MITSUBISHI

MOTION CONTROLLER (SV43)

Programming Manual

type A172SHCPUN, A171SHCPUN
A273UHCPU(32 axis feature)
A173UHCPU(S1)

MITSUBISHI ELECTRIC

INTORODUCTION

Thank you for purchasing the Mitsubishi Motion Controller/Personal Machine Controller. This instruction manual describes the handling and precautions of this unit. Incorrect handing will lead to unforeseen events, so we ask that you please read this manual thoroughly and use the unit correctly.
Please make sure that this manual is delivered to the final user of the unit and that it is stored for future reference.

Precautions for Safety

Please read this instruction manual and enclosed documents before starting installation, operation, maintenance or inspections to ensure correct usage. Thoroughly understand the machine, safety information and precautions before starting operation.
The safety precautions are ranked as "Warning" and "Caution" in this instruction manual.

! **WARNING**	When a dangerous situation may occur if handling is mistaken leading to fatal or major injuries.
! **CAUTION**	When a dangerous situation may occur if handling is mistaken leading to medium or minor injuries, or physical damage.

Note that some items described as cautions may lead to major results depending on the situation. In any case, important information that must be observed is described.

For Safe Operation

1. Prevention of electric shocks

WARNING

- Never open the front case or terminal covers while the power is ON or the unit is running, as this may lead to electric shocks.
- Never run the unit with the front case or terminal cover removed. The high voltage terminal and charged sections will be exposed and may lead to electric shocks.
- Never open the front case or terminal cover at times other than wiring work or periodic inspections even if the power is OFF. The insides of the control unit and servo amplifier are charged and may lead to electric shocks.
- When performing wiring work or inspections, turn the power OFF, wait at least ten minutes, and then check the voltage with a tester, etc. Failing to do so may lead to electric shocks.
- Always ground the control unit, servo amplifier and servomotor with Class 3 grounding. Do not ground commonly with other devices.
- The wiring work and inspections must be done by a qualified technician.
- Wire the units after installing the control unit, servo amplifier and servomotor. Failing to do so may lead to electric shocks or damage.
- Never operate the switches with wet hands, as this may lead to electric shocks.
- Do not damage, apply excessive stress, place heavy things on or sandwich the cables, as this may lead to electric shocks.
- Do not touch the control unit, servo amplifier or servomotor terminal blocks while the power is ON, as this may lead to electric shocks.
- Do not touch the internal power supply, internal grounding or signal wires of the control unit and servo amplifier, as this may lead to electric shocks.

2. For fire prevention

CAUTION

- Install the control unit, servo amplifier, servomotor and regenerative resistor on inflammable material. Direct installation on flammable material or near flammable material may lead to fires.
- If a fault occurs in the control unit or servo amplifier, shut the power OFF at the servo amplifier's power source. If a large current continues to flow, fires may occur.
- When using a regenerative resistor, shut the power OFF with an error signal. The regenerative resistor may abnormally overheat due to a fault in the regenerative transistor, etc., and may lead to fires.
- Always take heat measures such as flame proofing for the inside of the control panel where the servo amplifier or regenerative resistor is installed and for the wires used. Failing to do so may lead to fires.

3. For injury prevention

⚠ CAUTION

- Do not apply a voltage other than that specified in A172SHCPUN/A171SHCPUN user's manual, A273UHCPU user's manual, A173UHCPU(S1) user's manual or the instruction manual for the product you are using on any terminal. Doing so may lead to destruction or damage.
- Do not mistake the terminal connections, as this may lead to destruction or damage.
- Do not mistake the polarity (+/–), as this may lead to destruction or damage.
- The servo amplifier's heat radiating fins, regenerative resistor and servo amplifier, etc., will be hot while the power is ON and for a short time after the power is turned OFF. Do not touch these parts as doing so may lead to burns.
- Always turn the power OFF before touching the servomotor shaft or coupled machines, as these parts may lead to injuries.
- Do not go near the machine during test operations or during operations such as teaching. Doing so may lead to injuries.

4. Various precautions

Strictly observe the following precautions.
Mistaken handling of the unit may lead to faults, injuries or electric shocks.

(1) System structure

⚠ CAUTION

- Always install a leakage breaker on the control unit and servo amplifier power source.
- If installation of a magnetic contactor for power shut off during an error, etc., is specified in the instruction manual for the servo amplifier, etc., always install the magnetic contactor.
- Install an external emergency stop circuit so that the operation can be stopped immediately and the power shut off.
- Use the control unit, servo amplifier, servomotor and regenerative resistor with the combinations listed in A172SHCPUN/A171SHCPUN user's manual or the instruction manual for the product you are using. Other combinations may lead to fires or faults.
- If safety standards (ex., robot safety rules, etc.,) apply to the system using the control unit, servo amplifier and servomotor, make sure that the safety standards are satisfied.
- If the operation during a control unit or servo amplifier error and the safety direction operation of the control unit differ, construct a countermeasure circuit externally of the control unit and servo amplifier.
- In systems where coasting of the servomotor will be a problem during emergency stop, servo OFF or when the power is shut OFF, use dynamic brakes.
- Make sure that the system considers the coasting amount even when using dynamic brakes.
- In systems where perpendicular shaft dropping may be a problem during emergency stop, servo OFF or when the power is shut OFF, use both dynamic brakes and magnetic brakes.
- The dynamic brakes must be used only during emergency stop and errors where servo OFF occurs. These brakes must not be used for normal braking.
- The brakes (magnetic brakes) assembled into the servomotor are for holding applications, and must not be used for normal braking.
- Construct the system so that there is a mechanical allowance allowing stopping even if the stroke end limit switch is passed through at the max. speed.

⚠ CAUTION

- ⚠ Use wires and cables that have a wire diameter, heat resistance and bending resistance compatible with the system.
- ⚠ Use wires and cables within the length of the range described in A172SHCPUN/ A171SHCPUN user’s manual or the instruction manual for the product you are using.
- ⚠ The ratings and characteristics of the system parts (other than control unit, servo amplifier, servomotor) must be compatible with the control unit, servo amplifier and servomotor.
- ⚠ Install a cover on the shaft so that the rotary parts of the servomotor are not touched during operation.
- ⚠ There may be some cases where holding by the magnetic brakes is not possible due to the life or mechanical structure (when the ball screw and servomotor are connected with a timing belt, etc.). Install a stopping device to ensure safety on the machine side.

(2) Parameter settings and programming

⚠ CAUTION

- ⚠ Set the parameter values to those that are compatible with the control unit, servo amplifier, servomotor and regenerative resistor model and the system application. The protective functions may not function if the settings are incorrect.
- ⚠ The regenerative resistor model and capacity parameters must be set to values that conform to the operation mode, servo amplifier and servo power unit. The protective functions may not function if the settings are incorrect.
- ⚠ Set the mechanical brake output and dynamic brake output validity parameters to values that are compatible with the system application. The protective functions may not function if the settings are incorrect.
- ⚠ Set the stroke limit input validity parameter to a value that is compatible with the system application. The protective functions may not function if the setting is incorrect.
- ⚠ Set the servomotor encoder type (increment, absolute position type, etc.) parameter to a value that is compatible with the system application. The protective functions may not function if the setting is incorrect.
- ⚠ Set the servomotor capacity and type (standard, low-inertia, flat, etc.) parameter to values that are compatible with the system application. The protective functions may not function if the settings are incorrect.
- ⚠ Set the servo amplifier capacity and type parameters to values that are compatible with the system application. The protective functions may not function if the settings are incorrect.
- ⚠ Use the program commands for the program with the conditions specified in the instruction manual.
- ⚠ Set the sequence function program capacity setting, device capacity, latch validity range, I/O assignment setting, and validity of continuous operation during error detection to values that are compatible with the system application. The protective functions may not function if the settings are incorrect.
- ⚠ Some devices used in the program have fixed applications, so use these with the conditions specified in the instruction manual.
- ⚠ The input devices and data registers assigned to the link will hold the data previous to when communication is terminated by an error, etc. Thus, an error correspondence interlock program specified in the instruction manual must be used.
- ⚠ Use the interlock program specified in the special function unit's instruction manual for the program corresponding to the special function unit.

(3) Transportation and installation

⚠ **CAUTION**

⚠ Transport the product with the correct method according to the weight.
⚠ Use the servomotor suspension bolts only for the transportation of the servomotor. Do not transport the servomotor with machine installed on it.
⚠ Do not stack products past the limit.
⚠ When transporting the control unit or servo amplifier, never hold the connected wires or cables.
⚠ When transporting the servomotor, never hold the cables, shaft or detector.
⚠ When transporting the control unit or servo amplifier, never hold the front case as it may fall off.
⚠ When transporting, installing or removing the control unit or servo amplifier, never hold the edges.
⚠ Install the unit according to A172SHCPUN/A171SHCPUN user's manual, A273UHCPU user's manual, A173UHCPU(S1) user's manual or the instruction manual for the product you are using in a place where the weight can be withstood.
⚠ Do not get on or place heavy objects on the product.
⚠ Always observe the installation direction.
⚠ Keep the designated clearance between the control unit or servo amplifier and control panel inner surface or the control unit and servo amplifier, control unit or servo amplifier and other devices.
⚠ Do not install or operate control units, servo amplifiers or servomotors that are damaged or that have missing parts.
⚠ Do not block the intake/outtake ports of the servomotor with cooling fan.
⚠ Do not allow conductive matter such as screw or cutting chips or combustible matter such as oil enter the control unit, servo amplifier or servomotor.
⚠ The control unit, servo amplifier and servomotor are precision machines, so do not drop or apply strong impacts on them.
⚠ Securely fix the control unit and servo amplifier to the machine according to A172SHCPUN/A171SHCPUN/A273UHCPU/A173UHCPU(S1) user's manual or the instruction manual for the product you are using. If the fixing is insufficient, these may come off during operation.
⚠ Always install the servomotor with reduction gears in the designated direction. Failing to do so may lead to oil leaks.
⚠ Store and use the unit in the following environmental conditions.

Environment	Conditions	
	Control unit/servo amplifier	Servomotor
Ambient temperature	0°C to +55°C (With no freezing)	0°C to +40°C (With no freezing)
Ambient humidity	According to each instruction manual.	80%RH or less (With no dew condensation)
Storage temperature	According to each instruction manual.	–20°C to +65°C
Atmosphere	Indoors (where not subject to direct sunlight). No corrosive gases, flammable gases, oil mist or dust must exist.	
Altitude	1000m or less above sea level.	
Vibration	According to each instruction manual.	

