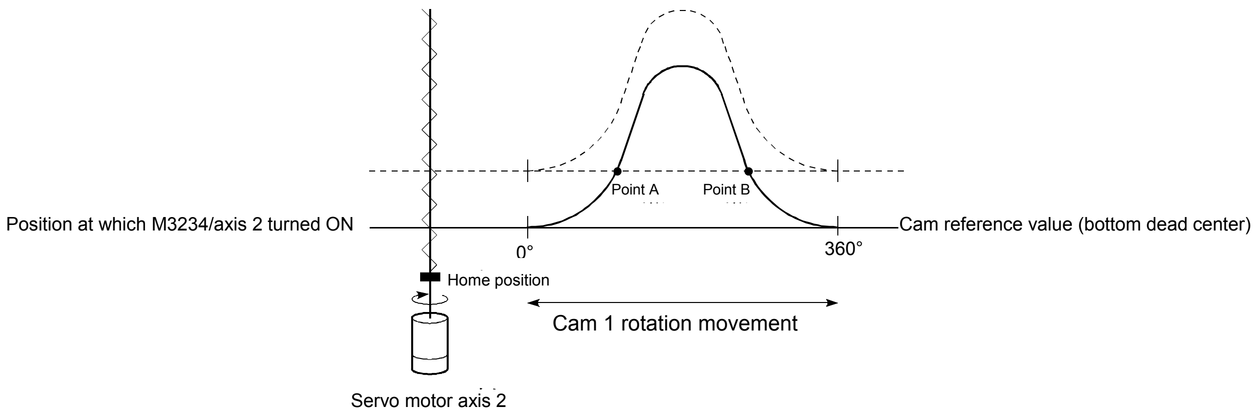
Sequence programs

Conditions are determined based on the type of output module used with the advanced synchronous control program, however, in this program example, the current value when switching becomes the cam bottom dead center position.



Caution when creating advanced synchronous control programs

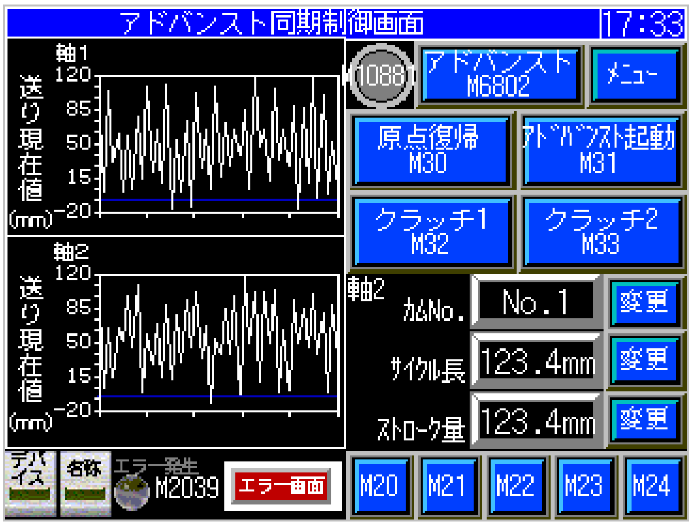
If using cams, advanced synchronous control switching setting M3075 is turned ON when the cam reference value (bottom dead center) setting M3234/axis 2 is ON.
There is no need to turn M3234/axis 2 ON and OFF each time if the reference value has been determined. If turned ON and OFF recklessly, the reference value will change, leading to potential trouble, and therefore caution is advised.

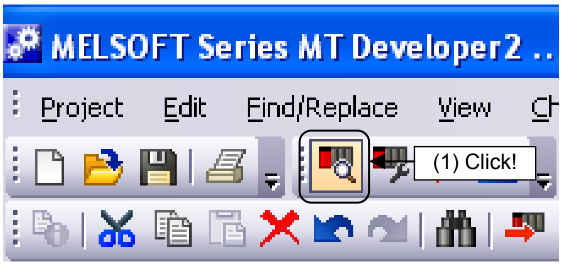


If the axis stops at point A or B on the solid line during movement, by turning M3234/axis 2 ON and OFF, the cam reference value (bottom dead center) will change to a stopped position, and the axis movement will change to that of the broken line when the next movement is made.

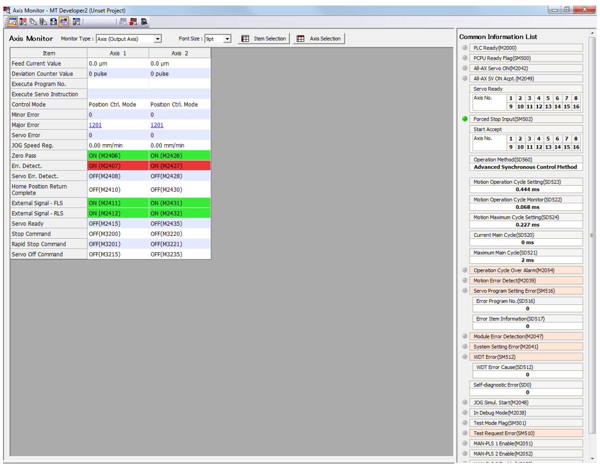
10.10 Demonstration Machine Operation

Demonstration machine operation panel





(1) Click the monitor tool button.



(2) The monitor window axis monitor appears.

Go to next page

From previous page

Set the Q PLC CPU and Q motion CPU to "RUN".

[Zeroing execution
and positioning at standby point]

Press [ADVANCED M6802] at the demonstration machine operation panel.

Press the zeroing [ZEROING M30] button: Movement starts in the zeroing direction, and positioning is performed at the standby point P address (0.0).

The feed current value will be 10000.0 µm for both axes 1 and 2.

[Switching to advanced synchronous
control and clutch operation]

Press [ADVANCED START M31] at the demonstration machine operation panel.

Press the [CLUTCH 1 M32] and [CLUTCH 2 M33] buttons, and ensure that clutch operation is possible.

- Press [CLUTCH 1 M32] during operation with advanced synchronous control.
  The clutch opens, and the movement in the X-axis direction stops.

CAUTION

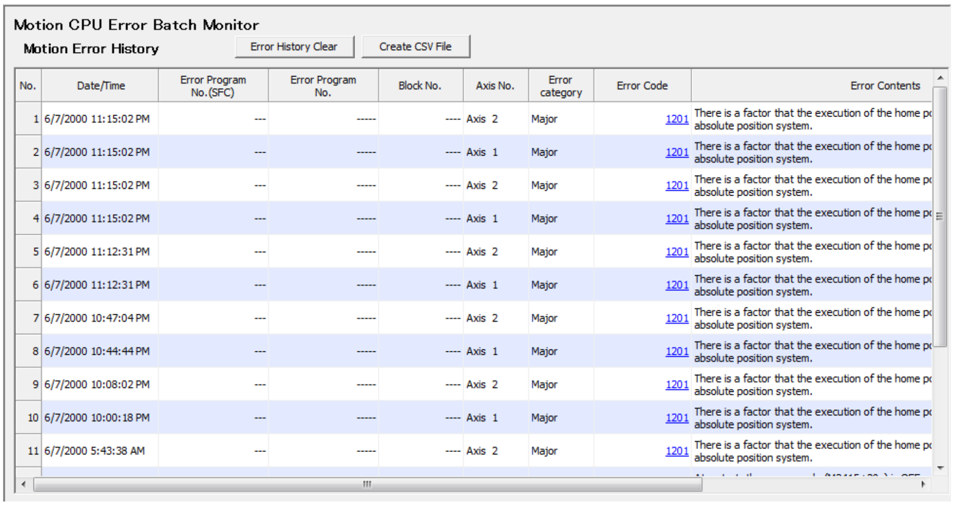
- When the clutch is turned OFF, the X-axis movement stops, and when the clutch is turned ON, the X-axis direction position changes only by the amount that the axis did not move when it stopped.

Error check operation

Motion CPU error batch monitor screen

Select [Monitor] → [Motion CPU error batch monitor] → [Motion CPU error batch monitor] on the [Online] menu.

↓



Motion CPU Error Batch Monitor

Motion Error History

Error History Clear | Create CSV File

No.	Date/Time	Error Program No.(SFC)	Error Program No.	Block No.	Axis No.	Error category	Error Code	Error Contents
1	6/7/2000 11:15:02 PM	---	-----	----	Axis 2	Major	1201	There is a factor that the execution of the home pc absolute position system.
2	6/7/2000 11:15:02 PM	---	-----	----	Axis 1	Major	1201	There is a factor that the execution of the home pc absolute position system.
3	6/7/2000 11:15:02 PM	---	-----	----	Axis 2	Major	1201	There is a factor that the execution of the home pc absolute position system.
4	6/7/2000 11:15:02 PM	---	-----	----	Axis 1	Major	1201	There is a factor that the execution of the home pc absolute position system.
5	6/7/2000 11:12:31 PM	---	-----	----	Axis 2	Major	1201	There is a factor that the execution of the home pc absolute position system.
6	6/7/2000 11:12:31 PM	---	-----	----	Axis 1	Major	1201	There is a factor that the execution of the home pc absolute position system.
7	6/7/2000 10:47:04 PM	---	-----	----	Axis 2	Major	1201	There is a factor that the execution of the home pc absolute position system.
8	6/7/2000 10:44:44 PM	---	-----	----	Axis 1	Major	1201	There is a factor that the execution of the home pc absolute position system.
9	6/7/2000 10:08:02 PM	---	-----	----	Axis 2	Major	1201	There is a factor that the execution of the home pc absolute position system.
10	6/7/2000 10:00:18 PM	---	-----	----	Axis 1	Major	1201	There is a factor that the execution of the home pc absolute position system.
11	6/7/2000 5:43:38 AM	---	-----	----	Axis 2	Major	1201	There is a factor that the execution of the home pc absolute position system.

Go to next page

From previous page

[Changing the stroke amount]

- Set the stroke amount in the 1.0 to 120.0 mm range at the demonstration machine operation panel.
- Ensure that the stroke amount is changed.

[Speed change]

Press M20 (2000 mm/min), M21 (1000 mm/min), M22 (500 mm/min), and M23 (temporary stop) to change the command generation axis speed.

Press M24 to change the command generation axis speed in analog.

[Set the cam No. to "2".]

- Set the cam No. to "2" at the demonstration machine operation panel.
- "2" displays for the "Execute cam No."

[Content to be checked (cam No.2)]

- Watch the stop status.
- Check the details monitor at each module.
- Change the stroke amount.
- Watch the movement when the speed is changed (2000 mm/min, 1000 mm/min, 500 mm/min, temporary stop).
- Watch the movement when the clutch is turned OFF.

[Set the cam No. to "3".]

- Set the cam No. to "3" at the demonstration machine operation panel.
- "3" displays for the "Execute cam No."

[Content to be checked (cam No.3)]

- Watch the stop status.
- Check the details monitor at each module.
- Change the stroke amount.
- Watch the movement when the speed is changed (2000 mm/min, 1000 mm/min, 500 mm/min, temporary stop).
- Watch the movement when the clutch is turned OFF.

[Cycle length setting]

- Press ADVANCED START to end advanced startup.
- Press ADVANCED M6802 to end all operations.
- Set the cycle length again at the demonstration machine operation panel.
- Press ADVANCED M6802 again, and then press ZEROING M30 to perform zeroing.
- Set in the same manner for cam No.2 and No.3.

Practice is complete when all of these operations are finished.

Point

If the cycle length < the stroke amount, a 31.1 35.1 2-axis alarm may occur at the servo amp.

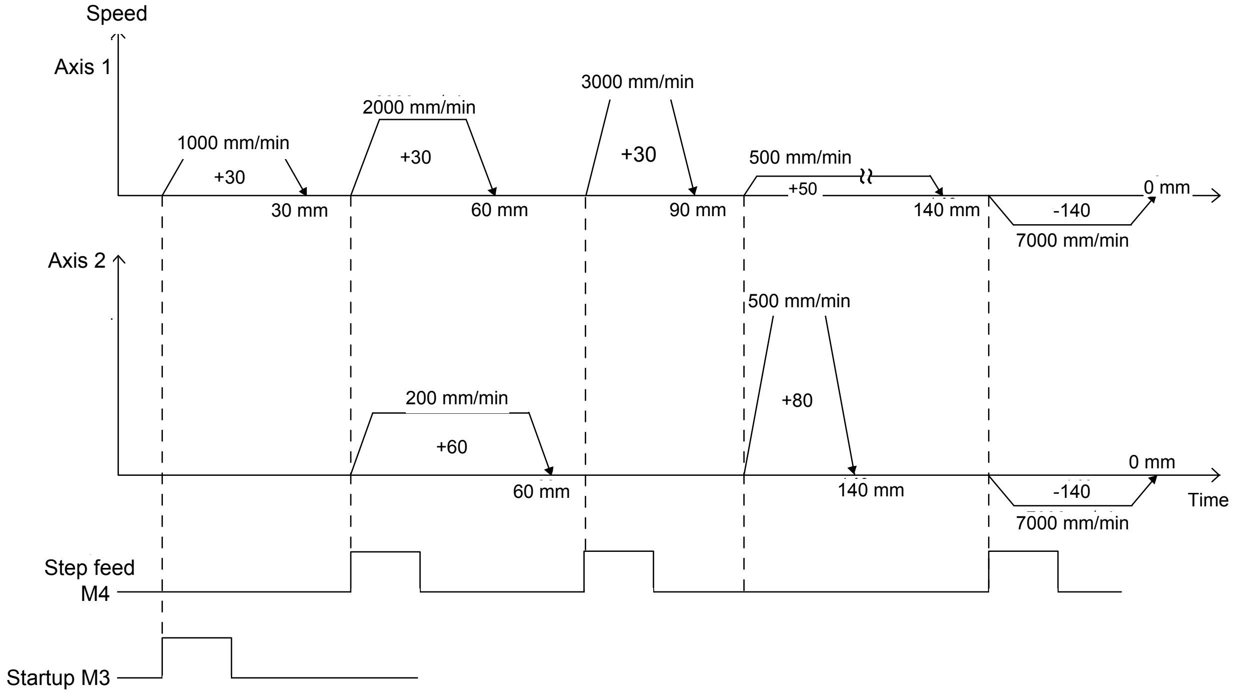
Appendices

Appendix 1 Application Practice in SV22 Real Mode

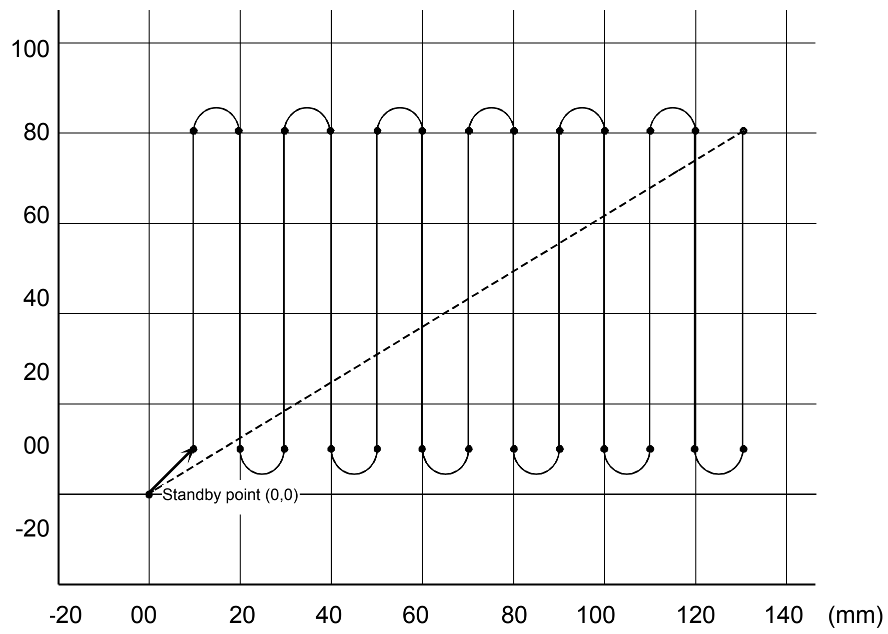
Appendix 1.1 Practice Content

Perform continuous positioning at multiple points.
SV13 operation is the same as that for SV22 in real mode, and therefore this practice applies to both.

Continuous positioning (1) operation diagram



Continuous positioning (2) operation diagram

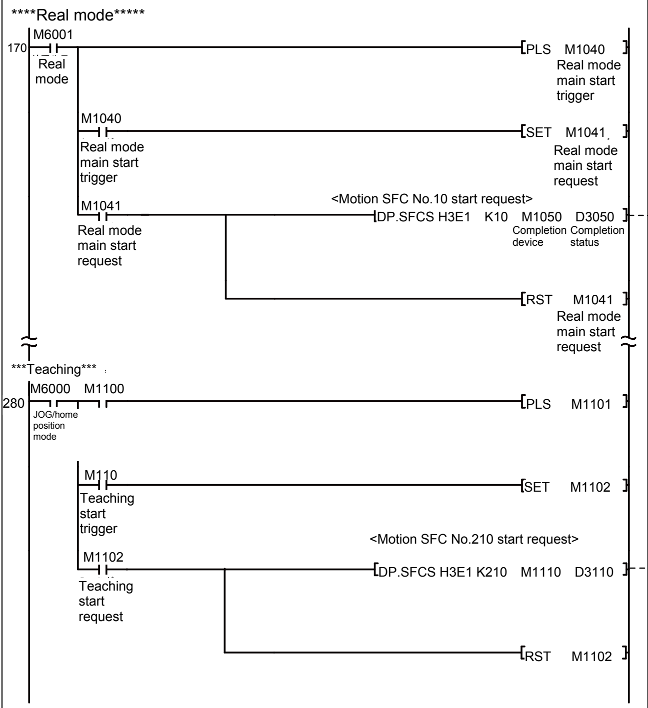


Appendix 1.2 Practice Motion SFC Programs

These sequence/motion SFC programs have been created for operation purposes on the assumption that MT Works2 (for Q172DSCPU) be used.
Refer to section 9.2 for an explanatory drawing of the demonstration machine operation panel.
Refer to section 9.5 for details on initial processing, JOG operation, zeroing, standby point positioning, point selection positioning, and address indirect designation positioning.

Appendix 1.2.1 Program list

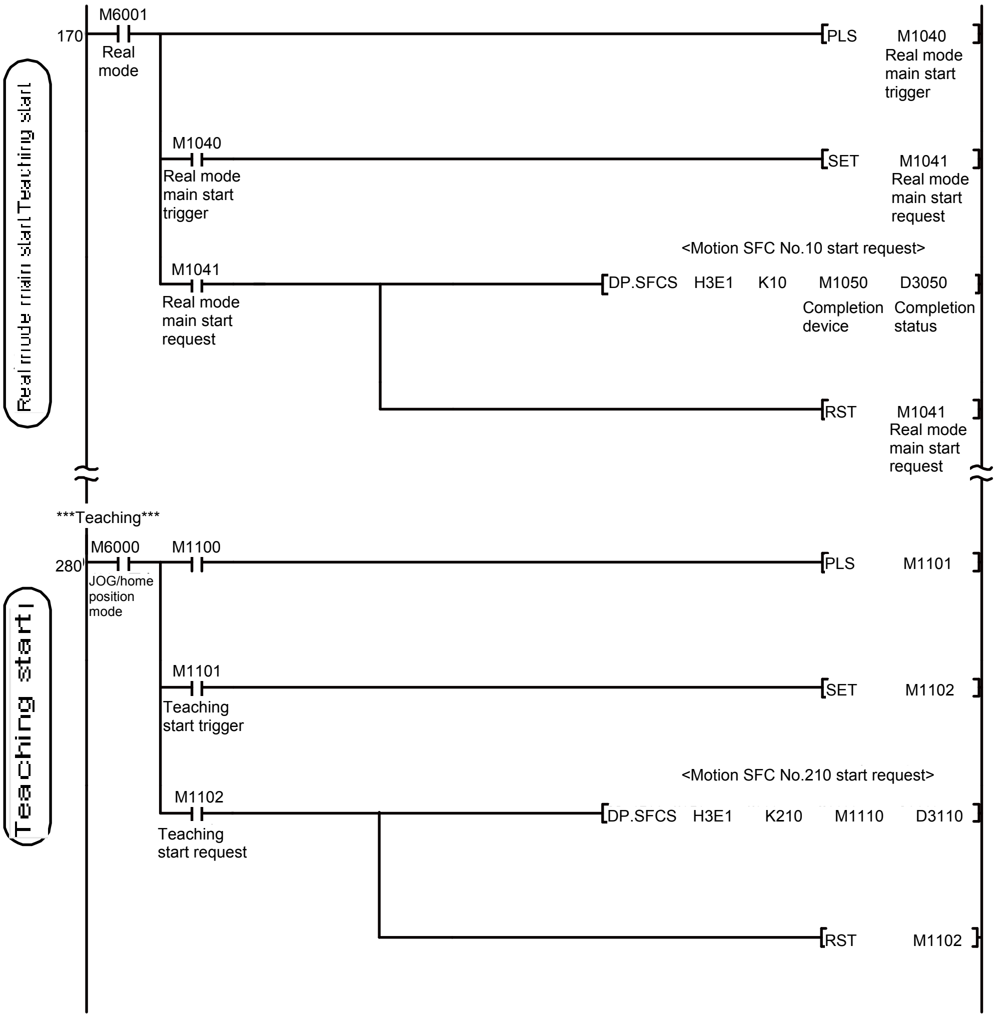
The sequence program and motion SFC program used for practice are shown in the following list.
Refer to the respective descriptions of each program in this manual for details.

Constant execution	Started with sequence program	Started with motion SFC program
• Sequence program  	• [Real mode main] Motion SFC program No.10 • [Teaching] Motion SFC program No.210	• [Standby positioning] Motion SFC program No.20 • [Point selection] Motion SFC program No.30 • [Address designation] Motion SFC program No.40 • [Continuous positioning (1)] Motion SFC program No.50 • [Continuous positioning (2)] Motion SFC program No.60 • [Teaching playback] Motion SFC program No.70 • [Fixed feed] Motion SFC program No.80 • [Fixed feed advance] Motion SFC program No.220 • [Speed change] Motion SFC program No.200

Motion SFC program parameters

No.	Program name	Automatic start	END operation	No. of transitions	Execution timing
10	Real mode main	No	-	3	Normal
20	Standby point positioning	No	-	3	Normal
30	Point selection	No	-	3	Normal
40	Address indirect designation	No	-	3	Normal
50	Continuous positioning (1)	No	-	3	Normal
60	Continuous positioning (2)	No	-	3	Normal
70	Teaching playback	No	-	3	Normal
80	Fixed feed	No	-	3	Normal
200	Speed change	No	-	3	Normal
210	Teaching	No	-	3	Normal
220	Fixed feed advance	No	Continuous	2	Event (0.8 ms)

• **Q03UD sequence program**



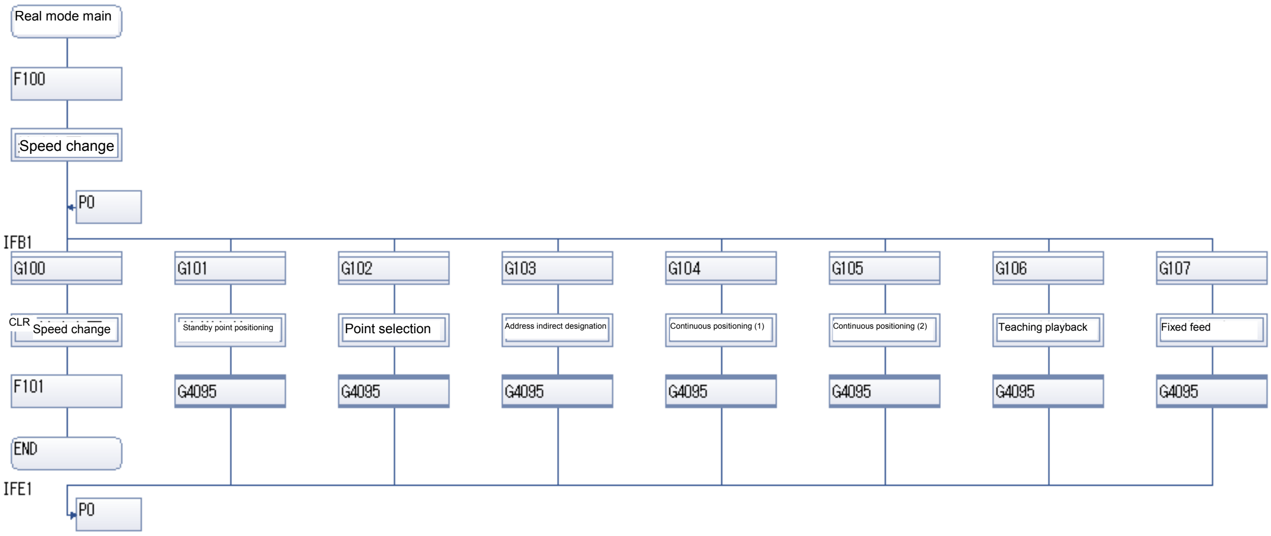
Appendix 1.2.2 Main routine motion SFC program (real mode operation)

This is the main executed motion SFC program when performing operation in real mode.
Other motion SFC programs used to perform various types of operation when in real mode from this main routine motion SFC program are started as sub-routines.

(1) Motion SFC program started from main routine motion SFC program

Motion SFC program No.	Program name	Reference section
20	Standby point positioning	9.9
30	Point selection	9.9
40	Address indirect designation	9.9
50	Continuous positioning (1)	Appendix 1.2.3
60	Continuous positioning (2)	Appendix 1.2.4
70	Teaching playback	Appendix 1.2.5
80	Fixed feed	Appendix 1.2.6
200	Speed change	Appendix 1.2.7
210	Teaching	Appendix 1.2.5
220	Fixed feed advance	Appendix 1.2.6

(2) Program example



Appendix 1.2.3 Continuous positioning (1)

This is an example of a program used to perform positioning at multiple points based on respective conditions.
The standby method if the flow is branched, and M-codes that can be used to control auxiliary machinery with sequence programs are set.

(1) Multiple servo program execution order control

To execute servo programs in the order 50 → 51, 52 → 53 → 54 → 56, 57, by using a "WAIT" type transition after the motion control step (servo program), the system waits until the servo program currently running is complete before proceeding to the next motion control step (servo program).
Furthermore, if the program is interrupted during consecutive execution, execution is resumed from the interrupted servo program.

(2) Example of servo program with M-code

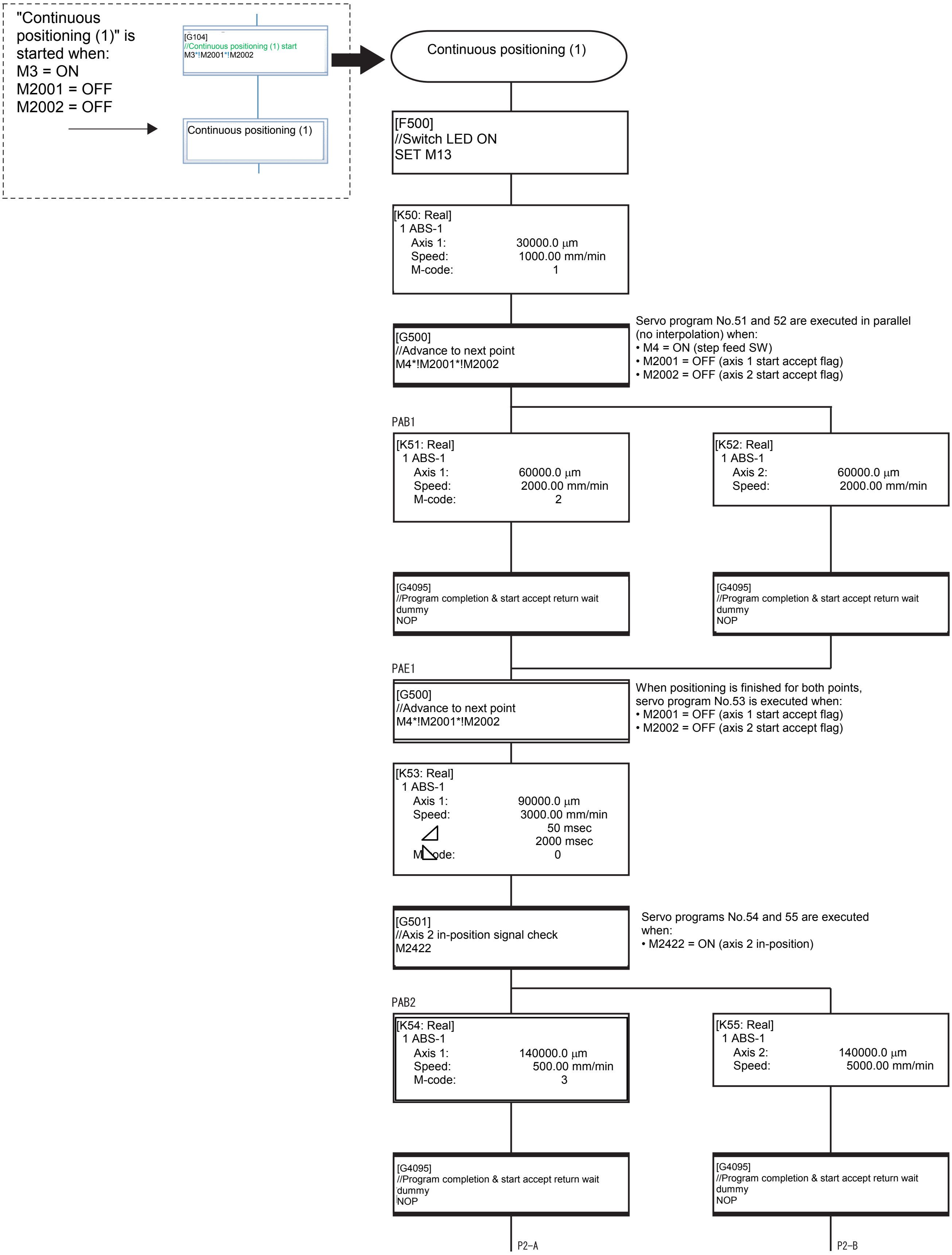
M-codes 0 to 255 are added to servo programs, and by running these programs, M-code Nos. are entered in the M-code monitor register.
Data is also sent to the PLC CPU by setting auto refresh (user setting), and therefore if monitored with the sequence program comparison command, the M-code No. is known, allowing the operation determined beforehand to be performed.

```
[K  50 : Real Axis ]
   1   ABS-1
        Axis                      1
        ->Address          100000.0  μm
        Speed               1000.00  mm/min
        M-code                    1
```

M-code 1 is added.

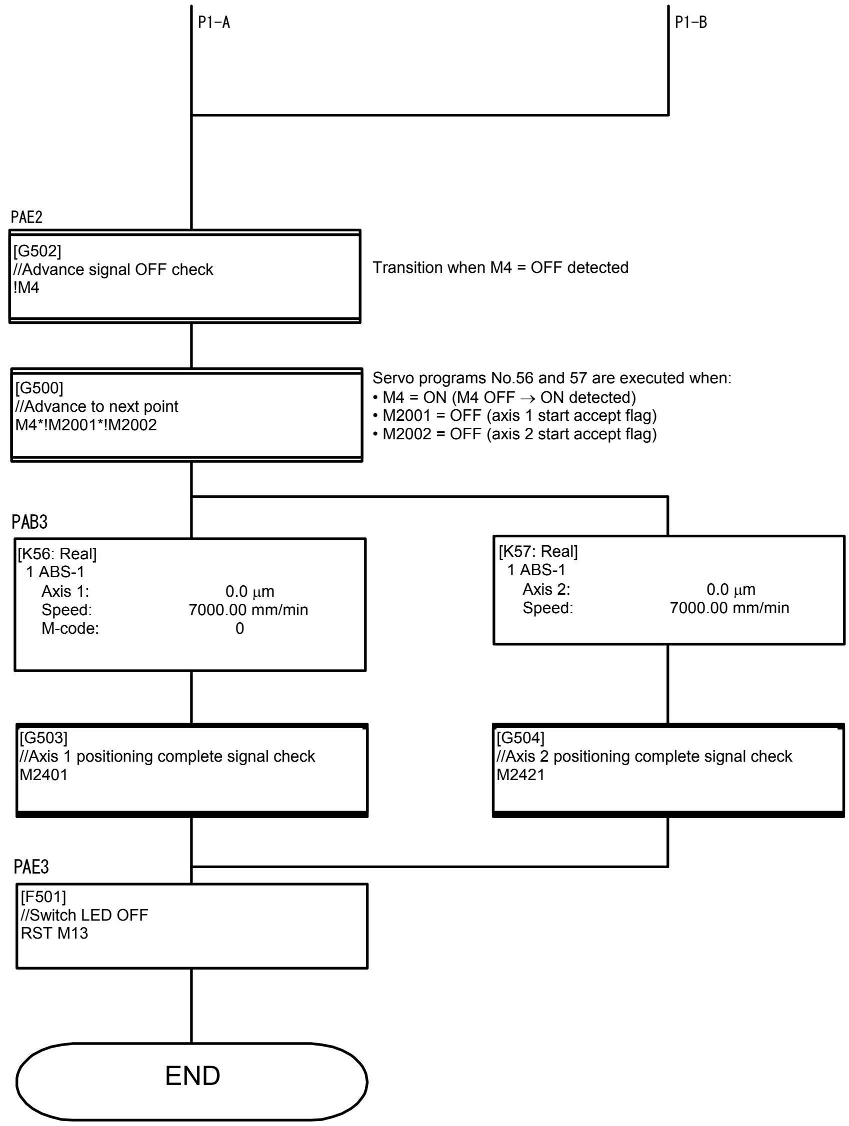
(3) Motion SFC program

[Real mode main] program



(Go to next page)

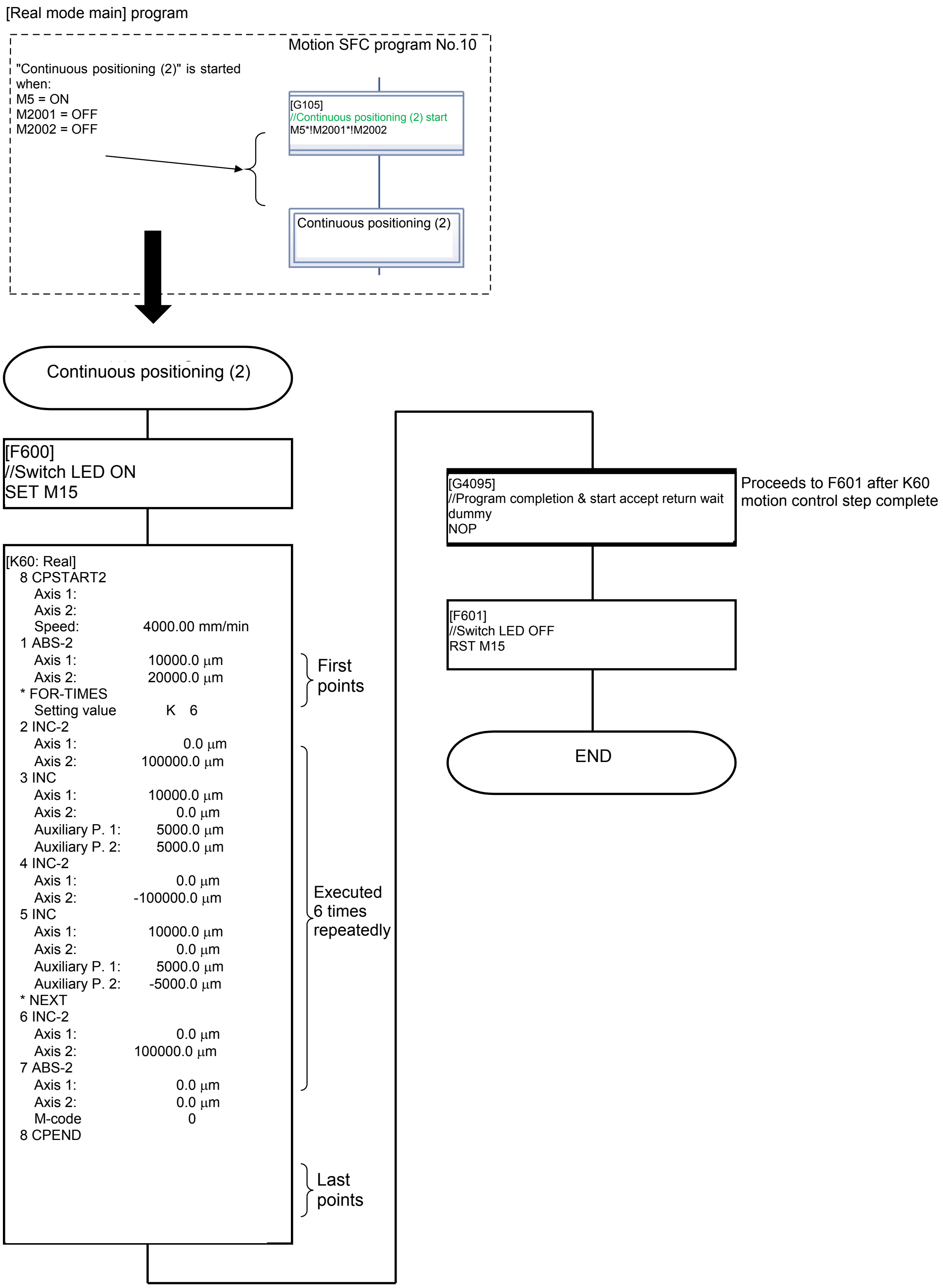
(From previous page)



Appendix 1.2.4 Continuous positioning (2)

This is an example of a program used to perform continuous interpolation between multiple points with 2-axis constant speed control.
Even with independent servo programs, multiple operations are possible if the operation pattern is fixed.

[Motion SFC program]

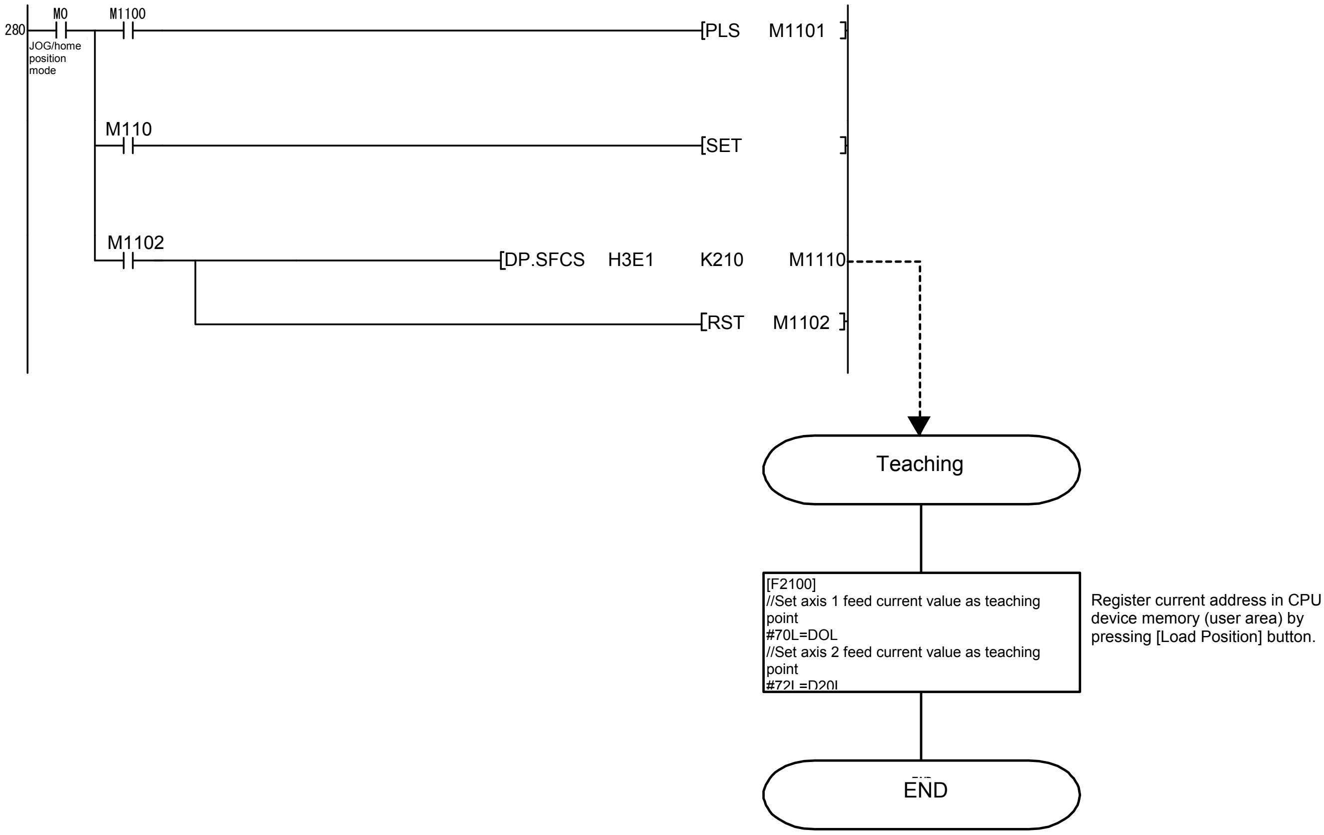


Appendix 1.2.5 Teaching, teaching playback

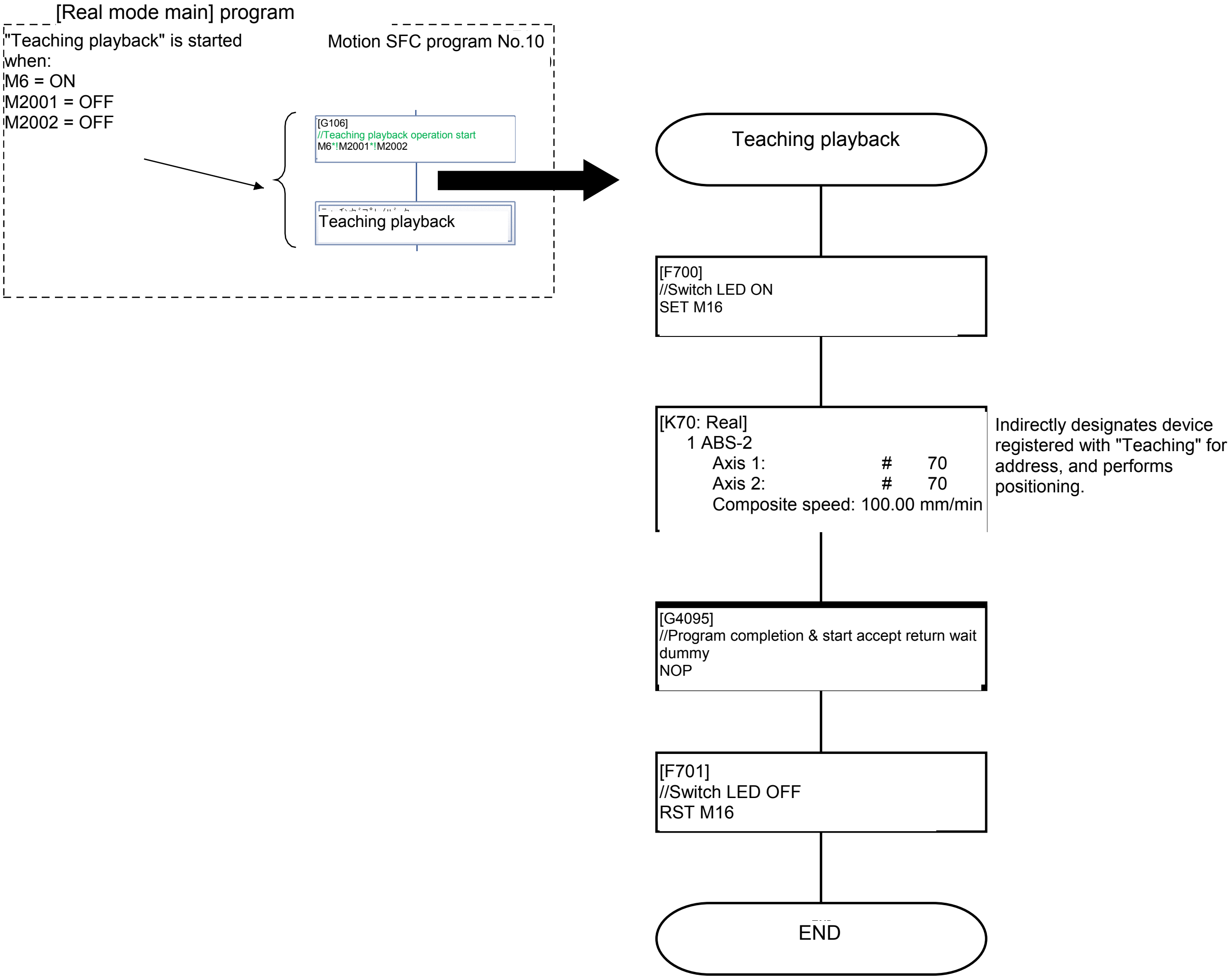
Teaching programs are used to register positions (with push button operation) to which axes are moved to manually with JOG operation and so on, and teaching playback programs are used to perform position at registered addresses.

Motion SFC program No.210 [teaching]

Register the current address by pressing the [Load Position] button on the demonstration machine operation panel.



Motion SFC program No.70 [teaching playback]
Perform positioning at the address registered with teaching.

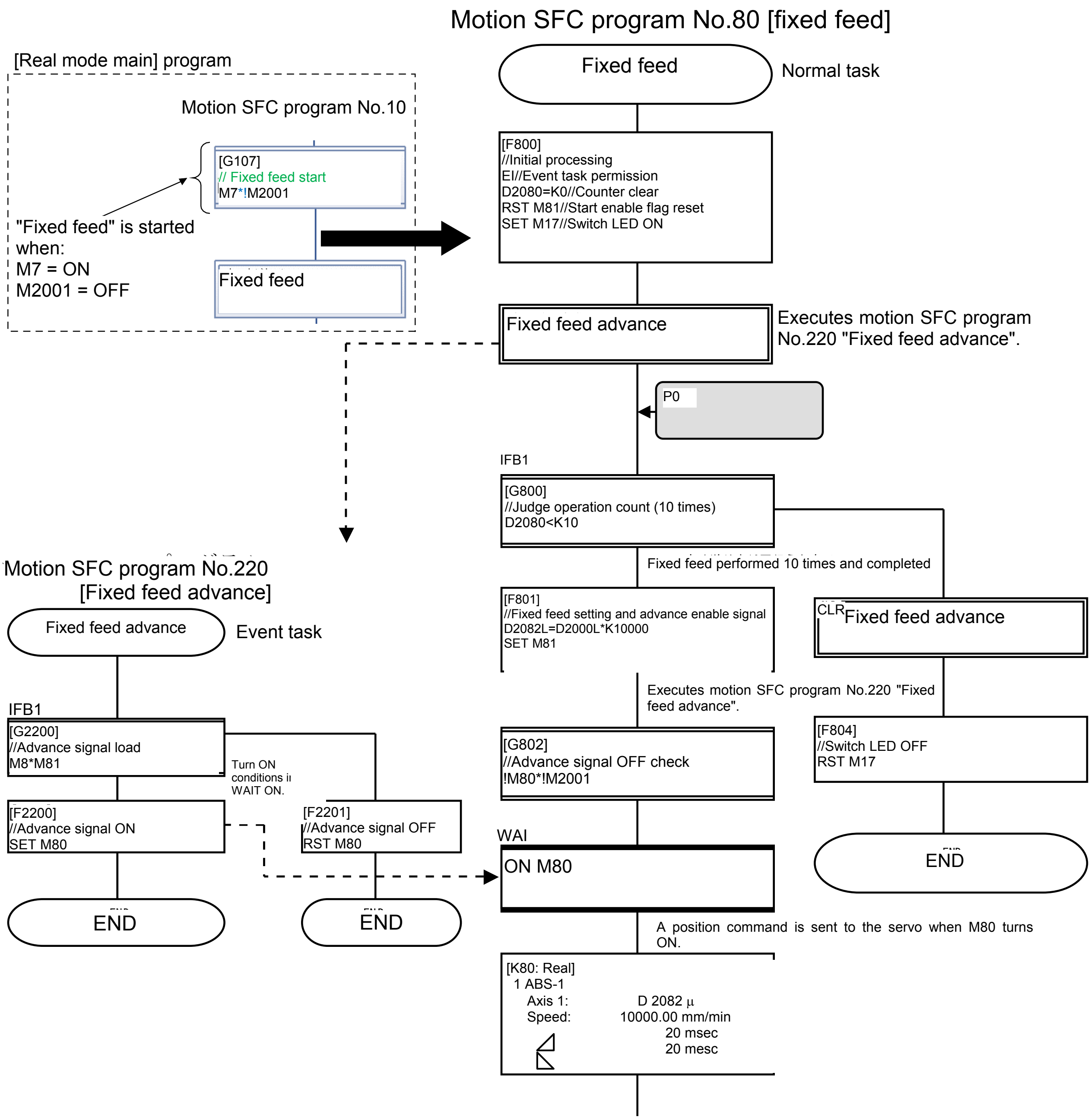


Appendix 1.2.6 Fixed feed, fixed feed advance

Operations in which workpieces of fixed length are fed at fixed timing such as when inputting signals are known as fixed feed.
If there are many fixed feed, and the interval between signals is short, there may be times when it is necessary to shorten the start time between signal input and the start of operation.
With this program example, the following effective functions are used in such a case.

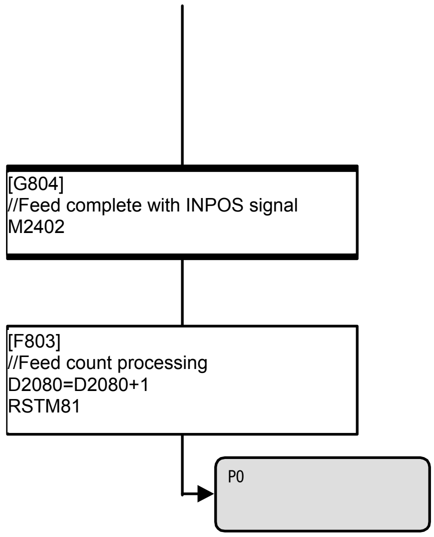
- WAIT-ON(WAIT-OFF) command: Performs start preparations for the next motion control step beforehand.

- Event tasks: Periodically runs a motion SFC program at a fixed cycle (0.8 ms).



(Go to next page)

(From previous page)

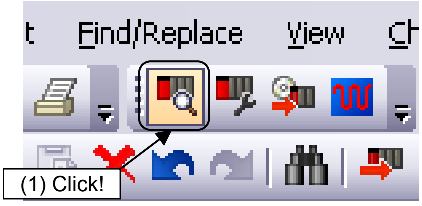


The task type and operating conditions for each program are set in the "Program Parameters". "Program Parameters" are located in the peripheral tool "Options" → "SFC Parameter Settings" → "Program Parameters".

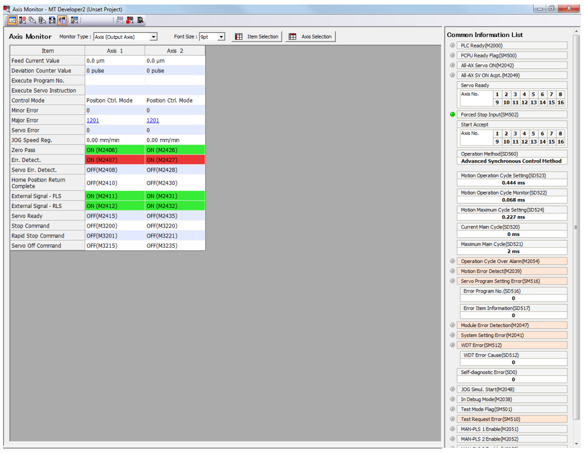
Appendix 1.3 Demonstration Machine Operation

Appendix 1.3.1 Operation

Servo motors are run, and servo motor operation is monitored with MT Works2.



(1) Click the monitor tool button.



(2) The monitor window Current Value Expansion Monitor appears.

Go to next page

[Teaching/
teaching playback]

From previous page

- Teaching

Press [JOG] at the demonstration machine operation panel.

Enable the JOG operation screen [JOG/Home Pos.] button.

- Perform JOG operation using the "JOG Operation" [←X], [X→], [↑Y], and [↓Y] buttons.
- Turn ON "Teaching" position load, and register the position moved to with JOG operation.

[Teaching] program (motion SFC program No.210)

```
Teaching

[F2100]
  //Set axis 1 feed current value as teaching
 #70L=D0L
  //Set axis 2 feed current value as teaching
 #72L=D20L

END
```

With "Position Load" ON, start motion SFC program [Teaching].
Substitute the axis 1 and 2 current values (D0, D20) for #70 and #72.

- Teaching playback

Change to the Real screen.

- Press [Real Mode Main] to turn ON the running lamp.
- By pressing [Teaching] on the screen, positioning is performed at the registered address.

[Teaching playback] program (motion SFC program No.70)

```
Teaching playback

[F 700]
//Switch LED ON
SET M16

[K70: Real]
1 ABS-2 (composite)
  Axis                    1
   └ Address         #   70   μm
  Axis                    2
   └ Address         #   72   μm
  Composite speed   1000.00   mm/min
```

Perform positioning at the #70 and #72 addresses registered with teaching.

Go to next page

From previous page

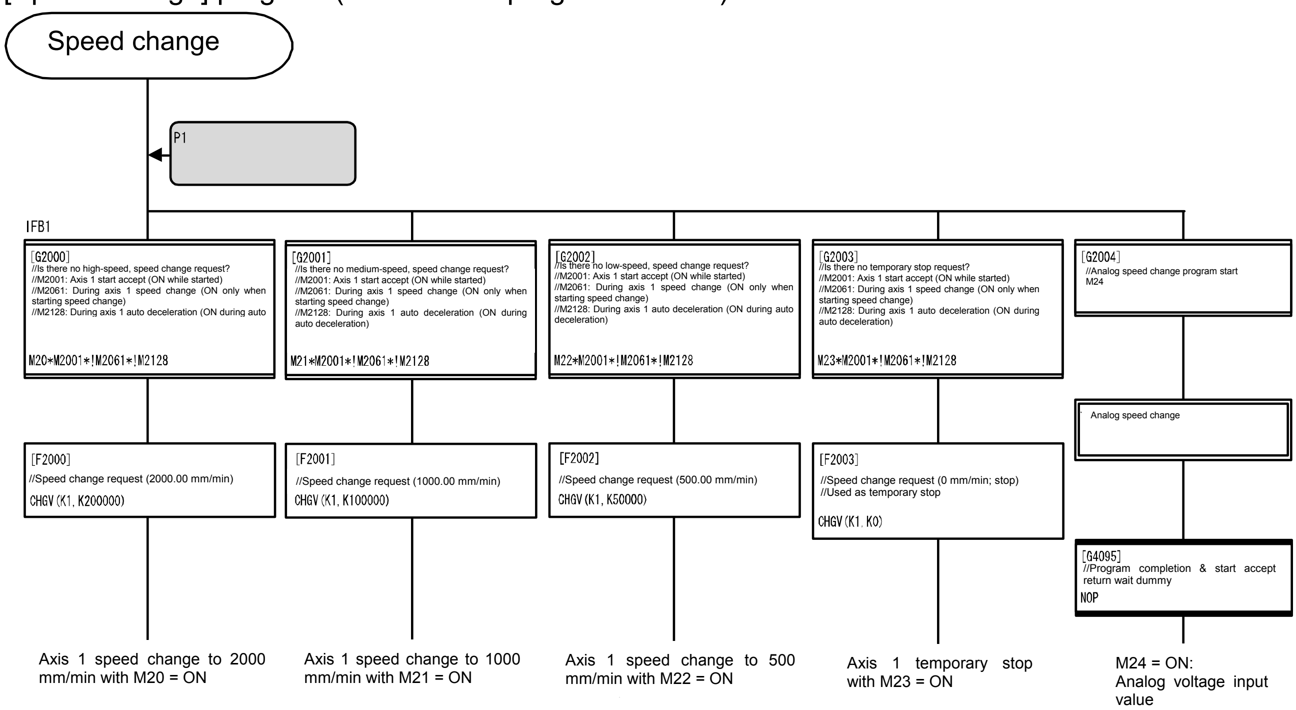
[Mid-operation check details]

Speed change/temporary stop during operation (operation during continuous positioning, constant speed control, speed control)

- By turning ON the [2000] touch panel speed change switch, the speed changes to 2000 mm/min.
- By turning ON the [1000] touch panel speed change switch, the speed changes to 1000 mm/min.
- By turning ON the [500] touch panel speed change switch, the speed changes to 500 mm/min.
- By turning ON the [0] touch panel speed change switch, movement stops temporarily.
- By turning ON the [Analog] touch panel speed change switch, speed changes to analog speed.

(The speed may be changed multiple times during operation. However, do not perform operation during zeroing or during deceleration. A minor error will occur.)

[Speed change] program (motion SFC program No.200)



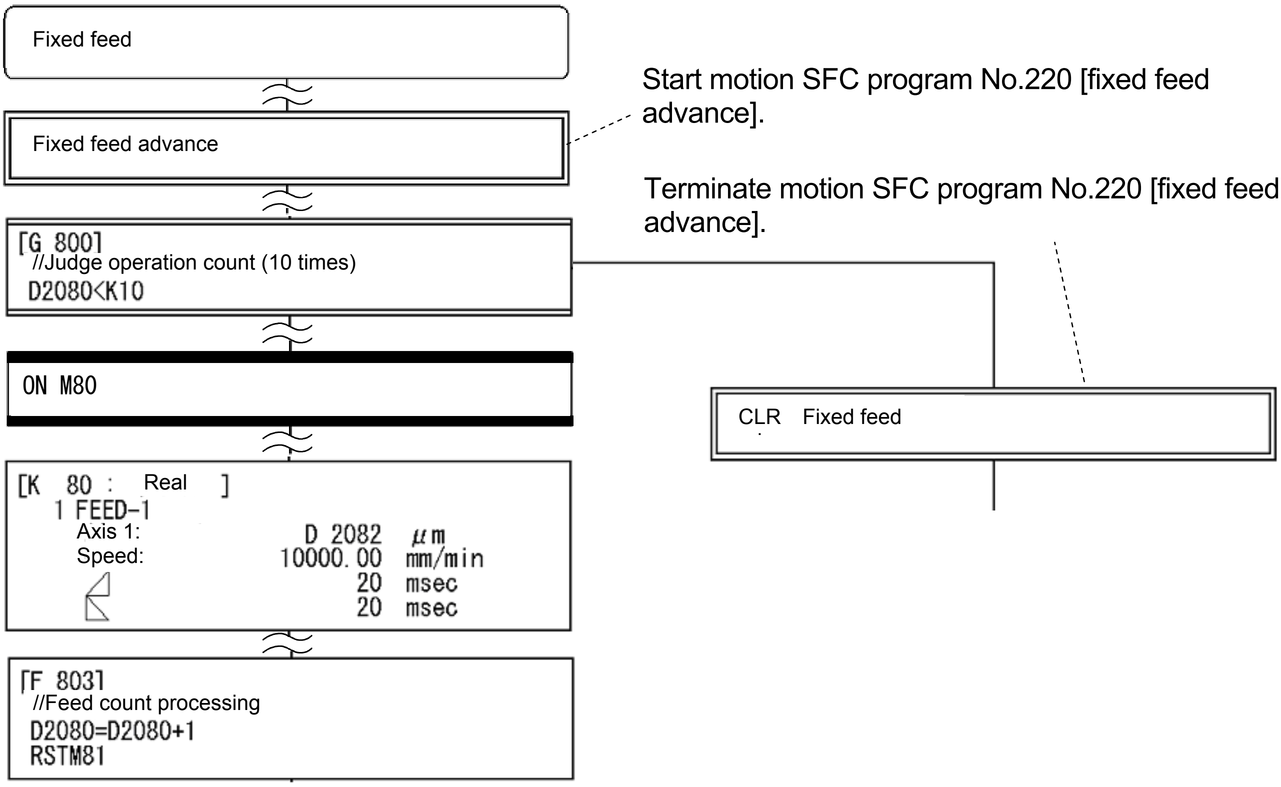
Go to next page

From previous page

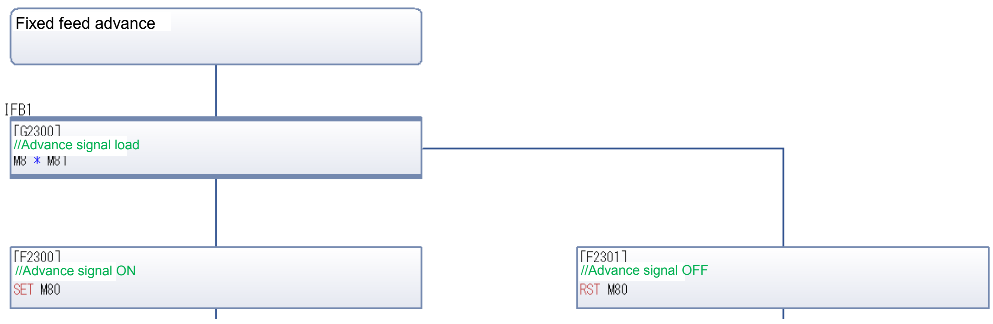
[Fixed feed, fixed feed advance]

- Fixed feed, fixed feed advance
  Change to the Real screen.
  - Press [Real Mode Main] to turn ON the running lamp.
  - Set the fixed feed amount to "10" at the touch panel.
  - Press the "Fixed Feed" [Permit fixed] button to permit fixed feed operation. → Operation will not start yet.
  - Fixed feed is performed once each time the "Fixed Feed" [Execute fixed] button is pressed, and stops after ten times.
  - Fixed feed operation is performed with the fixed feed and fixed feed advance motion SFC program.

[Fixed feed] program (motion SFC program No.80)



[Fixed feed advance] program (motion SFC program No.220)



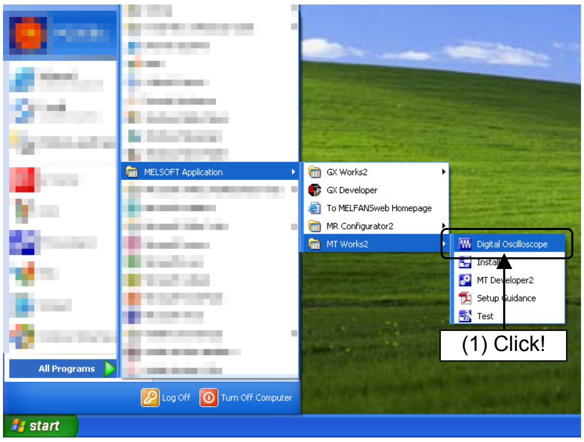
Operation complete

Appendix 2 Digital Oscilloscope

Position commands, position droop, motor speed, motor current, and speed commands and so on can be traced with the MT Works2 digital oscilloscope.

Refer to the performance specifications (digital oscilloscope) in the MT Developer2 Help.

(1) Communication settings

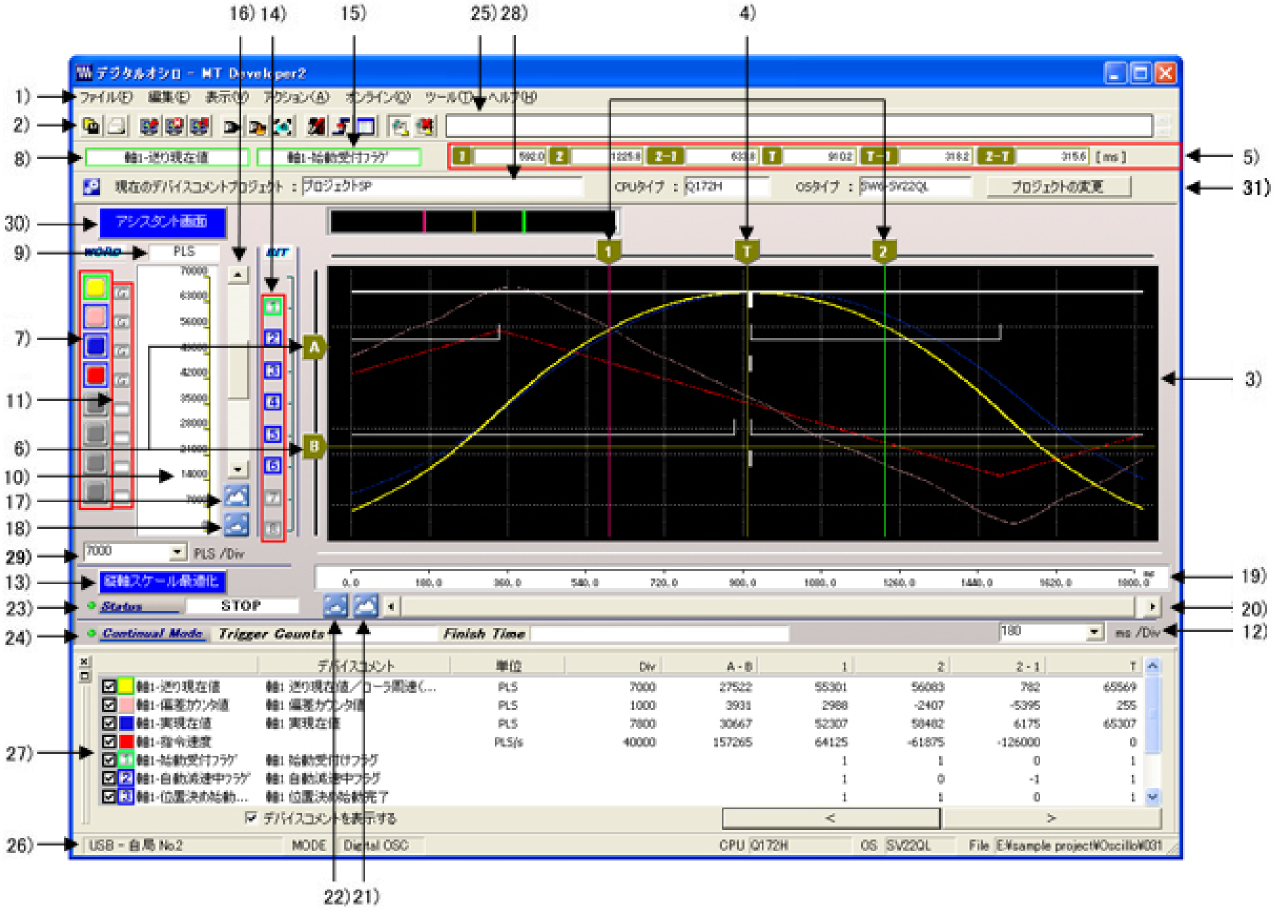


(1) Click the Windows [start] button, and then select [All Programs] → [MELSOFT Application] → [MT Works2] → [Digital Oscilloscope].

Go to next page

From previous page

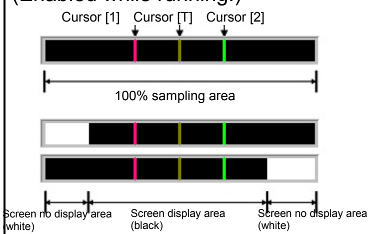
(2) A Digital Oscilloscope window appears.

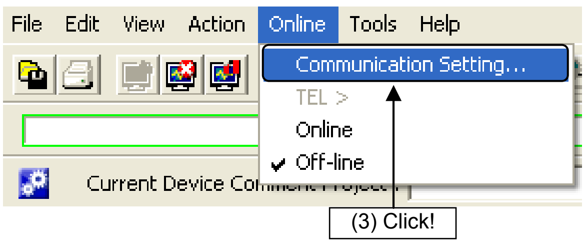


No.	Item	Details
1)	Menu bar	This menu is used to perform each function.
2)	Toolbar	Displays tool buttons used to perform each function.
3)	Waveform display area	Displays word data and bit data waveforms.
4)	X-axis cursors [1], [2], [T]	Displays X-axis cursors [1] and [2], and trigger cursor[T].
5)	X-axis cursor position display field	Displays X-axis cursor [1] and [2] and trigger cursor[T] position (time), and the time between cursors. (Unit: msec)
6)	Y-axis cursors [A], [B]	Displays Y-axis cursors [A] and [B]
7)	Word waveform selection button	Selects the word waveform subject to operation.
8)	Word waveform item name display field	Displays the probe name for the word waveform selected with the word waveform selection button.
9)	Word waveform item unit display field	Displays the data unit for the word waveform selected with the word waveform selection button.
10)	Word waveform selection item scale display field	Displays the data scale value for the word waveform selected with the word waveform selection button.
11)	GND level button	Displays the GND(0) existence, and changes between the word waveform and GND level display.
12)	X-axis 1 Division setting field (Displays only in FIXED grid mode.)	Changes the X-axis 1 Division setting.
13)	Y-axis scale optimization button (Displays only in FIXED grid mode.)	Automatically adjusts Y-axis divisions so that the selected word waveform can be displayed inside a single screen.
14)	Bit waveform selection button	Selects the bit waveform subject to operation.
15)	Bit waveform selection item display field	Displays the probe name for the bit waveform selected with the word waveform selection button.
16)	Y-axis waveform scrollbar	Scrolls the word waveform selected with the word waveform selection button in the Y-axis direction.
17)	Vertical waveform enlarge button ( )	Enlarges the scale of the word waveform selected with the word waveform selection button.
18)	Vertical waveform reduce button ( )	Reduces the scale of the word waveform selected with the word waveform selection button.

Go to next page

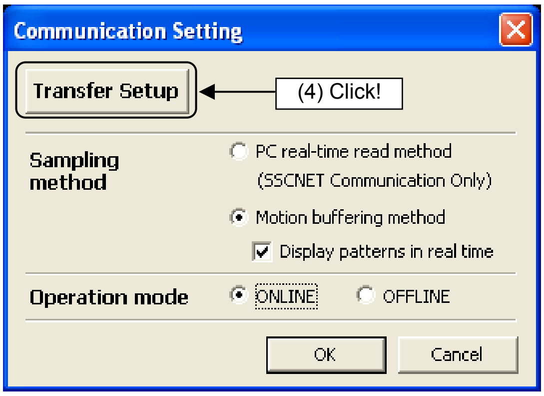
From previous page

No.	Item	Details
19)	X-axis (time) scale display field	Displays the X-axis (time axis) scale.
20)	X-axis waveform scrollbar	Scrolls through the entire waveform in the X-axis direction.
21)	Horizontal waveform enlarge button ( )	Enlarges the entire waveform in the horizontal direction.
22)	Horizontal waveform reduce button ( )	Reduces the entire waveform in the horizontal direction.
23)	Status display field	Displays the status when sampling.
24)	Continual mode status display field	Displays the status during execution in trigger type Continual mode.
25)	File comment display field	Displays a comment for the currently displayed file.
26)	Status bar	Displays digital oscilloscope status information.
27)	Docking window — Cursor window	Displays cursor position data and the difference between cursors as the X-axis and Y-axis cursors move.
28)	MAP window	Displays which area of the 100% sampling data is the data area (X-axis range) displayed in the graph display field with a black band. *:The display area is only the X-axis scale range. The Y-axis scale display area is not applicable. By left-clicking any position in the MAP window, a graph displays with the clicked X-axis position as the center (vicinity). (Enabled while running.) 
29)	Word waveform scale mode display/change field (Displays only in AUTO grid mode.) 	Displays/changes the data scale mode for the word waveform selected with the word waveform selection button. • Manual scale [FIX] button: If the word waveform scale mode is changed to MANUAL, enlarge/reduce (range adjustment) the Y-axis scale, scroll the Y-axis (display area), and adjust the GND(0) position, and then press the FIX button to set the scale.
	Y-axis 1 Division setting field (Displays only in FIXED grid mode.)	Changes the Y-axis 1 division setting for the selected word waveform.
30)	Assistant screen display button	Displays the Assistant screen. The display changes from [STOP -> Assistant screen] while running.
31)	Device comment project bar	Displays the set content for the current device comment project.

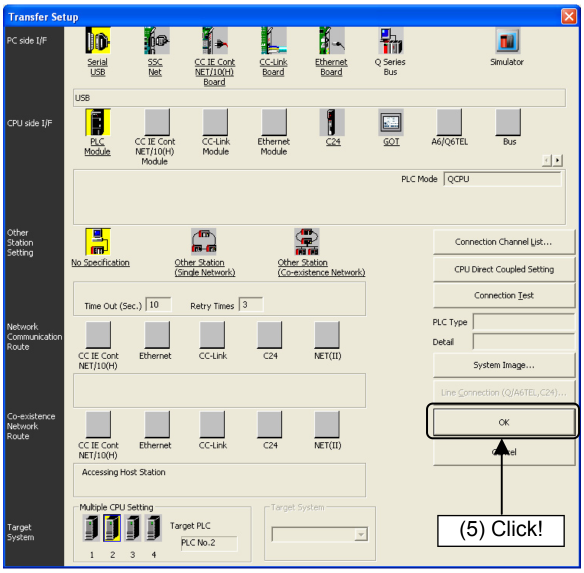


(3) Click [Communication Setting…] on the digital oscilloscope [Online] menu to specify communication settings.

Go to next page



(4) A Communication Settings dialog box appears.
Select "Motion buffering method" (select the check box to display waveforms in real time) for the "Sampling method", and select "ONLINE" for the "Operation mode".
When settings are complete, click the Transfer Setup button.

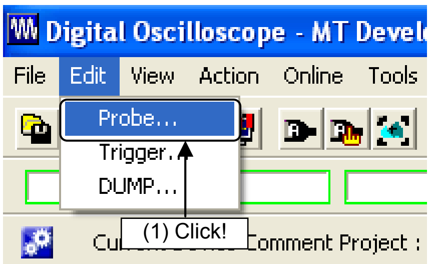


(5) Specify the following settings at the Transfer Setup dialog box that appears, and then click the OK button.

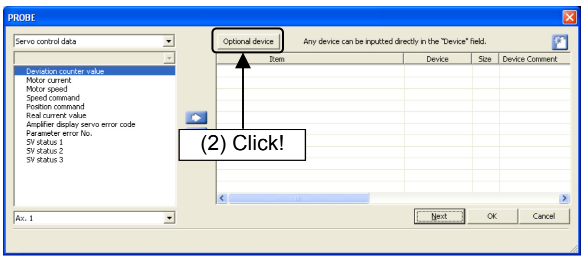
- Computer I/F: Serial USB
- CPU I/F: CPU module
- Other station designation: No other station designation
- Applicable system: Multiple CPU designation No.2 CPU

(6) The display then returns to the Communication Setting dialog box. Click the OK button.

(2) Waveform measurement



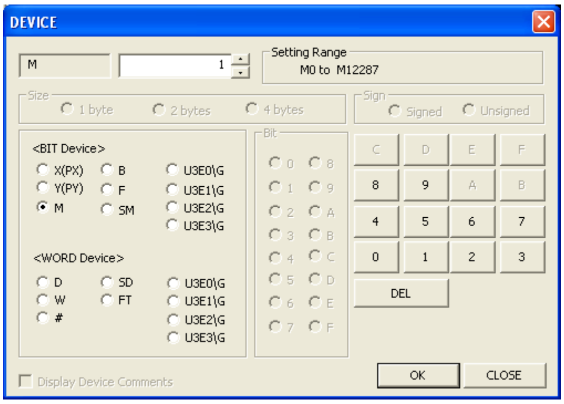
(1) Select the item to be probed.
Click [Probe] on the [Edit] menu at the Digital Oscilloscope window.



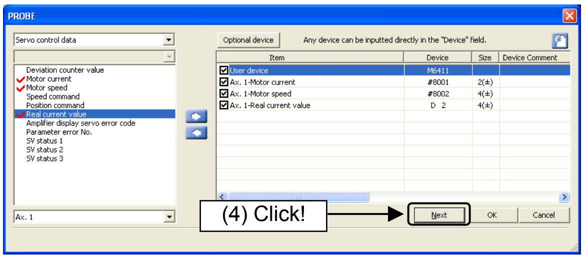
(2) Click the Optional device button at the PROBE screen that appears.

Go to next page

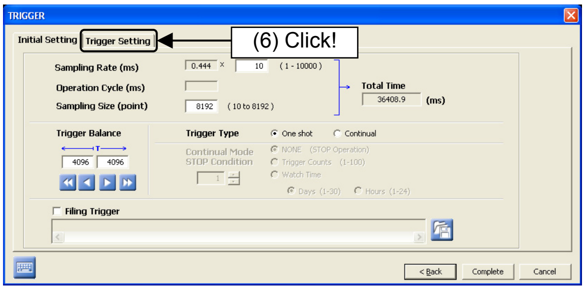
From previous page



(3) Select the check box and use the ten-key pad to enter "M1" at the DEVICE screen, and then click the OK button.

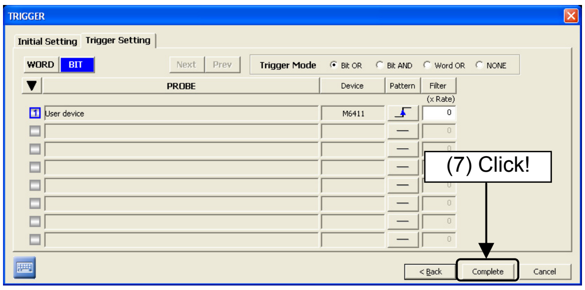


(4) The display then returns to the PROBE screen. Select the item to be set, and then click [arrow button] to register.
Register the "Motor current", "Motor speed", and "Feed current value" here.
Click the Next button.



(5) Set the trigger at the TRIGGER screen that appears.
Specify the default settings as follows.
- Sampling Rate: 0.444 x 10 (msec)
- Sampling Size: 8192
- Trigger Type: Select "One shot".

(6) Click the "Trigger Setting" tab.

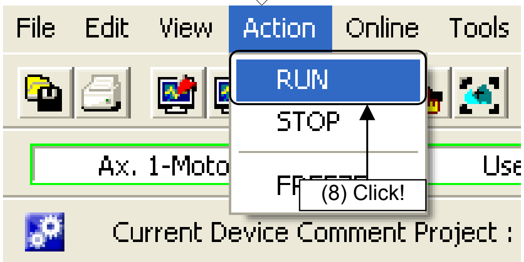


(7) Specify the trigger settings as follows.
- Trigger Mode: Bit OR
- Pattern: [pattern icon] (OFF→ON (startup))

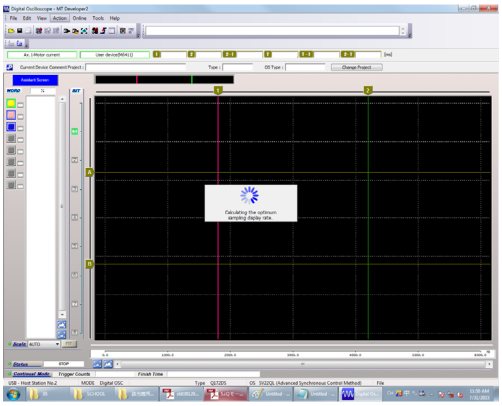
Click the Complete button.

Go to next page

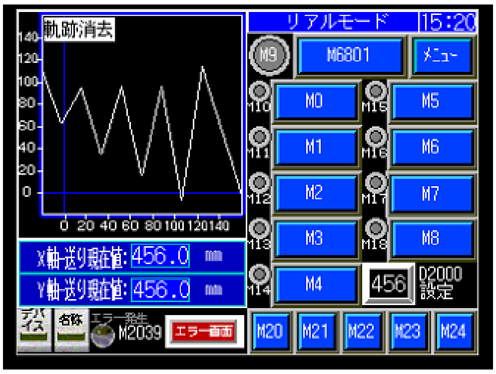
From previous page



(8) Click [RUN] on the [Action] menu at the Digital Oscilloscope window.
Sampling is started.

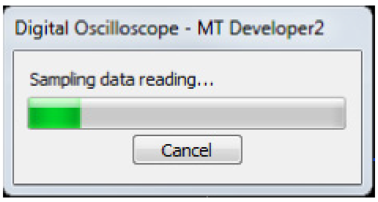


(9) The system waits for the trigger, and "Sampling before trigger" appears in the display area MAP.

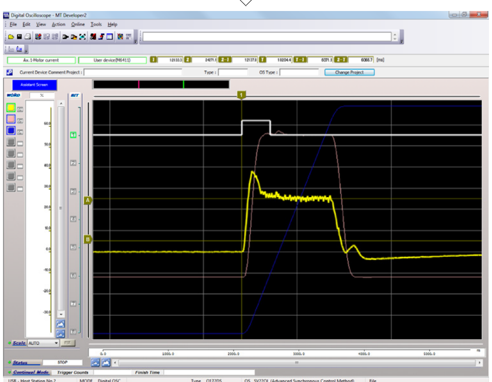


(10) Press Standby Point at the demonstration machine operation panel to perform positioning to the standby point.

(11) Align the digital switch to "30" and press Position Movement to perform point positioning.
The trace monitor is executed.



(12) Once buffering is complete when the trigger is established, a buffering data read progress bar is displayed.



(13) The waveform displays once buffering data reading is complete.

Appendix 3 Windows Computer Operation

Appendix 3.1 MELSOFT MT Works2 Installation Procedure

This section describes the installation and uninstallation procedures for MT Developer2.

■ Product configuration

Model name	Software name	Qty
SW1DNC-MTW2-J (Japanese edition package)	MELSOFT MT Works2 (MT Developer2) (1 licensed product) CD-ROM	1
	Installation procedure manual	1
	Software usage agreement	1
	Software registration guidance	1
	License agreement	1
	Information	1

■ Operating environment

Item		Software name
Computer	Computer	Personal computer running Windows®
	OS	Microsoft® Windows® 7 Starter [no Service Pack/1]*2 Microsoft® Windows® 7 Home Premium [no Service Pack/1]*2*3 Microsoft® Windows® 7 Professional [no Service Pack/1]*2*3 Microsoft® Windows® 7 Ultimate [no Service Pack/1]*2*3 Microsoft® Windows® 7 Enterprise [no Service Pack/1]*2*3 Microsoft® Windows Vista® Home Basic [no Service Pack/1]*2 Microsoft® Windows Vista® Home Premium [no Service Pack/1]*2 Microsoft® Windows Vista® Business [no Service Pack/1]*2 Microsoft® Windows Vista® Ultimate [no Service Pack/1]*2 Microsoft® Windows Vista® Enterprise [no Service Pack/1]*2 Microsoft® Windows® XP Professional [Service Pack 2/3] Microsoft® Windows® XP Home Edition [Service Pack 2/3] Microsoft® Windows® 2000 Professional [Service Pack 4]
	CPU	Desktop computer: Intel® Celeron® processor 2.8GHz or faster recommended Notebook computer: Intel® Pentium® M processor 1.7GHz or faster recommended
	Required memory	1 GB or more recommended
	Video card	Video card compatible with Microsoft® DirectX® 9.0c or later
Available hard disk space		When installing MT Developer2: available HDD space of 1 GB or more When running MT Developer2: available virtual memory of 512 MB or more
Disk drive		3.5 inch (1.44 MB) floppy disk drive *1 CD-ROM compatible disk drive
Display		Resolution: 1024 x 768 or higher
Communication interface		RS-232 port USB port SSC I/F card (A30CD-PCF) *4 SSC I/F board (A10BD-PCF) *4 Ethernet port

*1: Required if installing this OS software with a floppy disk.
*2: Compatible with 32-bit edition.
*3: Compatible with 64-bit edition.
*4: A30CD-PCF and A10BD-PCF are not compatible with the 64-bit edition of Windows® 7.

(1) Installation procedure
Install MT Developer2 on the computer.

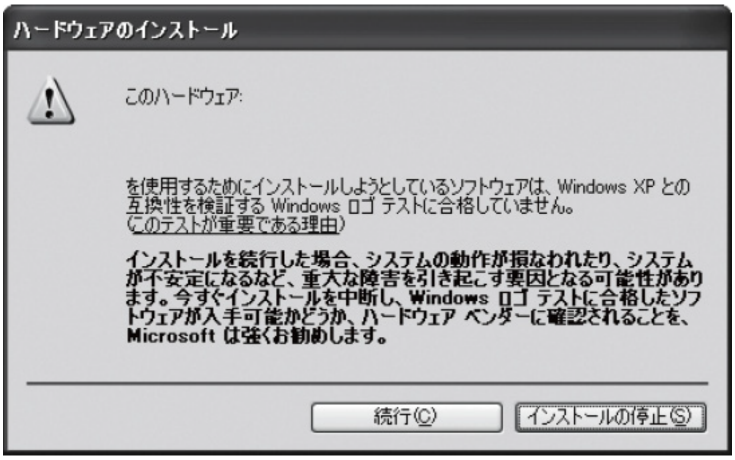
1) Insert the CD-ROM in the CD-ROM drive. Double-click "Setup.exe" (may also appear as "Setup") in the CD-ROM.