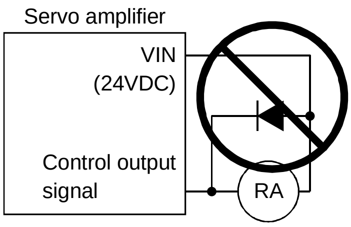
⚠ CAUTION

- ⚠ When coupling with the synchronization encoder or servomotor shaft end, do not apply impact such as by hitting with a hammer. Doing so may lead to detector damage.
- ⚠ Do not apply a load larger than the tolerable load onto the servomotor shaft. Doing so may lead to shaft breakage.
- ⚠ When not using the unit for a long time, disconnect the power line from the control unit or servo amplifier.
- ⚠ Place the control unit and servo amplifier in static electricity preventing vinyl bags and store.
- ⚠ When storing for a long time, contact the Service Center or Service Station.

(4) Wiring

⚠ CAUTION

- ⚠ Correctly and securely wire the wires. Reconfirm the connections for mistakes and the terminal screws for tightness after wiring. Failing to do so may lead to run away of the servomotor.
- ⚠ After wiring, install the protective covers such as the terminal covers to the original positions.
- ⚠ Do not install a phase advancing capacitor, surge absorber or radio noise filter (option FR-BIF) on the output side of the servo amplifier.
- ⚠ Correctly connect the output side (terminals U, V, W). Incorrect connections will lead the servomotor to operate abnormally.
- ⚠ Do not connect a commercial power supply to the servomotor, as this may lead to trouble.
- ⚠ Do not mistake the direction of the surge absorbing diode installed on the DC relay for the control signal output of brake signals, etc. Incorrect installation may lead to signals not being output when trouble occurs or the protective functions not functioning.



- ⚠ Do not connect or disconnect the connection cables between each unit, the encoder cable or sequence expansion cable while the power is ON.
- ⚠ Securely tighten the cable connector fixing screws and fixing mechanisms. Insufficient fixing may lead to the cables combing off during operation.
- ⚠ Do not bundle the power line or cables.

(5) Trial operation and adjustment

⚠ CAUTION

- ⚠ Confirm and adjust the program and each parameter before operation. Unpredictable movements may occur depending on the machine.
- ⚠ Extreme adjustments and changes may lead to unstable operation, so never make them.
- ⚠ When using the absolute position system function, on starting up, and when the controller or absolute value motor has been replaced, always perform a home position return.

(6) Usage methods

⚠ CAUTION

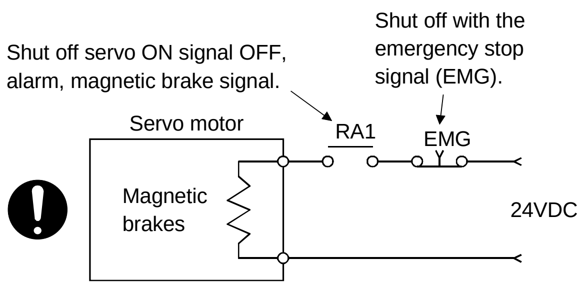
- ⚠ Immediately turn OFF the power if smoke, abnormal sounds or odors are emitted from the control unit, servo amplifier or servomotor.
- ⚠ Always execute a test operation before starting actual operations after the program or parameters have been changed or after maintenance and inspection.
- ⚠ The units must be disassembled and repaired by a qualified technician.
- ⚠ Do not make any modifications to the unit.
- ⚠ Keep the effect or magnetic obstacles to a minimum by installing a noise filter or by using wire shields, etc. Magnetic obstacles may affect the electronic devices used near the control unit or servo amplifier.
- ⚠ When using the CE mark-compatible equipment, refer to "EMC Installation Guidelines" (data number IB(NA)-*****-*) for the motion controller and to the corresponding EMC guideline data for the servo amplifier, inverter and other equipment.
- ⚠ Use the units with the following conditions.

Item	Conditions
Input power	According to A172SHCPUN/A171SHCPUN/ A273UHCPU/A173UHCPU(S1) user's manual.
Input frequency	According to A172SHCPUN/A171SHCPUN/ A273UHCPU/A173UHCPU(S1) user's manual.
Tolerable momentary power failure	According to A172SHCPUN/A171SHCPUN/ A273UHCPU/A173UHCPU(S1) user's manual.

(7) Remedies for errors

⚠ CAUTION

- ⚠ If an error occurs in the self diagnosis of the control unit or servo amplifier, confirm the check details according to the instruction manual, and restore the operation.
- ⚠ If a dangerous state is predicted in case of a power failure or product failure, use a servomotor with magnetic brakes or install a brake mechanism externally.
- ⚠ Use a double circuit construction so that the magnetic brake operation circuit can be operated by emergency stop signals set externally.
- ⚠ If an error occurs, remove the cause, secure the safety and then resume operation.
- ⚠ The unit may suddenly resume operation after a power failure is restored, so do not go near the machine. (Design the machine so that personal safety can be ensured even if the machine restarts suddenly.)



(8) Maintenance, inspection and part replacement

⚠ CAUTION
⚠ Perform the daily and periodic inspections according to the instruction manual. ⚠ Perform maintenance and inspection after backing up the program and parameters for the control unit and servo amplifier. ⚠ Do not place fingers or hands in the clearance when opening or closing any opening. ⚠ Periodically replace consumable parts such as batteries according to A172SHCPUN/ A171SHCPUN user's manual, A273UHCPU user's manual, A173UHCPU(S1) user's manual or the instruction manual for the product you are using.

⚠ CAUTION
⚠ Do not touch the lead sections such as ICs or the connector contacts. ⚠ Do not place the control unit or servo amplifier on metal that may cause a power leakage or wood, plastic or vinyl that may cause static electricity buildup. ⚠ Do not perform a megger test (insulation resistance measurement) during inspection. ⚠ When replacing the control unit or servo amplifier, always set the new unit settings correctly. ⚠ When the controller or absolute value motor has been replaced, carry out a home position return operation using one of the following methods, otherwise position displacement could occur. 1) After writing the servo data to the PC using peripheral device software, switch on the power again, then perform a home position return operation. 2) Using the backup function of the peripheral device software, load the data backed up before replacement. ⚠ After maintenance and inspections are completed, confirm that the position detection of the absolute position detector function is correct. ⚠ Do not short circuit, charge, overheat, incinerate or disassemble the batteries. ⚠ The electrolytic capacitor will generate gas during a fault, so do not place your face near the control unit or servo amplifier. ⚠ The electrolytic capacitor and fan will deteriorate. Periodically change these to prevent secondary damage from faults. Replacements can be made by the Service Center or Service Station.

(9) Disposal

⚠ CAUTION
⚠ Dispose of this unit as general industrial waste. ⚠ Do not disassemble the control unit, servo amplifier or servomotor parts. ⚠ Dispose of the battery according to local laws and regulations.

(10) General cautions

⚠ CAUTION
⚠ All drawings provided in the instruction manual show the state with the covers and safety partitions removed to explain detailed sections. When operating the product, always return the covers and partitions to the designated positions, and operate according to the instruction manual.

Revisions

*The manual number is given on the bottom left of the back cover.

Print Date	*Manual Number	Revision
Feb., 2000	IB(NA)-0300014-A	First edition

CONTENTS

1. GENERAL DESCRIPTION

This manual describes the positioning control parameters, positioning-dedicated devices, positioning methods and other information required to execute positioning control with the motion controller (SV43). The motion controller (SV43) uses the NC language (EIA) (hereafter referred to as the "motion program") as a programming language.
The motion controller (SV43) can exercise the following positioning control.

Applicable CPU	Number of Axes Controlled in Positioning Control
A172SHCPUN	8
A171SHCPUN	4
A273UHCPU (32 axis feature)	32
A173UHCPU(S1)	32

In this manual, the above CPUs are collectively referred to as the "servo system CPUs".
The following software packages are used to make system settings, and to set, test and monitor the servo parameters and motion programs.

- SW2SRX-GSV43P software package
- SW2NX-GSV43P software package

} .... Abbreviated to "GSV43P"

⚠ CAUTION

- ⚠ When designing the system, provide external protective and safety circuits to ensure safety in the event of trouble with the motion controller.
- ⚠ There are electronic components which are susceptible to the effects of static electricity mounted on the printed circuit board. When handling printed circuit boards with bare hands you must ground your body or the work bench.
  Do not touch current-carrying or electric parts of the equipment with bare hands.
- ⚠ Make parameter settings within the ranges stated in this manual.
- ⚠ Use the program instructions that are used in programs in accordance with the conditions stipulated in this manual.
- ⚠ Some devices for use in programs have fixed applications: they must be used in accordance with the conditions stipulated in this manual.

Conventions Used in this Manual

Positioning signals are always indicated in the following order: signal for A172SHCPUN → signal for A171SHCPUN → signal for A273UHCPU (32 axes feature) → signal for A173UHCPU(S1). If only one positioning signal is indicated, this means that the signal is used in common by every CPUs.
The explanatory text is written with reference to the A172SHCPU: if you are not using an A172SHCPUN, the positioning signals should be read as the positioning signals for the CPU you are using.
(For the positioning signals used with each CPU, refer to Appendix 6.)

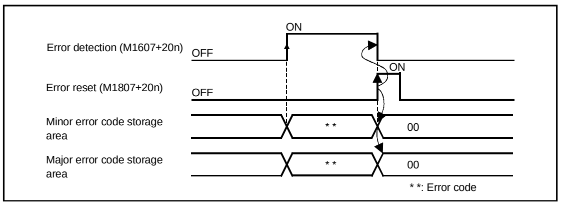
A172SHCPUN/A171SHCPUN

A273UHCPU (32 axis feature) /A173UHCPU(S1)

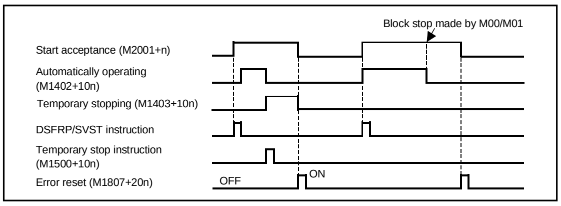
3. POSITIONING SIGNALS

3.1.24 Error reset command (M1807+20n/M3207+20n)

(1) The error reset command is used to clear the minor error code or major error code storage area of an axis for which the error detection signal has come ON (M1607+20n), and to reset the error detection signal (M1607+20n).



(2) The motion program running status is reset if the error is reset during a temporary stop (M1403+10n) made by the stop command (M1800+20n) during an automatic start or if the error is reset during a block stop made by M00/M01.