2) Follow the screen instructions to select or enter the required items.

CAUTION

The following message may appear before the product installation is complete.
The driver must be installed.

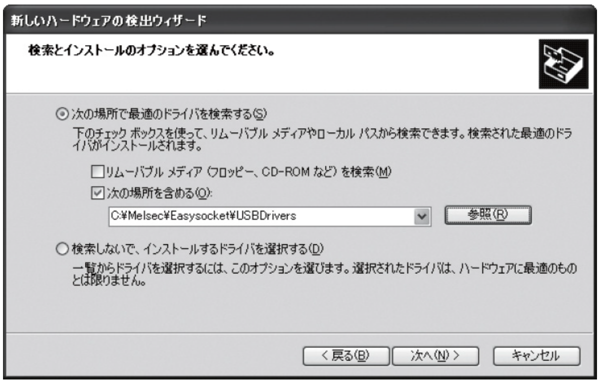
When using Windows® XP



Select "Continue" and install the driver.

(2) USB driver installation procedure
It is necessary to install a USB driver to perform USB communication with the motion CPU.

(When using Windows® XP)



1) Connect the computer and PLC CPU, and then turn ON the PLC CPU.

2) A "New hardware search wizard start" dialog box appears. Select "Install from list or specific location (details)".

3) A "Select search and install options" dialog box appears. Select "Search for best driver in following location". Select the "Include following location" check box, and then set "Easysocket USBdrivers" in the folder in which MT Developer2 was installed. If multiple MELSOFT products have been installed, browse the installation direction for the first product. ("\Melsec\Easysocket\USBDrivers" or "\[installation folder specified when installing]\Easysocket\USBDrivers")

POINT

If unable to install the USB driver, check the following settings.

- If "Block - Do not install unsigned driver software" is selected in [Control Panel] - [System] - [Hardware] - [Driver Signature], it may not be possible to install the USB driver.
- Select "Ignore - Install software and do not check", or "Warning - Select operation each time", and then install the USB driver.

Appendix 4 Q173DCPU and Q172DCPU Comparison

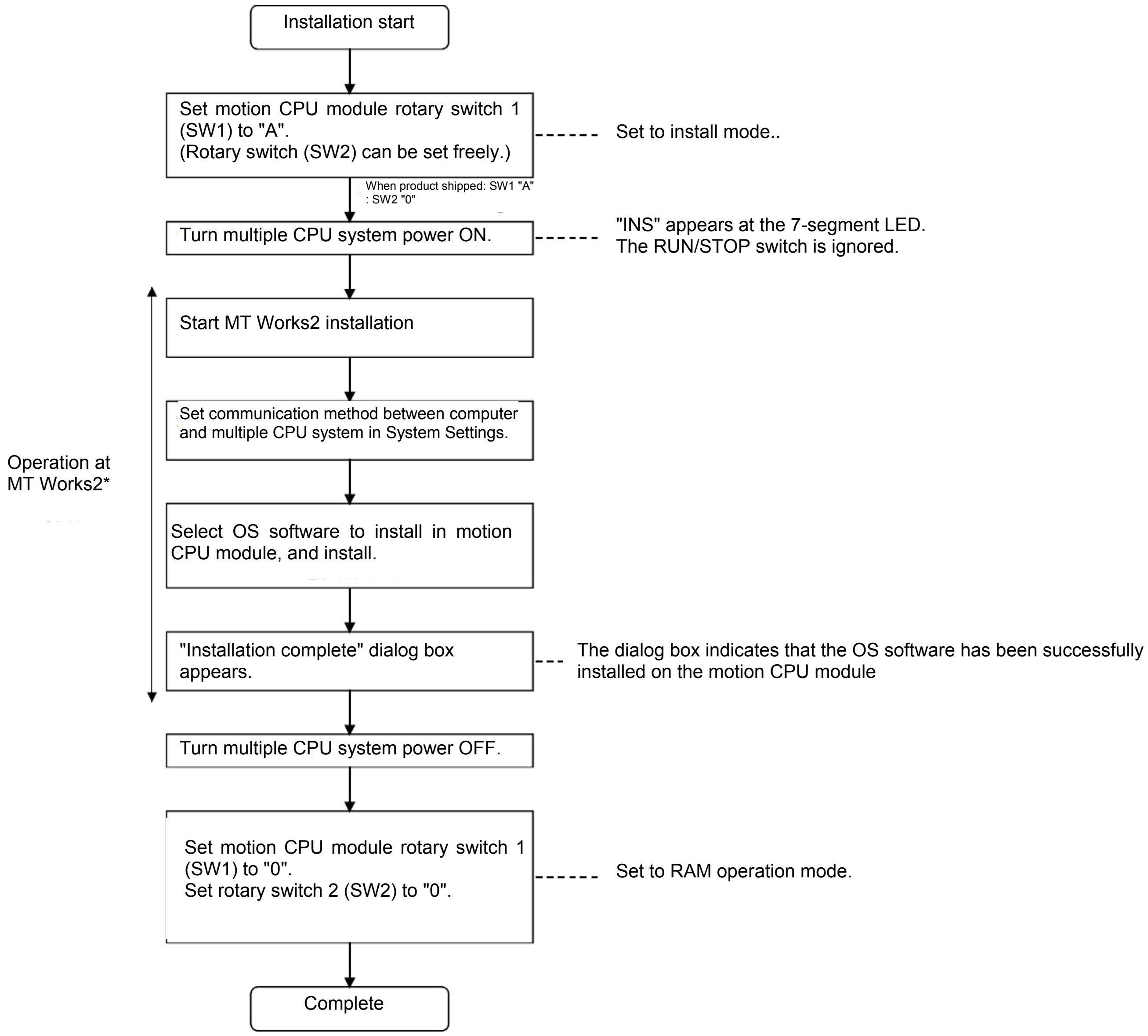
Item		Q173DSCPU	Q172DSCPU	Q173DCPU	Q172DCPU
Number of control axes		32 axes	16 axes	32 axes	8 axes
Operation cycle (default)	SV13	0. 22 ms/1 to 4 axes 0. 44 ms/5 to 10 axes 0. 88 ms/11 to 24 axes 1. 77 ms/25 to 32 axes	0. 22 ms/1 to 4 axes 0. 44 ms/5 to 10 axes 0. 88 ms/11 to 16 axes	0. 44 ms/1 to 6 axes 0. 88 ms/7 to 8 axes	0. 44 ms/1 to 3 axes 0. 88 ms/4 to 8 axes
	SV22	0. 44 ms/1 to 6 axes 0. 88 ms/7 to 16 axes 1. 77 ms/17 to 32 axes	0. 44 ms/1 to 6 axes 0. 88 ms/7 to 16 axes	0. 44 ms/1 to 4 axes 0. 88 ms/5 to 8 axes	0. 88 ms/1 to 5 axes 1.77 ms/6 to 8 axes
OS software media		CD-ROM		FD (2)	
OS software model (SV13/SV22)		SW□DNC-SV□Q□			
Peripheral device I/F		USB/RS-232/Ethernet (via PLC CPU), peripheral I/F (motion CPU control)			
Main base unit		Main base unit (Q35B/Q38B/Q312B)			
	Attachment to panel	Tightening with unit securing screw			
	DIN rail	Unusable			
No.1 CPU module		Universal model (QnUD(E)(H)CPU)			
Attachment order for CPU modules from No.2		No restriction			
Empty CPU slots		Can be set between CPU modules.			
PLC CPU module		Universal model (QnUD(E)(H)CPU)			
Motion CPU module		Q173DSCPU/Q172DSCPU		Q173DCPU(-S1)/Q172DCPU(-S1)	
	Motion CPU module combination	Combination with Q173DCPU(-S1)/Q172DCPU(-S1) possible		Combination with Q173DSCPU/Q172DSCPU possible	
	Attachment on main base unit	Tightening with motion CPU module securing screw			
	Function selection switch	Rotary switch 1, rotary switch 2			
	RESET/L.CLR switch	None			
	LED display	7-segment LED status display			
External battery		Add Q6BAT if continuous power outage continues for 1 month or more.			
	Battery holder unit	Required			
External forced stop input		• Uses motion CPU module EMI terminal. • Uses device specified with external forced stop input in system settings.			
External forced stop input cable		Required			
Motion module		Q172DLX/Q172DEX/Q173DPX/Q173DSXY*1			
	Attachment on main base unit	Q172DLX/Q172DEX/Q173DPX: Installation not possible in I/O slots 0 to 2.			
Multiple CPU high speed transmission memory for CPU high speed transfer		Equipped			
Devices	Number of internal relays (M)	12288			
Devices	No. of latch relays (L)	None (M latch possible with latch setting)			
Devices	Number of special relays (M)	-			
Devices	Number of special relays (SM)	2256			
Devices	Number of special registers (D)	-			
Devices	Number of special registers (SD)	2256			
Devices	Number of motion registers (#)	12288			
Devices	Multiple CPU shared devices (U□\G)	Max. 14336			
Dedicated motion sequence commands	SV13/SV22	D(P).DDRD, D(P).DDWR, D(P).SFCS, D(P).SVST, D(P).CHGT, D(P).CHGT2, D(P).CHGV, D(P).CHGA, D(P).GINT		D(P).DDRD, D(P).DDWR, D(P).SFCS, D(P).SVST, D(P).CHGT, D(P).CHGV, D(P).CHGA, D(P).GINT	
	SV43	-		D(P).DDRD, D(P).DDWR, D(P).SFCS, D(P).SVST, D(P).CHGT, D(P).CHGV, D(P).CHGA	
	Interlock conditions	Multiple commands can be executed in succession with no interlock conditions due to high speed interrupt accept flag from CPU□ to self CPU. □: CPU No.			

Item		Q173DSCPU	Q172DSCPU	Q173DCPU	Q172DCPU
Motion module	SV13	Q172DLX, Q173DPX, Q173DSXY		Q172DLX, Q173DPX, Q173DSXY*1	
	SV22	Q172DLX, Q172DEX, Q173DPX, Q173DSXY		Q172DLX, Q172DEX, Q173DPX, Q173DSXY*1	
	SV43	-		Q172DLX, Q173DPX	
System settings		• QnUD(E)(H)CPU is No.1 CPU. • The only main base units that can be used are multiple CPU high speed main base units (Q35DB/Q38DB/Q312DB). • Q172DLX, Q172DEX, Q173DPX cannot be installed in I/O slots 0 to 2.			
Latch clear		Remote operation			
RUN/STOP		Remote operation, RUN/STOP switch			
Boot operation from ROM		• ROM writing is performed in RAM operation mode/ROM operation mode. • MT Developer□ data can be written to the ROM.			
CPU shared memory	Multiple CPU high speed transmission area	Yes			
	Access with multiple CPU shared memory	Possible			
Auto refresh	Shared memory	CPU shared memory multiple CPU high speed transmission area			
	Auto refresh settings	32 range setting possible			
	Multiple CPU high speed refresh function	Yes			
Latch range setting	Latch (1)	Clearing possible with remote latch clear latch clear (1), latch clear (1) (2).			
	Latch (2)	Clearing possible with remote latch clear latch clear (1) (2).			
All clear function		Executed in install mode.			
Multiple CPU related error clearance		Turn M2039 OFF.			

*1: Q173DCPU-S1/Q172DCPU-S1 only

Appendix 5 OS Software Installation Procedure

It is necessary to install OS software for the motion CPU module using MT Works2.
The installation procedure is as follows.



*: Follow the MT Works2 screen instructions to install.
Refer to MT Works2 Help for details.

POINT

(1) The motion CPU module product condition is as follows when shipped.
- Q173DSCPU/Q172DSCPU
  The OS software (SV22) has already been installed when the product is shipped.
  The latest OS software can be downloaded from MELFANSweb.
- Q173DCPU(-S1)/Q172DCPU(-S1)
  The OS software has not been installed when the product is shipped.
  It is necessary to install the OS software before starting the system.

(2) If changing the rotary switch setting, always turn the power OFF beforehand.

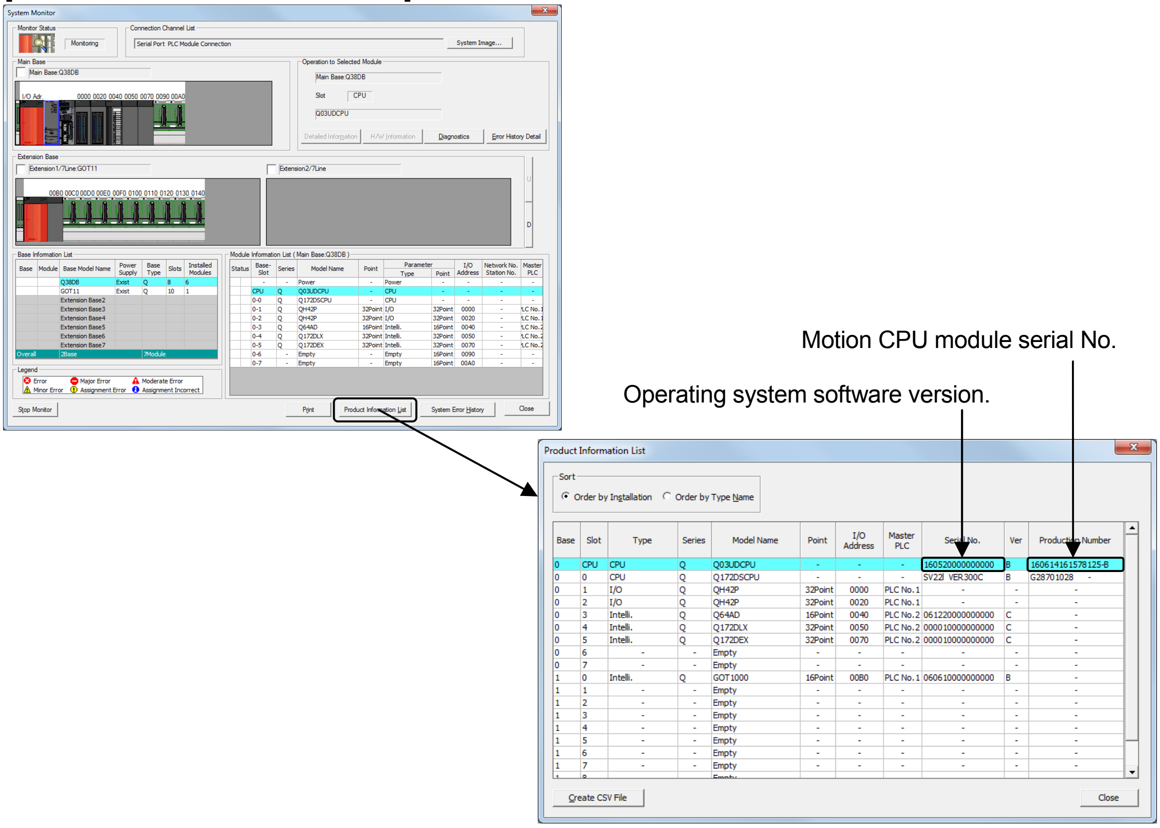
(3) Even if the software is installed, programs, parameters, and absolute position data written to the motion CPU module is not rewritten.
However, if using the software security key function with the Q173DSCPU/Q172DSCPU, if the software security key embedded in the OS software already installed in the motion CPU differs from that in the OS software to be installed, an all clear is performed at the same time as the installation. A backup using MT Developer2 is recommended beforehand.

(4) Do not perform the following during installation. The motion CPU module may malfunction.
- Do not turn the multiple CPU system power OFF.
- Do not set the PLC CPU module "RUN/STOP/RESET" switch to "RESET".
- Do not turn the computer power OFF.
- Do not disconnect the cable connected to the computer.

(5) If installing multiple motion CPU modules on which the OS software has not been installed on the same base unit, and then installing the OS software, set all motion CPU modules on which the software has not been installed to install mode (set rotary switch 1 (SW1) to "A"), and then perform the installation.
Note: If the power is turned ON for motion CPU modules on which the OS software has not been installed, the system will not function normally.
It is necessary to install the OS software before starting the system.

OS software version check

The motion CPU OS software version can be checked at the GX Works2 system monitor.
Click [Diagnosis] - [System Monitor] at GX Works2 to display the System Monitor screen, and then click the [Product information list device] button.



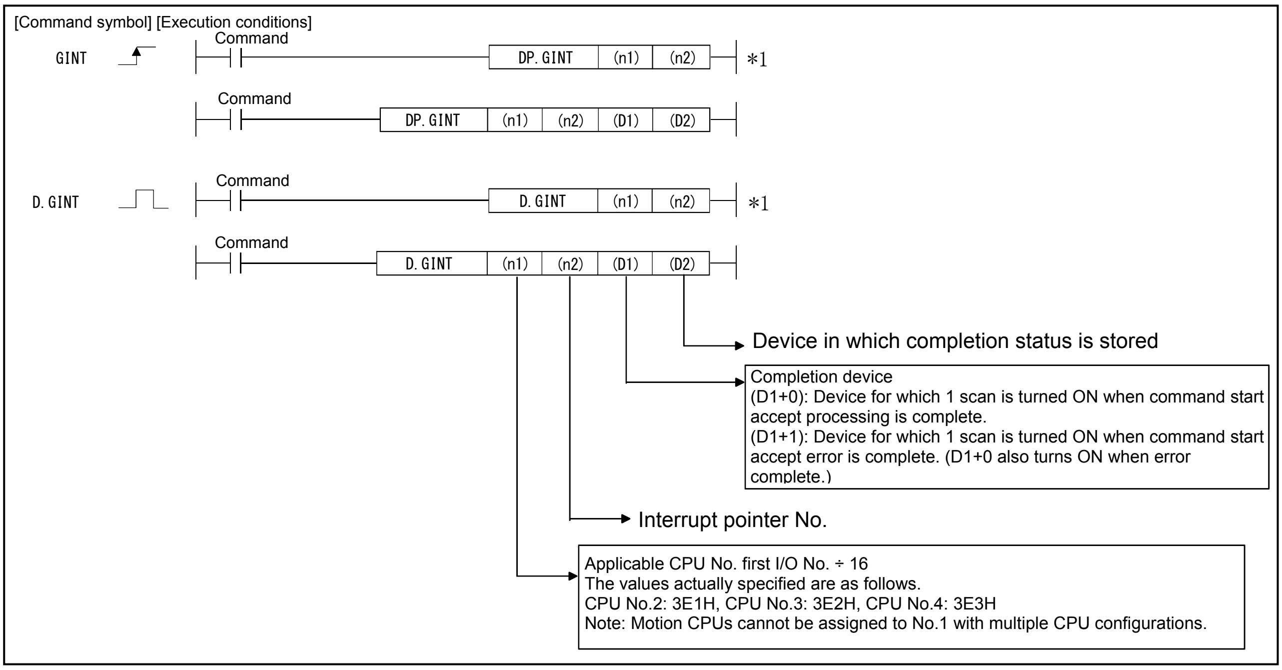
Remarks
The "Motion CPU module serial No." and "OS software version" displayed at the GX Works2 system monitor (Product information list device) applies to those motion CPU modules manufactured from the beginning of October, 2007.

Appendix 6 Dedicated Motion Sequence Commands

This section describes SFCS commands and GINT commands used to issue servo program start requests, DDRD commands, and DDWR commands.

Appendix 6.1 GINT Interrupt Commands to Other CPUs

This is a command used to trigger an interrupt for Q motion CPUs.



*1: This command can be omitted if both (D1) and (D2) are omitted.

(1) GINT command program example

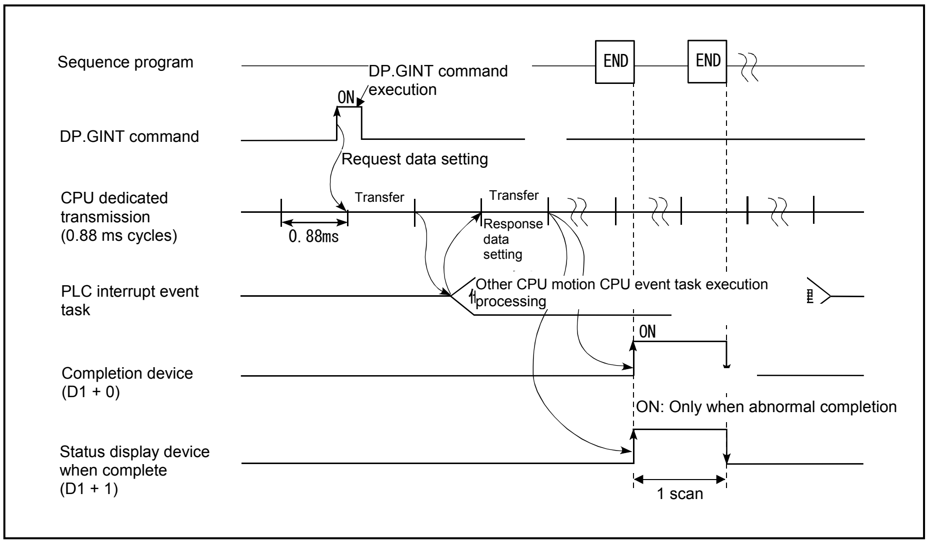
This program used to trigger an interrupt pointer No.10 interrupt for the motion CPU (No.2) when M0 turns ON.

<Example 1> Program if completion device, completion status omitted

<Example 2> Program if completion device, completion status used

(2) **Execution timing**

The following is an overview of operation between CPUs when executing the DP.GINT command.



(3) Error content

In the following cases, an abnormal termination occurs, and an error code is stored in the device specified at the completion status storage device (D2).

If the completion status storage device (D2) is omitted, no error is detected and processing is not performed, and therefore caution is advised.

Completion status * (Error code) (H)	Error cause	Remedy
0010	The command request from the PLC CPU to the motion CPU exceeds the permissible value.	Check the program, and then change to the correct sequence program.
2082	The interrupt pointer No. set with the D(P).GINT command lies outside the 0 to 15 range.	
2100	The number of command requests issued from the PLC CPU to the motion CPU simultaneously is 33 or more with D(P).GINT, and therefore the motion CPU is unable to process.	

*: 0000H (normal)

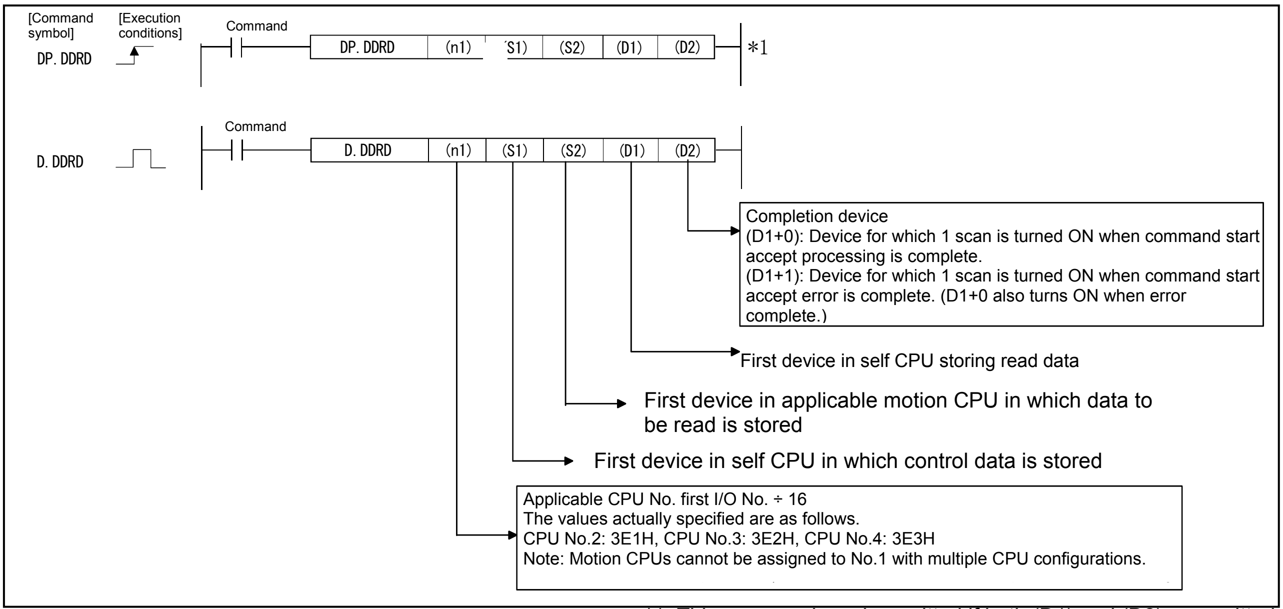
In the following cases, an operation error occurs, the diagnostic error flag (SM0) turns ON, and the error code is stored in the diagnostic error register (SD0).

Error code *	Error cause	Remedy
4350	The specified applicable CPU module is incorrect. (1) A reserved CPU No. was specified. (2) An uninstalled CPU No. was specified. (3) The applicable CPU module first No. I/O No. ÷ 16(n1) lies outside the 3E0H to 3E3H range.	Check the program, and then change to the correct sequence program.
4351	Cannot be executed at the specified applicable CPU module. (1) The command name is incorrect. (2) An unsupported command was specified at the applicable CPU module.	
4352	The number of specified command devices is incorrect.	
4353	A device that cannot be used with the specified command has been specified.	

*: 0 (normal)

Appendix 6.2 Read Command from DDRD Q Motion CPU Device

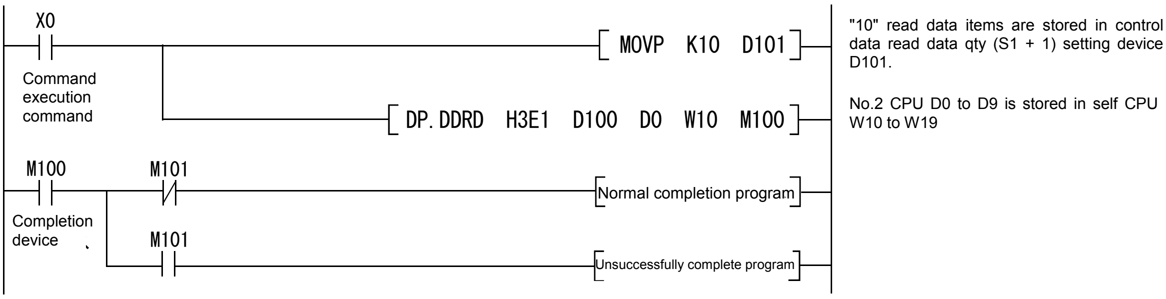
This command is used to read device data inside the Q motion CPU directly from the Q PLC CPU.



*1: This command can be omitted if both (D1) and (D2) are omitted.

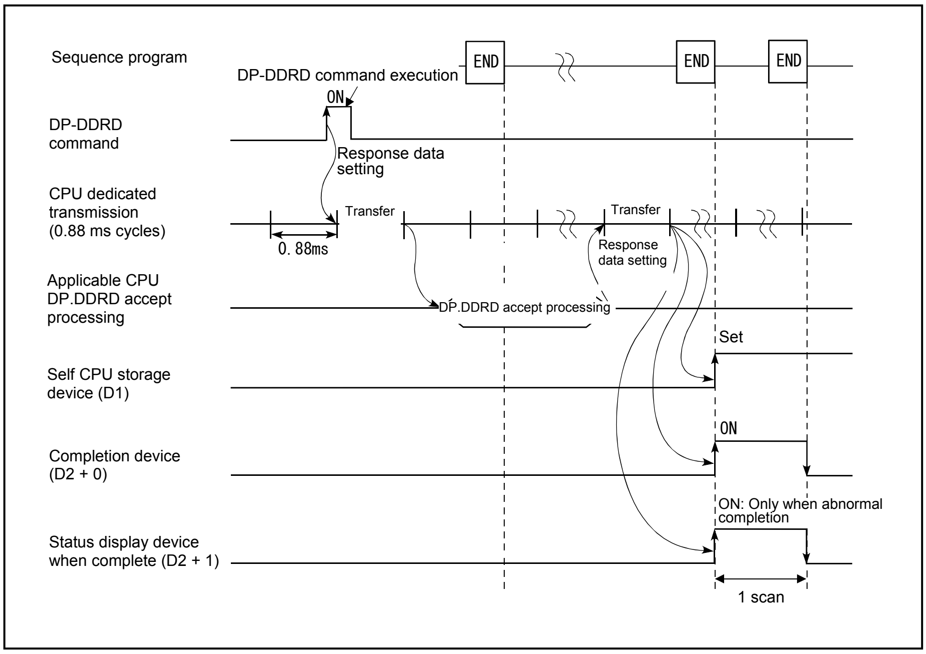
(1) DDRD command program example

This program is used to store a 10 word piece of data from the No.2 CPU D0 to the self CPU W10 and onward when X0 is ON.



(2) Execution timing

The following is an overview of operation between CPUs when executing the DP.DDRD command.



(3) **Error content**

In the following cases, an abnormal termination occurs, and an error code is stored in the device specified at the completion status storage device (S1 + 0).

Completion status * (Error code) (H)	Error cause	Remedy
0010	The command request from the PLC CPU to the motion CPU exceeds the permissible value.	Check the program, and then change to the correct sequence program.
2001	The specified device cannot be used with the motion CPU. Or it lies outside the device range.	
2081	The number of read data items set with the D(P).DDRD command is illegal.	
2100	The number of command (D(P).DDRD/D(P).DDWR combined) requests issued from the PLC CPU to the motion CPU simultaneously is 65 or more, and therefore the motion CPU is unable to process.	

*: 0000H (normal)

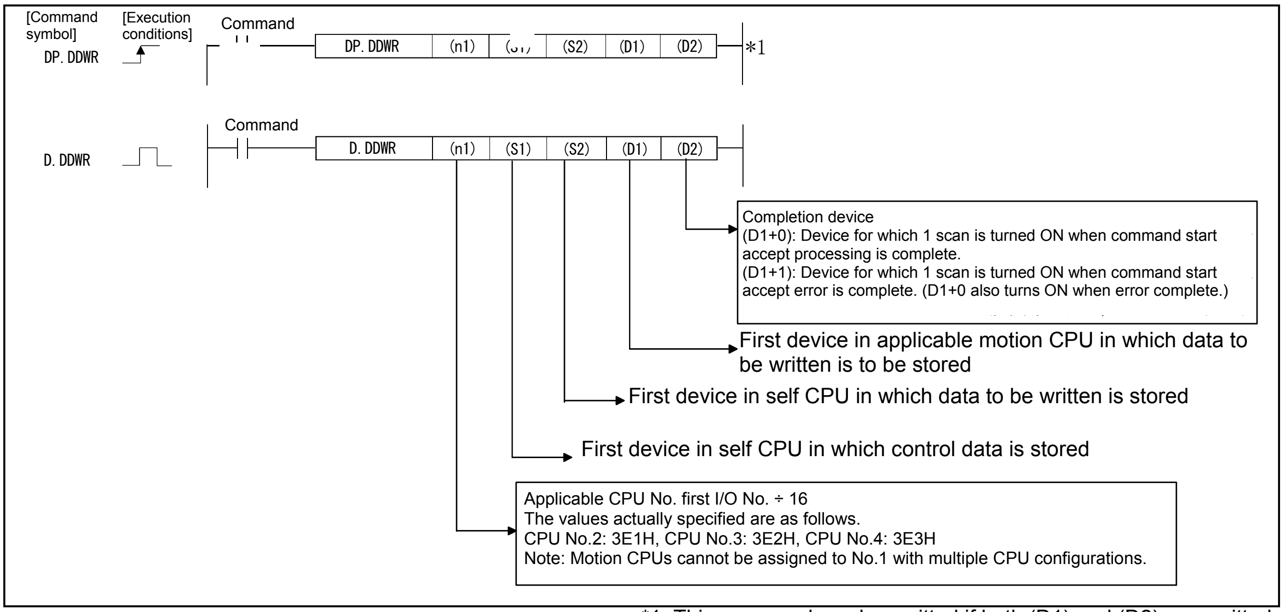
In the following cases, an operation error occurs, the diagnostic error flag (SM0) turns ON, and the error code is stored in the diagnostic error register (SD0).

Error code *	Error cause	Remedy
4101	The number of read data items lies outside the read data storage device range.	Check the program, and then change to the correct sequence program.
4350	The specified applicable CPU module is incorrect. (1) A reserved CPU No. was specified. (2) An uninstalled CPU No. was specified. (3) The applicable CPU module first No. I/O No. ÷ 16(n1) lies outside the 3E0H to 3E3H range.	
4351	Cannot be executed at the specified applicable CPU module. (1) The command name is incorrect. (2) An unsupported command was specified at the applicable CPU module.	
4352	The number of specified command devices is incorrect.	
4353	A device that cannot be used with the specified command has been specified.	
4354	A character string that cannot be handled with the specified command has been specified.	
4355	The number of read data items lies outside the 1 to 20 range.	

*: 0 (normal)

Appendix 6.3 Read Command from DDWR Q Motion CPU Device

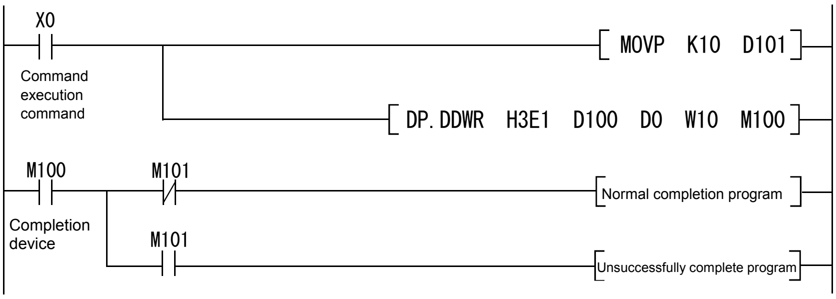
This command is used to write device data inside the Q motion CPU directly from the Q PLC CPU.



*1: This command can be omitted if both (D1) and (D2) are omitted.

(1) DDWR command program example

This program is used to store a 10 word piece of data from the self CPU D0 to the No.2 CPU W10 and onward when X0 is ON.

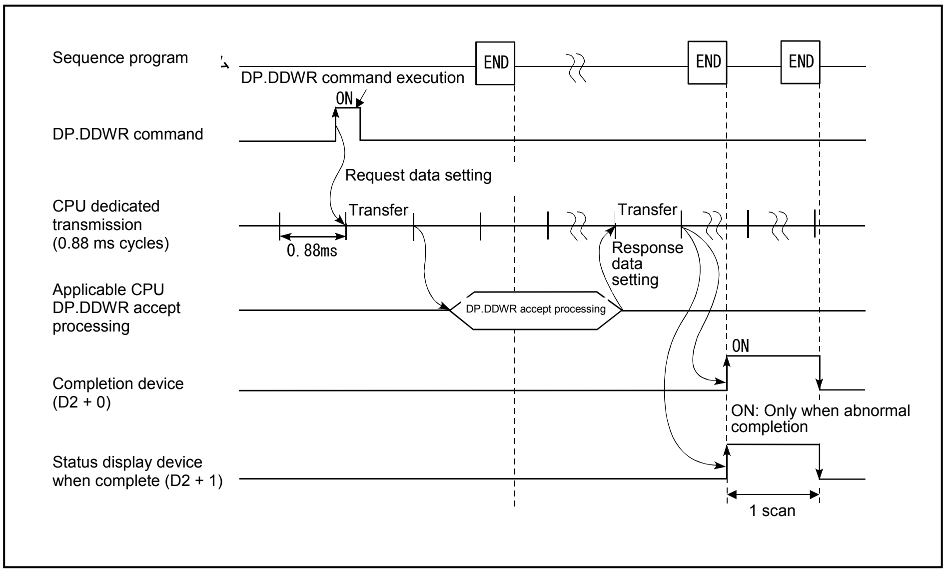


"10" write data items are stored in control data write data qty (S1 + 1) setting device D101.

Self CPU D0 to D9 is stored in No.2 CPU W10 to W19.

(2) Execution timing

The following is an overview of operation between CPUs when executing the DP.DDWR command.



(3) Error content

In the following cases, an abnormal termination occurs, and an error code is stored in the device specified at the completion status storage device (S1 + 0).

Completion status * (Error code) (H)	Error cause	Remedy
0010	The command request from the PLC CPU to the motion CPU exceeds the permissible value.	Check the program, and then change to the correct sequence program.
2001	The specified device cannot be used with the motion CPU. Or it lies outside the device range.	
2080	The number of write data items set with the D(P).DDWR command is illegal.	
2100	The number of command (D(P).DDRD/D(P).DDWR combined) requests issued from the PLC CPU to the motion CPU simultaneously is 65 or more, and therefore the motion CPU is unable to process.	

*: 0000H (normal)

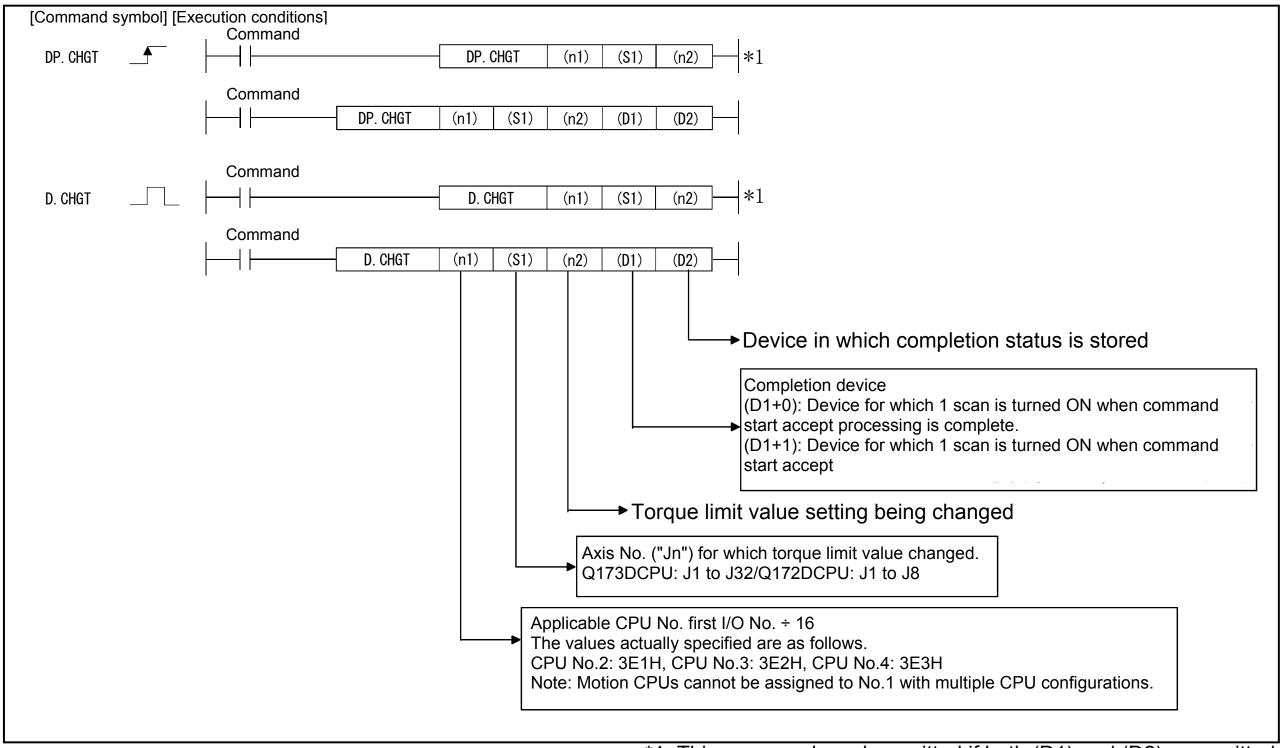
In the following cases, an operation error occurs, the diagnostic error flag (SM0) turns ON, and the error code is stored in the diagnostic error register (SD0).

Error code *	Error cause	Remedy
4101	The number of write data items lies outside the write data storage device range.	Check the program, and then change to the correct sequence program.
4350	The specified applicable CPU module is incorrect. (1) A reserved CPU No. was specified. (2) An uninstalled CPU No. was specified. (3) The applicable CPU module first No. I/O No. ÷ 16(n1) lies outside the 3E0H to 3E3H range.	
4351	Cannot be executed at the specified applicable CPU module. (1) The command name is incorrect. (2) An unsupported command was specified at the applicable CPU module.	
4352	The number of specified command devices is incorrect.	
4353	A device that cannot be used with the specified command has been specified.	
4354	A character string that cannot be handled with the specified command has been specified.	
4355	The number of write data items lies outside the 1 to 20 range.	

*: 0: Normal

Appendix 6.4 CHGT Torque Limit Value Change Request Command

This command is used to change the torque limit values when in real mode, regardless of whether the machine is running or is stopped.

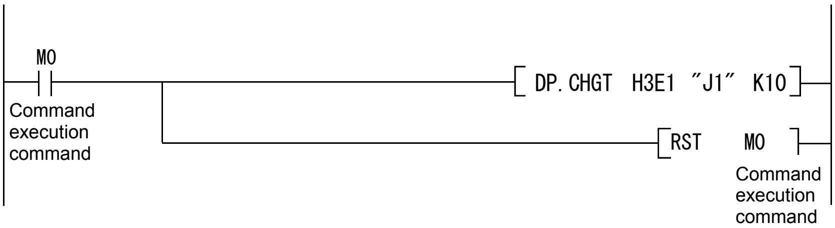


*1: This command can be omitted if both (D1) and (D2) are omitted.

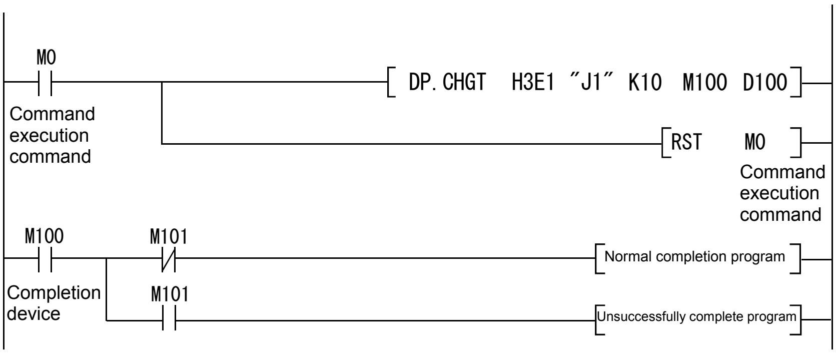
(1) CHGT command program example

This program is used to change the motion CPU (No.2) axis 1 torque limit value to 10% when M0 turns ON.

<Example 1> Program if completion device, completion status omitted

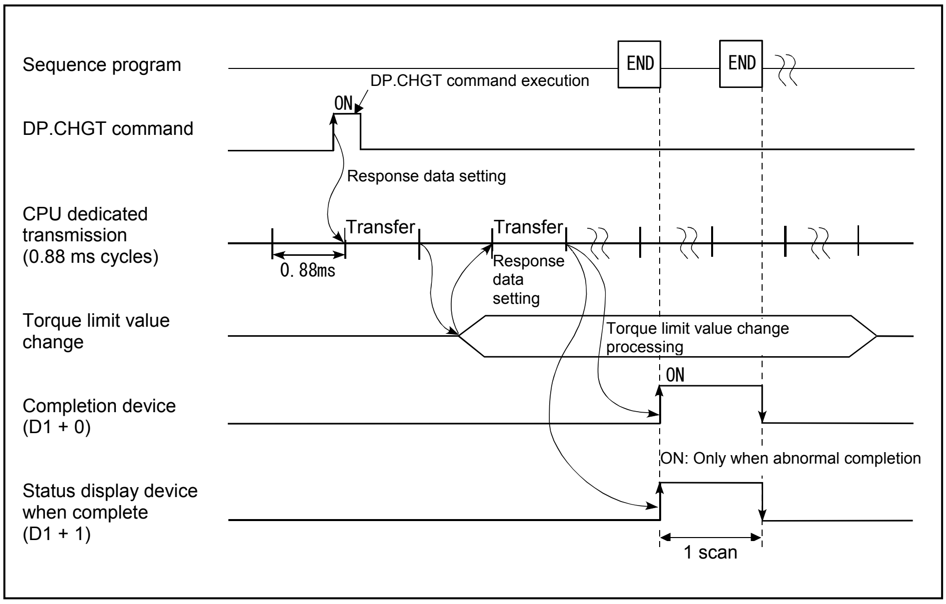


<Example 2> Program if completion device, completion status used



(2) Execution timing

The following is an overview of operation between CPUs when executing the DP.CHGT command.



(3) Error content

In the following cases, an abnormal termination occurs, and an error code is stored in the device specified at the completion status storage device (D2).
If the completion status storage device (D2) is omitted, no error is detected and processing is not performed, and therefore caution is advised.

Completion status * (Error code) (H)	Error cause	Remedy
0010	The command request from the PLC CPU to the motion CPU exceeds the permissible value.	Check the program, and then change to the correct sequence program.
2205	The axis No. specified with the D(P).CHGT command is illegal.	

*: 0000H (normal)

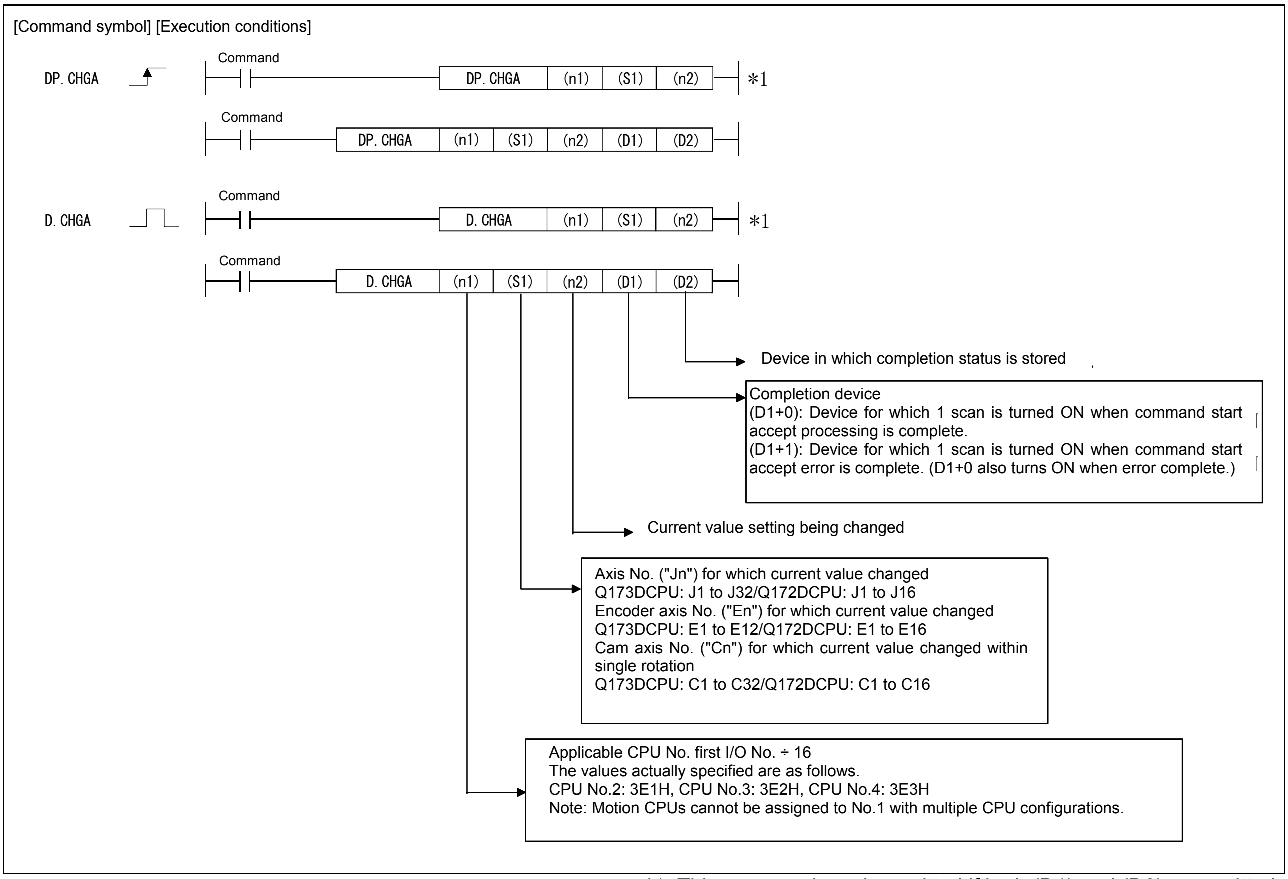
In the following cases, an operation error occurs, the diagnostic error flag (SM0) turns ON, and the error code is stored in the diagnostic error register (SD0).

Error code *	Error cause	Remedy
4350	The specified applicable CPU module is incorrect. (1) A reserved CPU No. was specified. (2) An uninstalled CPU No. was specified. (3) The applicable CPU module first No. I/O No. ÷ 16(n1) lies outside the 3E0H to 3E3H range.	Check the program, and then change to the correct sequence program.
4351	Cannot be executed at the specified applicable CPU module. (1) The command name is incorrect. (2) An unsupported command was specified at the applicable CPU module.	
4352	The number of specified command devices is incorrect.	
4353	A device that cannot be used with the specified command has been specified.	
4354	A character string that cannot be handled with the specified command has been specified.	

*: 0 (normal)

Appendix 6.5 CHGA Current Value Change Command

This command is used to change the current value of the stopped axis.

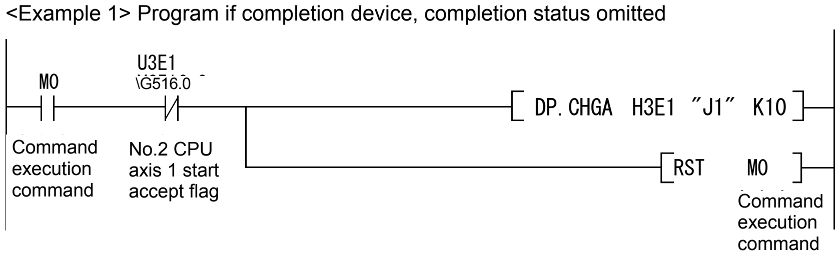


*1: This command can be omitted if both (D1) and (D2) are omitted.

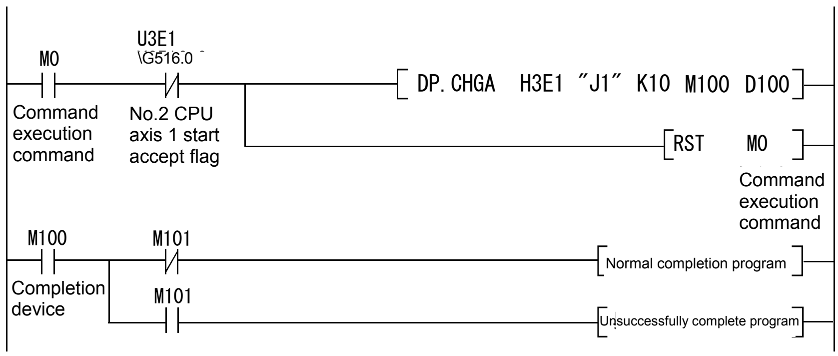
(1) CHGA command program example

This program is used to change the motion CPU (No.2) axis 1 current value to 10 when M0 turns ON.

<Example 1> Program if completion device, completion status omitted

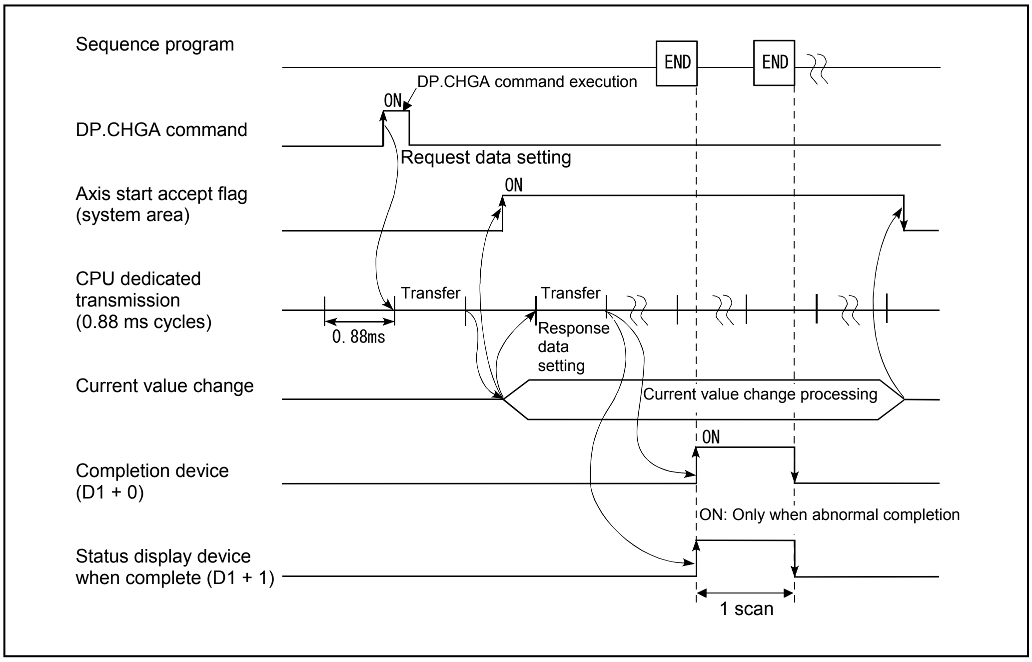


<Example 2> Program if completion device, completion status used



(2) Execution timing

The current value for the specified axis is changed when the CHGA command execution command turns ON.



(3) **Error content**

In the following cases, an abnormal termination occurs, and an error code is stored in the device specified at the completion status storage device (D2).

If the completion status storage device (D2) is omitted, no error is detected and processing is not performed, and therefore caution is advised.

Completion status * (Error code) (H)	Error cause	Remedy
0010	The command request from the PLC CPU to the motion CPU exceeds the permissible value.	Check the program, and then change to the correct sequence program.
2100	The number of command (D(P).SVST/D(P).CHGA combined) requests issued from the PLC CPU to the motion CPU simultaneously is 65 or more, and therefore the motion CPU is unable to process.	
2203	The axis No. specified with the D(P).CHGA command is illegal.	

*: 0000H (normal)

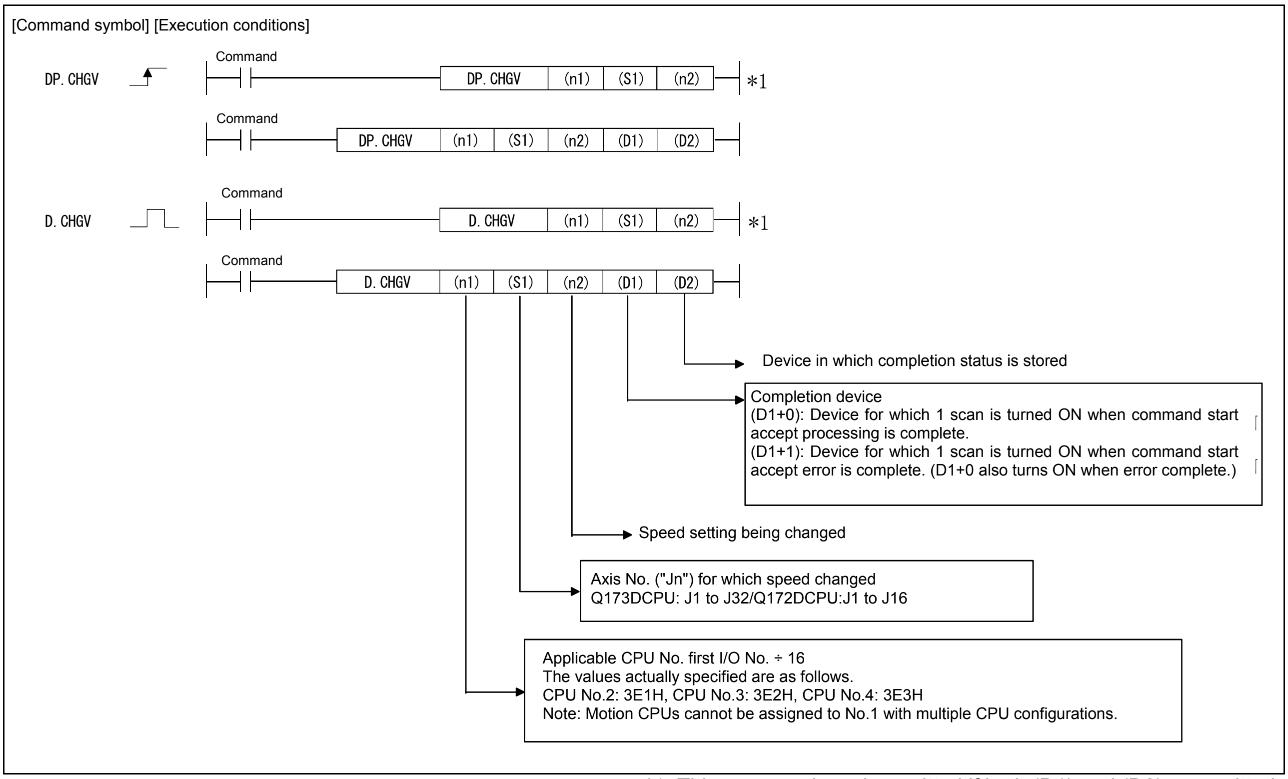
In the following cases, an operation error occurs, the diagnostic error flag (SM0) turns ON, and the error code is stored in the diagnostic error register (SD0).

Error code *	Error cause	Remedy
4350	The specified applicable CPU module is incorrect. (1) A reserved CPU No. was specified. (2) An uninstalled CPU No. was specified. (3) The applicable CPU module first No. I/O No. ÷ 16(n1) lies outside the 3E0H to 3E3H range.	Check the program, and then change to the correct sequence program.
4351	Cannot be executed at the specified applicable CPU module. (1) The command name is incorrect. (2) An unsupported command was specified at the applicable CPU module.	
4352	The number of specified command devices is incorrect.	
4353	A device that cannot be used with the specified command has been specified.	
4354	A character string that cannot be handled with the specified command has been specified.	

*: 0 (normal)

Appendix 6.6 CHGV Speed Change Command

This command is used to change the speed during positioning or during JOG operation.

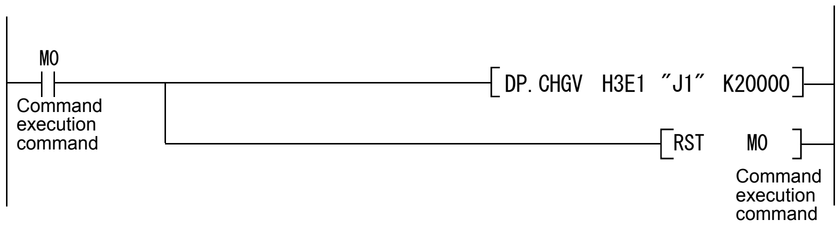


*1: This command can be omitted if both (D1) and (D2) are omitted.

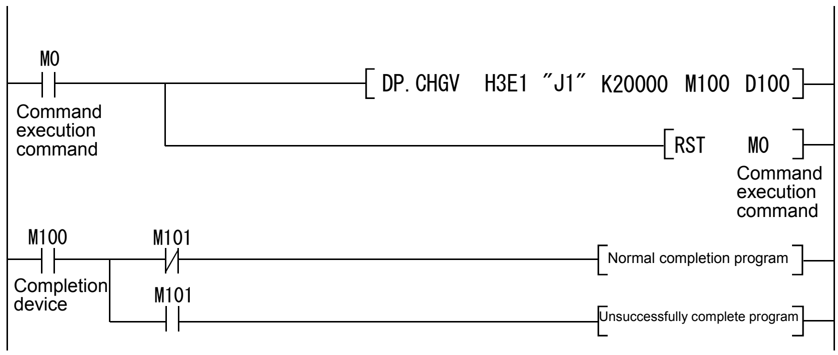
(1) CHGA command program example

This program is used to change the motion CPU (No.2) axis 1 positioning speed to 20000 when M0 turns ON.

<Example 1> Program if completion device, completion status omitted

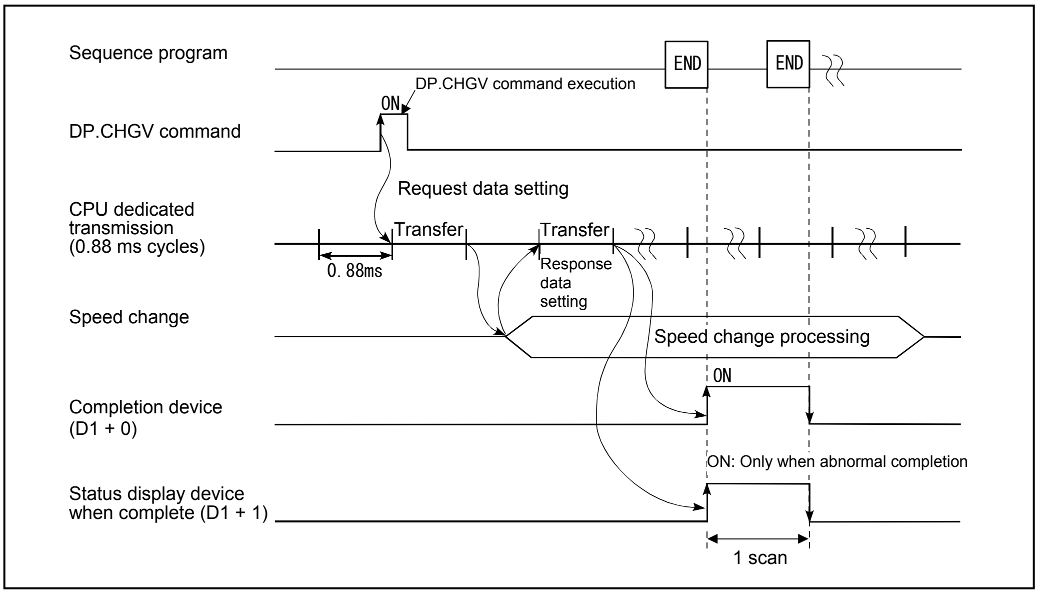


<Example 2> Program if completion device, completion status used



(2) Execution timing

The speed for the specified axis is changed when the CHGV command execution command turns ON.



(3) Error content

In the following cases, an abnormal termination occurs, and an error code is stored in the device specified at the completion status storage device (D2).
If the completion status storage device (D2) is omitted, no error is detected and processing is not performed, and therefore caution is advised.

Completion status * (Error code) (H)	Error cause	Remedy
0010	The command request from the PLC CPU to the motion CPU exceeds the permissible value.	Check the program, and then change to the correct sequence program.
2204	The axis No. specified with the D(P).CHGV command is illegal.	

*: 0000H (normal)

In the following cases, an operation error occurs, the diagnostic error flag (SM0) turns ON, and the error code is stored in the diagnostic error register (SD0).

Error code *	Error cause	Remedy
4350	The specified applicable CPU module is incorrect. (1) A reserved CPU No. was specified. (2) An uninstalled CPU No. was specified. (3) The applicable CPU module first No. I/O No. ÷ 16(n1) lies outside the 3E0H to 3E3H range.	Check the program, and then change to the correct sequence program.
4351	Cannot be executed at the specified applicable CPU module. (1) The command name is incorrect. (2) An unsupported command was specified at the applicable CPU module.	
4352	The number of specified command devices is incorrect.	
4353	A device that cannot be used with the specified command has been specified.	
4354	A character string that cannot be handled with the specified command has been specified.	

*: 0 (normal)

Appendix 7 Operation Control Programs (Details)

Appendix 7.1 Device Descriptions

Word devices and bit devices are described below.

(1) Word device description

	Device description			Device No. (n) designation range
	16-bit integer type	32-bit integer type (n is even number)	64-bit Floating-point type (n is even number)	
Data register	Dn	DnL	DnF	0 to 8191
Link register	Wn	WnL	Wn:F	0 to 1FFF
Special register	SDn	SDnL	SDnF	0 to 2255*1
Motion register	#n	#nL	#nF	0 to 12287
Multiple CPU area devices	U□\Gn	U□\GnL	U□\GnF	10000 to (10000+p-1)*2 □: CPU No. (No.1 CPU: 3EO No.2 CPU: 3E1 No.3 CPU: 3E2 No.4 CPU: 3E3) It is not possible to specify a CPU No. greater than the number of multiple CPUs.
Coasting timer	-	FT	-	-

*1: Indirect designation is not possible for the 2000 to 2255 range.
*2: p is the number of multiple CPU high speed transmission area user setting areas for each CPU.

(a) An L is added to 32-bit integer type, and an F (for link registers: F) to 64-bit floating-point type to distinguish them.
(b) The device number is specified with an even number for 32-bit integer type and 64-bit floating-point type. (Device numbers cannot be specified with an odd number.)
(c) Coasting timer FT counts up every 888 [us]. (The coasting timer is a 32-bit integer type.)

(2) Bit device description

	Device description	Device No. (n) designation range
Input relay	Xn/PXn	0 to 1FFF*1
Output relay	Yn/PYn	0 to 1FFF
Internal relay	Mn	0 to 12287
Multiple CPU area devices	U□\Gn	10000.0 to (10000+p-1).F*2 □: CPU No. (No.1 CPU: 3EO No.2 CPU: 3E1 No.3 CPU: 3E2 No.4 CPU: 3E3) It is not possible to specify a CPU No. greater than the number of multiple CPUs.
Link relay	Bn	0 to 1FFF
Annunciator	Fn	0 to 2047
Special relay	SMn	0 to 2255*3

*1: With input devices (PXn + 0 to PXn + F) assigned to the motion CPU built-in interface (DI), the PXn + 4 to PXn + F range is fixed at 0 and cannot be used. (n = first input number)
*2: p is the number of multiple CPU high speed transmission area user setting areas for each CPU.
*3: Indirect designation is not possible for the 2000 to 2255 range.

(a) If used as batch bit data with DIN and DOUT, n is specified with a multiple of 16.
(b) If using multiple CPU area devices as batch bit data, specify as word devices without specifying bits.

(3) **Device No. indirect designation**

Device Nos. (n) can be designated indirectly for the above word device and bit device descriptions.

(a) Device No. (n) indirect designation with word device
- Word devices for which device Nos. are designated indirectly cannot be used.
- Indirect designation is possible with 16-bit integer type and 32-bit integer type word devices. The 64-bit floating-point type cannot be used.

<Description example>

Good example	Bad example
#(D10)	#(D(D5))
D(#10L)F	D(#4F)

(b) Device No. (n) indirect designation with operational expression
- Indirect designation is possible with a calculation method using the following data and operators.

Usable data	16-bit integer type word devices
	32-bit integer type word devices
	16-bit integer type constants
	32-bit integer type constants
Usable operators	Addition: +
	Subtraction: -
	Multiplication: *
	Division: /
	Remainder: %
	Sign inversion: -

- Word devices for which device Nos. are designated indirectly cannot be used.
- Only a single operator can be used.

<Description example>

Good example	Bad example
#(D10-K5)	#(D(D5)F+K20)
D(#10L%H6L)F	D(#4L<<K2)

*: If performing device No. indirect designation using the results obtained with other than the above calculations, describe by separating into two blocks as shown below.

```
D0 = SHORT(ASIN(#0F))
W0 = #(D0)
```

POINT

For details on the multiple CPU high speed transmission area user setting points, refer to Chapter 2 of the "Q173D(S)CPU/Q172D(S) CPU Motion Controller Programming Manual (Common Edition)".

Appendix 7.2 Constant Description

16-bit integer type, 32-bit integer type, and 64-bit floating-point type constants are described below.

	16-bit integer type	32-bit integer type	64-bit floating-point type
Decimal notation	K-32768 to K32767	K-2147483648L to K2147483647L	K-1.79E+308 to K-2.23E-308, K0.0, K2.23E-308 to K1.79E+308
Hexadecimal notation	H0000 to HFFFF	H00000000L to HFFFFFFFFL	-

(1) L is added to 32-bit integer type constants, 64-bit floating-point type constants contain a decimal point and added index portion (E) in order to clearly identify the data type.

(2) If the data type is omitted, the values will be regarded as the minimum type.

(3) K is added at the beginning if expressed in decimal notation, and H is added if expressed in hexadecimal notation. K can be omitted.

(4) 64-bit floating-point type constants cannot be expressed in hexadecimal notation.

F/FS	G
○	○

Appendix 7.3 Binary Operation

Appendix 7.3.1 Substitution: =

Format	(D)=(S)

No. of basic steps	4

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(S)	-	○	○	○	○	○	○	○	○	-	-
(D)	-	○	○	○	-	-	-	-	-	-	-

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S)	Word device/constant/calculation method for which substitution performed	(D) data type
(D)	Word device for which the operation result is stored	

[Function]

(1) The data value specified with (S) is substituted for the word device specified with (D).

(2) If the (S) and (D) data types differ, the (S) data type is converted to (D) and then substituted. (If (D) is a 16-bit integer type or 32-bit integer type constant, and (S) is a 64-bit floating-point type constant, the decimal portion of (S) is cut.)

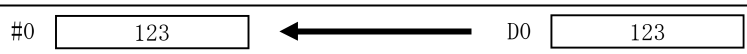
[Error]

(1) An operation error occurs in the following cases.
- When the (S) data lies outside the (D) data range.
- When either (D) or (S) is an indirect designation device, and the device No. lies outside the range.

[Program example]

(1) Program in which the D0 value is substituted for #0

```
#0 = D0
```



(2) Program in which K123456.789 is substituted for D0L

```
D0L = K123456.789
```

64-bit floating-point type constants are converted to 32-bit integer type constants and then substituted.

(3) Program in which the result of adding K123 and #0 is substituted for W0

```
W0 = K123+#0
```

F/FS	G
○	○

Appendix 7.3.2 Addition: +

Format	(S1)+(S2)

No. of basic steps	4

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(S1)	-	○	○	○	○	○	○	○	○	-	-
(S2)	-	○	○	○	○	○	○	○	○	-	-

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S1)	Augend data	Larger data type of (S1) and (S2)
(S2)	Addition data	

[Function]

(1) Data specified with (S2) is added to data specified with (S1).

(2) If the (S1) and (S2) data types differ, conversion is made to the larger of the two and then the operation is performed.

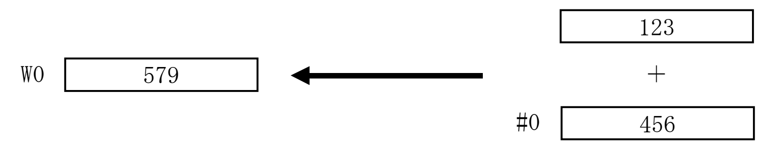
[Error]

(1) An operation error occurs in the following cases.
- Either (S1) or (S2) is an indirect designation device, and the device No. lies outside the range.

[Program example]

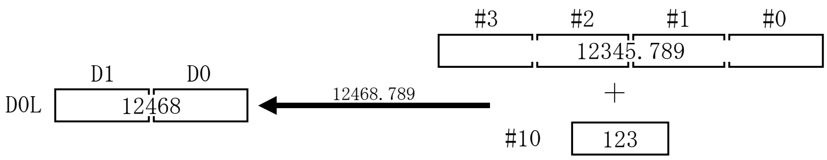
(1) Program in which the result of adding K123 and #0 is substituted for W0

W0 = K123 + #0



(2) Program in which the result of adding #0F and #10 is substituted for D0L

D0L = #0F + #10



Addition is performed with the 64-bit floating-point type, the result is converted to a 32-bit integer type constant and then substituted.

F/FS	G
○	○

Appendix 7.3.3 Subtraction: -

Format	(S1)-(S2)

No. of basic steps	4

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(S1)	-	○	○	○	○	○	○	○	○	-	-
(S2)	-	○	○	○	○	○	○	○	○	-	-

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S1)	Minuend data	Larger data type of (S1) and (S2)
(S2)	Subtraction data	

[Function]

(1) Data specified with (S2) is subtracted from data specified with (S1).

(2) If the (S1) and (S2) data types differ, conversion is made to the larger of the two and then the operation is performed.

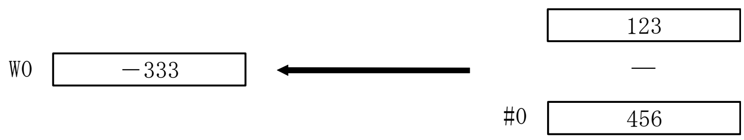
[Error]

(1) An operation error occurs in the following cases.
- Either (S1) or (S2) is an indirect designation device, and the device No. lies outside the range.

[Program example]

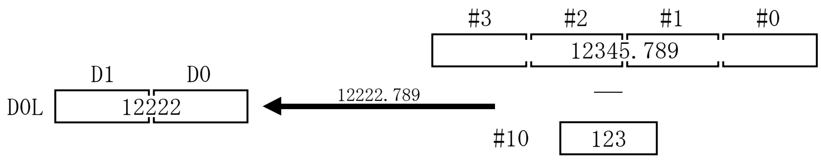
(1) Program in which the result of subtracting #0 from K123 is substituted for W0

```
W0 = K123 - #0
```



(2) Program in which the result of subtracting #10 from #0F is substituted for D0L

```
D0L = #0F - #10
```



Subtraction is performed with the 64-bit floating-point type, the result is converted to a 32-bit integer type constant and then substituted.

F/FS	G
○	○

Appendix 7.3.4 Multiplication: *

Format	(S1)*(S2)

No. of basic steps	4

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(S1)	-	○	○	○	○	○	○	○	○	-	-
(S2)	-	○	○	○	○	○	○	○	○	-	-

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S1)	Factor data	Larger data type of (S1) and (S2)
(S2)	Multiplication data	

[Function]

(1) Data specified with (S1) is multiplied by data specified with (S2).

(2) If the (S1) and (S2) data types differ, conversion is made to the larger of the two and then the operation is performed.

(3) The motion SFC program processed the multiplication result with the type specified at (2). An overflow occurs if the multiplication result exceeds the range for numerical values processed with each type, however, an operational error does not occur.
By converting setting data with a type conversion command, overflows can sometimes be prevented. (See program examples (3), (4).)

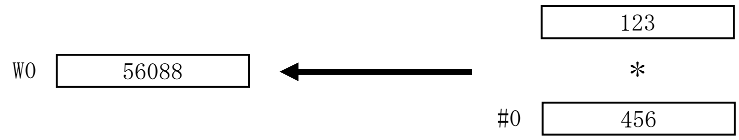
[Error]

(1) An operation error occurs in the following cases.
- Either (S1) or (S2) is an indirect designation device, and the device No. lies outside the range.

[Program example]

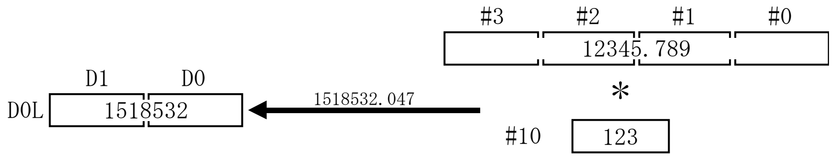
(1) Program in which the result of multiplying #0 by K123 is substituted for W0

```
W0 = K123 * #0
```



(2) Program in which the result of multiplying #10 by #0F is substituted for D0L

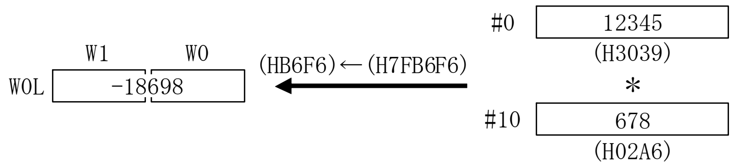
```
D0L = #0F * #10
```



Multiplication is performed with the 64-bit floating-point type, the result is converted to a 32-bit integer type constant and then substituted.

(3) Program in which the result of multiplying #10 by #0 is substituted for W0L

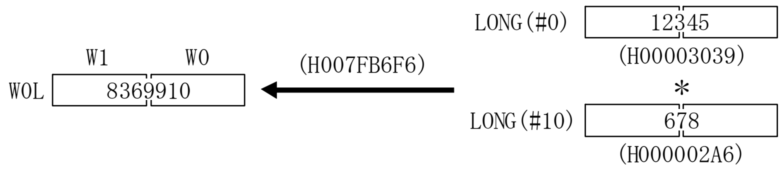
```
W0L = #0 * #10
```



Both setting data items are 16-bit integer type, and therefore the multiplication result is processed as 16-bit integer type.
An overflow occurs, and the operation result is the latter 16 bits of the multiplication result.

(4) Program in which #0 and #10 are converted to 32-bit integer type, and the multiplication result is substituted for W0L

```
W0L = LONG(#0) * LONG(#10)
```



Even if the device value is the same as program example (3), the multiplication result is processed as a 32-bit integer type with the type conversion command, and therefore no overflow occurs.

F/FS	G
○	○

Appendix 7.3.5 Division: /

Format	(S1)/(S2)

No. of basic steps	4

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(S1)	-	○	○	○	○	○	○	○	○	-	-
(S2)	-	○	○	○	○	○	○	○	○	-	-

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S1)	Divisor data	Larger data type of (S1) and (S2)
(S2)	Division data	

[Function]

(1) Data specified with (S1) is divided by data specified with (S2) to obtain the quotient.

(2) If the (S1) and (S2) data types differ, conversion is made to the larger of the two and then the operation is performed.

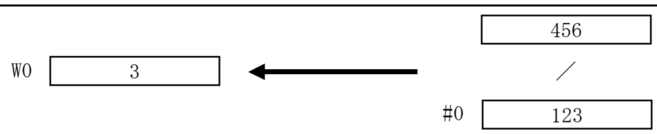
[Error]

(1) An operation error occurs in the following cases.
- When (S2) is 0
- Either (S1) or (S2) is an indirect designation device, and the device No. lies outside the range.

[Program example]

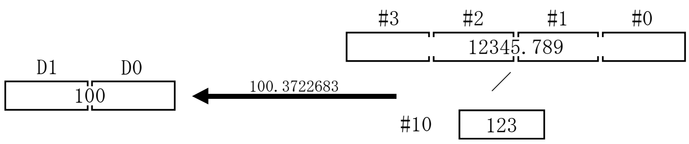
(1) Program in which K456 is divided by #0, and the quotient is substituted for W0

```
W0 = K456 / #0
```



(2) Program in which #0F is divided by #10, and the quotient is substituted for D0L

```
D0L = #0F / #10
```



Division is performed with the 64-bit floating-point type, the quotient is converted to a 32-bit integer type constant and then substituted.

F/FS	G
○	○

Appendix 7.3.6 Remainder: %

Format	(S1)%(S2)

No. of basic steps	4

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(S1)	-	○	○	-	○	○	○	-	○	-	-
(S2)	-	○	○	-	○	○	○	-	○	-	-

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S1)	Divisor data	Larger data type of (S1) and (S2) (integer type)
(S2)	Division data	

[Function]

(1) Data specified with (S1)is divided by (S2) to obtain the remainder

(2) If the (S1) and (S2) data types differ, conversion is made to the larger of the two and then the operation is performed.

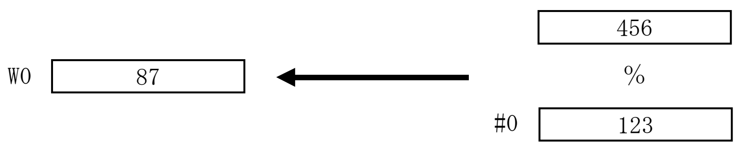
[Error]

(1) An operation error occurs in the following cases.
- When (S2) is 0
- Either (S1) or (S2) is an indirect designation device, and the device No. lies outside the range.

[Program example]

(1) Program in which K456 is divided by #0, and the remainder is substituted for W0

```
W0 = K456 % #0
```



F/FS	G
○	○

Appendix 7.4 Bit Operation

Appendix 7.4.1 Bit inversion (complement): ~

Format	~(S)	No. of basic steps	2

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(S)	-	○	○	-	○	○	○	-	○	-	-

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S)	Data for which bit inversion is performed	(S) data type (integer type)

[Function]

(1) Obtains the bit inversion value for data specified with (S).

[Error]

(1) An operation error occurs in the following cases.
- When (S) is an indirect designation device, and the device No. lies outside the range.

[Program example]

(1) Program in which the #0 bit inversion value is obtained, and then substituted for D0

```
D0 = ~#0
```

	b15 ··· b0			b15 ··· b0
D0	1 1 0 1 1 0 1 0 1 1 0 0 1 0 1 1	←	#0	0 0 1 0 0 1 0 1 0 0 1 1 0 1 0 0

F/FS	G
○	○

Appendix 7.4.2 Bit logical product: &

Format	(S1)&(S2)

No. of basic steps	4

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(S1)	-	○	○	-	○	○	○	-	○	-	-
(S2)	-	○	○	-	○	○	○	-	○	-	-

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S1)	Data for which a logical product operation is performed for each bit	Larger data type of (S1) and (S2) (integer type)
(S2)		

[Function]

(1) Obtains the logical product for each bit for data specified with (S1) and data specified with (S2).

(2) If the (S1) and (S2) data types differ, conversion is made to the larger of the two and then the operation is performed. When doing so, conversion is performed with symbols, and therefore caution is advised.

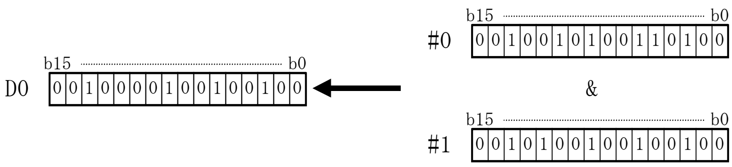
[Error]

(1) An operation error occurs in the following cases.
- Either (S1) or (S2) is an indirect designation device, and the device No. lies outside the range.

[Program example]

(1) Program in which the logical product of #0 and #1 is obtained, and then substituted for D0

```
D0 = #0 & #1
```



F/FS	G
○	○

Appendix 7.4.3 Bit logical sum: |

Format	(S1) | (S2)

No. of basic steps	4

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(S1)	-	○	○	-	○	○	○	-	○	-	-
(S2)	-	○	○	-	○	○	○	-	○	-	-

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S1)	Data for which a logical sum operation is performed for each bit	Larger data type of (S1) and (S2) (integer type)
(S2)		

[Function]

(1) Obtains the logical sum for each bit for data specified with (S1) and data specified with (S2).

(2) If the (S1) and (S2) data types differ, conversion is made to the larger of the two and then the operation is performed. When doing so, conversion is performed with symbols, and therefore caution is advised.

[Error]

(1) An operation error occurs in the following cases.
- Either (S1) or (S2) is an indirect designation device, and the device No. lies outside the range.

[Program example]

(1) Program in which the logical sum of #0 and #1 is obtained, and then substituted for D0

```
D0 = #0 | #1
```

```
                                   b15 ························ b0
                               #0  0 0 1 0 0 1 0 1 0 0 1 1 0 1 0 0
    b15 ························ b0
D0  0 0 1 0 1 1 0 1 0 0 1 1 0 1 0 0  ←          |
                                   b15 ························ b0
                               #1  0 0 1 0 1 0 0 1 0 0 1 0 0 1 0 0
```

F/FS	G
○	○

Appendix 7.4.4 Bit exclusive logical sum: ^

Format	(S1)^(S2)

No. of basic steps	4

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(S1)	-	○	○	-	○	○	○	-	○	-	-
(S2)	-	○	○	-	○	○	○	-	○	-	-

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S1)	Data for which an exclusive logical sum operation is performed for each bit	Larger data type of (S1) and (S2) (integer type)
(S2)		

[Function]

(1) Obtains the exclusive logical sum for each bit for data specified with (S1) and data specified with (S2).

(2) If the (S1) and (S2) data types differ, conversion is made to the larger of the two and then the operation is performed. When doing so, conversion is performed with symbols, and therefore caution is advised.