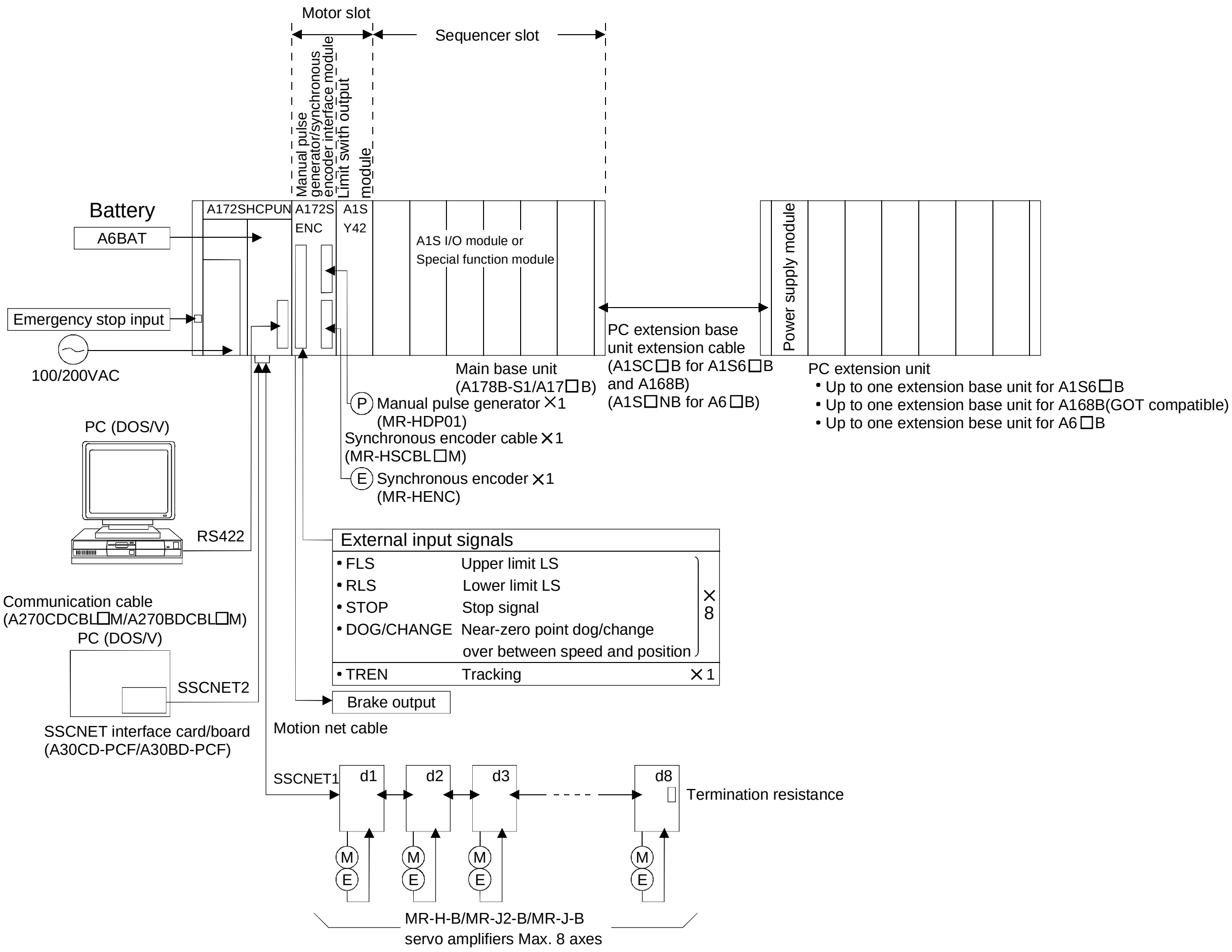
(3) When the error reset command is switched on during automatic operation (M1402+10n ON), the above reset processing is performed after stop processing is executed under the temporary stop command (M1500+10n).

3 - 19

1.1 System Configuration

1.1.1 A172SHCPUN system overall configuration

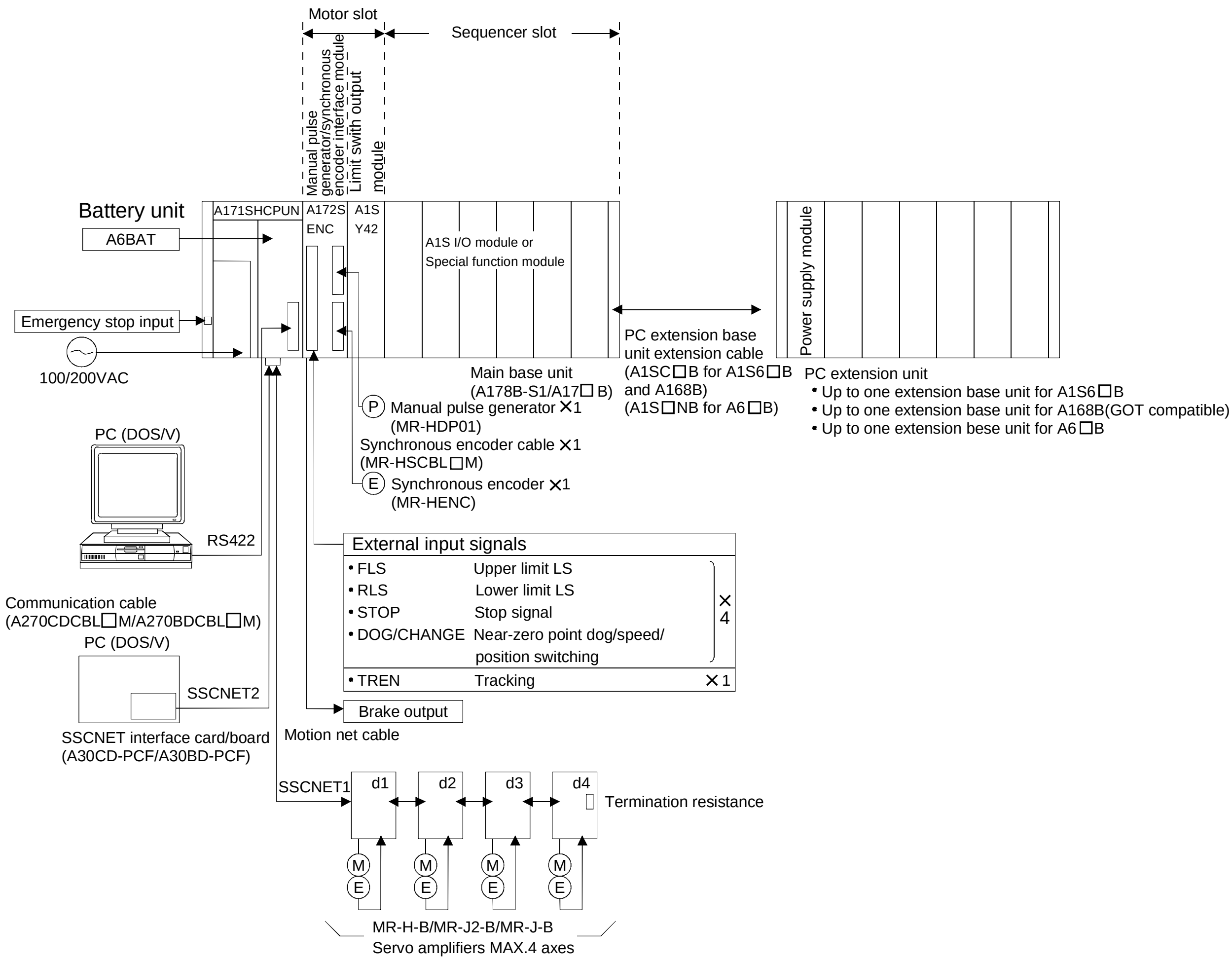
An example system configuration with A172SHCPUN is shown below.



NOTES
(1) When using the PC extension base and bus connection type GOT, choose the A168B as the PC extension base. When not using the PC extension base, the bus connection type GOT can be connected directly to the PC extension base connector of the main base unit. (2) Use motion slots to mount PC A1S I/O modules if necessary. (3) When the power supply to the servo system CPU is switched ON and OFF, erroneous process outputs may temporarily be made due to the delay between the servo system CPU power supply and the external power supply for processing (especially DC), and the difference in startup times. For example, if the power supply to the servo system CPU comes ON after the external power supply for processing comes ON at a DC output module, the DC output module may temporarily give erroneous outputs when the power to the servo system CPU comes ON. Accordingly a circuit that ensures that the power supply to the servo system CPU comes ON first should be constructed.

1.1.2 A171SHCPUN system overall configuration

An example system configuration with A171SHCPUN is shown below.



NOTES

(1) When using the PC extension base and bus connection type GOT, choose the A168B as the PC extension base.
When not using the PC extension base, the bus connection type GOT can be connected directly to the PC extension base connector of the main base unit.

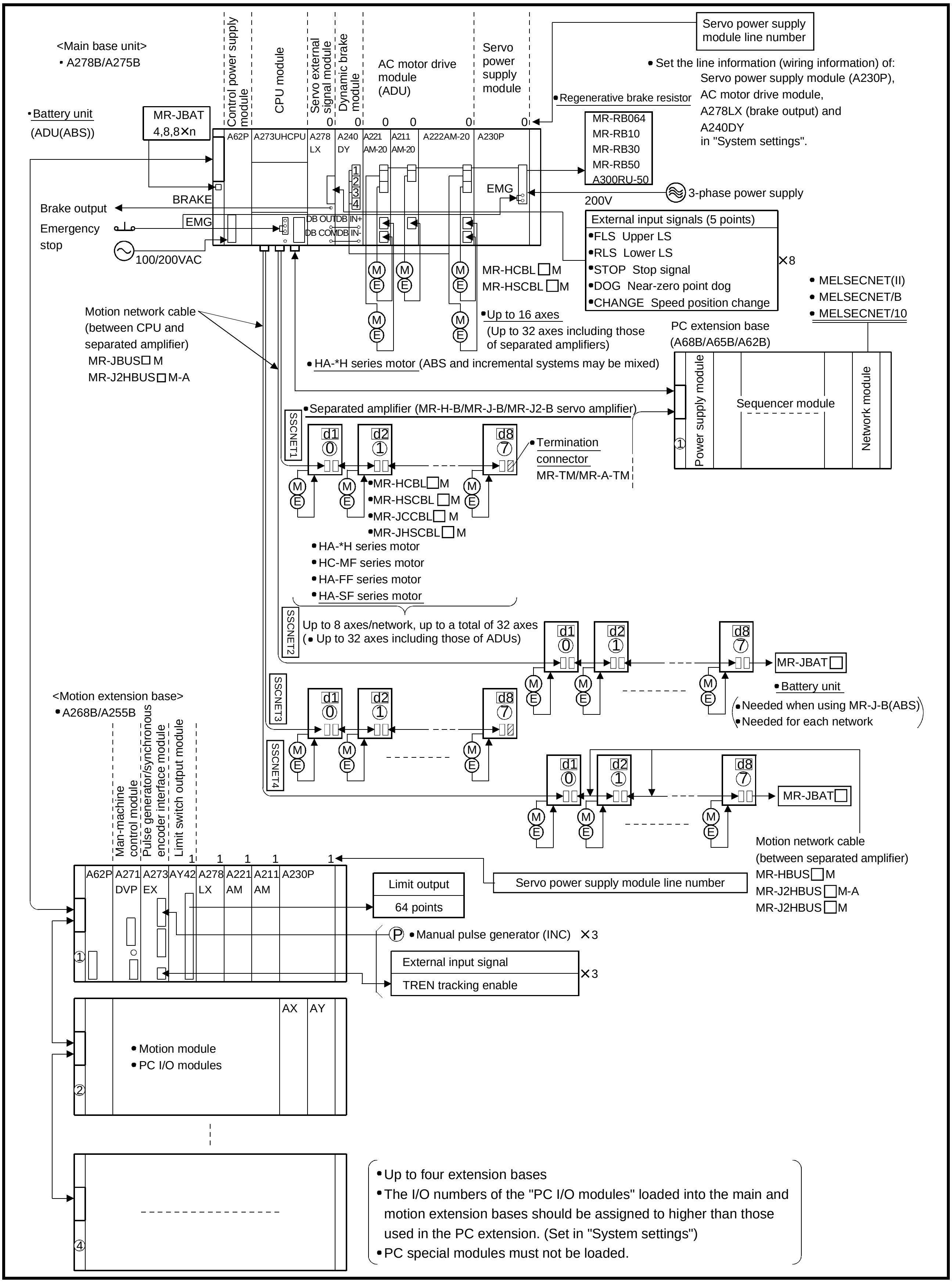
(2) Use motion slots to mount PC A1S I/O modules if necessary.

(3) Though A172SENC has external input signals for 8 axes, make settings for the first 4 axes (PXO to PXOF).

(4) When the power supply to the servo system CPU is switched ON and OFF, erroneous process outputs may temporarily be made due to the delay between the servo system CPU power supply and the external power supply for processing (especially DC), and the difference in startup times.
For example, if the power supply to the servo system CPU comes ON after the external power supply for processing comes ON at a DC output module, the DC output module may temporarily give erroneous outputs when the power to the servo system CPU comes ON. Accordingly a circuit that ensures that the power supply to the servo system CPU comes ON first should be constructed.

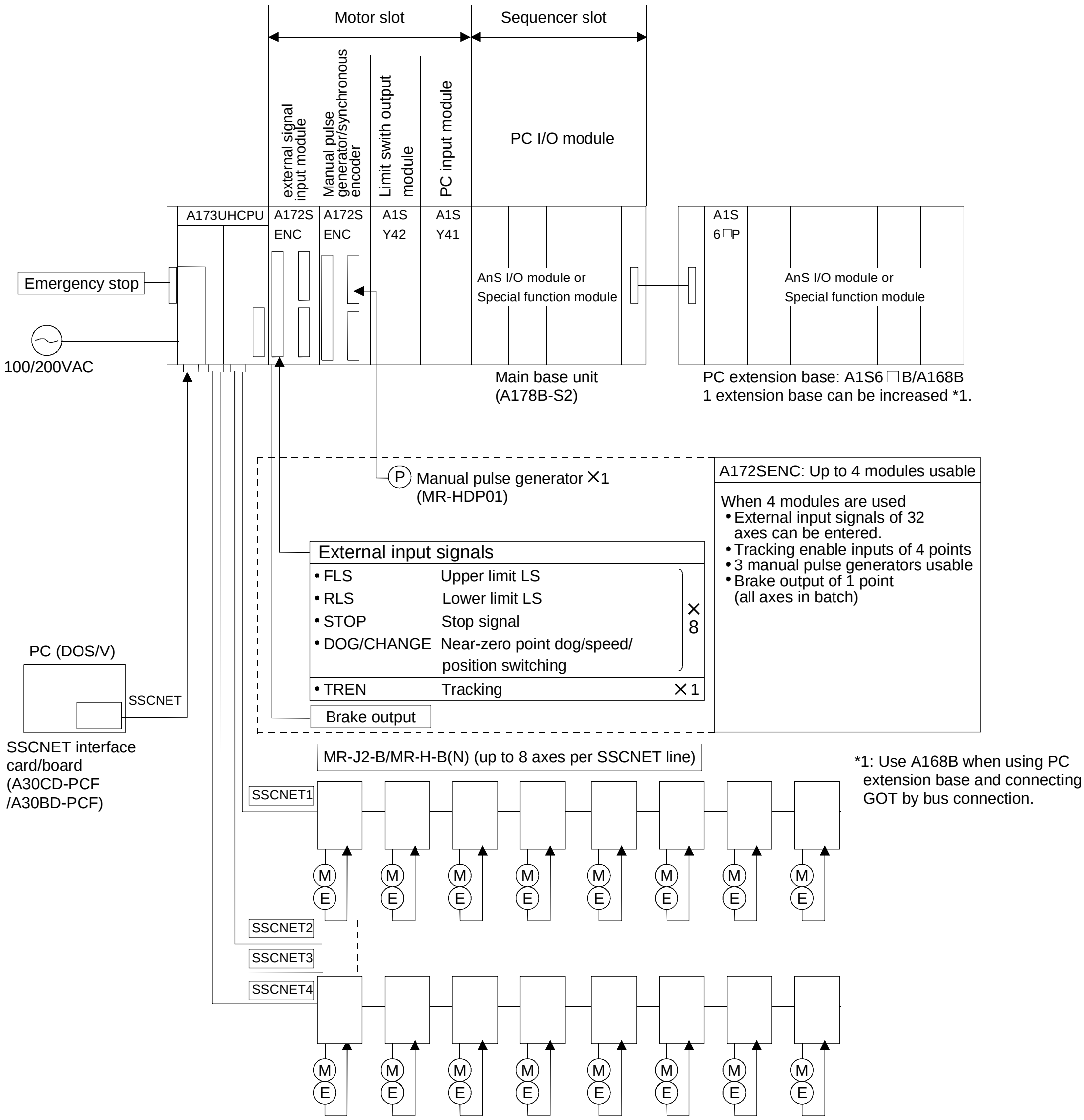
1.1.3 A273UHCPU (32 axis feature) system overall configuration

The system configuration example of the motion controller (SV43) is shown below.



1.1.4 A173UHCPU(S1) system overall configuration

An example system configuration with A173UHCPU(S1) is shown below.



NOTES
When the power supply to the servo system CPU is switched ON and OFF, erroneous process outputs may temporarily be made due to the delay between the servo system CPU power supply and the external power supply for processing (especially DC), and the difference in startup times. For example, if the power supply to the servo system CPU comes on after the external power supply for processing comes on at a DC output module, the DC output module may temporarily give erroneous outputs when the power to the servo system CPU comes on. Accordingly a circuit that ensures that the power supply to the servo system CPU comes on first should be constructed.

1.1.5 System configuration precautions

The following table summarizes the notes on system configuration, system setup items, and relative checks that differ from those of the A171SCPU.

<table>
<tr><th>Product Name</th><th>Module Name</th><th>Number of Available Modules</th><th>System Setup Item</th><th>Relative Check</th><th>Notes and Remarks</th></tr>
<tr><td>Separated amplifier</td><td>MR-J2-B
MR-H-B
MR-J-B</td><td>• Max. 8 axes for A172SHCPUN
• Max. 4 axes for A171SHCPUN</td><td>1. MR-J2-B allows the use of the following motors with high-resolution encoders.
• HC-MF***W1 (32768PLS)
• HA-FF***W1 (32768PLS)
• HC-SF**2W2 (131072PLS)
2. [Allowable travel value during power-off]
When ABS motor is used, set the allowable travel value during servo amplifier power-off by rpm (rotations per minute).
This setting value is used for checking when the servo amplifier is switched ON.<table><tr><th>Setting range</th><th>Default value</th></tr><tr><td>0 to 16383 (rpm)</td><td>10 (rpm)</td></tr></table></td><td></td><td>• Connect the servo amplifier to the 'SSCNET1' interface.
• The setting range changes for high-resolution encoder support.</td></tr>
<tr><td rowspan="2">Manual pulse generator /synchronous encoder interface module</td><td>A172SENC</td><td>1</td><td>1. External signals
(1) Set the axis numbers which use external signals FLS, RLS, STOP, and DOG/CHANGE for A172SENC CTRL connector signals PX0 to PX1F.
The axes which do not use external signals may be left unspecified.<table><tr><th>CPU unit</th><th>Setting range</th><th>Default value</th></tr><tr><td>A172SHCPUN</td><td>Set axes 1 to 8 for PX0 to PX1F.</td><td>Axes 1 to 8 are set.</td></tr><tr><td>A171SHCPUN</td><td>Set axes 1 to 4 for the first half (PX0 to PX0F).</td><td>Axes 1 to 4 are set.</td></tr></table></td><td>• The same axis number must not be set.</td><td rowspan="2">• The external signal setup window has been improved for a better understanding.
• The conventional A171SENC can also be used for A171SHCPUN and A172SHCPUN. However, it must be set as A172SENC during system setting.</td></tr>
<tr><td>A171SENC</td><td>0</td><td>Settings cannot be made.</td><td></td></tr>
<tr><td>Man/machine control module</td><td>A271DVP</td><td>0</td><td>Not available. Settings cannot be made.</td><td></td><td></td></tr>
<tr><td>PC CPU I/O module (motion slot)</td><td>A1SX**
A1SY**
A1SH42</td><td>Up to 256 I/O points (total)</td><td>1. Set the number of points and the starting I/O number for PC CPU I/O modules to be mounted on the motion extension base unit.
The number to be set must not precede the I/O numbers for use by the PC extension base unit.<table><tr><th>CPU unit</th><th>Effective setting range</th><th>Default value</th></tr><tr><td>A172SHCPUN</td><td>X/Y0~X/Y3FF</td><td>——</td></tr><tr><td>A171SHCPUN</td><td>X/Y0~X/Y1FF</td><td>——</td></tr></table></td><td>• The total number of points must be less than or equal to 256.
• The starting I/O number plus number of occupied points must be less than or equal to X/Y800.</td><td>• Though settings can be made within a range of X/Y0 to X/Y7FF, they must be made in the range defined in the left-hand column.</td></tr>
<tr><td rowspan="2">PC extension base unit</td><td>A1S68B
A1S65B</td><td>Up to 1 stage</td><td rowspan="2" colspan="2"></td><td>• Use this unit for systems capable of one-stage extension.</td></tr>
<tr><td>A168B</td><td>Up to 3 stages</td><td>• Use this base in a system having two or more extension bases.</td></tr>
</table>

POINT

1. When using the existing A171SCPU user program and parameters, perform the following procedure:
   (1) Start the peripheral S/W package by A172SHCPUN or A171SHCPUN, then read the sequence file and servo file created for A171SCPU via the File Read function.
   ↓
   (2) Display the System Setup screen.
   The existing system status is displayed with the following alert:
   (Start by A172SHCPUN)

Dialog	
Replaces A171SCPU with A172SHCPUN.	• • • The character string "A171SHCPUN" is displayed only when A171SHCPUN is used for startup.
Replaces A171SENC with A172SENC.	• • • This message is displayed only when A171SENC has been set.
YES NO	

   ↓
   (3) Select “YES” and the existing settings will be replaced with those for the startup CPU module.
   Select “NO” and the existing A171SCPU settings will remain in effect.
   (4) Utilization of motion program
      (a) The handling of the variable type changes.
      When a variable has no representation of the type, it is handled as a 32-bit integer type in the A171SCPU.
      A variable is handled as a 16-bit integer type in the A172SHCPUN/ A171SHCPUN.
      "L" or ":L" is added when a variable is handled as a 32-bit integer type in the A172SHCPUN/A171SHCPUN.
      Example:
      1) For A171SCPU
      #0 ..... [D1,D0] 32-bit integer type
      2) For A172SHCPUN/A171SHCPUN
      #0 ..... [D0] 16-bit integer type
      When handled as 32-bit integer type
      #0:L ..... [D1,D0]
      For more information, refer to "6.6 Method for Setting the Positioning Data".
      (b) Add a return code to the last line of a program.
      The GSV43P edit screen changes.
      Before utilizing the program created on SW2SRX-GSV43 Ver. F/SW2NX-GSV43P Ver. B or earlier, add a return code to the last line of the program.
      After utilization, make an error check for each program number.
      The program may not be displayed properly in the presence of an error.