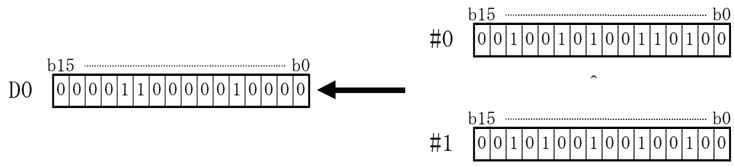
[Error]

(1) An operation error occurs in the following cases.
- Either (S1) or (S2) is an indirect designation device, and the device No. lies outside the range.

[Program example]

(1) Program in which the exclusive logical sum of #0 and #1 is obtained, and then substituted for D0

```
D0 = #0 ^ #1
```



F/FS	G
○	○

Appendix 7.4.5 Bit right shift: >>

Format	(S1)>>(S2)	No. of basic steps	4

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(S1)	-	○	○	-	○	○	○	-	○	-	-
(S2)	-	○	○	-	○	○	○	-	○	-	-

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S1)	Data for which a right shift is performed	(S1) data type (integer type)
(S2)	Number of right shifts	

[Function]

(1) Data specified with (S1) is shifted to the right by the number of times in the data specified with (S2).

(2) If the uppermost bit of (S1) is 1, 1 is entered for the uppermost bit in the right shift result. If the uppermost bit of (S1) is 0, 0 is entered for the uppermost bit in the right shift result.

(3) The result is 0 when (S1) is a 16-bit integer type, and (S2) is negative number or 16 or higher.

(4) The result is 0 when (S1) is a 32-bit integer type, and (S2) is negative number or 32 or higher.

[Error]

(1) An operation error occurs in the following cases.
- Either (S1) or (S2) is an indirect designation device, and the device No. lies outside the range.

[Program example]

(1) Program in which #0 is shifted two bits to the right, and then substituted for D0

```
D0 = #0 >> K2
```

	b15															b0			b15															b0
D0	0	0	0	0	1	0	0	1	0	1	0	0	1	1	0	1	←	#0	0	0	1	0	0	1	0	1	0	0	1	1	0	1	0	0

F/FS	G
○	○

Appendix 7.4.6 Bit left shift: <<

Format	(S1)<<(S2)	No. of basic steps	4

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(S1)	-	○	○	-	○	○	○	-	○	-	-
(S2)	-	○	○	-	○	○	○	-	○	-	-

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S1)	Data for which a left shift is performed	(S1) data (integer type)
(S2)	Number of left shifts	

[Function]

(1) Data specified with (S1) is shifted to the left by the number of times in the data specified with (S2).

(2) 0 is entered for the lowermost bit in the left shift result.

(3) The result is 0 when (S1) is a 16-bit integer type, and (S2) is negative number or 16 or higher.

(4) The result is 0 when (S1) is a 32-bit integer type, and (S2) is negative number or 32 or higher.

[Error]

(1) An operation error occurs in the following cases.
- Either (S1) or (S2) is an indirect designation device, and the device No. lies outside the range.

[Program example]

(1) Program in which #0 is shifted one bit to the left, and then substituted for D0

```
D0 = #0 << K1
```

b15 ... b0 / b15 ... b0

D0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | ← #0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 |

F/FS	G
○	○

Appendix 7.5 Bit Device Status

Appendix 7.5.1 ON (contact A): (none)

Format	(S)	No. of basic steps	2

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(S)	○	-	-	-	-	-	-	-	-	-	-

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S)	Bit device used for bit conditional expression	Logical type (true/false)

[Function]

(1) When the bit device specified with (S) with the bit conditional expression is ON(1), true is returned, and when OFF(0), false is returned.

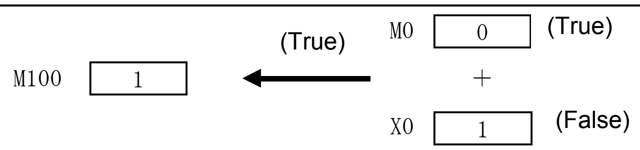
[Error]

(1) An operation error occurs in the following cases.
- When (S) is an indirect designation device, and the device No. lies outside the range.

[Program example]

(1) Program in which M100 is set when either M0 or X0 is ON(1)

```
SET M100 = M0 + X0
```



F/FS	G
○	○

Appendix 7.5.2 OFF (contact B): !

Format	!(S)	No. of basic steps	2

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(S)	○	-	-	-	-	-	-	-	-	-	-

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S)	Bit device used for bit conditional expression	Logical type (true/false)

[Function]

(1) When the bit device specified with (S) with the bit conditional expression is OFF(0), true is returned, and when ON(1), false is returned.

[Error]

(1) An operation error occurs in the following cases.
- When (S) is an indirect designation device, and the device No. lies outside the range.

[Program example]

(1) Program in which M100 is reset when M0 is OFF(0)

```
RST M100 = !M0
```

M100 [0] ← !M0 [0] (True)

F/FS	G
○	○

Appendix 7.6 Bit Device Control

Appendix 7.6.1 Device set: SET

Format	SET(D)=(S)

No. of basic steps	4

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(D)	○	-	-	-	-	-	-	-	-	-	-
(S)	○	-	-	-	-	-	-	-	-	○	○

*1: Writing is not possible for PX, and therefore cannot be used for (D).
*2: M2001 to M2032 cannot be used for (D).

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(D)	Bit data for which device setting is performed	Bit logical type (true/false)
(S)	Condition data which determines whether device setting is performed	

[Function]

(1) Sets bit data specified with (D) when the data specified with (S) is true.

(2) (S) can be omitted. The format is "SET(D)" at this time, and device setting is performed unconditionally.

(3) If set as a transition condition in the last block of the transition program, the data true/false specified with (S) is returned as logical type data. In this case, (S) cannot be omitted.

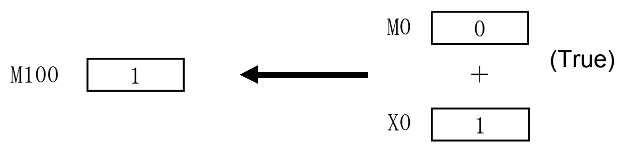
[Error]

(1) An operation error occurs in the following cases.
- When either (D) or (S) is an indirect designation device, and the device No. lies outside the range.

[Program example]

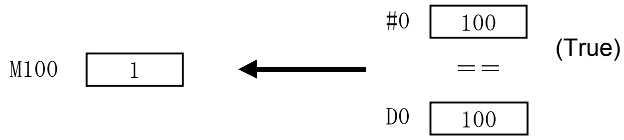
(1) Program in which M100 is set when either M0 or X0 is 1

```
SET M100 = M0 + X0
```



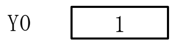
(2) Program in which M100 is set when #0 and D0 match

```
SET M100 = #0 == D0
```



(3) Program in which Y0 is set unconditionally

```
SET Y0
```



F/FS	G
○	○

Appendix 7.6.2 Device reset: RST

Format	RST(D)=(S)

No. of basic steps	4

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(D)	○	-	-	-	-	-	-	-	-	-	-
(S)	○	-	-	-	-	-	-	-	-	○	○

*1: Writing is not possible for PX, and therefore cannot be used for (D).
*2: M2001 to M2032 cannot be used for (D).

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(D)	Bit data for which device resetting is performed	Bit logical type (true/false)
(S)	Condition data which determines whether device resetting is performed	

[Function]

(1) Resets bit data specified with (D) when the data specified with (S) is true.

(2) (S) can be omitted. The format is "RST(D)" at this time, and device resetting is performed unconditionally.

(3) If set as a transition condition in the last block of the transition program, the data true/false specified with (S) is returned as logical type data. In this case, (S) cannot be omitted.

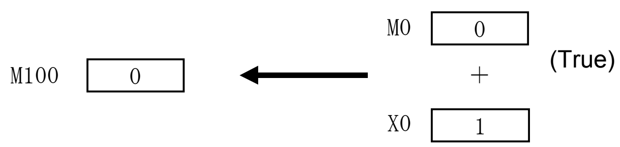
[Error]

(1) An operation error occurs in the following cases.
- When either (D) or (S) is an indirect designation device, and the device No. lies outside the range.

[Program example]

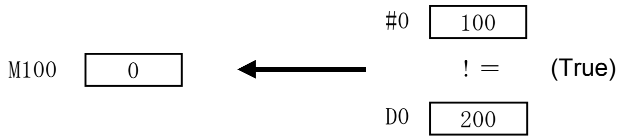
(1) Program in which M100 is reset when either M0 or X0 is 1

```
RST M100 = M0 + X0
```



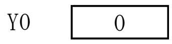
(2) Program in which M100 is reset when #0 and D0 do not match

```
RST M100 = #0 != D0
```



(3) Program in which Y0 is reset unconditionally

```
RST Y0
```



F/FS	G
○	○

Appendix 7.6.3 Device output: DOUT

Format	DOUT (D),(S)	No. of basic steps	4

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(D)	○	-	-	-	-	-	-	-	-	-	-
(S)	-	○	○	-	○	○	○	-	○	-	-

*1: PX and special relays cannot be used for (D).
*2: The range including M2000 to M2127 cannot be used for (D).

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(D)	Output destination bit data	Batch bit
(S)	Output source data	

[Function]

(1) Outputs bit data specified with (S) to bit data specified with (D).

(2) Device Nos. for bit data specified with (D) are specified in multiples of 16.

(3) If the (S) type is 16-bit integer type, (S) data is output sequentially from the lowermost bit in 16 points beginning with the bit device specified with (D).

(4) If the (S) type is 32-bit integer type, (S) data is output sequentially from the lowermost bit in 32 points beginning with the bit device specified with (D).

[Error]

(1) An operation error occurs in the following cases.
- When either (D) or (S) is an indirect designation device, and the device No. lies outside the range.
- When (D) is an indirect designation device, and the device No. is not a multiple of 16.

[Program example]

(1) Program in which D0 data is output to Y0 to YF.

```
DOUT Y0,D0
```

YF ……… Y0
0 0 1 0 0 1 1 1 0 0 0 0 1 1 1 1 ← D0 0 0 1 0 0 1 1 1 0 0 0 0 1 1 1 1
b15 ……… b0

F/FS	G
○	○

Appendix 7.6.4 Device input: DIN

Format	DIN (D),(S)

No. of basic steps	4

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(D)	-	○	○	-	-	-	-	-	-	-	-
(S)	○	-	-	-	-	-	-	-	-	-	-

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(D)	Input destination data	(D) data type (integer type)
(S)	Input origin bit data	

[Function]

(1) Inputs bit data specified with (S) to data specified with (D).

(2) Device Nos. for bit data specified with (S) are specified in multiples of 16.

(3) If the (D) type is 16-bit integer type, (D) data is input sequentially from the lowermost bit in 16 points beginning with the bit device specified with (S).

(4) If the (D) type is 32-bit integer type, (D) data is input sequentially from the lowermost bit in 32 points beginning with the bit device specified with (S).

[Error]

(1) An operation error occurs in the following cases.
- When either (D) or (S) is an indirect designation device, and the device No. lies outside the range.
- When (S) is an indirect designation device, and the device No. is not a multiple of 16.

[Program example]

(1) Program in which X0 to XF data is input to D0.

```
DIN D0,X0
```

b15 .... b0 XF X0

D0 0 0 1 0 0 1 1 1 0 0 0 0 1 1 1 1 ← 0 0 1 0 0 1 1 1 0 0 0 0 1 1 1 1

F/FS	G
○	○

Appendix 7.6.5 Bit device output: OUT

Format	OUT (D) = (S)

No. of basic steps	4

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(D)	○	-	-	-	-	-	-	-	-	-	-
(S)	○	-	-	-	-	-	-	-	-	○	○

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(D)	Bit device for which device output is performed	Bit
(S)	Device output conditions data	

[Function]

(1) Bit devices specified with (D) are set when the data specified with (S) is true, and bit devices specified with (D) are reset when the data specified with (S) is false.

(2) If set as a transition condition in the last block of the transition program, the data true/false specified with (S) is returned as logical type data.

(3) (S) cannot be omitted.

[Error]

(1) An operation error occurs in the following cases.
- When either (D) or (S) is an indirect designation device, and the device No. lies outside the range.

[Program example]

(1) Program in which M100 turns ON when M0 is ON, and turns OFF when M0 is OFF.

```
OUT M100 = M0
```

(2) Program in which M100 turns ON when both M0 and M1 are ON, and turns OFF in all other cases.

```
OUT M100 = M0 * M1
```

(3) M100 turns ON when the D0 and D2000 values match, and turns OFF when they do not.

```
OUT M100 = (D0 == D2000)
```

F/FS	G
○	○

Appendix 7.7 Logical Operations

Appendix 7.7.1 Logical affirmation: (none)

Format	(S)	No. of basic steps	-

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(S)	○	-	-	-	-	-	-	-	-	○	○

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S)	Data for which logical affirmation is performed	Logical type (true/false)

[Function]

(1) Returns logical type data true/false specified with (S) as is. (Logical affirmation)

[Error]

(1) An operation error occurs in the following cases.
- When (S) is an indirect designation device, and the device No. lies outside the range.

[Program example]

(1) Program in which M100 is set when either M0 or X0 is ON(1)

```
SET M100 = M0 + X0
```

```
                        (True)   M0 [ 0 ] (False)
M100 [ 1 ]  <-------                  +
                                 X0 [ 1 ] (True)
```

F/FS	G
○	○

Appendix 7.7.2 Logical negation: !

Format	!(S)	No. of basic steps	2

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(S)	○	-	-	-	-	-	-	-	-	○	○

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S)	Data for which logical negation is performed	Logical type (true/false)

[Function]

(1) Performs logical negation for data specified with (S).

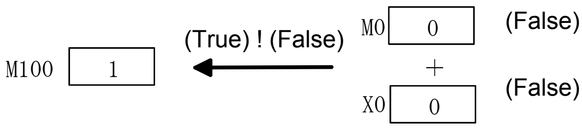
[Error]

(1) An operation error occurs in the following cases.
- When (S) is an indirect designation device, and the device No. lies outside the range.

[Program example]

(1) Program in which M100 is set when "either M0 or X0 is ON(1)" (when both M0 and X0 are OFF(0))

```
SET M100 = !(M0 + X0)
```



F/FS	G
○	○

Appendix 7.7.3 Logical product: *

Format	(S1)*(S2)

No. of basic steps	4

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(S1)	○	-	-	-	-	-	-	-	-	○	○
(S2)	○	-	-	-	-	-	-	-	-	○	○

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S1)	Data for which logical product operation is performed	Logical type (true/false)
(S2)		

[Function]

(1) Obtains the logical product for data specified with (S1) and data specified with (S2).

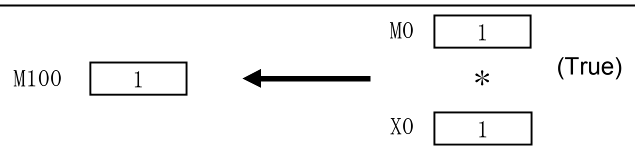
[Error]

(1) An operation error occurs in the following cases.
- When (S) is an indirect designation device, and the device No. lies outside the range.

[Program example]

(1) Program in which M100 is set when both M0 and X0 are 1

```
SET M100 = M0 * X0
```



F/FS	G
○	○

Appendix 7.7.4 Logical sum: +

Format	(S1)+(S2)	No. of basic steps	4

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(S1)	○	-	-	-	-	-	-	-	-	○	○
(S2)	○	-	-	-	-	-	-	-	-	○	○

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S1)	Data for which logical sum operation is performed	Logical type (true/false)
(S2)		

[Function]

(1) Obtains the logical sum for data specified with (S1) and data specified with (S2).

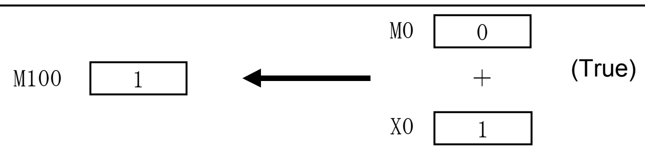
[Error]

(1) An operation error occurs in the following cases.
- Either (S1) or (S2) is an indirect designation device, and the device No. lies outside the range.

[Program example]

(1) Program in which M100 is set when either M0 or X0 is 1

```
SET M100 = M0 + X0
```



F/FS	G
○	○

Appendix 7.8 Comparison Operations

Appendix 7.8.1 Match: ==

Format	(S1)==(S2)

No. of basic steps	4

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(S1)	-	○	○	○	○	○	○	○	○	-	-
(S2)	-	○	○	○	○	○	○	○	○	-	-

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S1)	Data for which comparison is made	Logical type (true/false)
(S2)		

[Function]

(1) Data specified with (S1) is compared with data specified with (S2), and the result is true if they match.

(2) If the (S1) and (S2) data types differ, conversion is made to the larger of the two and then the comparison is made.

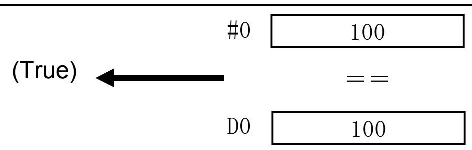
[Error]

(1) An operation error occurs in the following cases.
- Either (S1) or (S2) is an indirect designation device, and the device No. lies outside the range.

[Program example]

(1) Program in which a comparison is made when #0 and D0 match

```
#0 == D0
```



F/FS	G
○	○

Appendix 7.8.2 Mismatch !=

Format	(S1)!=(S2)

No. of basic steps	4

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(S1)	-	○	○	○	○	○	○	○	○	-	-
(S2)	-	○	○	○	○	○	○	○	○	-	-

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S1)	Data for which comparison is made	Logical type (true/false)
(S2)		

[Function]

(1) Data specified with (S1) is compared with data specified with (S2), and the result is true if they do not match.

(2) If the (S1) and (S2) data types differ, conversion is made to the larger of the two and then the comparison is made.

[Error]

(1) An operation error occurs in the following cases.
- Either (S1) or (S2) is an indirect designation device, and the device No. lies outside the range.

[Program example]

(1) Program in which a comparison is made when #0 and D0 do not match

```
#0 != D0
```

#0 100

(True) ← ! =

D0 20

F/FS	G
○	○

Appendix 7.8.3 Less than: <

Format	(S1)<(S2)

No. of basic steps	4

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(S1)	-	○	○	○	○	○	○	○	○	-	-
(S2)	-	○	○	○	○	○	○	○	○	-	-

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S1)	Data for which comparison is made	Logical type (true/false)
(S2)		

[Function]

(1) If the data specified with (S1) is less than the data specified with (S2), the result is true.

(2) If the (S1) and (S2) data types differ, conversion is made to the larger of the two and then the comparison is made.

[Error]

(1) An operation error occurs in the following cases.
- Either (S1) or (S2) is an indirect designation device, and the device No. lies outside the range.

[Program example]

(1) Program in which a comparison is made to determine whether #0 is less than D0

```
#0 < D0
```

#0 10

(True) ← <

D0 20

F/FS	G
○	○

Appendix 7.8.4 Less than or equal to: <=

Format	(S1)<=(S2)	No. of basic steps	4

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(S1)	-	○	○	○	○	○	○	○	○	-	-
(S2)	-	○	○	○	○	○	○	○	○	-	-

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S1)	Data for which comparison is made	Logical type (true/false)
(S2)		

[Function]

(1) If the data specified with (S1) is less than or equal to the data specified with (S2), the result is true.

(2) If the (S1) and (S2) data types differ, conversion is made to the larger of the two and then the comparison is made.

[Error]

(1) An operation error occurs in the following cases.
- Either (S1) or (S2) is an indirect designation device, and the device No. lies outside the range.

[Program example]

(1) Program in which a comparison is made to determine whether #0 is less than or equal to D0

```
#0 <= D0
```

#0 [10]

(True) ← <=

D0 [20]

F/FS	G
○	○

Appendix 7.8.5 Greater than: >

Format	(S1)>(S2)

No. of basic steps	4

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(S1)	-	○	○	○	○	○	○	○	○	-	-
(S2)	-	○	○	○	○	○	○	○	○	-	-

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S1)	Data for which comparison is made	Logical type (true/false)
(S2)		

[Function]

(1) If the data specified with (S1) is greater than the data specified with (S2), the result is true.

(2) If the (S1) and (S2) data types differ, conversion is made to the larger of the two and then the comparison is made.

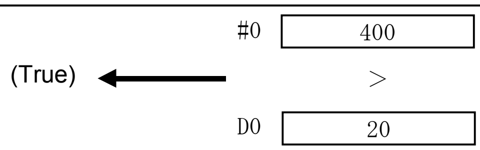
[Error]

(1) An operation error occurs in the following cases.
- Either (S1) or (S2) is an indirect designation device, and the device No. lies outside the range.

[Program example]

(1) Program in which a comparison is made to determine whether #0 is greater than D0

```
#0 > D0
```



F/FS	G
○	○

Appendix 7.8.6 Greater than or equal to: >=

Format	(S1)>=(S2)

No. of basic steps	4

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(S1)	-	○	○	○	○	○	○	○	○	-	-
(S2)	-	○	○	○	○	○	○	○	○	-	-

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S1)	Data for which comparison is made	Logical type (true/false)
(S2)		

[Function]

(1) If the data specified with (S1) is greater than or equal to the data specified with (S2), the result is true.

(2) If the (S1) and (S2) data types differ, conversion is made to the larger of the two and then the comparison is made.

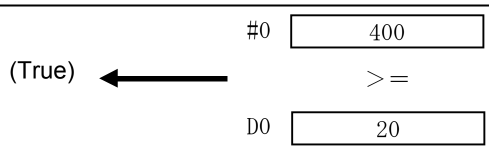
[Error]

(1) An operation error occurs in the following cases.
- Either (S1) or (S2) is an indirect designation device, and the device No. lies outside the range.

[Program example]

(1) Program in which a comparison is made to determine whether #0 is greater than or equal to D0

```
#0 >= D0
```



F/FS	G
○	○

Appendix 7.9 Dedicated Motion Functions (CHGV/CHGT)

Appendix 7.9.1 Speed change request: CHGV

Format	CHGV((S1),(S2))	**No. of basic steps**	4

[Usable data]

Setting data	**Usable data**										
	Bit device	**Word device**				**Constant**			**Calculation formula**	**Bit conditional expression**	**Comparative conditional expression**
		16-bit integer type	**32-bit integer type (L)**	**64-bit floating-point type (F)**	**Coasting timer**	**16-bit integer type (K/H)**	**32-bit integer type (K/H, L)**	**64-bit floating-point type (K)**			
(S1)	-	-	-	-	-	○	-	-	-	-	-
(S2)	-	○	○	-	-	○	○	-	○	-	-

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S1)	Axis No. for which speed change request is made	-
(S2)	Specified speed	

[Function]

(1) Speed change is performed using the following procedure.

(a) The speed change accepting flag (M2061 to M2092) corresponding to the axis specified with (S1) is turned ON.

(b) The speed of the axis specified with (S1) is changed to the speed specified with (S2).

(c) The speed change accepting flag is turned OFF.

(2) The range of axis Nos. that can be set for (S1) is as follows.

Q172DSCPU	Q173DSCPU
1 to 16	1 to 32

When performing interpolation control, set one of the interpolation axes. If performing linear interpolation control, speed change is performed based on the positioning speed designation method set in the servo program.

Positioning speed designation method	Operation
Composite speed designation	The speed is changed so that the composite speed becomes the speed specified with (S2).
Major axis reference	The speed is changed so that the major axis speed becomes the speed specified with (S2).
Reference axis speed designation	The speed is changed so that the reference axis speed becomes the speed specified with (S2).

(3) Operation is performed as follows based on the designated speed symbol set for (S2).

Designated speed symbols	Operation
Positive	Speed change
0	Temporary stop
Negative	Reversal

(4) The range for the designated speed that can be set for (S2) is as follows.

(a) Real mode

	mm		inch		degree		PLS	
	Setting range	Unit	Setting range	Unit	Setting range	Unit	Setting range	Unit
Speed change request	0 to 600000000	$\times 10^{-2}$ mm/min	0 to 600000000	$\times 10^{-3}$ inch/min	0 to 2147483647	$\times 10^{-3}$ *1 degrees/min	0 to 2147483647	PLS/s
Reversal request	-1 to -600000000	$\times 10^{-2}$ mm/min	-1 to -600000000	$\times 10^{-3}$ inch/min	-1 to -2147483647	$\times 10^{-3}$ *1 degrees/min	-1 to -2147483647	PLS/s

*1: If the degree axis speed 10 times designation is enabled in the fixed parameters, the unit will be $x10^{-2}$ [degrees/min].

(b) Virtual mode

	PLS	
	Setting range	Unit
Speed change request	0 to 2147483647	PLS/s
Reversal request	-1 to -2147483647	PLS/s

(5) The speed changed with the CHGV command is valid only for servo programs that have been started.

(6) Speed change is not performed when deceleration of the axis specified with (S1) is stopped.

(7) Speed change is not performed if the axis specified with (S1) is currently subject to speed/torque control.

(8) By specifying a negative speed during startup and then issuing a speed change request, the axis starts to decelerate from that point onward, and is able to return in the reverse direction when deceleration is complete. The following operations are performed with servo commands.

Control mode	Servo command	Operation
Linear control	ABS-1 INC-1 ABS-2 INC-2 ABS-3 INC-3 ABS-4 INC-4	The travel direction is reversed when deceleration is complete, the axis returns to the positioning start point at the specified absolute value for speed, and then stops (standby). When performing circular interpolation, the axis returns in a circular locus.
Circular interpolation control	ABS CIRCULAR INC CIRCULAR	
Fixed feed	FEED-1 FEED-2 FEED-3	
Constant speed control	CPSTART1 CPSTART2 CPSTART3 CPSTART4	The travel direction is reversed when deceleration is complete, the axis returns to the previous point at the specified absolute value for speed, and then stops (standby).
Speed control (I)	VF VR	The travel direction at the specified absolute value for speed when deceleration is complete is reversed. The axis does not stop until a stop command is input.
Speed control (II)	VVF VVR	
Speed/position control	VPF VPR VPSTART	Reversal is not possible. The request is deemed to be a normal speed change request. A minor error [305]* occurs, and speed is controlled at the speed limit value.
Fixed-pitch feed control	PFSTART	
Fixed position stop speed control	PVF PVR	
Speed switching control	VSTART	
JOG operation		
High-speed oscillation	OSC	The speed cannot be changed. A minor error [310]* occurs.
Zeroing	ZERO	The speed cannot be changed. A minor error [301]* occurs.

*: Minor error [301]: Speed change was performed during zeroing.
Minor error [305]: The set speed lies outside the 0 to speed limit value range.
Minor error [310]: Speed change was performed during high-speed oscillation.

[Control details]

(a) If the speed is changed to a negative speed, the control indicated in the above table is performed based on the control mode during startup.

(b) The command speed when returning is the change speed absolute value.

(c) The status when the axis is standing by at the return position is as follows.

1) Signal status

• Start accept (M2001+n)	ON (no change to before CHGV execution)
• Positioning start complete (M2400+20n)	ON (no change to before CHGV execution)
• Positioning complete (M2401+20n)	OFF
• In-position (M2402+20n)	ON
• Command in-position (M2403+20n)	OFF
• Speed change "0" accepting flag (M2240+n)	ON

2) If starting again, change the speed to a positive speed.
3) If terminating positioning, turn the stop command ON.
4) If a negative speed change is performed again, it is ignored.

(d) Operation is as follows if during reversal in speed control mode.
1) If returning the travel direction again, change the speed to a positive speed.
2) If stopping, turn the stop command ON.
3) If a negative speed change is performed again, speed change is performed in the reversal direction.

(e) Changes to negative speeds are not performed for axes for which the stroke limit is disabled.

[Error]

(1) An operation error occurs in the following cases, and speed change is not performed.
- When the (S1) designated axis No. lies outside the range.
- When (S2) is an indirect designation device, and the device No. lies outside the range.

(2) A minor error occurs in the following cases, and speed change is not performed.
- When zeroing is performed for the axis specified with (S1). (Minor error: 301)
- When changes to negative speeds are performed for axes for which the stroke limit is disabled. (Minor error: 310)

POINT

Speed changes are ignored even if performed when the axis specified with (S1) is decelerating.
No error occurs at this time.

(3) A minor error occurs in the following case, and control is performed at the speed limit value.
- When the absolute value for the speed specified with (S2) is greater than the speed limit value. (Minor error: 305)

POINT

If the negative change speed absolute value exceeds the speed specified in the servo program during constant speed control, reversal control is performed at the speed specified in the program (speed clamp control for speed change during constant speed control). No error occurs at this time.

[Program example]

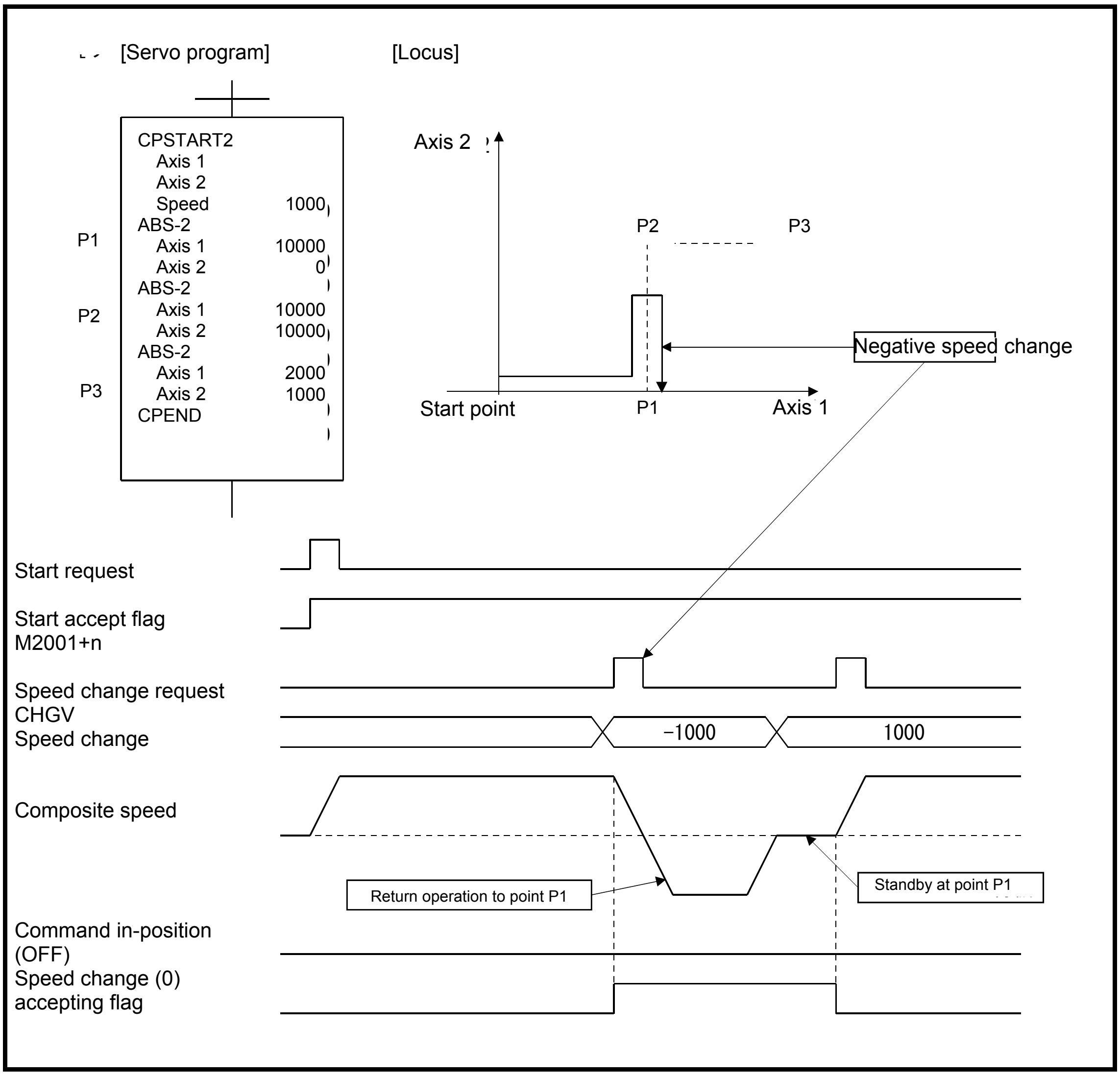
(1) Program in which the axis 2 positioning speed is changed

CHGV(K2,K10)

(2) Reversal program in which the axis 1 positioning speed is changed to a negative value

CHGV(K1,K-1000)

The operation when a reversal request is made when performing constant speed control is as follows.



By changing the speed to a negative value while performing positioning at P2 as shown above, the axis returns to P1 along the locus specified in the program and stands by at P1.

POINT

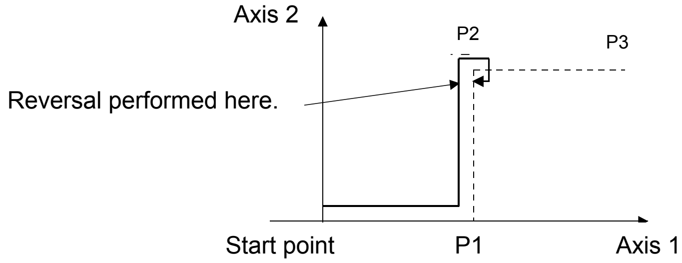
<u>• Speed change precautions</u>

(1) If speed change is performed during the period up until the "Positioning start complete signal" status turns ON when issuing a servo program start request, the speed change may be invalid. If the speed change is performed around the same time as the program is started, create the program in such a way that the speed change is always performed after the "Positioning start complete signal" status turns ON.

(2) If the M-code FIN signal wait function is used when performing constant speed control, and a reversal request is made while the axis is stopped and waiting for FIN, the request is ignored.

(3) If a reversal request is made immediately before P2 and P2 is passed while decelerating as in the example on the previous page, the axis returns to P2.

(4) The response time from the point the CHGV command is executed until the speed actually starts to change is delayed by a maximum of the length of the operation cycle.



F/FS	G
○	○

Appendix 7.9.2 Torque limit value change request: CHGT

Format	CHGT((S1),(S2))

No. of basic steps	4

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(S1)	-	-	-	-	-	○	-	-	-	-	-
(S2)	-	○	○	-	-	○	○	-	○	-	-

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S1)	Axis No. for which torque limit value change request made	-
(S2)	Specified torque limit value	

[Function]

(1) The axis torque limit value specified with (S1) is changed to the torque limit value specified with (S2).

(2) When in real mode, if servo startup is complete for the axis in question, the torque limit value is changed at any time regardless of whether the servo is starting, stopped, ON, or OFF.

(3) The range for the axis No. that can be set for (S1) is as follows.

Q172DSCPU	Q173DSCPU
1 to 16	1 to 32

(4) The range for the torque limit value that can be set for (S2) is 1 to 1000 [%].

(5) The relationship with the torque limit value specified in the servo program is as follows.

When started

When the servo starts normally, a command is issued specifying the torque limit value for the starting axis servo based based either on "P.torque" set in the servo program, or the "Torque limit value" in the specified parameter block. This torque limit value is applied to the travel amount for the interpolating axis when starting interpolation.

↓

By executing the CHGT command, a command is issued specifying the torque limit value set only for the designated axis.

↓

Subsequently, the torque limit value specified for the servo when starting the servo program or when starting JOG operation is valid only if it is lower than the torque limit value changed with the CHGT command. Clamp processing for this torque limit value is performed for each axis.

When starting

(a) Even if the following settings are specified, the torque limit value is not changed to a value higher than that changed with the CHGT command.
   - Torque limit value at midway point when performing constant speed control or speed switching control
   - Torque limit value at the moment position control switching is performed when performing speed/position switching control
   - Torque limit value when performing speed control

(b) With the CHGT command, it is also possible to change to a torque limit value higher than that set in the servo program or in the parameter block.

(6) The torque limit value changed with the CHGT command is valid only while the servo amp control power is ON.

[Error]

(1) An operation error occurs in the following cases, and the torque limit value is not changed.
   - When the (S1) designated axis No. lies outside the range.
   - When (S2) is an indirect designation device, and the device No. lies outside the range.

(2) A minor error occurs in the following cases, and the torque limit value is not changed.
   - When the torque limit value specified with (S2) lies outside the 1 to 1000 [%] range. (Minor error: 311)
   - When the CHGT command is issued for axes that have not been started (Minor error: 312)

[Program example]

(1) Program in which the axis 2 torque limit value is changed to 10 [%]

```
CHGT(K2, K10)
```

POINT

(1) The CHGT command has no effect (is ignored) if issued while in virtual mode. If the torque limit value is changed during operation in virtual mode, perform after setting the "Torque limit value setting device" in the output module parameters for the mechanical system program.
(2) The time from the point the CHGT command is executed until the torque limit value is actually transferred to the servo amp is delayed by a maximum of the length of the operation cycle.

F/FS	G
○	○

Appendix 7.9.3 Torque limit value individual change request: CHGT2

Format	CHGT2((S1),(S2),(S3))

No. of basic steps	5

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(S1)	-	-	-	-	-	○	-	-	-	-	-
(S2)	-	○	○	-	-	○	○	-	○	-	-
(S3)	-	○	○	-	-	○	○	-	○	-	-

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S1)	Axis No. for which torque limit value change request made	-
(S2)	Plus direction torque limit value (×0.1 [%]	
(S3)	Minus direction torque limit value (×0.1 [%]	

[Function]

(1) The axis torque limit value specified with (S1) is changed to the plus direction torque limit value specified with (S2) and minus direction torque limit value specified with (S3).
The plus direction torque limit value is used to power the servo motor forward rotation (CW) and control the reverse rotation (CCW) regenerative torque, and the minus direction torque limit value is used to power the servo motor reverse rotation (CCW) and control the forward rotation (CCW) regenerative torque.

(2) If servo startup is complete for the axis in question, the torque limit value is changed at any time regardless of whether the servo is starting, stopped, ON, or OFF.

(3) If a CHGT2 command is executed for the mechanical system output module in virtual mode, set 300 [%] for the output module torque limit value. If the torque limit value for the output module is set by indirect designation with a device, a minor error (error code: 6260) occurs, and the torque limit value is not changed individually.

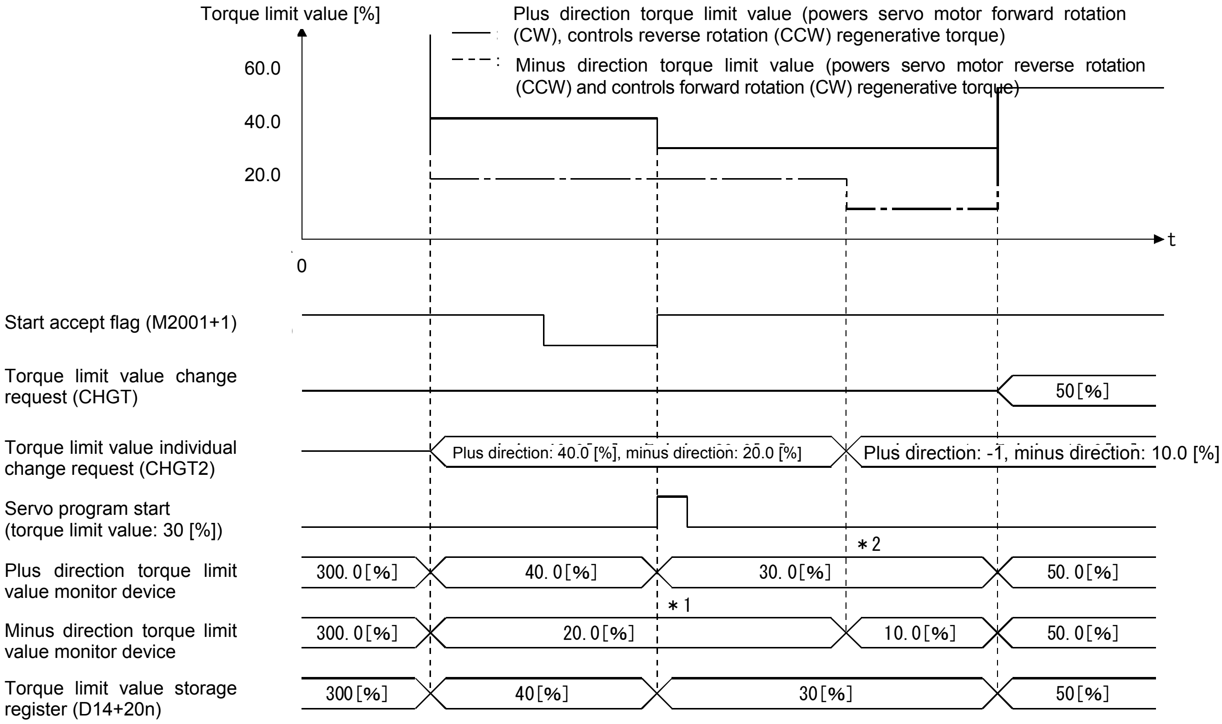
(4) The range for the axis No. that can be set for (S1) is as follows.

Q172DSCPU	Q173DSCPU
1 to 16	1 to 32

(5) (S2) and (S3) cannot be omitted. If changing only one of the torque limit values, set -1 for the setting data for which no change is required.

(6) The range for the torque limit value that can be set for (S2) and (S3) is 1 to 10000 (x 0.1 [%]).

(7) For details on the relationship between the torque limit value specified in the servo program and the torque limit value change request command, refer to the "Q173D(S)CPU/Q172D(S) CPU Motion Controller (SV13/SV22) Programming Manual (Real Mode Edition)".
Operation when the CHGT2 and CHGT commands are combined is as follows.



*1: The torque limit value specified in the servo program is clamped with the minus direction torque limit value changed with CHGT2.
*2: -1 is set for the CHGT2 plus direction torque limit value, and therefore there is no change.

(8) When performing speed/torque control, it is not possible to change to the speed set in the servo data, the speed set in the torque control data, or to a torque limit value higher than the value used when performing torque control. If either the (S2) or (S3) value specified with the CHGT2 command is greater than the torque limit value used when performing speed and torque control, a minor error (error code: 319) occurs, and the torque limit value is not changed individually.

(9) By setting a plus direction torque limit value monitor device and minus direction torque limit value monitor device in the servo data settings extended parameters, the plus and minus direction torque limit values can be monitored.

[Error]

(1) An operation error occurs in the following cases, and the torque limit value is not changed.
- When the (S1) designated axis No. lies outside the range.
- Either (S2) or (S3) is an indirect designation device, and the device No. lies outside the range.