   * Other than system setup data and motion program data can be used without change.

1.2 Table of Software Package

Use	Peripheral devices	Peripheral software package			Unit OS software package model name			
			Model name	Applicable Version	For A172SH CPUN	For A171SH CPUN	For A273UH CPU (32 axis feature)	For A173UH CPU
For machine tool peripheral	DOS/V	English	SW2SRX-GSV43PE	From 00A on	SW0SRX-SV43C	SW0SRX-SV43F	SW2SRX-SV43U	SW2SRX-SV43A

1.3 Positioning Control by the Servo System CPU

A servo system CPU can execute positioning control and sequence control for 8 axes (when using A172SHCPUN), 4 axes (when using A171SHCPUN) or 32 axes (when using A273UHCPU (32 axis feature) or A173UHCPU) by means of a multi-axis positioning control CPU (hereafter called the "PCPU") and a sequence control CPU (hereafter called the "SCPU").
Sequence control capabilities are equivalent to those of the A2SHCPU's I/O and memory enhanced version (when using A172SHCPUN), to those of the A2SHCPU (when using A171SHCPUN), or to those of the A3U (when using A273UHCPU or A173UHCPU).

(1) Control handled by the SCPU
  (a) Sequence control
    The SCPU controls I/O modules and special function modules in accordance with the sequence program.
    (The method for executing a sequence program is the same as in the A2SHCPU's I/O and memory enhanced version, the A2SHCPU and the A3U.)

  (b) Start of positioning start in accordance with sequence program, and setting of positioning data
    1) The SCPU requests motion programs to be executed by the DSFRP instruction (up to 3 axes for interpolation) or by the SVST instruction (up to 4 axes for interpolation).
    2) The SCPU make a home position return or speed change using the DSFLP instruction or CHGA/CHGV instruction.
    3) The SCPU performs JOG operation.
    4) The SCPU sets the data required to execute manual pulse generator operation.

(2) Control handled by the PCPU
  (a) The PCPU executes motion programs requested to be run by the DSFRP/SVST instruction from the sequence program to exercise the preset positioning control.
    Positioning control data are the positioning control parameters and the positioning data set in motion programs.

  (b) The PCPU changes the set home position return or positioning speed set in the DSFLP/CHGA/CHGV instruction from the sequence program.

  (c) The PCPU performs positioning with a manual pulse generator.

[Executing Positioning Control with a Servo System CPU]

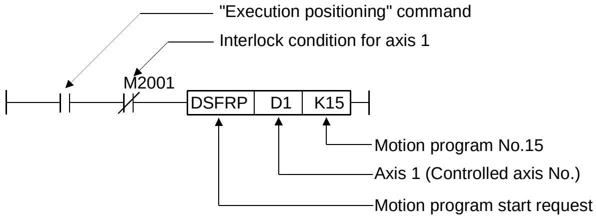
The servo system CPU executes positioning control in accordance with the motion programs designated by the sequence program of the SCPU.
An overview of the method used for positioning control is presented below.

Servo System CPU System

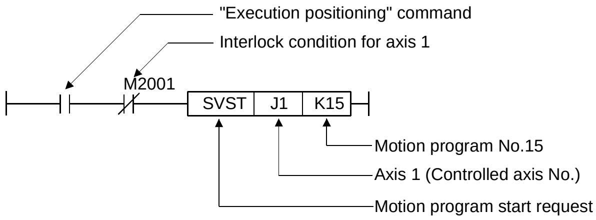
SCPU Control

Sequence program .... Created and modified using a peripheral device*1

Example: DSFRP instruction (A273UHCPU (32 axIs feature) and A173UHCPU: unusable



Example: SVST instruction



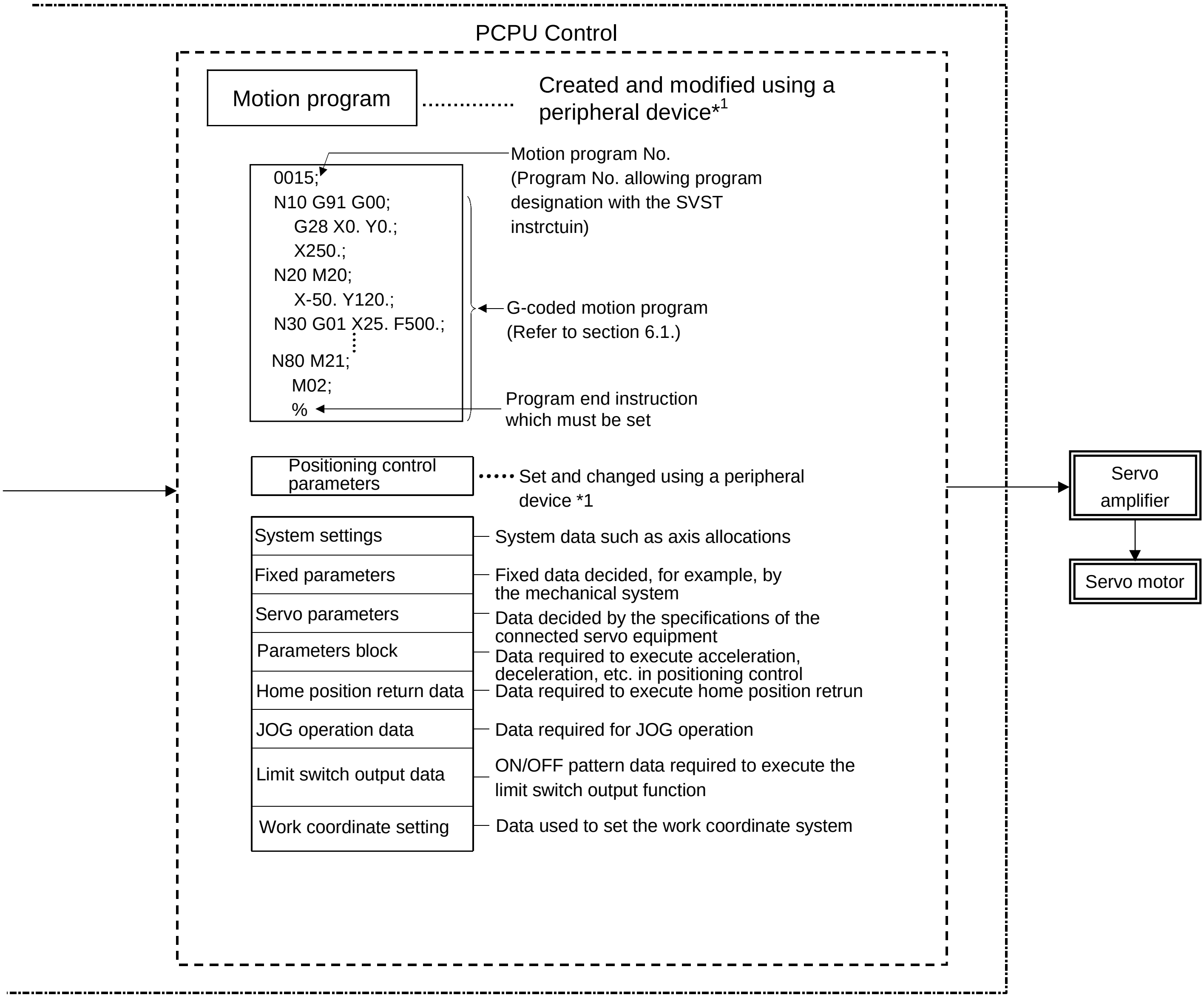
1) In the sequence program, the motion program number and controlled axis number are set with the DSFRP/SVST instruction.
2) When the DSFRP/SVST instruction is executed, the PCPU is requested to execute the program with the designated servo program number.

Request for execution of motion program

(1) Motion programs and positioning control parameters are set using a peripheral device.

(2) Positioning is started by the sequence program (DSFRP/SVST instruction).
   (a) The motion program number and controlled axis number are designated by the DSFRP/SVST instruction.
      1) The motion program number can be set either directly or indirectly.
      2) The controlled axis number can only be set directly.

(3) The positioning specified by the designated motion program is executed.



REMARK

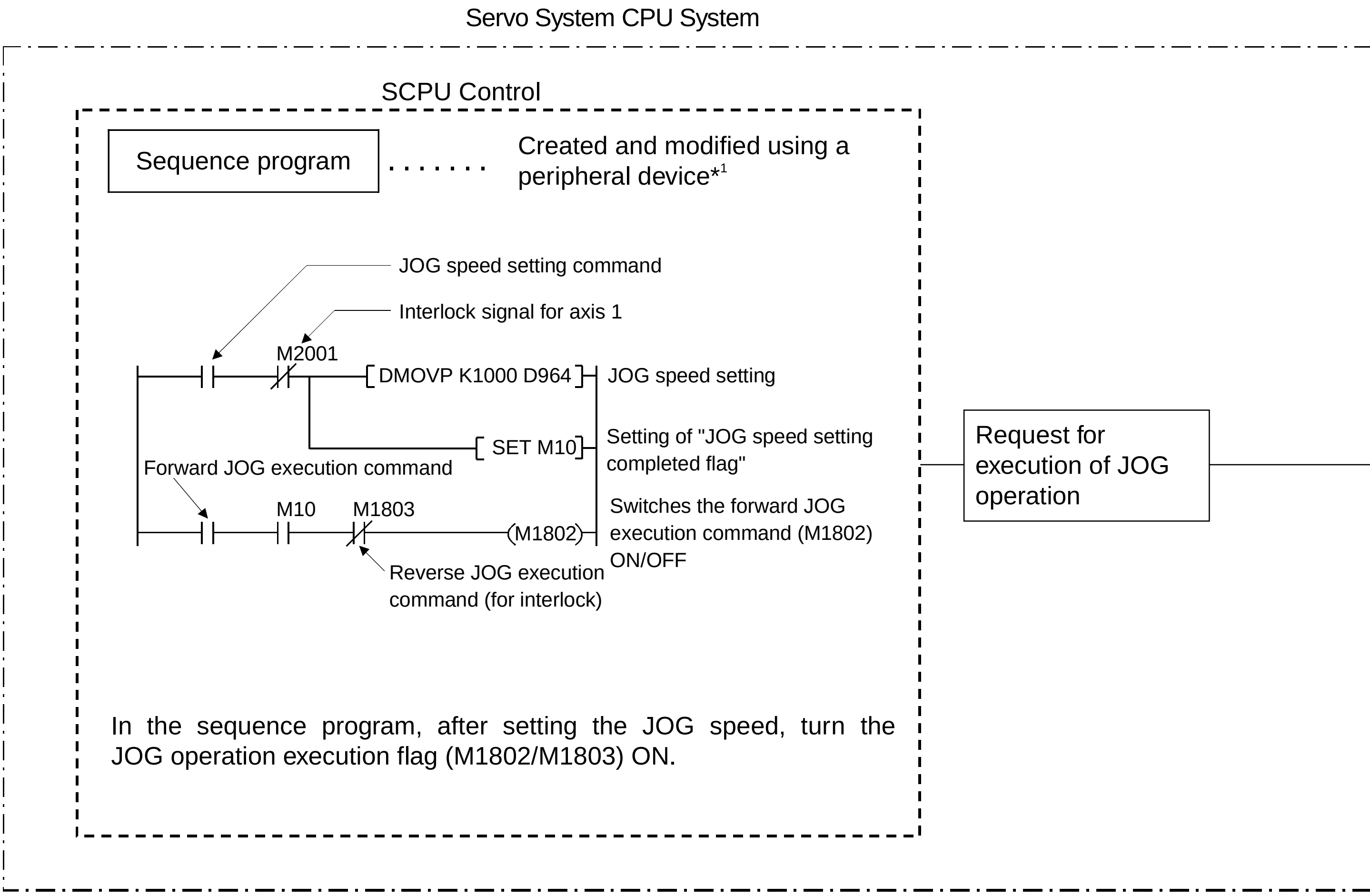
*1: Any of the following peripheral devices, running the GSV43P software, can be used.

- An IBM PC/AT or 100% compatible machine in which PC-DOS 5.0 or a later version has been installed (hereafter called an "IBM PC")

IBM is a registered trade mark of International Business Machines Corporation

[Executing JOG Operation with a Servo System CPU]

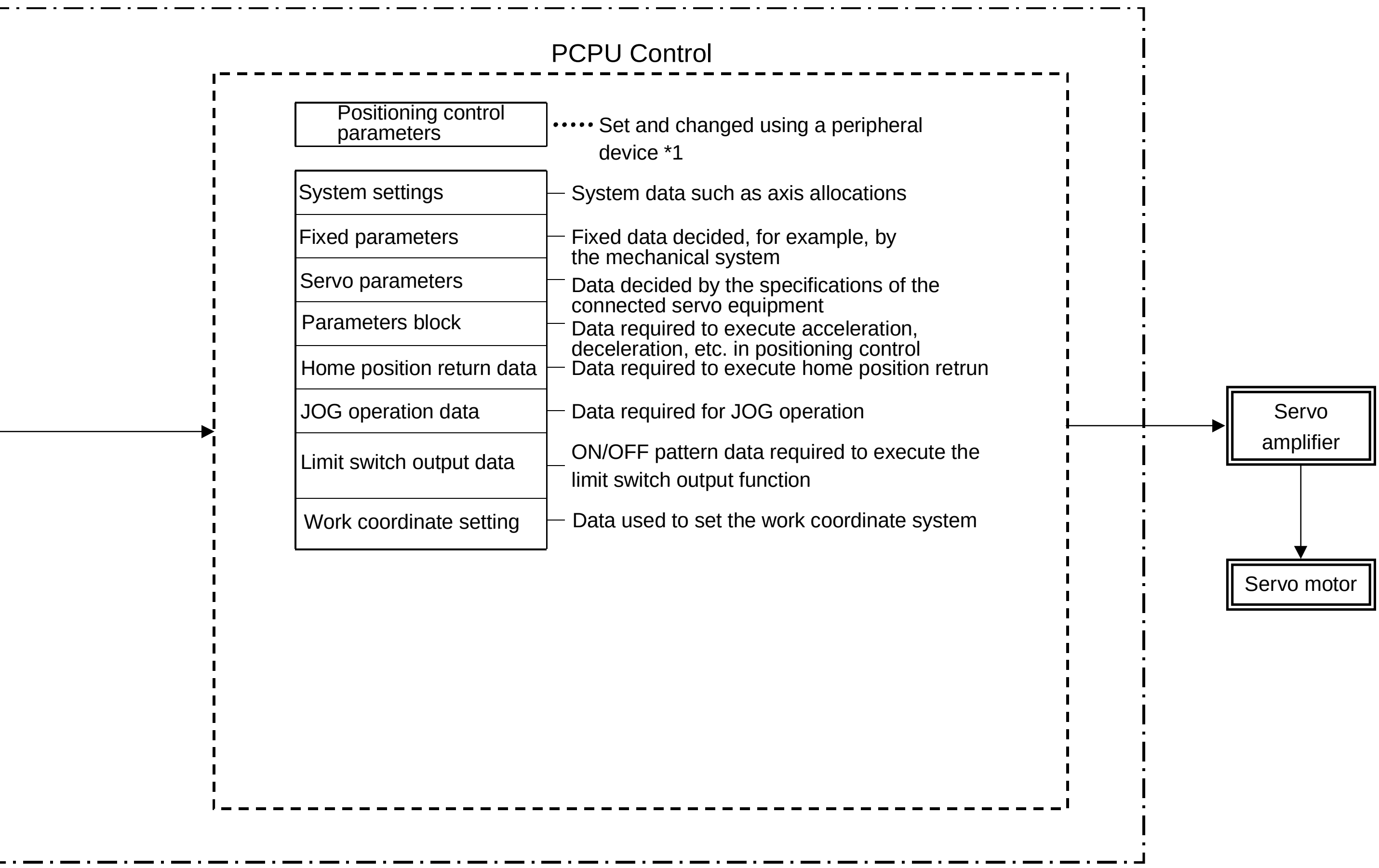
The servo system CPU can be used to perform JOG operation on a designated axis in accordance with a sequence program.
An overview of JOG operation is presented below.



(1) Set the positioning control parameters using a peripheral device.

(2) Using the sequence program, set the JOG speed in the JOG operation speed setting register for each axis.

(3) JOG operation is executed while the JOG operation execution flag is kept ON by the sequence program.

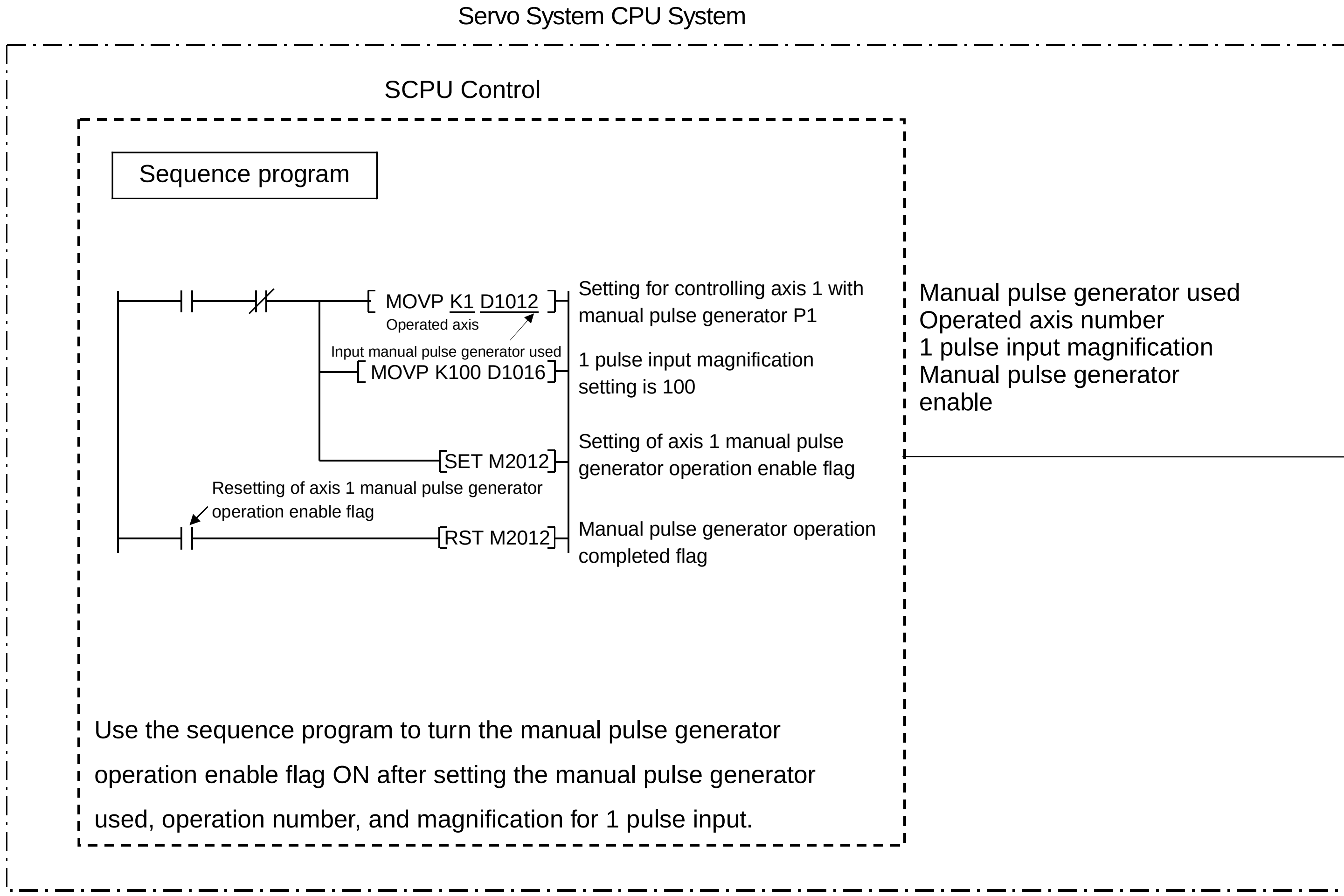


REMARK

*1: Any of the following peripheral devices, running the GSV43P software, can be used.
- IBM PC

[Executing Manual Pulse Generator Operation with a Servo System CPU]

When executing positioning control with a manual pulse generator connected to an A172SENC or A171SENC, manual pulse generator operation must be enabled by the sequence program.
An overview of positioning control using manual pulse generator operation is presented below.