(2) A minor error occurs in the following cases, and the torque limit value is not changed.
- When the torque limit value specified with (S2) or (S3) lies outside the 0.1 to 1000.0 [%] range.
(Minor error: 311)
- When the CHGT2 command is issued for axes that have not been started (Minor error: 312)
- When the (S2) or (S3) value is greater than the torque limit value when performing speed/torque control if a CHGT2 command is executed for an axis for which speed/torque control is being performed. (Minor error: 319)
- When a CHGT2 command is executed for an axis for which the torque limit value is designated indirectly with a device at the output module when in virtual mode. (Minor error: 6260)

[Program example]

(1) Program in which the axis 2 torque limit value is changed individually to 20.0 [%] for the plus direction and 10.0 [%] for the minus direction.

```
CHGT2(K2, K200, K100)
```

POINT
The time from the point the CHGT2 command is executed until the torque limit value is actually transferred to the servo amp is delayed by a maximum of the length of the operation cycle.

F/FS	G
○	○

Appendix 7.9.4 Target position change request: CHGP

Format	CHGP((S1),(S2),(S3))

No. of basic steps	6

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(S1)	-	-	-	-	-	○	-	-	-	-	-
(S2)	-	○	-	-	-	○	-	-	-	-	-
(S3)	-	○	-	-	-	-	-	-	-	-	-

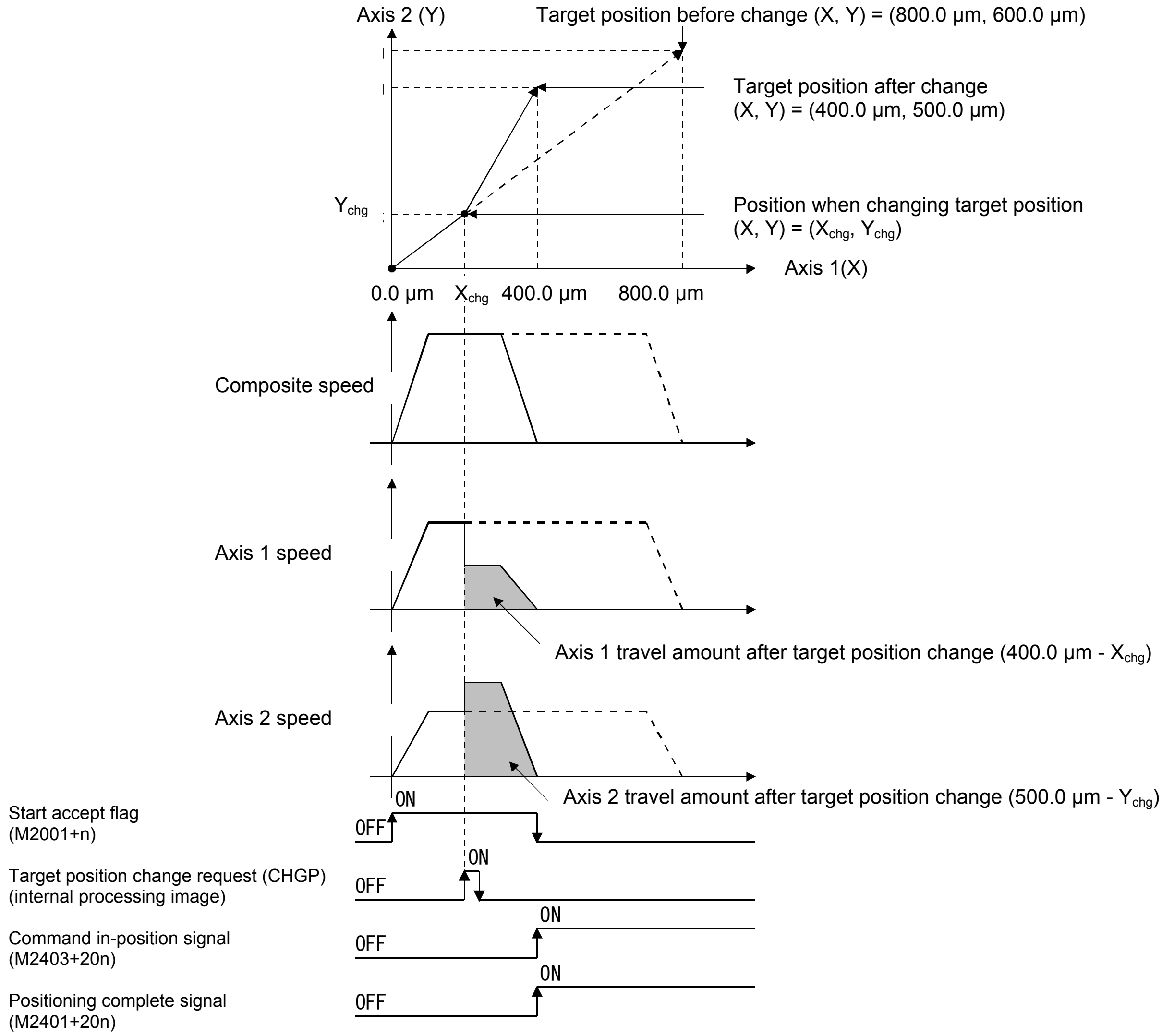
○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S1)	Axis No. for which target position change request is made	-
(S2)	Change address designation method 0: Address designation 1: Travel amount designation	
(S3)	First number of device for which target position change value is stored	

[Overview]

When a target position change request is issued, the target position is changed while executing positioning commands. The new target position can be specified with an absolute address, or with the relative travel amount from the feed current value when executing the target position change request.
If a request for a target position change to (X, Y) = (400.0 μm, 500.0 μm) by specifying an absolute address while performing linear interpolation from the positioning start position (X, Y) = (0.0 um, 0.0 um) to (X, Y) = (800.0 μm, 600.0 μm), operation is as follows.



[Function]

(1) The target position is changed for the axis specified with (S1). Depending on the method specified with (S2), the target position after the change is calculated with the value stored in the device specified with (S3).

> **POINT**
>
> (1) The CHGP command is valid only for axes that have been started.
> (2) Target position change is not performed when deceleration of the specified axis is stopped.
> (3) The time from the point the CHGP command is executed until the target position is actually changed is delayed by a maximum of the length of the operation cycle.
> (4) By executing the CHGP command when making a servo program start request (when positioning start complete signal (M2400+20n) is OFF), the target position change is invalid. If the target position change is performed around the same time as the servo program is started, create the program in such a way that the target position change is performed after the "Positioning start complete signal" status turns ON.

(2) The range of axis Nos. that can be set for (S1) is as follows.
When performing interpolation control, set one of the interpolation axes.

Q172DSCPU	Q173DSCPU
1 to 16	1 to 32

(3) By setting (S2), the target position is as follows.

(a) When (S2) is set to 0 (address designation method), the target position value stored in the device specified with (S3) is set as the target position.

(b) When (S2) is set to 1 (travel value designation method), the position from the feed current value when executing the CHGP command to the position following travel of the target position change amount stored in the device specified with (S3) is set as the target position.

> **Point**
>
> By setting (S2) to 1 (travel value designation method) and executing the CHGP command with a normal task, variations in the changed target position may occur as a result of variations in the command accept timing. By executing with the same fixed cycle task as the operation cycle, variations can be controlled.

(4) The first device in which the target position change value is stored is specified in (S3).
Set an even number for the first device, and set the target position change value as follows.

<table>
<tr><th rowspan="3">Offset</th><th rowspan="3">Name</th><th colspan="5">Setting range</th></tr>
<tr><th rowspan="2">mm</th><th rowspan="2">inch</th><th rowspan="2">PLS</th><th colspan="2">degrees</th></tr>
<tr><th>Address designation</th><th>Travel value designation</th></tr>
<tr><td>+0</td><td rowspan="2">Target pos. change value 1</td><td rowspan="8">-2147483648 to 2147483647 (×10^{-1} [µm])</td><td rowspan="8">-2147483648 to 2147483647 (×10^{-5} [inch])</td><td rowspan="8">-2147483648 to 2147483647 ([PLS])</td><td rowspan="8">0 to 35999999 (×10^{-5} [degree])</td><td rowspan="8">-2147483648 to 2147483647 (×10^{-5} [degrees])</td></tr>
<tr><td>+1</td></tr>
<tr><td>+2</td><td rowspan="2">Target pos. change value 2</td></tr>
<tr><td>+3</td></tr>
<tr><td>+4</td><td rowspan="2">Target pos. change value 3</td></tr>
<tr><td>+5</td></tr>
<tr><td>+6</td><td rowspan="2">Target pos. change value 4</td></tr>
<tr><td>+7</td></tr>
</table>

(a) Set a positioning address or travel value for the target position change value based on the (S2) setting.

(b) Set the target position change values in ascending order among the interpolation axes.
(Example) If making a target position change request while an INC-3 command is being executed

```
[K100] INC-3
        Axis    3,    3000PLS
        Axis    4.    4000PLS
        Axis    1,    4000PLS
        Speed         10000PLS/s
```

The axis Nos. corresponding to target position change values 1 to 4 are as follows.

Target position change value 1	Axis No.1 setting
Target position change value 2	Axis No.3 setting
Target position change value 3	Axis No.4 setting
Target position change value 4	No setting required

(5) The CHGP command can be executed for both real mode programs and virtual mode programs.

(6) When executing the CHGP command, the following operations are performed with servo commands during execution.

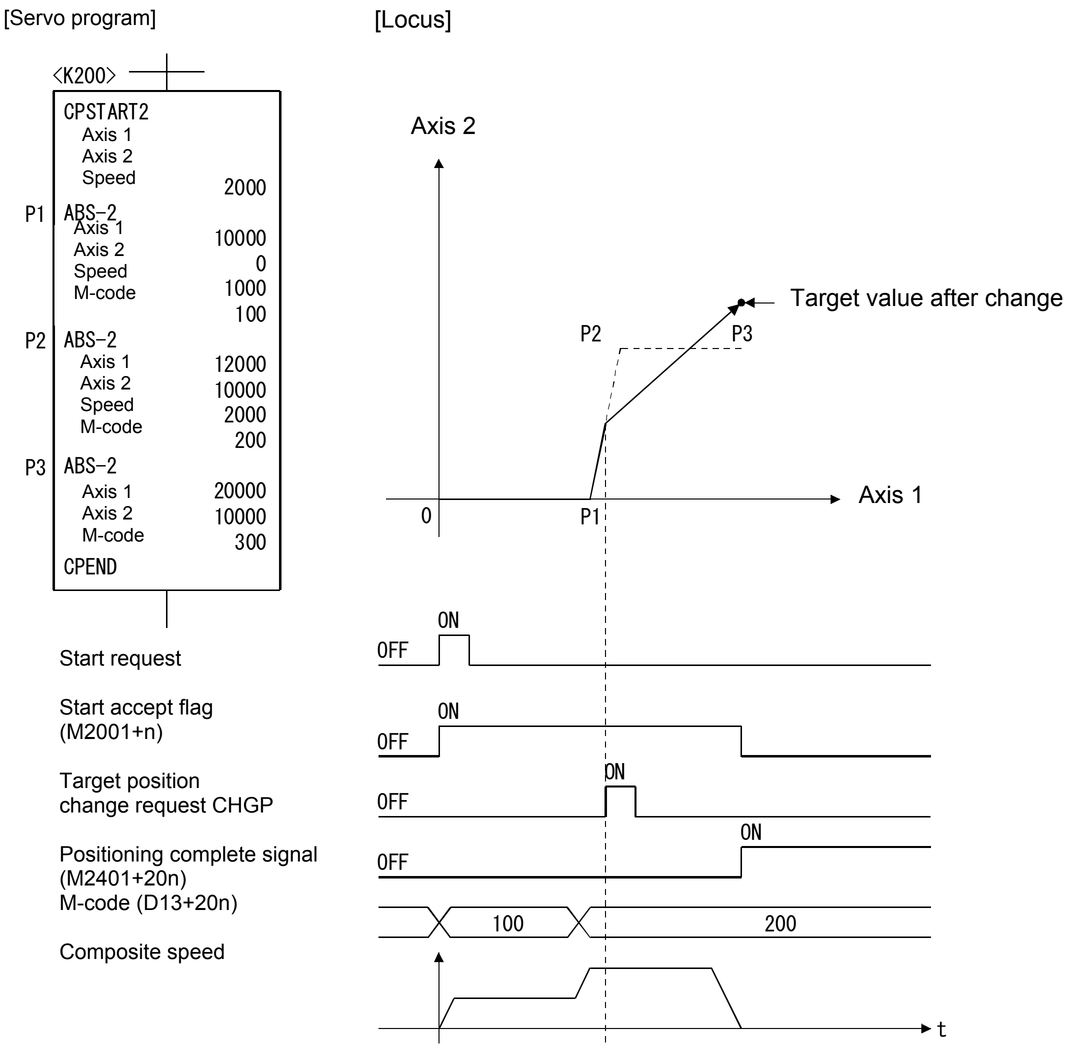
Control	Servo command	Operation
Linear	ABS−1 INC−1 ABS−2 INC−2 ABS−3 INC−3 ABS−4 INC−4	By executing the CHGP command, positioning is performed with linear interpolation control to the target position from which the change was made from the feed current value when the command is executed.
Fixed feed	FEED-1 FEED-2 FEED-3	
Circular interpolation control	ABS CIRCULAR INC CIRCULAR	The target position change is ignored, and a minor error [330] occurs.
Helical interpolation control	ABS HELICAL INC HELICAL	
Constant speed control	CPSTART1 CPSTART2 CPSTART3 CPSTART4	By executing the CHGP command, positioning is performed with linear interpolation control to the target position from which the change was made from the feed current value when the command is executed. Positioning to the remaining points is not performed. (See item 10.)
Speed control (I)	VF VR	The target position change is ignored, and a minor error [330] occurs.
Speed control (II)	VVF VVR	
Speed/position control	VPF VPR VPSTART	
Fixed-pitch feed control	PFSTART	
Fixed position stop speed control	PVF PVR	
Speed switching control	VSTART	
JOG operation		
Speed/torque control		
High-speed oscillation	OSC	
Zeroing	ZERO	

(7) Operation following execution of the CHGP command is as follows.
- The automatic decelerating flag (M2128+n) turns ON when automatic deceleration to the target position following the change is processed.
- The command in-position signal (M2403+20n) turns ON when the absolute value for the difference between the target position following the change and the feed current value falls below the "command in-position range".
- The positioning complete signal (M2401+20n) turns ON when output of the command to the target position following the change is complete.

(8) After executing the CHGP command, the composite speed remains as is, and the speed of each axis changes based on the target position following the change. Consequently, the speed of each axis may change suddenly depending on the target position following the change, and therefore caution is advised.

(9) Processing is as follows if using reference axis speed designation or major axis reference designation with linear interpolation control.
  - The major axis is not reselected when changing the target position. The same major axis as that prior to the target position changed is used.
  - The positioning speed is recalculated based on the travel value for each axis following the target position change.
  - If the reference axis or major axis travel value becomes 0 due to the target position change, a minor error (error code: 264) occurs, and deceleration stops.

(10) By executing a CHGP command during constant speed control (CPSTART), positioning is performed at the changed target position.
Positioning is not performed at the points after the point being executed when a target position change request is made.

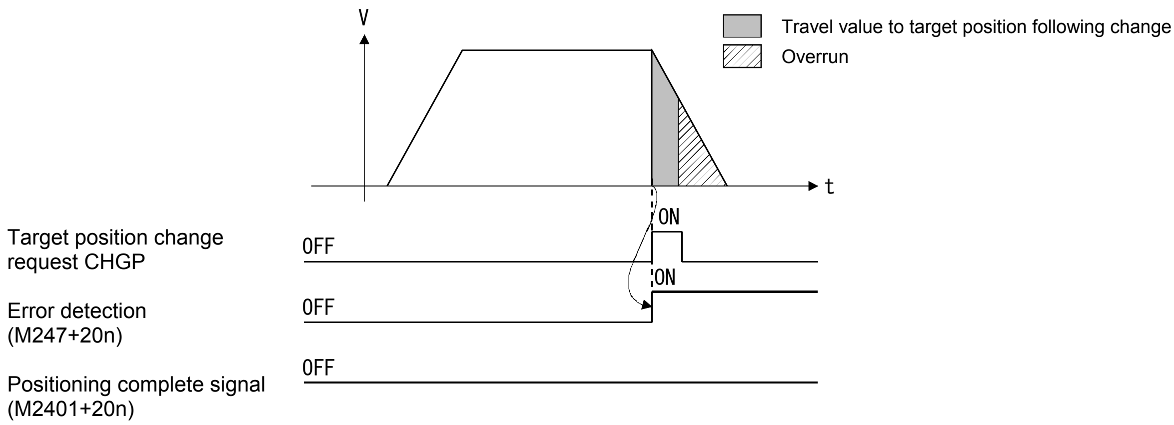


POINT

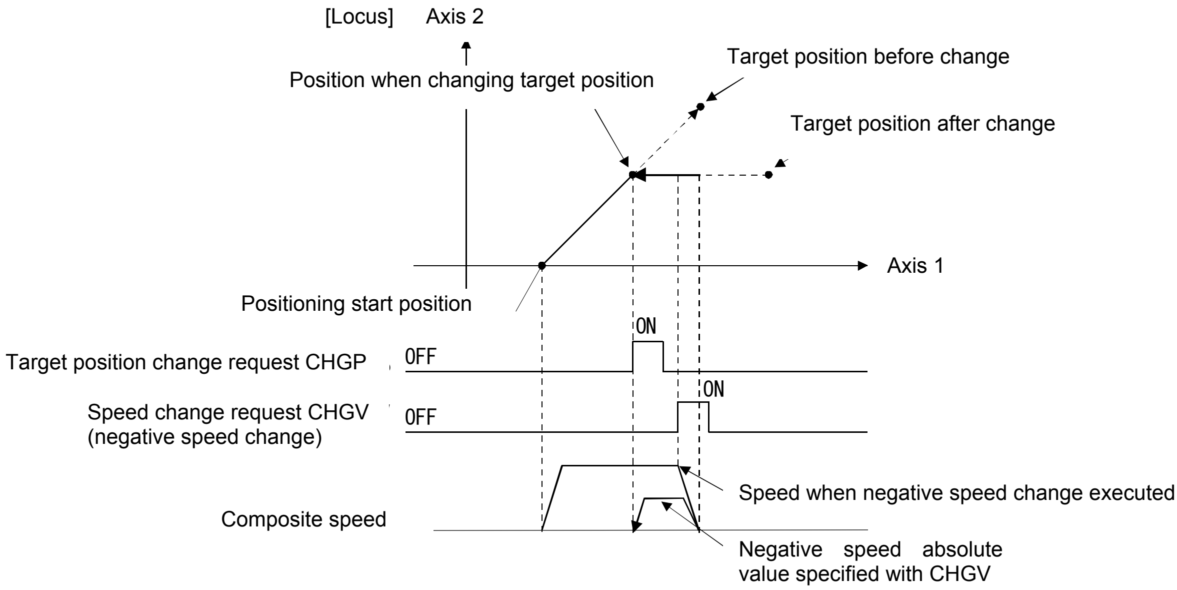
(1) By executing the CHGP command, setting items for the point for which positioning is currently being performed are taken over, and positioning is performed.
(2) The CHGP command is used to perform linear interpolation control for all axes specified with CPSTART, and therefore it is necessary to set target positions for all axes specified with CPSTART.
(3) If the CHGP command is executed while positioning at the circular interpolation or helical interpolation point when performing constant speed control, positioning at the circular interpolation and helical interpolation points is completed, and the target position is then changed at the same time as positioning at the linear interpolation point is started.

(11) The operation if a target position change request is made with the address designation method for axes for which the control unit is [degrees] is as follows.
- Positioning is performed at the address following the change with the current travel direction unchanged.
- If using the address designation method, set the change address from 0 to 35999999 × 10-5 [degrees]. If set outside the range, a minor error (error code: 260) occurs, and deceleration stops.

(12) By executing the CHGP command, the operation if the travel value to the target position following the change is smaller than the deceleration distance required to stop deceleration from the speed applied during control is as follows.
- A minor error (error code: 261) occurs, and deceleration stops the moment the CHGP command is executed.
- The difference between the travel value to the deceleration stoppage until the target position following the change is an overrun.
- The positioning complete signal (M2401+20n) does not turn ON.



(13) If a negative speed change is performed after executing the CHGP command, the axis decelerates to a speed of 0, and when decelerating is complete, the axis returns to and stops (stands by) at the position where the target position change (when CHGP command received) was made when performing linear interpolation.



[Error]

(1) An operation error occurs in the following cases, and the target position is not changed.
- When the (S1) designated axis No. lies outside the range.
- When a value outside the 0 to 1 range is specified with (S2).
- When (S3) is other than an even-numbered device.
- When the (S3) to (S3) + 7 device No. lies outside the range.

(2) A minor error occurs in the following cases, and the target position is not changed.
- When home zeroing is being performed for the relevant axis. (Minor error: 330)
- When executing a servo program that does permit the target position of the relevant axis to be changed. (Minor error: 330)
- When the target position following the change exceeds the stroke limit range. (Minor error: 262)
- When the FIN acceleration/deceleration or advanced S-curve acceleration/deceleration is set for the acceleration/deceleration system. (Minor error: 263)
- When the travel value for the reference axis or major axis becomes 0 if reference axis speed and major axis reference have been designated when performing linear interpolation control. (Minor error: 264)
- When the change address lies outside the 0 to 35999999 × 10-5 [degrees] if an address designation method target position change request for axes for which the control unit is [degrees]. (Minor error: 260)
- When the travel value to the target position following the target position change is smaller than that required to stop deceleration from the speed during control. (Minor error: 261)

[Program example]

(1) Program when changing the target position by travel value designation for axes 2 and 8 during positioning with ABS-2

[Servo program]

```
<K50>
ABS-2
  Axis 2        20000
  Axis 8        10000
  Speed          3000
```

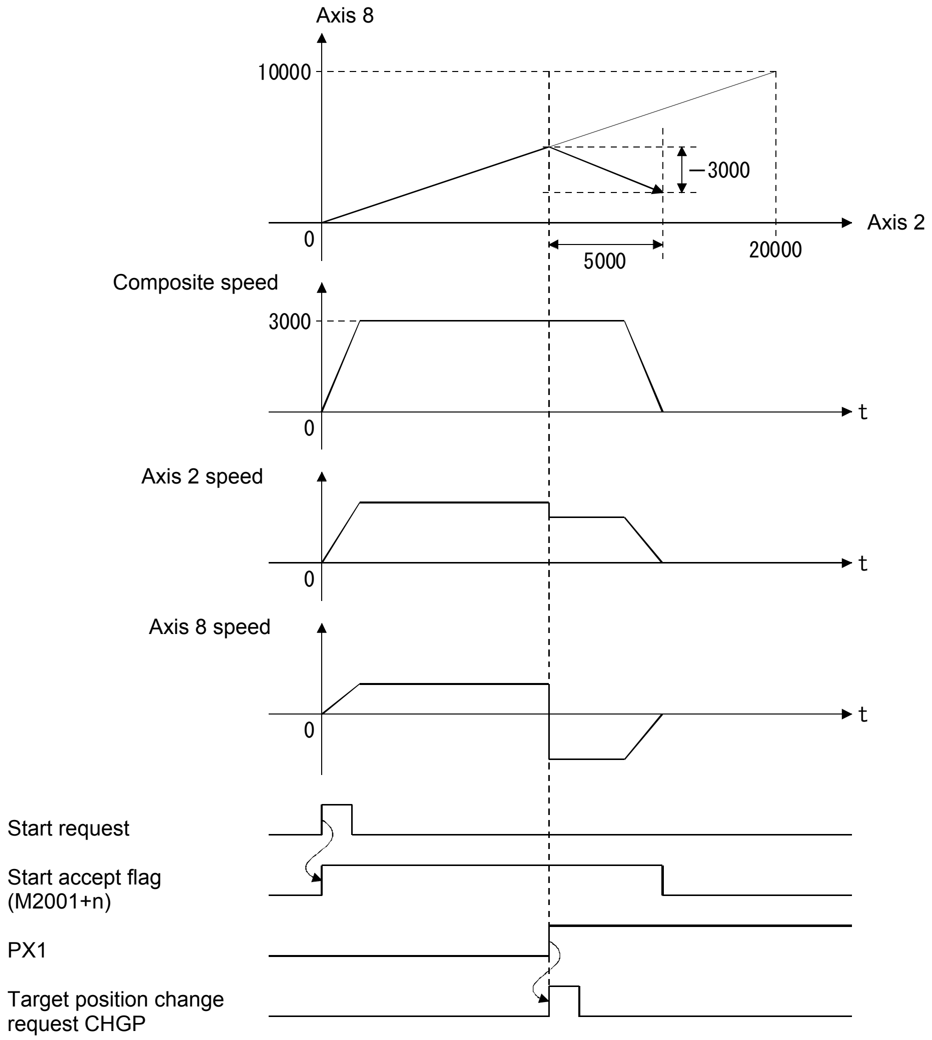
[Motion SFC program]

```
[F10]  D3000L=K5000    //Axis 2 target change position
       D3002L=K-3000   //Axis 8 target change position

[G10]  //Stand by until PX1 turns ON

[F20]  //Target position change by
       specifying travel value for
       axes 2, 8
```

[Locus]



F/FS	G
○	○

Appendix 7.10 Other Commands

Appendix 7.10.1 Event task authorized: EI

Format	EI	No. of basic steps	1

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
-	-	-	-	-	-	-	-	-	-	-	-

○: Setting possible

[Setting data]

There is no setting data.

[Function]

(1) Authorizes event task execution.

(2) Can only be used with normal tasks.

[Error]

(1) An operation error occurs in the following cases.
- When used with other than normal task.

[Program example]

(1) Authorizes event task execution.

```
EI
```

F/FS	G
○	○

Appendix 7.10.2 Event task prohibited: DI

Format	DI

No. of basic steps	1

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
-	-	-	-	-	-	-	-	-	-	-	-

○: Setting possible

[Setting data]

There is no setting data.

[Function]

(1) Prohibits event task execution.

(2) If an external interrupt or PLC interrupt occurs after executing the DI command, the corresponding event task is executed once when the EI command is executed. (If an external interrupt or PLC interrupt occurs multiple times while executing the DI command, the corresponding event task is executed once only when the EI command is executed.)

(3) Fixed cycle events are not executed during DI.

(4) The execution of NMI tasks cannot be prohibited.

(5) The status becomes the DI status when the multiple CPU system power is turned ON or reset.

[Error]

(1) An operation error occurs in the following cases.
- When used with other than normal task.

[Program example]

(1) Program in which event task execution is prohibited

```
DI
```

F/FS	G
○	○

Appendix 7.10.3 No processing: NOP

Format	NOP	No. of basic steps	1

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
-	-	-	-	-	-	-	-	-	-	-	-

○: Setting possible

[Setting data]

There is no setting data.

[Function]

(1) With a no processing command, there is no effect on operations performed thus far.

[Error]

(1) No processing: There is no NOP operation error.

F/FS	G
○	○

Appendix 7.10.4 Block transfer: BMOV

Format	BMOV (D),(S),(n)

No. of basic steps	6

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(D)	○	○	-	-	-	-	○	-	-	-	-
(S)	○	○	-	-	-	-	○	-	-	-	-
(n)	-	○	-	-	-	○	-	-	-	-	-

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(D)	First No. of transfer destination device	-
(S)	First No. of transfer origin device	
(n)	No. of transfer words	

[Function]

(1) The n word content from the device specified with (S) is batch transferred to the n word from the device specified with (D).

(2) Transfer is possible even if the transfer origin and transfer destination devices overlap. If transferring to the device with smaller number, data is transferred from (S), and if transferring to the device with larger number, data is transferred from (S) + (n - 1).

(3) By specifying Nn (cam No.) for (D) or (S), cam data can be batch transferred. It is necessary that cam data for the same cam No. already be registered in the motion controller. Ensure that the number of transfer words specified with (n) matches the specified cam No. resolution.

When writing cam data

The cam data storage area is rewritten.

- Transfer of data to the cam data area is also performed during cam operation. Data is not written while performing operation with the same cam No., and therefore caution is advised.

When reading cam data

- Cam data in the currently set condition is read.

(4) Devices that can be set for (D), (S), and (n) are as follows.

Setting data	Word device [*2]					Bit device [*2, *3]							Cam No. designation
	Dn	Wn	SDn	U□\Gn	#n	Mn	U□\Gn.m	Bn	Fn	SMn	Xn	Yn	Nn [*1]
(D)	○	○	-	○	○	○	-	○	○	-	○[*4]	○[*4]	○
(S)	○	○	○	○	○	○	-	○	○	○	○[*4]	○[*4]	○
(n)	○	○	-	○	○	-	-	-	-	-	-	-	-

*1: Nn indicates the cam No.
*2: Indirect designation is not possible for device Nos.
*3: Device Nos. for bit data are specified in multiples of 16.
*4: PX and PY cannot be set.

(5) The range for cam Nos. that can be set with Nn is as follows.

Q173D(S)CPU/Q172D(S)CPU
1 to 64
101 to 164
201 to 264
301 to 364

[Error]

(1) An operation error occurs in the following cases.
- When cam data for cam Nos. specified with (D) and (S) have not been registered in the motion controller.
- When the resolution for cam Nos. specified with (D) and (S) differs from the number of transfer words specified with (n).
- When (S) to (S) + (n - 1) lies outside the device range.
- When (D) to (D) + (n - 1) lies outside the device range.
- When (n) is 0 or a negative number.
- When PX and PY settings exist for (S) to (S) + (n - 1).
- When PX and PY settings exist for (D) to (D) + (n - 1).

When (n) is a word device designation.

(2) An error occurs in the following cases if motion SFC program conversion is performed at MT Developer□.
- When (S) to (S) + (n - 1) lies outside the device range.
- When (D) to (D) + (n - 1) lies outside the device range.
- When (n) is 0 or a negative number.
- When PX and PY settings exist for (S) to (S) + (n - 1).
- When PX and PY settings exist for (D) to (D) + (n - 1).

When (n) is a constant designation

- When (S) is a bit device, and the device No. is not a multiple of 16.
- When (D) is a bit device, and the device No. is not a multiple of 16.

[Program example]

(1) Program in which 5 word content from D0 is batch transferred from #10 to the 5 words

```
BMOV #10,D0,K5
```

		Batch transfer ←		
#10	12		D0	12
#11	34		D1	34
#12	56		D2	56
#13	78		D3	78
#14	90		D4	90

(2) Program in which 2048 word content from #0 is batch transferred to the cam No.2 (resolution 2048) data area

```
BMOV N2,#0,K2048
```

	Cam No.2 cam data	Batch transfer ←		
No.0 stroke ratio	H0000		#0	H0000
No.1 stroke ratio	H0005		#1	H0005
No.2 stroke ratio	H000A		#2	H000A
:	:		:	:
No.2047 stroke ratio	H0000		#2047	H0000

> **POINT**
>
> The cam stroke ratio is set in the 0 to 7FFFH range.

(3) Program in which 4 word content from X0 is batch transferred from #20 to the 4 words

```
BMOV #20,X0,K4
```

	b15 ― b0	Batch transfer ←	
#20	0 0 1 0 0 1 1 1 0 0 0 0 1 1 1 1	XF ― X0	0 0 1 0 0 1 1 1 0 0 0 0 1 1 1 1
#21	1 0 0 0 0 0 1 1 0 0 0 0 0 1 1 1	X1F ― X10	1 0 0 0 0 0 1 1 0 0 0 0 0 1 1 1
#22	0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1	X2F ― X20	0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1
#23	1 1 0 0 0 1 1 1 0 0 0 0 0 0 0 0	X3F ― X30	1 1 0 0 0 1 1 1 0 0 0 0 0 0 0 0

F/FS	G
○	○

Appendix 7.10.5 Same data block transfer: FMOV

Format	FMOV (D),(S),(n)

No. of basic steps	6

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(D)	○	○	-	-	-	-	○	-	-	-	-
(S)	○	○	-	-	-	○	-	-	-	-	-
(n)	-	○	-	-	-	○	-	-	-	-	-

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(D)	First No. of transfer destination device	-
(S)	Transfer data, or device No. in which data to be transferred is stored	
(n)	No. of transfer words	

[Function]

(1) The data or device content specified with (S) is (n) word transferred to the device specified with (D).

(2) Devices that can be set for (D), (S), and (n) are as follows.

Setting data	Word device *1					Bit device *1, *2						
	Dn	Wn	SDn	U□\Gn	#n	Mn	U□\Gn.m	Bn	Fn	SMn	Xn	Yn
(D)	○	○	-	○	○	○	-	○	○	-	○*3	○*3
(S)	○	○	○	○	○	○	-	○	○	○	○*3	○*3
(n)	○	○	-	○	○	-	-	-	-	-	-	-

*1: Indirect designation is not possible for device Nos.
*2: Device Nos. for bit data are specified in multiples of 16.
*3: PX and PY cannot be set.

[Error]

(1) An operation error occurs in the following cases.
- When (D) to (D) + (n - 1) lies outside the device range.
- When (n) is 0 or a negative number.
- When PX and PY settings exist for (D) to (D) + (n - 1).

When (n) is a word device designation.

(2) An error occurs in the following cases if motion SFC program conversion is performed with MT Developer□.
- When (D) to (D) + (n - 1) lies outside the device range.
- When (S) lies outside the device range.
- When (n) is 0 or a negative number.
- When PX and PY settings exist for (S).
- When PX and PY settings exist for (D) to (D) + (n - 1).

When (n) is a constant designation

- When (S) is a bit device, and the device No. is not a multiple of 16.
- When (D) is a bit device, and the device No. is not a multiple of 16.

[Program example]

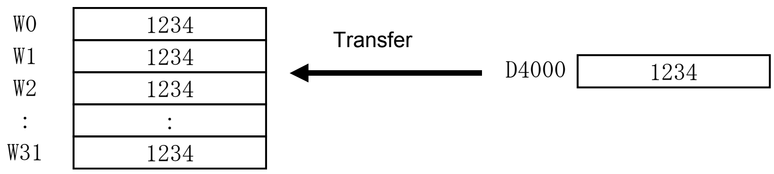
(1) Program in which all 3456H is set in the 100 word section from #10

```
FMOV #10,H3456,K100
```

m,km

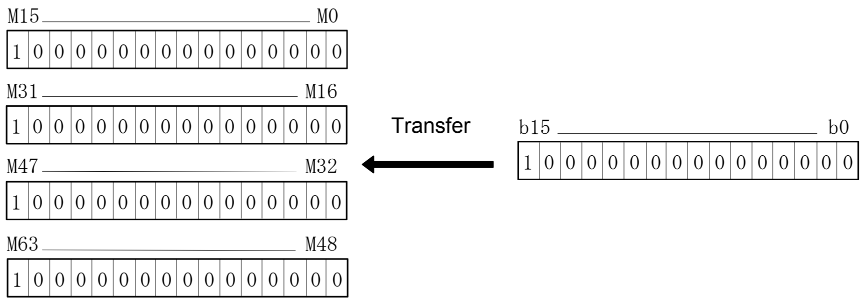
(2) Program in which the D4000 content is set in the 50 word section from W0

```
FMOV W0,D4000,K50
```



(3) Program in which all 8000H is set in the 4 word section from M0

```
FMOV M0,H8000,K4
```



F/FS	G
○	○

Appendix 7.10.6 Data writing to self CPU shared memory: MULTW

Format	MULTW (D),(S),(n),(D1)

No. of basic steps	8

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(D)	-	○	-	-	-	○	-	-	-	-	-
(S)	○	○	-	-	-	-	-	-	-	-	-
(n)	-	○	-	-	-	○	-	-	-	-	-
(D1)	○	-	-	-	-	-	-	-	-	-	-

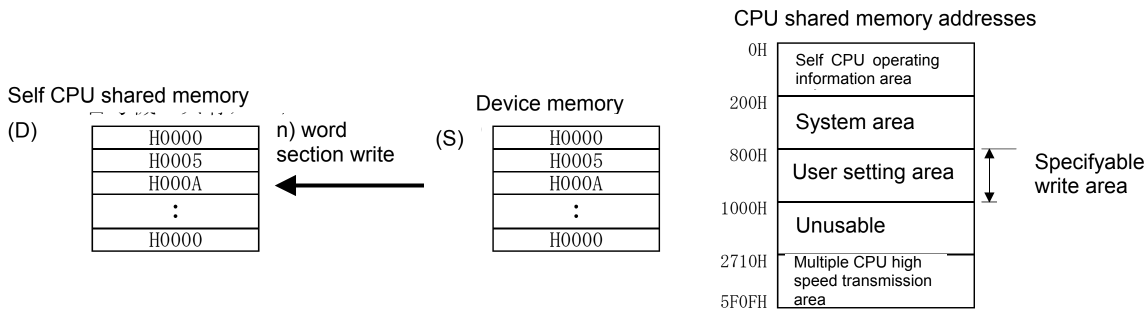
○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(D)	Self CPU shared memory address for write destination (800H to FFFH)	-
(S)	No. of first device in which write data is stored	
(n)	No. of write words (1 to 256)	
(D1)	Self CPU device turned ON when writing complete	

[Function]

(1) Writes the (n) word section of data for devices specified with the self CPU unit (S) and onward to the CPU shared memory address specified with the self CPU unit (D) and onward.
When writing is complete, the completed bit device specified with (D1) turns ON.



(2) Reset completed bit devices at the user program.

(3) Other MULTW commands cannot be processed until the MULTW command is executed and the completed bit device turns ON. If the MULTW command is executed again during the period of time from when the MULTW command is executed until the completed bit device turns ON, an error will occur for subsequently executed MULTW commands.

(4) Devices that can be set for (D), (S), (n), and (D1) are as follows.

Setting data	Word device *1					Bit device *1, *2						
	Dn	Wn	SDn	U□\Gn	#n	Mn	U□\Gn.m	Bn	Fn	SMn	Xn	Yn
(D)	○	○	-	○	○	-	-	-	-	-	-	-
(S)	○	○	-	○	○	○	-	○	○	-	○ *3	○ *3
(n)	○	○	-	○	○	-	-	-	-	-	-	-
(D1)	-	-	-	-	-	○	○	○	○	-	○ *4	○ *4

*1: Indirect designation is not possible for device Nos.
*2: Device Nos. for bit data are specified in multiples of 16.
*3: PX and PY cannot be set.
*4: PY setting is also possible. PX cannot be set.

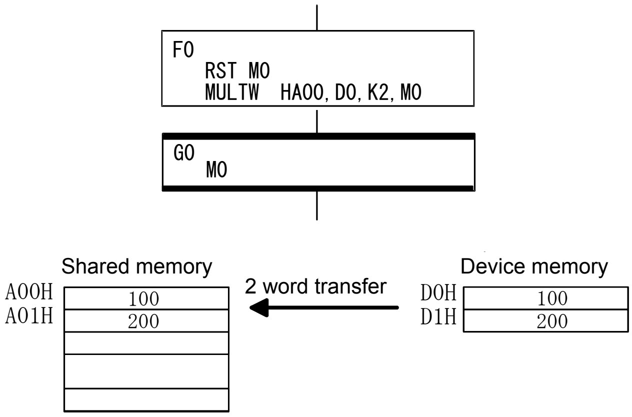
(5) With this command, processing time becomes longer in proportion to the number of write words (n), and execution tasks and the number of transfer words should be adjusted by referring to the operation processing time in order to prevent from obstructing the execution of motion operation.

[Error]

(1) An operation error occurs in the following cases.
- When the number of write words (n) lies outside the 1 to 256 range.
- When the write destination self CPU shared memory address (D) lies outside the CPU shared memory address (800H to FFFH) range.
- When the write destination self CPU shared memory address (D) + number of write words (n) lie outside the CPU shared memory address (800H to FFFH) range.
- The first device No. (S) in which the write data is stored + number of write words (n) lie outside the device range.
- When the MULTW command is executed again during the period of time from when the MULTW command is executed until the completed bit device turns ON.
- (D) is a device for which writing is not possible.
- When (S) is a bit device, and the device No. is not a multiple of 16.
- When PX and PY settings exist for (S) to (S) + (n - 1).

[Program example]

(1) Writes 2 words from D0 to shared memory A00H and onward, and processing proceeds to the next step after write completion is confirmed.



F/FS	G
○	○

Appendix 7.10.7 Data reading from shared memory: MULTR

Format	MULTR (D),(S1),(S2),(n)	No. of basic steps	7

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(D)	○	○	-	-	-	-	-	-	-	-	-
(S1)	-	○	-	-	-	○	-	-	-	-	-
(S2)	-	○	-	-	-	○	-	-	-	-	-
(n)	-	○	-	-	-	○	-	-	-	-	-

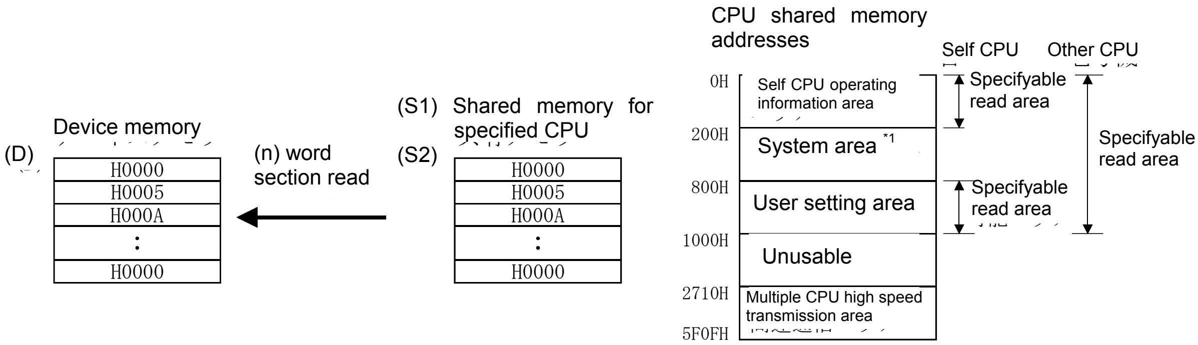
○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(D)	No. of first device in which read data is stored	-
(S1)	First I/O number of PLC CPU, motion CPU from which data is read (No.1 CPU: 3E0H, No.2 CPU: 3E1H, No.3 CPU: 3E2H, No.4 CPU: 3E3H)	
(S2)	CPU shared memory first address for data to be read (0H to FFFH)	
(n)	No. of read words (1 to 256)	

[Function]

(1) Reads (n) word data from the address specified with CPU shared memory (S2) in the applicable CPU specified with (S1), and stores it in the device specified with (D) onward.



*1: Read is not possible if the applicable CPU is a self CPU.

(2) Devices that can be set for (D), (S1), (S2), and (n) are as follows.

Setting data	Word device *1					Bit device *1, *2						
	Dn	Wn	SDn	U□\Gn	#n	Mn	U□\Gn.m	Bn	Fn	SMn	Xn	Yn
(D)	○	○	-	○	○	○	-	○	○	-	○*3	○*3
(S1)	○	○	-	○	○	-	-	-	-	-	-	-
(S2)	○	○	-	○	○	-	-	-	-	-	-	-
(n)	○	○	-	○	○	-	-	-	-	-	-	-

*1: Indirect designation is not possible for device Nos.
*2: Device Nos. for bit data are specified in multiples of 16.
*3: PX and PY cannot be set.

(3) If reading is completed successfully from the applicable CPU No. specified with (S1), read completion flags SM528 to SM531 (No.1 CPU: SM528, No.2 CPU: SM529, No.3 CPU: SM530, No.4 CPU: SM531) corresponding to the applicable CPU number turn ON. If reading is unsuccessful, the read completion flag for the applicable CPU No. specified with (S1) does not turn ON.

(4) With this command, processing time becomes longer in proportion to the number of read words (n), and execution tasks and the number of transfer words should be adjusted by referring to the operation processing time in order to prevent from obstructing the execution of motion operation.

(5) If multiple MULTR commands are executed simultaneously for the same CPU, read completion flag SM528 to SM531 for the applicable CPU turns ON depending on the result of the MULTR command executed last.

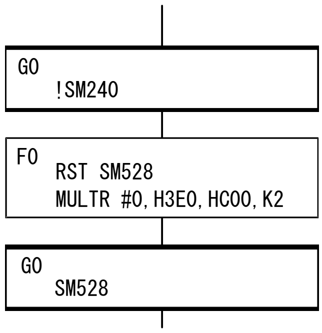
(6) Reset read completion flag SM528 to SM531 at the user program.

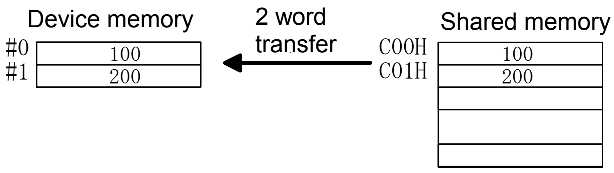
[Error]

(1) An operation error occurs in the following cases.
- When the number of read words (n) lies outside the 1 to 256 range.
- When the read data CPU shared memory first address (S2) lies outside the CPU shared memory address (0H to FFFH) range.
- When the read data CPU shared memory first address (S2) + number of read words (n) lie outside the CPU shared memory address (0H to FFFH) range.
- The first device No. (D) in which the read data is stored + number of read words (n) lie outside the device range.
- When other than 3E0H, 3E1H, 3E2H, or 3E3H is set with (S1).
- When the CPU performing reading is being reset.
- When an error is detected at the CPU performing reading.
- When (D) is a bit device, and the device No. is not a multiple of 16.
- When PX and PY settings exist for (D) to (D) + (n - 1).

[Program example]

(1) Confirms that the No.1 CPU is not being reset, reads 2 words to #0 onward from No.1 CPU shared memory C00H, and processing proceeds to the next step after write completion is confirmed.





F/FS	G
○	○

Appendix 7.10.8 Word data writing to intelligent function module: TO

Format	TO (D1),(D2),(S),(n)

No. of basic steps	7

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(D1)	-	○	-	-	-	○	-	-	-	-	-
(D2)	-	○	-	-	-	○	-	-	-	-	-
(S)	○	○	-	-	-	-	-	-	-	-	-
(n)	-	○	-	-	-	○	-	-	-	-	-

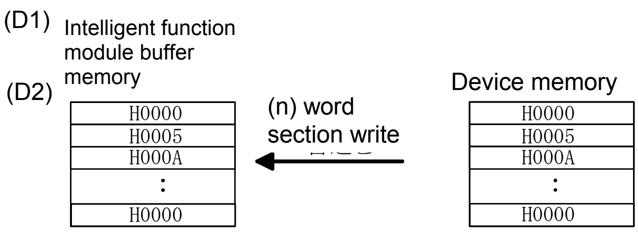
○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(D1)	Intelligent function module first I/O No. (000H to FF0H)	-
(D2)	First address in buffer memory to which data is written	
(S)	No. of first device in which write data is stored	
(n)	No. of write words (1 to 256)	

[Function]

(1) Writes (n) word data from the device specified with (S) to the address specified with (D2) in the buffer memory inside the intelligent function module managed by the self CPU specified with (D1) and onward.



(2) (D1) specifies the first I/O number for the module specified in the system settings.

Power unit	Q03UD CPU	Q173DS CPU	QX40	Q64AD	Q64DAN	
			First I/O No.	First I/O No.	First I/O No.	
			No. : 00H	No. : 10H	No. : 20H	

If the TO command is executed for the D/A conversion module (Q64DA) with the above mentioned system setting, (D1) will be 20H.

(3) Devices that can be set for (D), (D2), (S), and (n) are as follows.

Setting data	Word device *1					Bit device *1, *2						
	Dn	Wn	SDn	U□\Gn	#n	Mn	U□\Gn.m	Bn	Fn	SMn	Xn	Yn
(D1)	○	○	-	○	○	-	-	-	-	-	-	-
(D2)	○	○	-	○	○	-	-	-	-	-	-	-
(S)	○	○	-	○	○	○	-	○	○	-	○*3	○*3
(n)	○	○	-	○	○	-	-	-	-	-	-	-

*1: Indirect designation is not possible for device Nos.
*2: Device Nos. for bit data are specified in multiples of 16.
*3: PX and PY cannot be set.

(4) With this command, processing time becomes longer in proportion to the number of write words (n), and execution tasks and the number of transfer words should be adjusted by referring to the operation processing time in order to prevent from obstructing the execution of motion operation.

(5) Only the following analog modules can be used as motion CPU control modules.
- Analog input (Q68ADV, Q62AD-DGH, Q66AD-DG, Q68ADI, Q64AD, Q64AD-GH, Q68AD-G)
- Analog output (Q68DAVN, Q68DAIN, Q62DAN, Q62DA-FG, Q64DAN, Q66DA-G)

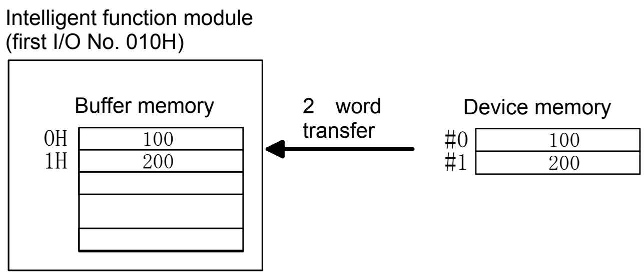
[Error]

(1) An operation error occurs in the following cases.
- When the number of write words (n) lies outside the 1 to 256 range.
- When unable to communicate with the intelligent function module when executing the command.
- When an intelligent function module error is detected when executing the command.
- When the I/O No. specified with (D1) is not an intelligent function module controlled by a self CPU.
- When the address specified with (D2) lies outside the buffer memory range.
- The first device No. (S) in which the write data is stored + number of write words (n) lie outside the device range.
- When (S) is a bit device, and the device No. is not a multiple of 16.
- When PX and PY settings exist for (S) to (S) + (n - 1).

[Program example]

(1) Writes 2 words from #0 to intelligent function module (first I/O No. 010H) buffer memory address 0H.

```
TO H010,H0,#0,K2
```



F/FS	G
○	○

Appendix 7.10.9 Word data reading from intelligent function module: FROM

Format	FROM (D),(S1),(S2),(n)

No. of basic steps	7

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(D)	○	○	-	-	-	-	-	-	-	-	-
(S1)	-	○	-	-	-	○	-	-	-	-	-
(S2)	-	○	-	-	-	○	-	-	-	-	-
(n)	-	○	-	-	-	○	-	-	-	-	-

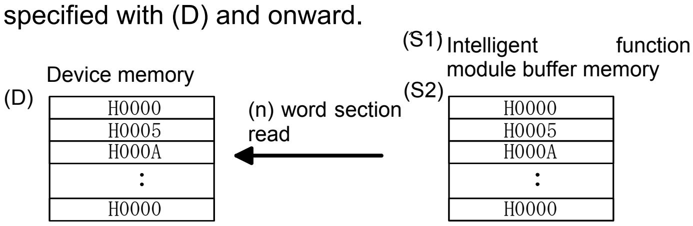
○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(D)	No. of first device in which read data is stored	-
(S1)	Intelligent function module first I/O No. (000H to FF0H)	
(S2)	First address in buffer memory from which data is read	
(n)	No. of read words (1 to 256)	

[Function]

(1) Reads (n) word data from the address specified with (S2) in the buffer memory inside the intelligent function module controlled by the self CPU specified with (S1), and writes it to the device specified with (D) and onward.



(2) (S1) specifies the first I/O number for the module specified in the system settings.

	Q03UD CPU	Q173DS CPU	QX40 First I/O No. No. : 00H	Q64AD First I/O No. No. : 10H	Q64DAN First I/O No. No. : 20H	

If the FROM command is executed for the A/D conversion module (Q64AD) with the above mentioned system setting, (S1) will be 10H.

(3) Devices that can be set for (D), (S1), (S2), and (n) are as follows.

Setting data	Word device *1					Bit device *1, *2						
	Dn	Wn	SDn	U□\Gn	#n	Mn	U□\Gn.m	Bn	Fn	SMn	Xn	Yn
(D)	○	○	-	○	○	○	-	○	○	-	○*3	○*3
(S1)	○	○	-	○	○	-	-	-	-	-	-	-
(S2)	○	○	-	○	○	-	-	-	-	-	-	-
(n)	○	○	-	○	○	-	-	-	-	-	-	-

*1: Indirect designation is not possible for device Nos.
*2: Device Nos. for bit data are specified in multiples of 16.
*3: PX and PY cannot be set.

(4) With this command, processing time becomes longer in proportion to the number of read words (n), and execution tasks and the number of transfer words should be adjusted by referring to the operation processing time in order to prevent from obstructing the execution of motion operation.

(5) Only the following analog modules can be used as motion CPU control modules.
- Analog input (Q68ADV, Q62AD-DGH, Q66AD-DG, Q68ADI, Q64AD, Q64AD-GH, Q68AD-G)
- Analog output (Q68DAVN, Q68DAIN, Q62DAN, Q62DA-FG, Q64DAN, Q66DA-G)

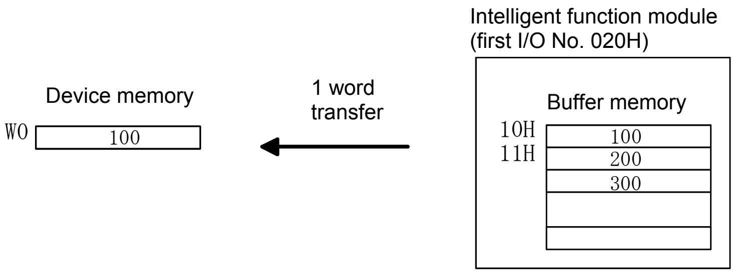
[Error]

(1) An operation error occurs in the following cases.
- When the number of read words (n) lies outside the 1 to 256 range.
- When unable to communicate with the intelligent function module when executing the command.
- When an intelligent function module error is detected when executing the command.
- When the I/O No. specified with (S1) is not an intelligent function module controlled by a self CPU.
- When the address specified with (S2) lies outside the buffer memory range.
- The first device No. (D) in which the read data is stored + number of read words (n) lie outside the device range.
- When (D) is a bit device, and the device No. is not a multiple of 16.
- When PX and PY settings exist for (D) to (D) + (n - 1).

[Program example]

(1) Reads 1 word from intelligent function module (first I/O No. 020H) buffer memory address 10H, and stores it in W0.

```
FROM W0,H020,H10,K1
```



F/FS	G
-	○

Appendix 7.10.10 Time wait: TIME

Format	TIME(S)	No. of basic steps	7

[Usable data]

Setting data	Usable data										
	Bit device	Word device				Constant			Calculation formula	Bit conditional expression	Comparative conditional expression
		16-bit integer type	32-bit integer type (L)	64-bit floating-point type (F)	Coasting timer	16-bit integer type (K/H)	32-bit integer type (K/H, L)	64-bit floating-point type (K)			
(S)	-	○	○	-	-	○	○	-	-	-	-

○: Setting possible

[Setting data]

Setting data	Details	Resultant data type
(S)	Wait time (0 to 2147483647) [ms]	Logical type (true/false)

[Function]

(1) Waits the length of time specified with (S).
When the elapsed time is less than the set time, the result is false, and when longer than or equal to the set time, the result is true, and processing continues.

(2) If (S) is specified with a 16-bit integer type word device and there are times when the time is specified between 32768 and 65535 [ms], perform 32-bit integer value conversion without symbol with ULONG. (See program example.)

[Error]

(1) An operation error occurs in the following cases.
- When (S) is an indirect designation device, and the device No. lies outside the range.
- When the data specified with (S) (device data when indirect designation) lies outside the 0 to 2147483647 range.

[Program example]

(1) Program with wait time of 60 seconds (when constant designation)

```
TIME K60000
```

(2) Program with 16-bit integer type indirect designation (#0) and in which the wait time can be between 32768 and 65535 [ms]

```
TIME ULONG(#0)
```

(3) Program in which the bit device is set (reset) when the specified time or longer has elapsed, and processing continues

```
SET M100 = TIME K60000
```

POINT

(1) If the wait time is designated indirectly with a word device, the device value is controlled with the value loaded first. The set time cannot be changed, even if the device value is changed during the wait time status.
(2) The TIME command is the equivalent of a conditional expression, and therefore can only be set in the last line of transition (G) programs.
(3) If transition programs (Gn) with same number for which the TIME command is set are used with multiple motion SFC programs, ensure that the programs are not run simultaneously. (If run simultaneously, the wait time for the program run first will be illegal.)
(4) If the transition program (Gn) is of another number, the TIME command can be executed simultaneously with multiple motion SFC programs. (The maximum number of simultaneous active steps is 256.)
(5) The wait time cannot be canceled during the wait time specified by executing the TIME command.

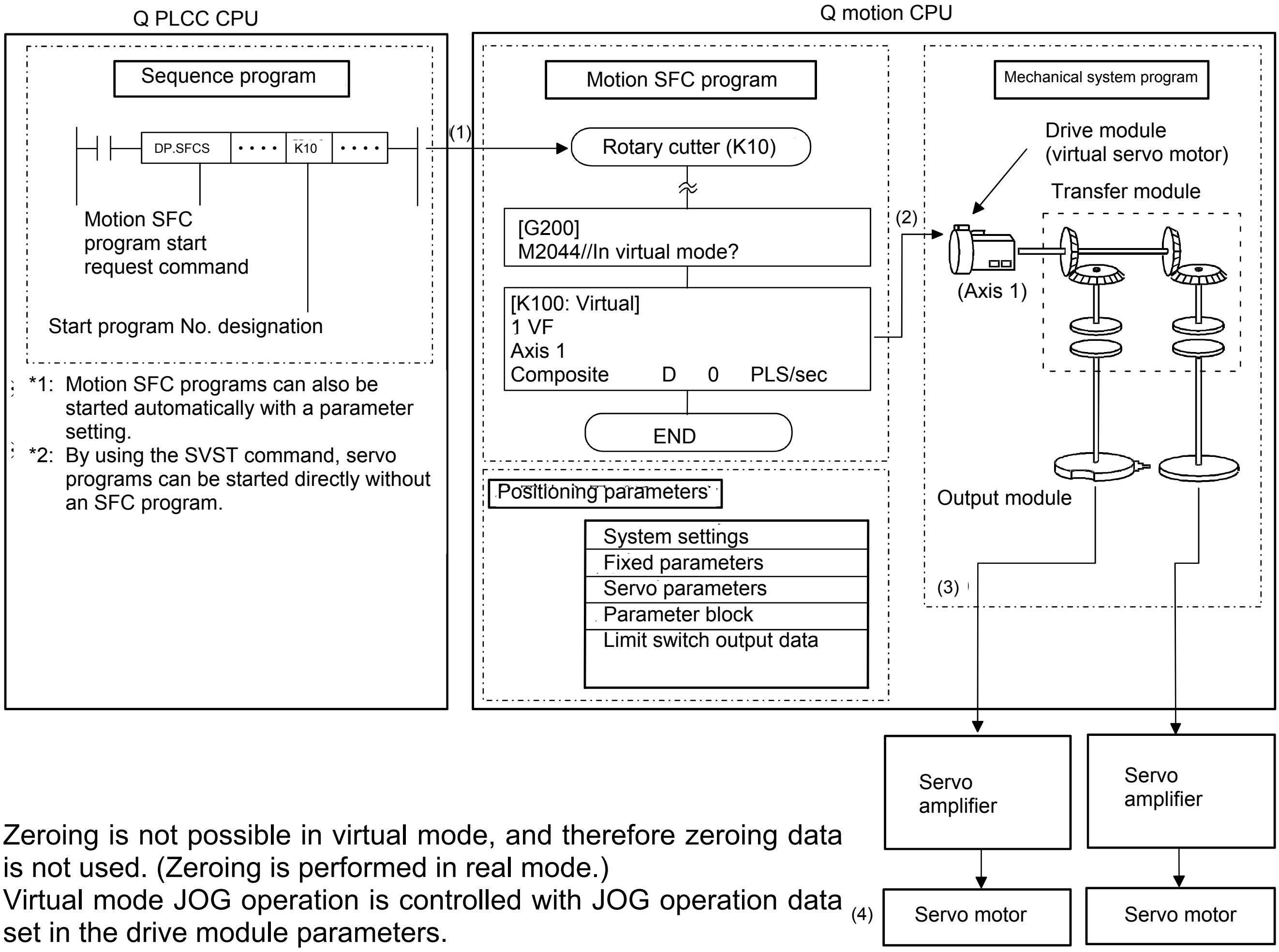
Appendix 8 Overview of Virtual Mode Control for SV22 Automatic Machines

(a) Virtual mode uses software to perform synchronous control processing with a mechanical system program comprised of a virtual main shaft and mechanical module. By using virtual mode, it is possible to switch from the previous synchronous control performed with a mechanical system using a main shaft, gears, and cams, etc. to positioning control using servo motors.

(b) With virtual mode, a mechanical system program is required in addition to the positioning parameters, servo programs, and motion SFC programs used with real mode.

(c) The procedure when performing positioning control in virtual mode is as follows.

1) A virtual mode motion SFC program start request is issued with a sequence program SFCS command.
↓
2) The mechanical system program virtual servo motors start.
↓
3) The results of operations performed through a transfer module are output to the servo amplifier set in the output module.
↓
4) Servo motors are controlled.



- Zeroing is not possible in virtual mode, and therefore zeroing data is not used. (Zeroing is performed in real mode.)
- Virtual mode JOG operation is controlled with JOG operation data set in the drive module parameters.
- External synchronous encoder pulses are input to the synchronous encoder input unit or manual pulse generator input unit, allowing the mechanical system program synchronous encoder to be operated.

Appendix 9 Glossary

A

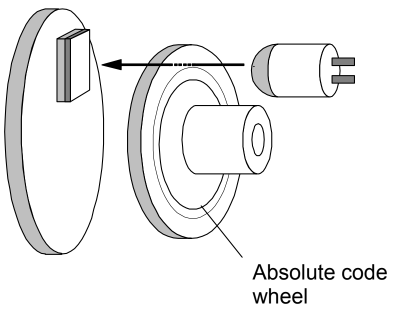
This means cam non-dimensional acceleration.
Non-dimensional acceleration is non-dimensional speed differentiated by non-dimensional time.
The maximum value is expressed with Am.
See "Am".
See "V".

5th power polynomial curve

This curve has five boundary conditions, is smooth, and possesses excellent characteristics.

Absolute encoder

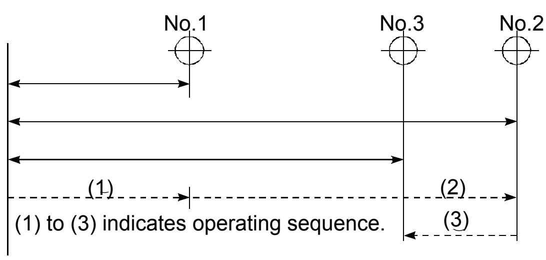
This is an absolute position detector that allows angular data contained in a single motor rotation to be output externally, and standard encoders allow 360 degrees to be extracted in 8 to 12 bits.
With incremental encoders, the axis position when a power outage occurs is lost, however, with absolute encoders, the axis position is retained, even in the event of a power outage.
See "Encoder".



Angular data contained in a single rotation is known the instant the power is turned back ON again, however, data for multiple rotations (how may rotations were made) is backed up with a battery.

Absolute mode

This is a method used to express the positioning address. This is an absolute address method.
This method expresses the distance from the reference 0. The positioning direction is determined automatically without being specified. There is also an incremental mode.



Absolute position system

By zeroing once when starting up positioning control devices, current values are backed up with a battery even when the power is turned OFF, and machine displacements are compensated.
Consequently, there is no need to perform zeroing after turning ON the power.
To construct this system, a servo amplifier compatible with servo motor with absolute position detector is required.

AC motor drive unit

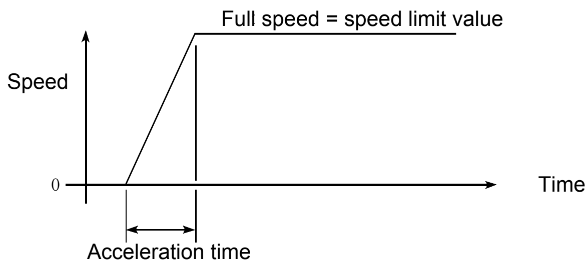
This is a built-in servo amplifier capable of being connected to and driving a single servo motor.

Acceleration

Acceleration is speed differentiated by time, and expresses the rate of change of speed.
Furthermore, acceleration is proportional to force.
See "A".

Acceleration time

This is the time taken to reach full speed from the stopped status with the motion controller.
The parameter acceleration time is the time taken to reach the speed limit value, and therefore becomes proportionally shorter if the set speed is low. It is determined by such factors as the machine inertia and motor torque, and load resistance torque.

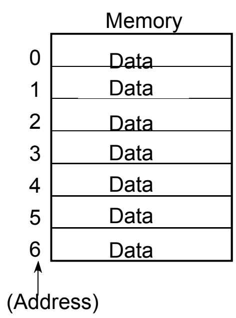


Actual current value

This is the actual servo travel amount pulse count calculated from feedback pulses.

Address

(1) Memory address Memory holds addresses, and data is written and read by specifying these addresses.



(2) Numerical value indicating the target position when performing positioning. Units are set in mm, inches, degrees, or pulses.

Am acceleration

This is the cam non-dimensional acceleration maximum value.
See "A".

Analog command

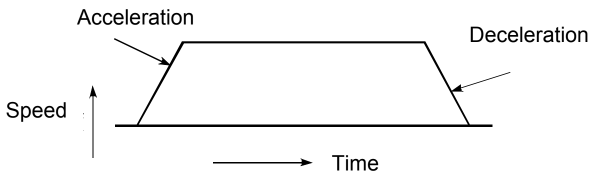
Converts command pulses inside the positioning module to analog voltage, and outputs the converted analog voltage to the servo motor drive module.
The motion controller contains no dedicated module capable of issuing this analog command.
A stand-alone MELSEC-A AD72 or AD70 analog output positioning module can be used.

Auto tuning

The responsiveness and stability of machines driven by servo motors is influenced by changes in the moment of inertia and rigidity resulting from changes in factors such as machine load.
This function is used to automatically adjust the speed loop gain and position loop gain based on the machine condition in order to maintain maximum machine performance.

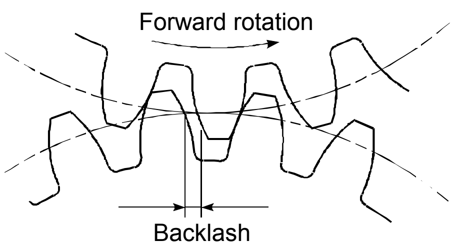
Automatic trapezoidal acceleration/deceleration

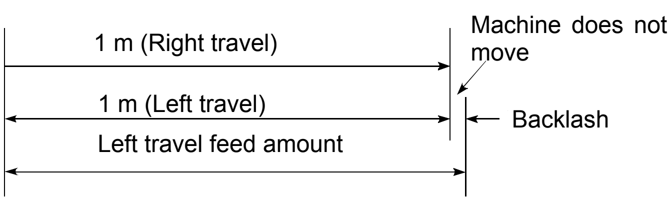
This is positioning movement in which the time and speed graph forms a trapezium.



Backlash compensation

Play (backlash) occurs as the movement direction changes from forward rotation to background rotation as the gears engage. The same phenomenon occurs even with screws, and it is not simply enough to feed an axis 1 m to the right when performing positioning and then feed 1 m back to the left to return the axis to its original position. The axis will not return to its original position until it has also been fed by the amount of play. This refers to the compensating of this play. This is similar to the "play" in car steering wheels.



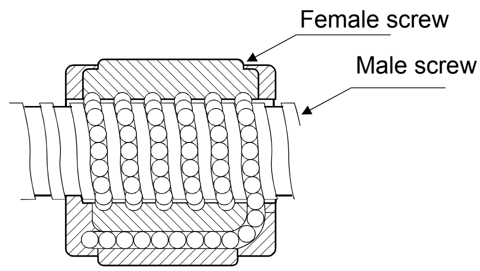


Backup function

(1) This function ensures that sequence programs and device statuses stored in the PLC CPU RAM memory are not forgotten even in the event of a power outage.
(2) This function is used on absolute position compatible systems to ensure that current values are not forgotten even in the event of a power outage.
(3) When replacing CPU modules, CPU data (servo programs, servo parameters, absolute position compatible data, etc.) is read by peripheral equipment, and then loaded following CPU replacement.

Ballscrew

This is a type of screw, and has balls in the engaging part similar to ball bearings. There is very little backlash, and it can rotate with very little force, and so is used for positioning.
See "Feed screw".



Base shut-off

The servo amplifier supplies power to the servo motor through power transistor switching.
Consequently, the base is shut off to stop power supply to the servo motor when the servo power turns OFF or when an alarm occurs. When this happens, servo motors are in a coasting condition.

Blank cover module

This is an empty module used to improve the appearance of vacant slots on the main base or expansion base.

Bottom dead center

This refers to the lower side of the machine installation route for the cam mechanism reciprocating motion.
This is the lower point of the cam.
See "Reciprocating cam".
See "Feed cam".

Cam

Machine element used to transfer anticipated movements through direct contact with a joint with contactor of simple shape such as a knife edge, roller, or planar shape.

Cam curve

The follower member motion curve moved with the cam can be set with a software package (SWOSRX-CAMP). There are various names of cam curves such as constant speed, constant acceleration, 5th power polynomial, cycloid, modified trapezoid, modified sine, modified constant velocity, trapecloid, double harmonic, and simple harmonic.

CAMP

CAMP is a software package (SW3RN－CAMP) used to create cams for virtual mode cam output.

CHANGE signal

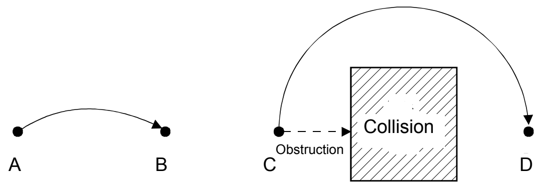
This is an external signal used to trigger position control while executing speed control.

Characteristics of cam curves

This is the speed and acceleration of cam curves.

Circular interpolation

Positioning is performed by running a horizontal direction motor and vertical direction motor simultaneously, the CPU performs the computations necessary to draw an arc, and interpolation is performed automatically.
Circles are created with auxiliary point designation, radius designation, and center point designation, and any obstructions found can be avoided.
See "Linear interpolation".



Command in-position

This turns ON when the difference detected between the positioning address (command position) and feed current value with a signal found in the positioning data fixed parameters matches the set value.
Detection is made a little before the positioning end point address, and it is used to carry out preparatory work, etc.

Constant speed control

With a single start command, positioning is performed to the end point at fixed speed while performing linear or circular specified positioning control to a predetermined pass point.
With a FOR/NEXT command, the same control as that for the pass point can be repeated.

Constant velocity curve

This curve is applied if necessary for axes to run at constant speed.

Continuous pass

This is control such as constant speed control in which a route is followed without interruption.

Control unit

This is one of the basic units of positioning data, and is specified in mm, inches, degrees, or pulses.

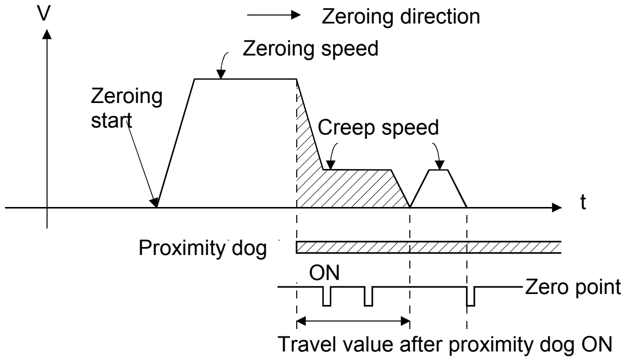
In Japan, mm or degrees? In the USA, inches or degrees?	Pulses can also be used!

COPY

This means copying a part from the Edit screen to another location.

Count type zeroing

The axis decelerates to creep speed when the proximity dog turns ON during zeroing, and after moving the travel value after the dog turns ON, the subsequent home position signal is set as the home position address.
The proximity dog length can be ignored.
See "Zeroing method".



Creep

This is a low speed at which the axis moves a little before reaching the home position when performing zeroing during positioning.
It is difficult to stop suddenly at a precise point when traveling at high speed, and therefore it is necessary to switch to creep speed.
See "Proximity dog type zeroing".

Current feed value

This is the number of calculated pulses corresponding to the travel distance output by the motion controller.

Current loop mode

This is also referred to as torque loop mode.
See "Position loop mode".

Current value

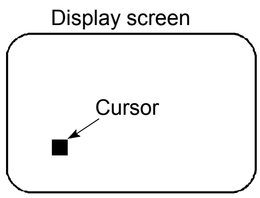
Current positioning control address

Current value change, current value rewrite

Refers to the teaching of temporary proximate values used for positioning when the machine is assembled and connected to the motion controller.
In addition, this function can be used to write temporary current values at such times as when current values are lost in the event of an accident, etc.
By then performing zeroing, the motion controller recognizes the home position. Changes to current values can be performed with a CHGA command during a positioning stoppage.

Cursor

Used to urge caution to the operator at display screens on peripheral equipment and CRTs, etc.



CUT

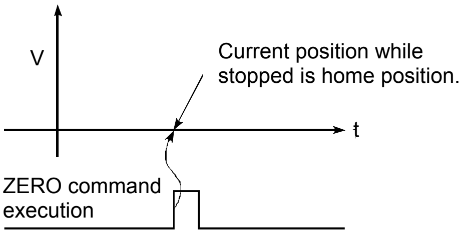
This means storing a part from the Edit screen to the system buffer.
Parts stored in the system buffer by cutting can be displayed on the Edit screen again by pasting.

Cycloid curve

Commonly abbreviated to CY curve, this curve has been known for many years as a continuous curve, and has little excitation frequency component, making it ideal for high speed. On the downside, it has high characteristic values such as speed, acceleration, and inertia torque.

Data set type zeroing

Sets the position at which the axis is currently stopped as the home position address.
No proximity dog switch is required.
See "Zeroing method".

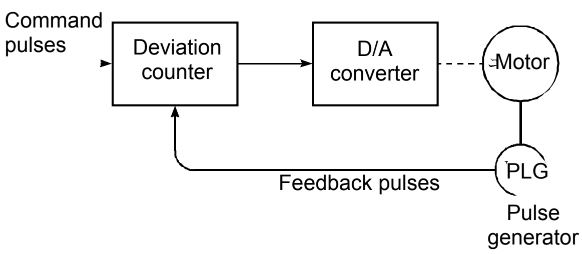


DELETE

This means deleting parts from the Edit screen.

Deviation counter

This counter is built in to the drive unit, and is used for positioning.
Feedback pulses are subtracted from motion controller command pulses, the command pulse and feedback pulse deviation value (droop pulses) are sent to the D/A converter, the motor is run, and if there are no command pulses, the motor is run until the number of droop pulses reaches 0.

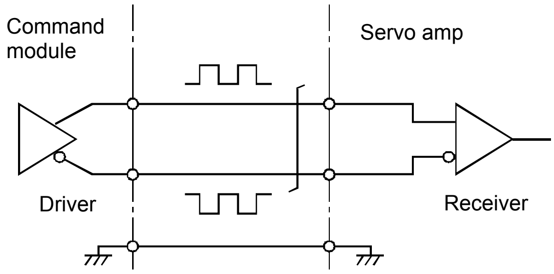


Differential gear

This is one transfer module in the virtual mode mechanical system program, and is used for auxiliary input for main shaft rotations.

Differential output

This is one type of encoder feedback pulse output.
If transferring a single signal, by transmitting signals with reversed polarity in pairs, the receipt side is able to judge by setting the signal logic, and its excellent noise resistant properties make it ideal for pulse train high speed signal transfer.



Digital bus connection

Commands output from the motion controller to servo amplifiers are generally in the form of a pulse train or analog output, however, this method involves issuing commands with digital values by connecting a bus line, facilitating the construction of highly reliable, high-speed, high-accuracy systems.

Direct clutch

This is one of the virtual mode mechanical system programs.
This transfer module clutch is a clutch with setting time of zero for which no smoothing time constant has been set.
See "Smoothing clutch".

Discontinuous curve

This is a constant speed curve or constant acceleration curve within a cam curve for which acceleration within an interval including both the start point and finish point is not continuous.

DOG signal

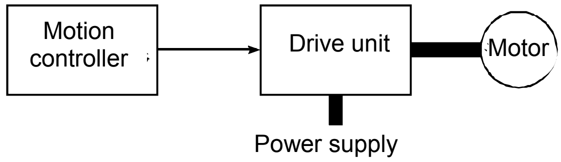
This refers to the home position proximity dog.

Drive module

This is one of the virtual mode mechanical system programs.
Refers to the pairing of a virtual servo motor and synchronous encoder used to rotate the main shaft and auxiliary input axes.

Drive unit

Commands (pulses, etc.) issued by the motion controller are of low voltage and current, resulting in insufficient energy to drive motors. This unit amplifies these commands to drive motors.

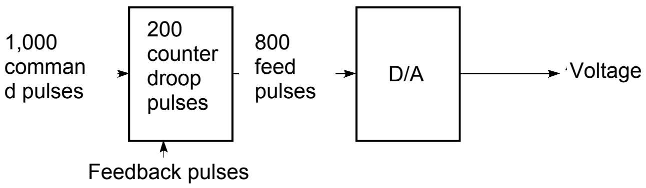


Drive unit ready

This signal indicates that the motor drive unit is ready.
The drive unit remains OFF if the power is OFF or if an accident occurs.

Droop pulses

As the machine has inertia (GD2), if positioning module speed commands are issued as is, the machine becomes delayed and is therefore unable to keep up.
In the case of servo motors, speed command pulses are accumulated in the deviation counter to delay them. Droop pulses are these accumulated pulses. When the machine stops, the deviation counter discharges all pulses to leave the count at 0.
To be exact, the difference between feed pulses and feedback pulses is droop pulses.



Dwell

Dwell refers to a condition in which the axis is temporarily stopped, with no follower member displacement over the passage of a certain period of time.

Dwell period

This is the input axis rotation angle when the output axis is stopped, and the sum of this and the index period is 360°.

Dwell time

It takes time to calculate deviation counter droop pulses immediately after positioning is finished. Positioning will be inaccurate if this time is too short, and so a longer time is used for the dwell time.

Dynamic brake

When the protective circuits are triggered by a power outage or emergency stop condition (EMG signal), the dynamic brake is used to short the circuit via a resistor between servo motor terminals, consume rotation energy as heat, and stop axes suddenly without coasting the motor.
Braking power is generated only while motors capable of obtaining brake torque greater than that of electromagnetic brakes are rotating, and as there is no holding power when motors are stopped, mechanical brakes are also used to prevent vertical axes from falling.

EIA

This is the EIA code (EIA standard) punched on the perforated paper tape used to instruct the NC unit to perform processing.
Other NC languages are ISO code (ISO standard) and JIS code (JIS standard).

EIA code

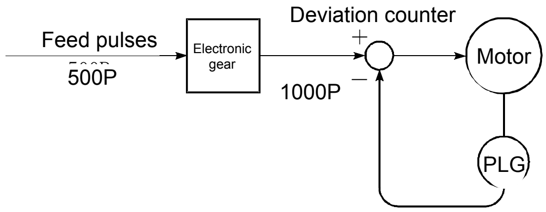
This is a tape code used for numerical control machine perforated paper tape stipulated by the Electronics Industries Association, and has 8 tracks including 6 bits used to show information, an odd number parity bit, and an EOB character (end of block).

Electromagnetic brake

Electromagnetic brakes are installed on motors to prevent vertical axes slipping during power outages or when accidents occur, and for protection when motors are stopped.
This is a non-excitation electromagnetic brake.

Electronic gear

This function is used for positioning, and allows the feed value per feedback pulse to be changed freely. The feed pulse and feedback pulse ratio, in other words pulse rate, is selected based on the machine, however, the advantage of this function is that it can be set freely regardless of this machine system.



Emergency stop

It is necessary to insert the emergency stop or stop program for safety purposes into the PLC program, and also install a circuit used to stop the machine outside the PLC.
This measure is taken in consideration of the rare event of a PLC defect occurring, or the emergency stop being disabled by the sequence program based on the timing at which the PLC power turns ON and OFF.
Note that it is better for input devices to use contact b because it allows wire damage and contact defects to be detected.
EMG signals should be used.

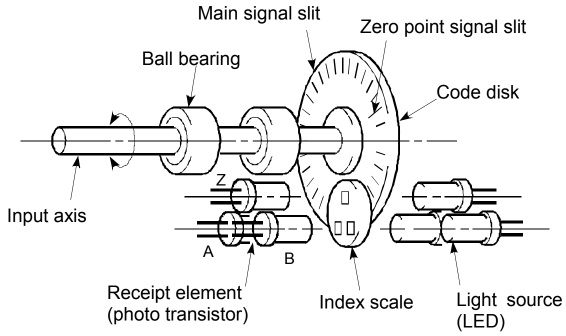
EMG signal

With all axes, the emergency stop external switch is contact b. Consequently, the power for the switch is normally ON.
By issuing this signal, all axes stop, the external emergency stop input flag (M9076) turns OFF, and the motor coasts.
Furthermore, addresses will be lost and so caution is required.

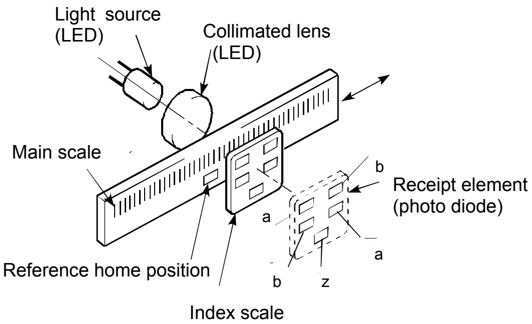
Encoder

Inputs position information to the control unit. Pulse generator, etc. Encoding device

The diagram shows an optical encoder.



Rotary encoder
(incremental)



Linear encoder

Linear encoders employ a binary output format, and are available in incremental and absolute types.

See "Absolute encoder".

See "Incremental encoder".

Error compensation

When a dimensional error exists at the machine, when the feed value is actually less than or greater than 1 m even although a 1 m command is sent from the module, the motion controller compensates for that error. For example, when the actual feed value is less than 1 m, extra pulses just enough to cover the shortfall are sent to perform the correct 1 m positioning.

External regenerative brake resistor

Referred to as regenerative brake.

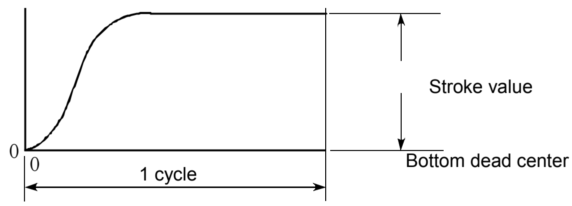
When moving machinery with a motor, power is normally supplied from the amplifier to the motor, however, when the motor is decelerating or driving a down load, the rotation energy held by the motor and machinery flows back (is regenerated) to the amplifier.

This regenerative energy is consumed by resistance, and regenerative control capability is exhibited using the regenerative brake torque obtained.

This is used if performing high-frequency acceleration/ deceleration.

Feed cam

Consecutive feed motions are made by the stroke amount from the lower stroke position (bottom dead center), facilitating conveyor feed and transfer device feed.



Feed forward control

Used to minimize motor delay and improve servo tracking in response to positioning control commands. (Disabled during auto tuning.)

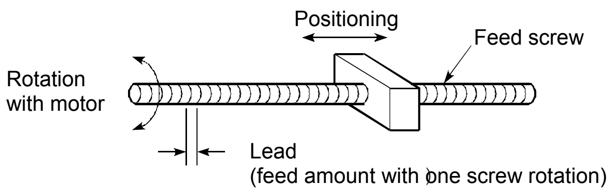
Set to 0 to 150%.

Feed pulse

These are pulses sent from the command device on the positioning module, etc. to the servo unit or stepping motor. These are also referred to as command pulses.

Feed screw

This is a piece of apparatus used to perform positioning by rotating a screw, and is the main screw. Ballscrews are commonly used to minimize backlash and dimensional error.



Feedback pulse

A command is issued during automatic control, and this pulse train is returned to confirm whether the machine is behaving in accordance with the command. If not, a correction command is issued. If a command with 10,000 pulses is issued, and 10,000 feedback pulses are returned, the balance should be 0. These are also referred to as return pulses.