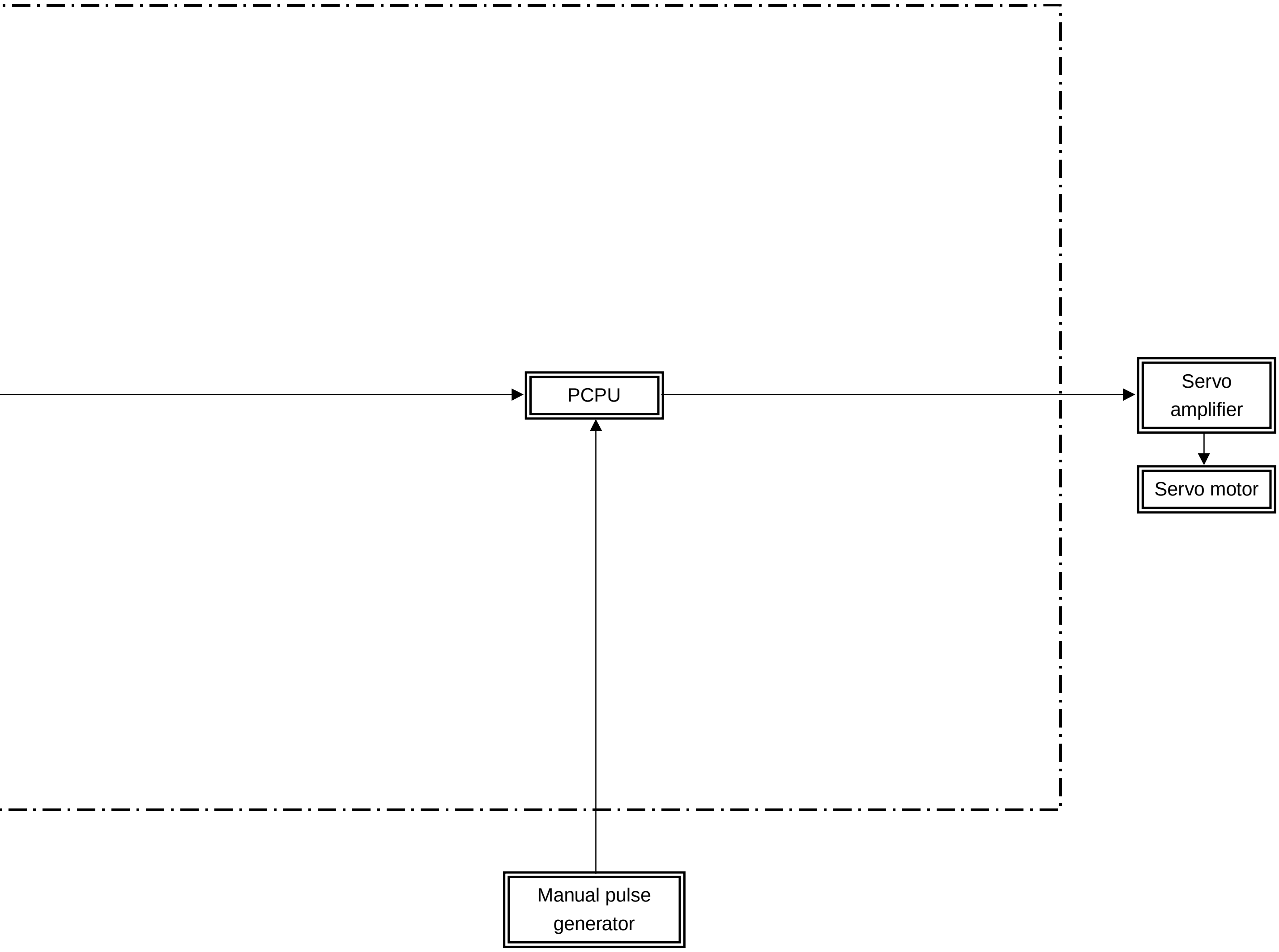


(1) Set the manual pulse generator used, operated axis number, and magnification for 1 pulse input by using the sequence program.

(2) Turn the manual pulse generator operation enable flag ON by using the sequence program.
.... manual pulse generator operation enabled

(3) Perform positioning by operating the manual pulse generator.

(4) Turn the manual pulse generator operation enable flag OFF by using the sequence program.
.... manual pulse generator operation completed


PCPU
Servo amplifier
Servo motor
Manual pulse generator

(1) Positioning control parameters
The positioning control parameters are classified into the eight types shown below.
Parameter data can be set and corrected interactively by using a peripheral device.

	Item	Description	Reference
1	System settings	The system settings set the modules used, axis numbers, etc.	Section 4.1
2	Fixed parameters	Fixed parameters are set for each axis. Their settings are predetermined by the mechanical system. They are used for servo motor control during positioning control.	Section 4.2
3	Servo parameters	Servo parameters are set for each axis. Their settings are predetermined by the type of servomotor connected. They are set to control the servomotors during positioning control.	Section 4.3
4	Home position return data	Home position return data is set for each axis. The return direction, return method, return speed, etc. are set for home position return.	Section 4.4
5	JOG operation	JOG operation data is set for each axis. The speed limit value and parameter block number are set for JOG operation.	Section 4.5
6	Parameter block	Up to 16 parameter blocks are set for acceleration, deceleration, speed control, etc. during positioning control. They are designated by the servo program, JOG operation data, and home position return data to easily change acceleration and deceleration (acceleration time, deceleration time, and speed limit value) during positioning control.	Section 4.6
7	Limit switch output data	Limit switch output data (ON/OFF pattern data) is set for each axis to be used when "USE" is set for the limit switch output setting in the fixed parameter. When positioning control takes place on an axis for which limit switch output data has been set, the set ON/OFF pattern of the axis is output to an external destination.	Section 7.1
8	Work coordinate data	Data used to set the work coordinate system. 6 different work coordinates can be set per axis. 1) G54 Work coordinate system 1 2) G55 Work coordinate system 2 3) G56 Work coordinate system 3 4) G57 Work coordinate system 4 5) G58 Work coordinate system 5 6) G59 Work coordinate system 6	Section 4.7

(2) Motion program
A motion program is designed to exercise positioning control and is requested to be started by the sequence program.
It comprises a motion program number, G code and positioning data.
For details, see Chapter 6.
- Motion program No. .... This number is designated in the sequence program.
- G code .... Indicates the type of positioning control.
- Positioning data .... Needed to execute the G code. Required data is predetermined for each G code.

(3) Sequence program
The sequence program serves to enable the execution of positioning control by motion programs, JOG operation, and manual pulse generator operation.
For details, see Chapter 5.

2. PERFORMANCE SPECIFICATIONS

2.1 SCPU Performance Specifications

Table 2.1.1 and 2.1.2 give the performance specifications of the SCPU.

Table 2.1.1 SCPU Performance Specifications (A172SHCPUN/A171SHCPUN)

<table>
<tr><th colspan="3">Item</th><th>A172SHCPUN</th><th>A171SHCPUN</th></tr>
<tr><td colspan="3">Control method</td><td colspan="2">Stored program repeated operation</td></tr>
<tr><td colspan="3">I/O control method</td><td colspan="2">Refresh method/direct method (selectable)</td></tr>
<tr><td colspan="3">Programming language</td><td colspan="2">Sequence control dedicated language
(Relay symbol language, logic symbol language, MELSAP II (SFC))</td></tr>
<tr><td colspan="2" rowspan="5">Number of instructions</td><td>Sequence instructions</td><td colspan="2">26</td></tr>
<tr><td>Basic instructions</td><td colspan="2">131</td></tr>
<tr><td>Applied instructions</td><td colspan="2">102</td></tr>
<tr><td>Special dedicated instructions</td><td colspan="2">12</td></tr>
<tr><td>Motion dedicated instructions</td><td colspan="2">6</td></tr>
<tr><td colspan="2" rowspan="2">Processing speed (μs)
(Sequence instruction)</td><td>Direct method</td><td colspan="2">0.25 to 1.9 μs/step</td></tr>
<tr><td>Refresh method</td><td colspan="2">0.25 μs/step</td></tr>
<tr><td colspan="3">Number of I/O points</td><td colspan="2">2048 (X/Y0 to X/Y7FF)</td></tr>
<tr><td colspan="3">Number of real I/O points</td><td>1024 (X/Y0 to X/Y3FF)</td><td>512 (X/Y0 to X/Y1FF)</td></tr>
<tr><td colspan="3">Watchdog timer (WDT)</td><td colspan="2">10 to 2000ms</td></tr>
<tr><td colspan="3">Memory size (internal RAM)</td><td>192 kbytes</td><td>64 kbytes</td></tr>
<tr><td colspan="2" rowspan="3">Program capacity</td><td>Main sequence program</td><td>Max. 30 k steps</td><td>Max. 14 k steps</td></tr>
<tr><td>Sub-sequence program</td><td>None</td><td>None</td></tr>
<tr><td>Micro computer program</td><td>Max. 58 kbytes</td><td>Max. 26 kbytes</td></tr>
<tr><td rowspan="20">Device</td><td colspan="2">No. of internal relays (M) (*1)</td><td>1000 (M0 to M999)</td><td rowspan="3">Total 2048 points common to M, L, S
(set with parameters)</td></tr>
<tr><td colspan="2">No. of latch relays (L)</td><td>1048 points (M1000 to M2047)</td></tr>
<tr><td colspan="2">No. of step relays (S)</td><td>0 point (none at initial status)</td></tr>
<tr><td colspan="2">No. of link relays (B)</td><td colspan="2">1024 points (B0 to B3FF)</td></tr>
<tr><td rowspan="3">Timers (T)</td><td>Points</td><td colspan="2">256 points</td></tr>
<tr><td>Specifications</td><td colspan="2"><table><tr><th></th><th>Time setting</th><th>Device</th></tr><tr><td>100 ms timer</td><td>0.1 to 3276.7s</td><td>T0 to T199</td></tr><tr><td>10 ms timer</td><td>0.01 to 327.67s</td><td>T200 to T255</td></tr><tr><td>100 ms elapsed time indicator</td><td>0.1 to 3276.7s</td><td>none at initial status</td></tr></table></td></tr>
<tr><td></td><td colspan="2">Set with parameters</td></tr>
<tr><td rowspan="3">Counters (C)</td><td>Points</td><td colspan="2">256 points</td></tr>
<tr><td>Specifications</td><td colspan="2"><table><tr><th></th><th>Setting range</th><th>Device</th></tr><tr><td>Normal counter</td><td>1 to 32767</td><td>C0 to C255</td></tr><tr><td>Interrupt program counter</td><td>1 to 32767</td><td>none at initial status</td></tr></table></td></tr>
<tr><td></td><td colspan="2">Set with parameters</td></tr>
<tr><td colspan="2">No. of data registers (D) (*1)</td><td colspan="2">1024 points (D0 to D1023)</td></tr>
<tr><td colspan="2">No. of link registers (W)</td><td colspan="2">1024 points (W0 to W3FF)</td></tr>
<tr><td colspan="2">No. of annunciators (F)</td><td colspan="2">256 points (F0 to F255)</td></tr>
<tr><td colspan="2">No. of file registers (R)</td><td colspan="2">Max. 8192 points (R0 to R8191) (set with parameters)</td></tr>
<tr><td colspan="2">No. of accumulators (A)</td><td colspan="2">2 points (A0, A1)</td></tr>
<tr><td colspan="2">No. of index registers (V, Z)</td><td colspan="2">2 points (V, Z)</td></tr>
<tr><td colspan="2">No. of pointers (P)</td><td colspan="2">256 points (P0 to P255)</td></tr>
<tr><td colspan="2">No. of interrupt pointers (I)</td><td colspan="2">32 points (I0 to I31)</td></tr>
<tr><td colspan="2">No. of special-function relays (M)</td><td colspan="2">256 points (M9000 to M9255)</td></tr>
</table>