See "Deviation counter".

File name

This is the name given when writing data or programs to a floppy disk or hard drive.

File names are made up of the system name and machine name, each with up to 8 characters, and a header is appended.

See "Machine name".

Fixed feed

This means obtaining the dimensions required to cut sheet and rod materials in the specified dimensions when performing positioning. The incremental method is commonly used.

There are three types: FEED-1, FEED-2, and FEED-3.

Follower member

This is a general term used to refer to the part that makes contact with the cam (rod which moves back and forth), or a load system after that point.

Formatting

Refers to the initialization of the hard drive or floppy drive disk, and involves the writing of computer rules and contents, etc. to the disk. Consequently, the disk memory capacity will be reduced by the amount required for formatting.

Disks are for general use, and therefore formatting is required to tailor them for the computer. Formatting need only be performed once at the beginning.

Forward limit switch signal

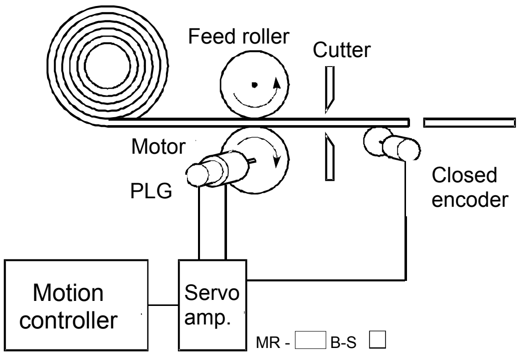
This is a positioning control device input signal used to report the triggering of the external upper limit switch (contact b configuration, power normally ON) for the travel range in which positioning control is performed.

This signal turns OFF when the external FLS signal (contact b) is OFF (not conducting), and the positioning operation has stopped.

Fully closed control

The machine travel mechanism is equipped with a closed encoder, and direct travel distance is detected, allowing transfer system mechanism (gears, ballscrews, timing belts, etc.) machine system errors between the motor and machine to be suppressed to a minimum.

This type of control is also ideally suited to positioning control for sliding mechanisms.



With closed encoders, the workpiece length is detected directly, ensuring a uniform workpiece cutting length regardless of feed roller slipping.

G-code

This is a standardized two-digit (00 to 99) number used to stipulate the NC unit axis control function, and is also referred to as G function.

Example

- G01 Linear interpolation
- G02 Circular interpolation (clockwise)
- G04 Dwell
- G28 Zeroing
- G50 Main shaft high speed setting

GD2

In mechanics, this is the same concept as moment of inertia, and is a format used to express the moment of inertia for gravitational unit systems (engineering units, etc.)

"GD^2" is one of these symbols with G representing gravity, and D representing the rotational diameter.

GD^2 = [gravity] x [rotational diameter]^2 (kgf/m^2)

The unit for moment of inertia used in catalogs is J (x 10^{-4} kg/m^2).

Consequently, it is given by GD^2 = 4 x J.

Gear

This is one transfer module in the virtual mode mechanical system program, and is used to branch main shaft rotations to the output module.

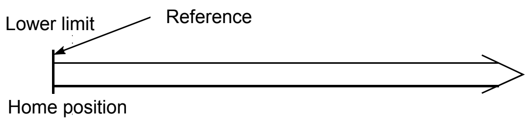
The gear ratio and rotation direction can be set.

Grid

Refers to useful reference horizontal and vertical lines used for arranging parts on the mechanical system editing screen.

Home position

This is the position used as the reference for positioning. Positioning cannot be started without a reference point.



In position

The droop pulse value (difference between position command value and position feedback from servo motor) in the deviation counter is detected with a signal in the positioning data servo parameters, and this in-position signal turns ON when the detected value matches the set value.

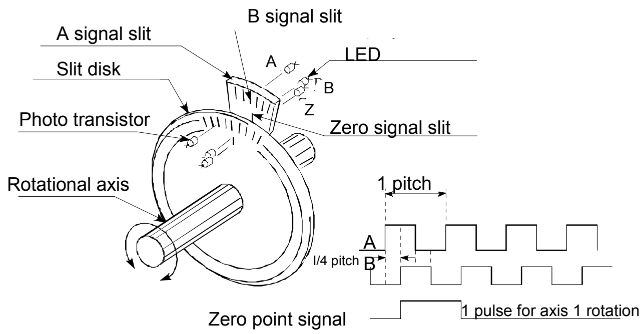
A few droop pulses are cut, allowing them to be used at such times as when staring subsequent positioning.

Incremental encoder

This is a device used simply to emit ON/OFF pulses as an axis rotates. Single-phase encoders emit only A pulses, and the axis rotation direction is unknown. Two-phase encoders emit both A and B pulses, allowing the system to judge that the motor is rotating in the forward direction if B turns ON while A is ON, and in the reverse direction if A turns ON while B is ON.

There are also encoders with zero point signals. Incremental encoders emit between 100 and 10,000 pulses per axis rotation, and are the most commonly used encoders.

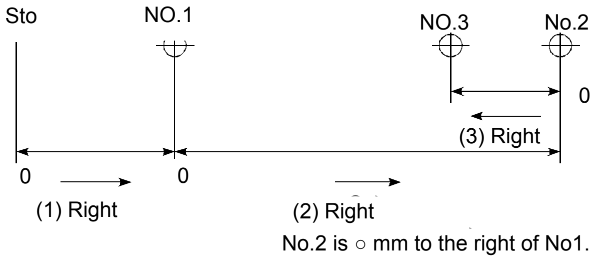
See "Encoder".



Output waveform 2 phase + home position output

Incremental mode

This mode is used for positioning, and expresses the position based on the specified direction and distance, with 0 as the stopping point. This is a relative address method. This mode is used for fixed feeding, etc. There is also an absolute mode.



Inertia

Behavior in which the current condition remains the same provided that the object is not acted upon by an external force. It is referred to as the moment of inertia.

Installation function

The motion controller internal OS (operating system) can be rewritten using peripheral equipment. Dedicated operating systems are used for the SV13 conveyance and assembly controller, SV22 automatic machine controller, SV43 processing machine peripheral equipment controller, and SV51 dedicated robot controller, and installing each OS facilitates use to match each machine.

Inverter

This is a device used to convert direct current to alternating current. In order to actually change the motor speed, a commercial frequency of 50Hz or 60Hz is first delivered by direct current, which is then converted to a an alternating current of 5Hz to 120Hz to control the motor speed.

Jerk

This is a further differentiation of acceleration by time, and expresses the rate of change of acceleration.

JOG

JOG operation This means moving a little at a time. Inching.
JOG operation is used for peripheral equipment test operation, and can be performed with a sequence program by writing parameters and the JOG speed.

KANA

KANA. Key found on peripheral equipment. Press to enter Katakana characters. Remember to reset this key when entering alphabet characters or numbers.

KPPS

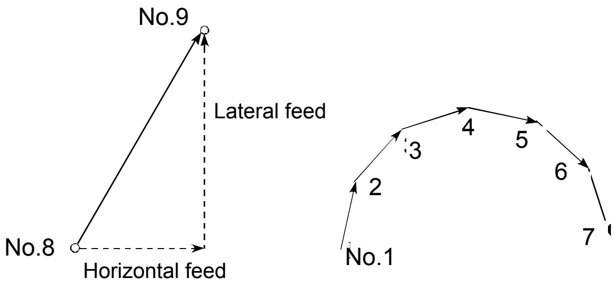
Kilo-pulse per second
This is the number of pulses per second. 80 KPPS means 80,000 pulses per second.

Line monitoring

This is the monitoring of the PLC and controller control status during operation.

Linear interpolation

Positioning is performed by running a horizontal direction (X) motor and vertical direction (Y) motor simultaneously, the CPU performs the computations necessary for axis travel to proceed in a straight line, and interpolation is performed automatically.
ABS-2 to ABS-4, and INC-2 to INC-4 can be used.
The following is an example of 2 axis linear interpolation.



Load inertia ratio

GD_L^2/GD_M^2
See "GD^2".

Low inertia motor

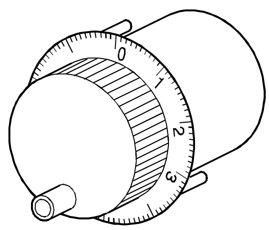
Used when wishing to accelerate and decelerate frequently.
In order to reduce the moment of inertia from standard motors to approximately one third, the rotor diameter is reduced, and the longitudinal direction is lengthened to cover torque.
A load inertia ratio of 1 or less is ideal.

Machine name

Maximum eight character code applied freely by the user from a file name. Alphabet characters (upper case), numbers, and one symbol are used. The first character must be an alphabet character.
See "File name".

Manual pulse generator

Pulses are generated by manually rotating a handle.



Margin

This is the cam and cam follower ratio of contact, and should normally be 60% or higher.

Master axis

This is the side at which positioning data is prioritized when performing interpolation during positioning.
It is an interpolation control unit set in the parameter block.

M-code
This is a signal used to trigger auxiliary functions such as drill change, clamping, unclamping, raising and lowering of electrodes, and all types of display that are performed together with positioning.
Codes 1 to 255 are assigned (1: clamp, 2: unclamp, etc.) and used by users.
M is an abbreviation of machine.

Mechanical support language
Synchronous control is performed, and therefore by using software to process synchronous control operations that were previously mechanically joined with mechanisms using a main shaft, gears, and cams, processing switches to positioning control (roller output, ballscrew output, rotary table output, cam output) with servo motors.
See "Mechanical system program".

Mechanical system program
This consists of a mechanical mechanism connection drawing connecting the drive module (virtual servo motor and synchronous encoder) and virtual main shaft, transfer module (gears, clutches, speed change gear, differential gear), output module (cams, rollers, ballscrews, rotary table) with the respective module parameters.

Model adaptive control
When performing actual operation, differences occur in the actual control state quantity relative to the ideal control state quantity.
Motion control enables optimum loop gain control based on those differences to ensure that control is always performed at maximum performance.

Modified constant velocity curve
Commonly abbreviated to MCV curve, this curve has a fixed speed interval in the middle of the curve, and is used when necessary to lower the maximum speed to reduce the pressure angle, or when a fixed speed portion is required.
It is applied to heavy loads traveling at medium speed.

Modified sine curve
Commonly abbreviated to MS curve, this is a commonly used standard curve. It has low maximum speed and small cam axis torque coefficient, and acceleration is comparatively low, and therefore is widely used when the nature of the load is unknown. It is applied to loads traveling at high speed.

Modified trapezoid curve
Commonly abbreviated to MT curve, this is a standard curve developed to minimize the maximum acceleration value, and is applied to light loads traveling at high speed.

Monitoring trace graph
This is a monitor function, and displays waveforms based on traced (recorded) position commands, position droop, motor speed, motor current, and speed command values during positioning.

Motion control
This refers to positioning control.

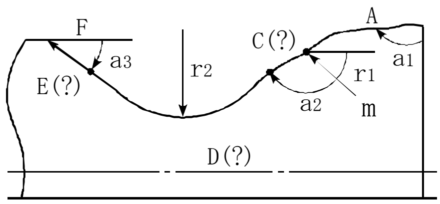
Multiplication ratio setting
This is the pulse rate.
See "Pulse rate".

No-dwell motion
At the operation start and end points, there is no dwell, acceleration is maintained at an arbitrary value, the reciprocating operation is repeated, and the acceleration (A) value becomes smaller.

Notch filter
This sets the notch frequency to match the machine system resonant frequency.

Numerical Control
This is the language punched on the paper tape used to instruct the NC unit to perform processing.
Other NC languages are EIA code (EIA standard), ISO code (ISO standard) and JIS code (JIS standard).

Numerical controller
Unit offering even more advanced positioning. 3 axes or more can be controlled with high accuracy and at high speed. Control for complex curves and curved surfaces is also possible.



One-dwell motion, dwell-rise-dwell motion
If used to double back on the same curve on the upward and downward journey for a movement involving a stop at only the start point or finish point of that journey, acceleration can be reduced, and movement becomes smoother.

Option slot
Slot into which a motion module or MELSEC-Q Series can be installed to suit the intended use.

Output module
This refers to a module used to run a servo motor in virtual mode.
The output module has rollers, ballscrews, rotary table, and cams.

Pancake motor
The axis direction dimension is 100 mm shorter than the standard shape,
and is used when there is little space to install the servo motor.

Parabolic curve
Commonly abbreviated to PB curve, it possesses the characteristic of having a non-dimensional maximum acceleration, facilitating minimum time control under the condition that the maximum acceleration value is suppressed.
On the downside, acceleration is discontinuous, and vibrations occur easily.

Parameter block

This allows changes to be made easily to control conditions with data such as that for acceleration and deceleration control used for positioning processing.

Parameters

Parameters stipulate PLC functions. Memory capacity, relay or timer types, status latch selection, and comment capacities and so on can be set by users as parameters. Default values are set to enable basic functionality. There are fixed parameters and servo parameters for positioning.

PASTE

This means redisplaying parts cut from the Edit screen and stored in the system buffer on the Edit screen again.

PCPU

This refers to the positioning control CPU that exists as the motion controller CPU configuration.
In addition, there is also a sequence control CPU known as an SCPU.

PG0 (PG zero)

See "Home position signal".

Plural harmonic motion

This is a cam curve,
examples of which are motions in which the acceleration pattern is the multiple perpendicular axis component of a uniform circular motion.
This has been improved to make it difficult to cause vibrations to "simple harmonic motions".

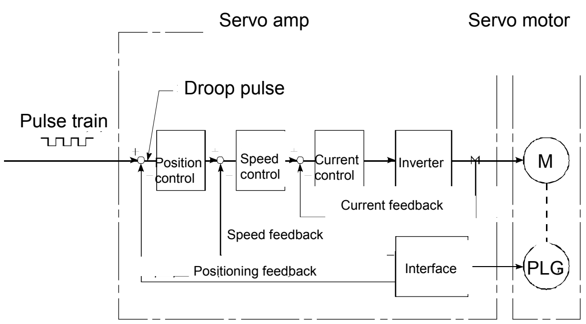
Position loop gain

Expresses the control response speed when performing positioning control at item 1 in the positioning data servo parameters. This value stipulates the number of deviation counter droop pulses during operation, and droop pulses will become smaller if the setting is high, allowing the settling time when the axis is stopped to be reduced. If too high, however, undulations will occur when the axis stops, resulting in slight vibrations. Droop pulses will increase in size if the value is small, allowing axes to come to a smooth halt as the settling time increases when the axis stops, however, the stopping error will increase.

$$\text{Position loop gain} = \frac{\text{Command pulse frequency}}{\text{Droop pulse}} \ (\text{sec}^{-1})$$

Position loop mode

This is one of the servo control modes used for positioning, and is used for position control.
In addition, there is also a speed loop mode used to perform speed control, and a torque loop mode used to perform torque control (current control).

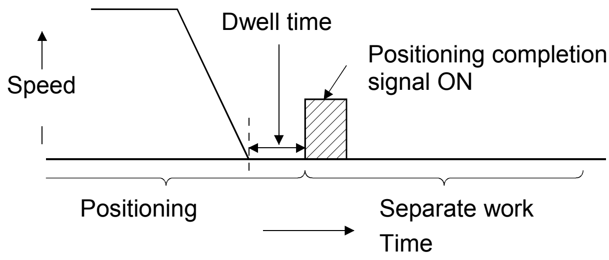


Positioning

This refers to traveling from a certain point to the predetermined next point.
For example, determining length in mm units, outputting a drilling position, etc.
Servo motors channel power from the motion controller issuing the position commands.

Positioning completion signal

This is signal Xn1 that turns ON when the positioning dwell time is complete.
The purpose of this signal is to begin other work (clamping, etc.) after positioning.



Positioning devices

These refer to I/O signals, internal relays, data registers, special relays, and special registers used to communicate signals between the SCPU (PLC CPU) and PCPU (positioning CPU).

Positioning parameters

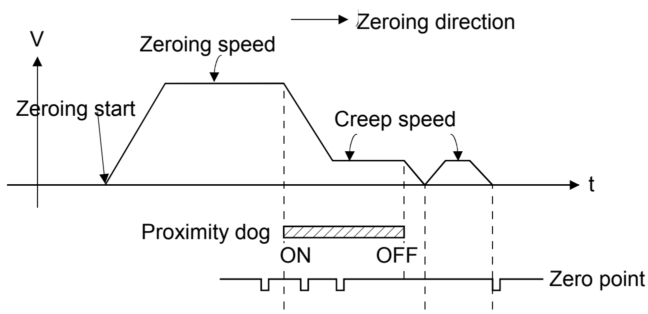
This is the basic data used for positioning control, and includes such information as system settings to match the servo motors and servo amps used, the control unit, travel value per pulse, speed limit value, upper and lower stroke limits, and acceleration/deceleration time.

Programmable controller ready

Signal indicating that the PLC CPU is ready. Special function modules are unable to function if this condition is not established.

Proximity dog type zeroing

The axis starts to decelerate when the proximity dog turns ON during zeroing, and after moving at creep speed until the proximity dog turns OFF, the first home position is set as the home position address. The length of the proximity dog is the point.
See "Zeroing method".

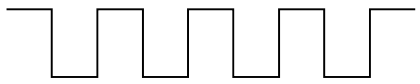


PTP point to point control

This refers to positioning control.
This is control in which pass points are specified at intervals on the route.
A request is made only to reach the target position, and control over the route during travel from a certain position to the next value is not required.

Pulse

(1) The turning ON and OFF of current (voltage) over a short period of time. The same term is applied to the human pulse. A pulse train is a series of pulses. The MELSEC AD71 is a unit that emits pulses.
The AD61 unit receives and counts pulses.



Pulse command

This command turns only 1 program cycle (1 scan) ON when conditions turn ON. With MELSEC-A, there is a PLS command that turns the 1 scan time ON with the leading edge when the signal is ON, and a PLF command that turns the 1 scan time ON with the trailing edge when the signal is OFF.

Pulse generator

This is a device used to generate pulses. For example, pulses are generated as the shaft attached to the motor axis rotates. Digital device
Single-phase types emit a single pulse train, and two-phase types emit two pulse trains with phase difference. 600 to 1 million pulses are emitted per axis rotation. Furthermore, one or two pulses with home position signal are emitted per axis rotation.
See "Encoder".

Pulse rate

This is a coefficient used for positioning which doubles, triples, halves, or thirds the feedback pulse per motor axis rotation, and is the ratio of feed pulses to feedback pulses.
For example, when there are 2,400 pulses per rotation and the pulse rate is 2, the result will be 1,200 pulses. The axis rotation per pulse when there are 2,400 pulses is 0.15°, however, this will be 0.3° with 1,200 pulses. Positioning accuracy drops as the pulse rate is increased.
See "Electronic gear".

Pulse train command

By continuously emitting the number of pulses corresponding to the machine travel distance from the motion controller to the servo motor servo amplifier, it is possible to perform positioning control proportional to the number of pulses.

Ready (M9074)

Condition in which the PCPU or servo amp is able to function normally after the power is turned ON.

Real mode

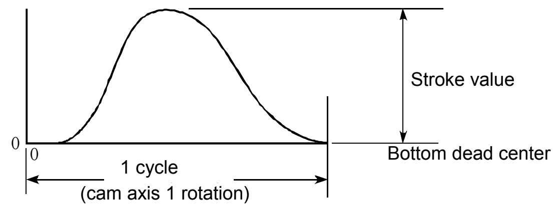
In this mode, servo motors are controlled directly with a servo program.

Real-time auto tuning

See "Auto tuning".

Reciprocating cam

Consecutive reciprocating motions are made by the stroke amount from the lower stroke position (bottom dead center), facilitating push/return movements, up/down movements, and left/right movements.

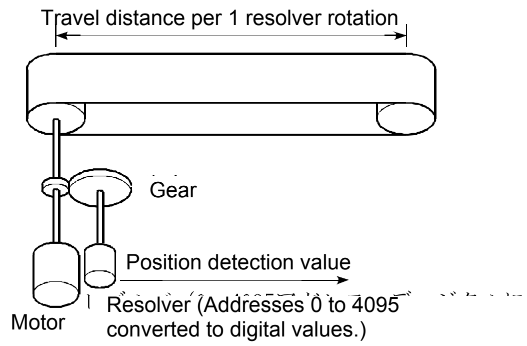


Regenerative brake option

This is an optional part, and is used to perform high-frequency acceleration and deceleration.
See "External regenerative brake resistor".

Resolver

This is a device used to resolve angle detection into two analog voltages.
Also referred to as a two-phase synchro, as opposed to single phase voltage input, the resolver converts a single rotation of the axis rotation angle to a perpendicular two-phase voltage (analog voltage), and then outputs it.



Reverse limit switch signal

This is a positioning control device input signal used to report the triggering of the external lower limit switch (contact b configuration, power normally ON) for the travel range in which positioning control is performed.
This signal turns OFF when the external RLS signal (contact b) is OFF (not conducting), and the positioning operation has stopped.

Roller

This is a cylindrical rotating object used to feed and roll paper or steel plate.
Roller output can be set as a virtual mode output module.

Rotary table

Performs positioning control while rotating the workpiece on a round table within a 360° range.

SCPU

This refers to the sequence CPU that exists as the motion controller CPU configuration.
In addition, there is also a positioning control CPU known as a PCPU.

Scroll

The CRT screen and so on changes repeatedly like a scroll.
The screen changes as the machine being controlled moves, and with key operations.

Sequence control

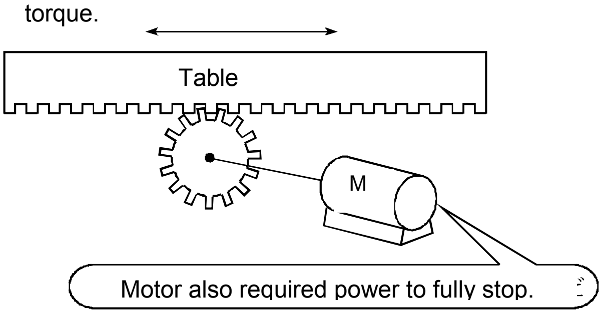
This refers to a sequence program used to control operations sequentially such as detecting the completion of a single movement with a switch, and using this signal to start the next operation.

Servo amplifier

There is a type built in to the controller base, and an externally installed type. The servo amplifier issues speed commands to the servo motor, and controls the servo motor with received feedback pulses.

Servo lock

Force used to hold the motor at the stop position is required for positioning with servo motors and stepping motors, etc. (The motor position will be lost if moved with external forces.)
This condition is referred to as servo lock or servo lock torque.

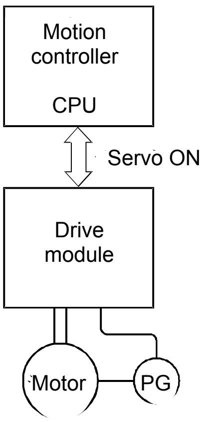


Servo motor

Motor that rotates reliably in response to commands.
These motors offer high responsiveness, high speed, and high accuracy, and are capable of frequent starting and stopping. They are produced in DC and AC types, and large capacity models are also available. They are equipped with pulse generators used to detect speed, and often perform feedback control.
In other words, they move in accordance with command values, and in such a manner as to minimize differences between command values and current values while detecting current values.

Servo on

Positioning is not performed when the drive unit is normal and this servo on is not ON.



Servo parameters

See "Positioning parameters".

Servo program

This is a program used to control servo motors,
and contains such commands as independent linear control, linear interpolation control, circular interpolation control, fixed feeding, speed control, constant speed control, and zeroing.

Servo response

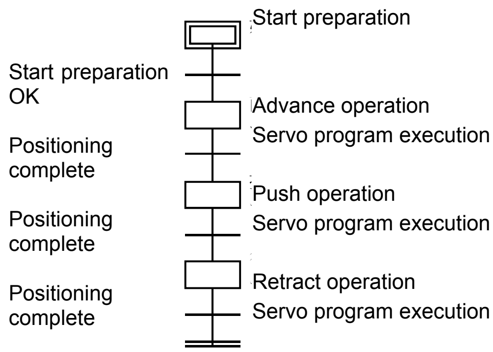
Sets auto turning responsiveness.
The optimum response can be selected based on the machine rigidity. The higher the machine rigidity, the higher responsiveness can be set, facilitating improved tracking in response to commands, as well as reduced settling time.

Settling time

This is the delay time from the time the stop command is complete until the servo motor stops (time until droop pulse becomes ±1).

SFC (sequential function chart)

This is the optimum structured programming method required to perform machine automatic control sequentially with a PLC.



Simple Harmonic motion

This is an example of a cam curve,
examples of which are motions in which the acceleration pattern is the single perpendicular axis component of a uniform circular motion.
This motion generally exhibits smooth characteristics, and is therefore applied to low speeds.
On the downside, acceleration is discontinuous, and vibrations occur easily.

Simultaneous start control

Two to three types of servo program are run with a START command to start multiple servo motors simultaneously.
Multiple axes specified in a special register are started simultaneously with a special relay with JOG operation.

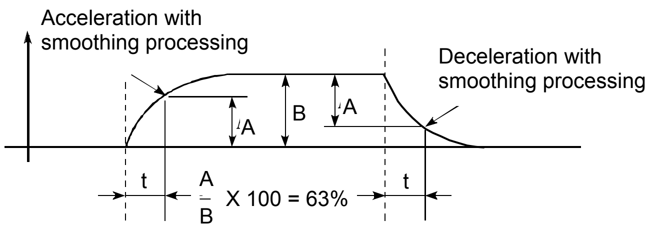
Skip function

This function allows subsequent positioning to be started even if an external STOP signal turns ON during positioning control, and the signal remains ON when stopped.
Subsequent positioning is started with an SVST command when the external STOP signal input disable flag is turned ON during deceleration, and the start accept flag turns OFF.

Slave axis
See "Master axis".

Smoothing clutch
This is a clutch for which a smoothing time constant is set as a virtual mode transfer module.
The rotation can be conveyed smoothly when the clutch is ON and OFF.
It is known as a direct clutch when the smoothing time constant is zero.

Smoothing time constant t



See "Smoothing clutch".

Speed change
See "DSFLP command".

Speed change control
Axes are positioned at the travel value end point while changing speed at the speed switching point during positioning control.

Speed change gear
This is one transfer module in the virtual mode mechanical system program, and is used to change the main shaft rotation speed and transfer it to the roller output module.

Speed control
Controls the speed for endless rotations in the same direction for conveyors, etc.
Using VF forward rotation and VR reverse rotation commands (position loop) and VVF forward rotation and VVR reverse rotation commands (speed loop), feed current values are zeroed at the same time as axis movement starts, axes are rotated at a previously set speed, and then decelerate when a stop command is received, without increasing or decreasing the feed current value.
Note that upper and lower stroke limits are ignored.

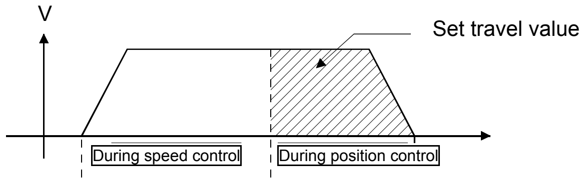
Speed integral compensation
Frequency responses are issued when performing positioning control at item 1 in the positioning data servo parameters, and transient characteristics are improved.
It is helpful to increase this value when the overshoot when accelerating or decelerating does not get any smaller even by adjusting the speed loop gain.
The unit is ms.

Speed limit value
This is the maximum positioning speed. By setting this value in the parameters, operation is performed with speed limit values even if a larger value is set due to a mistake in other data. Note that acceleration time and deceleration time are the speed limit value times.

Speed loop gain
Expresses the control response speed when performing speed control at item 1 in the positioning data servo parameters.
If the control system responsiveness drops and operation becomes unstable as the load inertia moment ratio increases, stability can be improved by increasing this setting.
If increased too much, the overshoot increases when accelerating, and motor vibration noises are emitted during operation or stoppages.

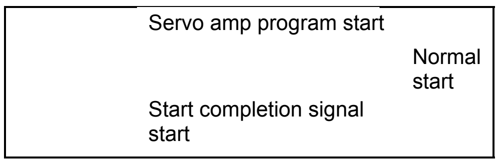
Speed loop mode
See "Position loop mode".

Speed/position control
Incremental positioning control is performed when external switching signals are received during speed control.



SSCNET
This is an abbreviation of Servo System Controller Network.
This is a connection method used to improve reliability between the motion controller and servo amp through high-speed serial communication.
Wiring work is simplified with a one-touch connection using a connector.

Start completion
This is a signal sent to immediately indicate that the motion controller has successfully started positioning.
It does not mean that positioning is complete.

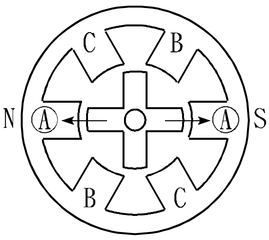


Starting axis
This is the axis to be started, and is axes 1 to 8/32.

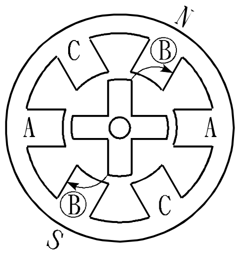
Status
This is a device used to express the condition, and collectively refers to signals that turn ON (1) in the clutch status, virtual mode status, and when making zeroing requests, etc.

Stepping motor

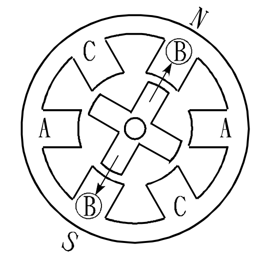
This is a motor that performs an angular rotation (e.g., 0.15°) with every pulse. Consequently, rotation proportional to the number of pulses can be obtained. Stepping motors are available in two to five-phase types, and with the three-phase type, the motor rotates by applying voltage in order from A to C. Most stepping motors are compact, and offer accurate rotation without feedback. Caution is advised with step outs, whereby the motor does not rotate accurately.



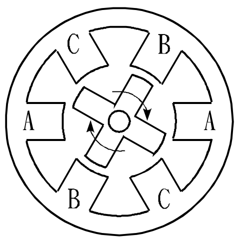
(1) First, the A phase is excited with a pulse.



(2) By then exciting the B phase, force moves in the direction indicated by the arrow.



(3) The nearest gear tooth is pulled toward the B phase, and the motor stops.



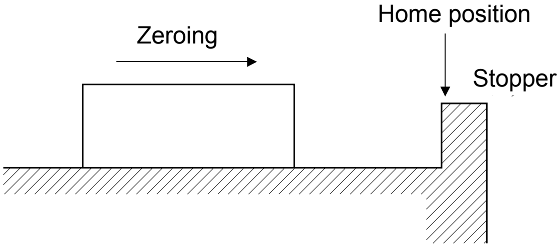
(4) By successively changing the excited phase, the rotor rotates in the clockwise direction.

STOP signal

This is a positioning control device input signal used to directly stop positioning from outside during operation. When the external STOP signal (contact a) is ON (conducting), operation stops and XnD turns ON.

Stopper-forced stop

This is a zeroing method using with positioning, and involves stopping the axis when it comes into contact with a stopper installed at the home position.
The motor will burn out and the stopper damaged if the axis remains against the stopper, and therefore various methods are used to prevent this such as equipping the system with a timer allowing the motor to be turned OFF when a fixed time has elapsed, or turning the motor OFF when the system detects that the motor torque has risen suddenly when the axis is against the stopper.

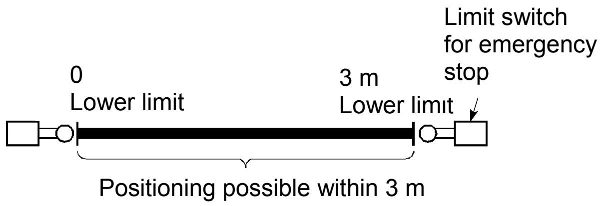


Stroke

This refers to the axis journey, and is the movement change over the distance from the point the axis starts moving until it next stops.

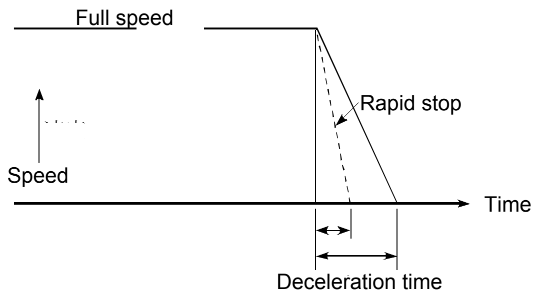
Stroke limit

This is the range in which positioning can be performed, or the movement range beyond which the machine will be damaged. If using a feed screw, the screw length is fixed, and if using fixed feed, this is the maximum dimension that is cut.
The upper and lower limits are set in the parameters, however, to ensure safety, the machine is installed with separate limit switches wired to external signal input modules, allowing axes to be stopped automatically.



Sudden stop

This is shorter than the deceleration time set in the parameters, and is the sudden stop deceleration time taken to stop.



SV13

This is a motion controller OS created for conveyance and assembly, is capable of 1 to 4 axis linear interpolation, 2 axis circular interpolation, CP control (constant speed control), and speed control, and is ideal for applications such as conveyance and assembly equipment.
Sequential control with SFC is possible.

SV22

This is a motion controller OS created for automatic machines, is capable of synchronous control of multiple servo motors, and controlling cams with software, and is ideal for applications such as automatic machines.

SV43

This is a motion controller OS created for processing machine peripheral equipment, is capable of linear interpolation, 2 axis circular interpolation, CP control (constant speed control), and speed control with an NC language (EIA), and is ideal for applications such as processing machines.

SV51

This is a motion controller OS created for dedicated robots, is capable of three-dimensional linear interpolation, three-dimensional circular interpolation, and three-dimensional CP control, and is ideal for applications such as dedicated robots (painting machines, transfer machines, etc.)
Sequential control with SFC is possible.

Synchronized control

This involves rotating the main shaft with a virtual mode drive module, and running the machine by synchronizing with multiple output modules (servo motors) through a transfer module.

Synchronous encoder

This is one type of virtual mode drive module. Pulses from encoders on external machines are input, and the system synchronizes with these pulses to drive the output module.

Teaching

This function is required for positioning, and involves the manual teaching of positions when addresses are unknown, or to align axes with the workpiece.
For example, it is troublesome to write the address for each point as data for complex addresses such as those in a picture, and so by tracing and teaching a model, positioning can be reproduced later.

Teaching playback function

This involves setting positioning points with the address teaching function, and simultaneously creating a servo program and setting positioning points with the program teaching function while actually moving the machine with the teaching unit (A30TU/A31TU).

Teaching unit

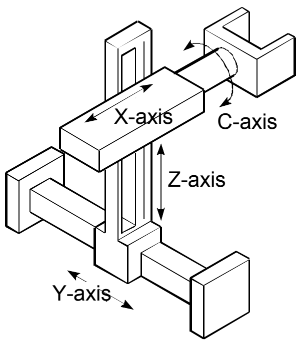
This unit is used to teach the data writing and reading, operation, and monitoring required for positioning.
There are two teaching units: A30TU and A31TU.

Three-dimensional cam

This cam uses three dimensional movements, and compared to planar cams, is generally more compact, and can be used as a positive cam for positive motion.

Three-dimensional interpolation CP control

XYZ-axes (3-axis orthogonal) and the C-axis (1 axis rotation) can be controlled at constant speed with a [CPSTART XYZ command] machine control servo command used with SV51 dedicated robots.



Top dead center

This refers to the upper side of the machine installation route for the cam mechanism reciprocating motion.

Torque

This is the size of a force acting on an axis multiplied by the arm length up to the line of action for that force.
N·m (kgf/m)

Torque loop mode

This is also referred to as current loop mode.
See "Position loop mode".

Torque ripple

This is the torque fluctuating range, or variations in torque.

Tracking

Travel values are entered from an external controller, and by adding these travel values to servo command values, positioning is performed at a relative speed with respect to the applicable object during travel.

Transmission module

This is one of the virtual mode mechanical system programs.
This is a module used to transfer drive module rotations to the output module, and is comprised of gears, clutches, speed change gear, and differential gear.

Trapecloid curve

Commonly abbreviated to TRP curve, residual vibrations after input is stopped can be suppressed, and seismic resistance is high.

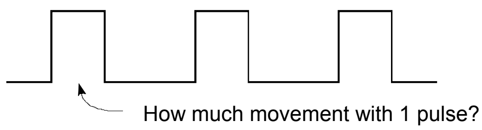
Travel

See "Stroke".

Travel per pulse

This is data calculated from the machine side, and stipulates how much the motor axis travels per pulse when the unit is mm, inches, or pulses when performing positioning. This corresponds to the position detection unit. Positioning accuracy higher than this is not possible.
Systems are normally designed with a travel value of one rotation per axis at the motor side as a reference, and therefore the travel value per pulse is calculated as follows.
Travel value per pulse =

$$\frac{\text{Pulse rate}}{\text{No. of pulses per pulse generator rotation}} \times \text{travel value per rotation}$$



Two-dwell motion

Motion with dwell at both ends of the journey

Unit setting

This refers to changing to the actual address unit or travel value unit for which positioning is to be performed.
Units are set in mm, inches, degrees, or pulses.

Unsymmetrical

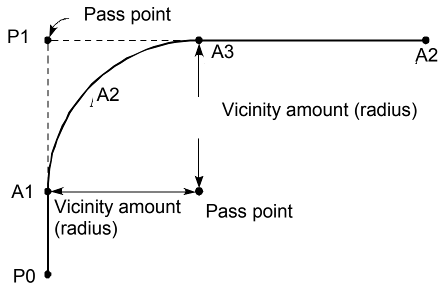
This is a cam curve in which the first half deceleration differs from the latter half ratio, and is mainly used to improve high-speed specification deceleration area characteristics.

V velocity

This means cam non-dimensional speed. This is non-dimensional displacement (motion displacement from start to finish expressed with 0 to 1) differentiated by non-dimensional time (motion time from start to finish expressed with 0 to 1).
See "Vm".

Vicinity passage

Refers to the performing of smooth pass point operations with SV51 dedicated robot three-dimensional interpolation CP control



When vicinity passage is zero
Locus is P0 → P1 → P2
When vicinity passage is specified
Locus is P0 → A1 → A2 → A3 → P2

Virtual auxiliary input

This is one of the virtual mode mechanical system programs,
and adds addition/subtraction rotations from the auxiliary shaft virtual servo motor or synchronous encoder to rotations from the main shaft.

Virtual main shaft

This is one of the virtual mode mechanical system programs.
This shaft is used to connect drive module rotations directly to the transfer module gear.

Virtual mode

This is a method used to move mechanical system program drive modules with a servo program or external encoder in order to drive the servo motor.
The mode used to drive servo motors directly with a servo program is called real mode.
See "Mechanical system program".

Virtual mode status

This is special relay M2044 used for monitoring, and is capable of confirming that the system is operating in virtual mode.

Virtual servo motor

This is one of the drive modules in the virtual mode mechanical system program, and is started with the servo program.
The main shaft is connected directly to the virtual servo motor.

Vm velocity

This is the cam non-dimensional speed maximum value.
See "V".

WDT error

This is an abbreviation of watchdog timer error, and indicates a PCPU defect.
M907 turns ON when an error occurs.

Window

Windows refers to selection menus displayed at the SW6RN-GSV22P or CAMP screen with peripheral equipment.

- Menu selection window
- Mode function selection window
- Sub function selection window
- Execution/setting selection window

Word

Expresses the data unit. With the MELSEC-A Series, 1 word represents 16 bits, and numerical values from -32,768 to 32,767 in decimal notation are handled. This is 0 to FFFF in hexadecimal notation.
However, there are also 32-bit commands, where 1 word represents 32 bits, and numerical values from -2,147,483,648 to 2,147,483,647 are handled. This is 0 to FFFFFFFF in hexadecimal notation.

Word devices

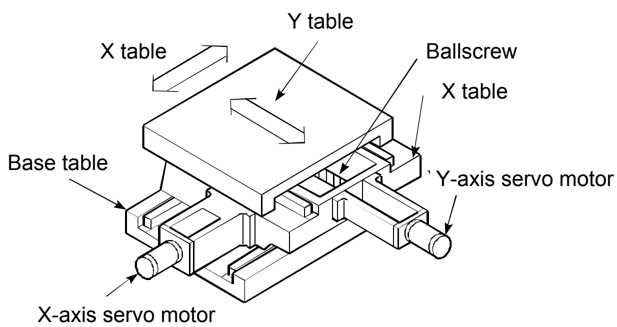
This is an element in the devices inside the PLC and holds data. In this device, 1 point is 1 word. The timer (T), counter (C), and all registers (D, R, W, Z, V, A), etc. are word devices.

X-axis

2D right/left lateral direction

XY table

This is a table moved in the X (lateral) and Y (longitudinal) directions so that positioning can be performed easily.
This is used when drilling holes in plates and drawing diagrams, etc.



Y-axis

2D forward/backward direction

Z phase

Also referred to as PG zero.
See "Home position signal".

Z-axis

3D up/down direction

Zero point signal

This is the pulse generator (encoder) PGO (detected once per rotation). It is also referred to as the Z phase.
See "Pulse generator".

Zero return data

This data is required by the motion controller to return to the home position. This is determined at the machine design stage, and involves changes to the machine design in order to be changed at a later date.
This is the reference point for home position positioning, and therefore zeroing is required at such times as when a power outage occurs during positioning, or an axis is moved manually with the power OFF because the current values held by the motion controller are no longer relevant. By performing zeroing, the machine searches for the proximity dog, moves, and then changes to creep speed, regardless of the current value.

Zeroing method

There is a proximity dog method, count method, and data set method.

Zeroing request

This request turns ON at the following times when using an incremental position system.
(1) When the power is turned ON.
(2) When the PLC ready signal turns ON.
(3) When parameters and zeroing data from peripheral equipment is written.
(4) When the following are selected while in peripheral equipment test mode.
Zeroing
Positioning
JOG operation
Manual pulse generator

The decision as to whether to perform zeroing at these times is made by the user.

Motion Controller School Textbook (Advanced Synchronous Control Edition) Windows PC Compatible MT Works2

MODEL	
MODEL CODE	
SH-030148ENG-A (1509) MEE	

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

When exported from Japan, this manual does not require application to the Ministry of Economy, Trade and Industry for service transaction permission.

Specifications subject to change without notice.