Table 2.1.1 SCPU Performance Specifications (Continued)

Item	A172SHCPUN	A171SHCPUN
No. of special-function registers (D)	256 points (D9000 to D9255)	
No. of expansion file register block	Max. 10 blocks (set by memory capacity)	Max. 2 blocks (set by memory capacity)
No. of comments	Max. 4032 (64 kbytes), 1 point = 16 bytes (Set in 64-point unit)	
Number of expansion comments (*2)	Max. 3968 points (63 kbytes), 1 point = 16 bytes (Set in 64-point unit)	
Self-diagnostic function	Watchdog error monitoring, memory/CPU/input/output/battery, etc. error detection	
Operation mode on error	Select stop/continue	
Output mode selection when switching from STOP to RUN	Select re-output operation status before STOP (default) or output after operation execution.	
Clock function	Year, month, day, hour, minute, day of the week (leap year automatic distinction)	
Program/parameter storage in ROM	Not possible	

(*1) Range of positioning dedicated devices differs depending on the OS. For details, see Chapter 3.

(*2) The expansion comments are not stored in the internal memory of the CPU.

Table 2.1.2 SCPU Performance Specifications (A273UHCPU/A173UHCPU(S1))

<table>
<tr><th colspan="3">Item</th><th>A273UHCPU</th><th>A173UHCPU</th><th>A173UHCPU-S1</th></tr>
<tr><td colspan="3">Control method</td><td colspan="3">Stored program repeated operation</td></tr>
<tr><td colspan="3">I/O control method</td><td colspan="3">Refresh method (partial direct I/O enabled by instruction)</td></tr>
<tr><td colspan="3">Programming language</td><td colspan="3">Sequence control dedicated language
(Relay symbol language, logic symbol language, MELSAP II (SFC))</td></tr>
<tr><td colspan="2" rowspan="5">Number of instructions</td><td>Sequence instructions</td><td colspan="3">22</td></tr>
<tr><td>Basic instructions</td><td colspan="3">252</td></tr>
<tr><td>Applied instructions</td><td colspan="3">252</td></tr>
<tr><td>Special dedicated instructions</td><td colspan="3">204</td></tr>
<tr><td>Motion dedicated instructions</td><td colspan="3">4</td></tr>
<tr><td colspan="3">Processing speed (µs) (Sequence instruction)</td><td colspan="3">0.15 µs/step</td></tr>
<tr><td colspan="3">Number of I/O points</td><td colspan="3">8192 (X/Y0 to X/Y1FFF)</td></tr>
<tr><td colspan="3">Number of real I/O points</td><td>2048 (X/Y0 to X/Y7FF)</td><td colspan="2">2048 (X/Y0 to X/Y7FF)
(Within the range of 1 expansion base unit)</td></tr>
<tr><td colspan="3">Watchdog timer (WDT)</td><td colspan="3">200ms</td></tr>
<tr><td colspan="3">Memory size (internal RAM)</td><td>For loaded memory cassette capacity (Max. 1024kbytes)</td><td>256 kbytes</td><td>1024kbytes</td></tr>
<tr><td colspan="2" rowspan="2">Program capacity</td><td>Main sequence program</td><td colspan="3">Max. 30 k steps</td></tr>
<tr><td>Sub-sequence program</td><td colspan="3">Max. 30 k steps</td></tr>
<tr><td rowspan="18">Device</td><td colspan="2">No. of internal relays (M) (*1)</td><td>8191
(M0 to M999,
M2048 to M8191)</td><td colspan="2" rowspan="3">Total 8191 points common to
M, L, S
(set with parameters)</td></tr>
<tr><td colspan="2">No. of latch relays (L)</td><td>1048 points (M1000 to M2047)</td></tr>
<tr><td colspan="2">No. of step relays (S)</td><td>0 point (none at initial status)</td></tr>
<tr><td colspan="2">No. of link relays (B)</td><td colspan="3">8192 points (B0 to B1FFF)</td></tr>
<tr><td rowspan="2">Timers (T)</td><td>Points</td><td colspan="3">2048 points (Initial status: 256 points)</td></tr>
<tr><td>Specifications</td><td colspan="3"><table>
<tr><th></th><th>Time setting</th><th>Device</th></tr>
<tr><td>100 ms timer</td><td>0.1 to 3276.7s</td><td>T0 to T199</td></tr>
<tr><td>10 ms timer</td><td>0.01 to 327.67s</td><td>T200 to T255</td></tr>
<tr><td>100 ms elapsed time indicator</td><td>0.1 to 3276.7s</td><td>none at initial status</td></tr>
<tr><td>Extended timer</td><td>Time set by word device (D, W, R)</td><td>T256 to T2047</td></tr>
</table>Set with parameters</td></tr>
<tr><td rowspan="2">Counters (C)</td><td>Points</td><td colspan="3">1024 points (Initial status: 256 points)</td></tr>
<tr><td>Specifications</td><td colspan="3"><table>
<tr><th></th><th>Setting range</th><th>Device</th></tr>
<tr><td>Normal counter</td><td>1 to 32767</td><td>C0 to C255</td></tr>
<tr><td>Interrupt program counter</td><td>C244 to 255</td><td>none at initial status</td></tr>
<tr><td>Extended counter</td><td>Count value set by word device (D, W, R)</td><td>C256 to C1023</td></tr>
</table>Set with parameters</td></tr>
<tr><td colspan="2">No. of data registers (D) (*1)</td><td colspan="3">8192 points (D0 to D8191)</td></tr>
<tr><td colspan="2">No. of link registers (W)</td><td colspan="3">8192 points (W0 to W1FFF)</td></tr>
<tr><td colspan="2">No. of annunciators (F)</td><td colspan="3">2048 points (F0 to F2047)</td></tr>
<tr><td colspan="2">No. of file registers (R)</td><td colspan="3">Max. 8192 points (R0 to R8191) (set with parameters)</td></tr>
<tr><td colspan="2">No. of accumulators (A)</td><td colspan="3">2 points (A0, A1)</td></tr>
<tr><td colspan="2">No. of index registers (V, Z)</td><td colspan="3">14 points (V, V1 to V6, Z, Z1 to Z6)</td></tr>
<tr><td colspan="2">No. of pointers (P)</td><td colspan="3">256 points (P0 to P255)</td></tr>
<tr><td colspan="2">No. of interrupt pointers (I)</td><td colspan="3">32 points (I0 to I31)</td></tr>
<tr><td colspan="2">No. of special-function relays (M)</td><td colspan="3">256 points (M9000 to M9255)</td></tr>
</table>

Table 2.1.2 SCPU Performance Specifications (Continued)

Item	A273UHCPU	A173UHCPU	A173UHCPU-S1
No. of special-function registers (D)	256 points (D9000 to D9255)		
No. of expansion file register block	Max. 46 blocks (set by memory cassette or memory capacity)	Max. 2 blocks (set by memory capacity)	Max. 46 blocks (set by memory capacity)
No. of comments	Max. 4032 (64 kbytes), 1 point = 16 bytes (Set in 64-point unit)		
Number of expansion comments (*2)	Max. 3968 points (63 kbytes), 1 point = 16 bytes (Set in 64-point unit)		
Self-diagnostic function	Watchdog error monitoring, memory/CPU/input/output/battery, etc. error detection	Watchdog error monitoring (watchdog timer fixed to 200msec)	
Operation mode on error	Select stop/continue		
Output mode selection when switching from STOP to RUN	Select re-output operation status before STOP (default) or output after operation execution.		
Clock function (*3)	Year, month, day, hour, minute, day of the week (leap year automatic distinction)		
Program/parameter storage in ROM	Not possible		
RUN-time start method	Initial start		
Latch (power failure compensation) range	L1000 to L2047 (default) (latch ranges can be set for L, B, T, C, D and W)		
Remote RUN and PAUSE contacts	From among X0 to X1FFF, one point can each be set as the RUN and PAUSE contacts.		
I/O assignment	The number of I/O points occupied and module type can be registered.		
Step run	Sequence program operation can be executed and stopped.		
Interrupt processing	Interrupt or cyclic interrupt signal can be used to run interrupt program.		
Data link	MELSECNET/10, MELSECNET(II)		

(*1) Range of positioning dedicated devices differs depending on the OS. For details, see Chapter 3.

(*2) The expansion comments are not stored in the internal memory of the CPU.

(*3) The year data read by the clock element is only the lower two digits of the year.
When used in sequence control, the year data must be compensated for by the sequence program in some applications of using the data.

2.2 PCPU Performance Specifications

Table 2.2.1 and 2.2.2 give the performance specifications of the PCPU.

Table 2.2.1 PCPU Performance Specifications (A172SHCPUN/A171SHCPUN)

Item		A172SHCPUN	A171SHCPUN
Number of control axes		8 axes (simultaneous: 2 to 4 axes, independent: 8 axes)	4 axes (simultaneous: 2 to 4 axes, independent: 4 axes)
Interpolation functions		Linear interpolation (4 axes max.), circular interpolation (2 axes)	
Control modes		PTP(point to point), constant speed control, high-speed oscillation control	
Control units		mm · inch · degree	
Programming language		Dedicated instructions (NC language (EIA))	
Motion program	Capacity	59kbytes	
	Number of points for positioning	Approx. 2700 points/axis (These values vary depending on the programs. Positioning data can be designated indirectly.)	
Program setting method		Setting with an IBM PC, running the GSV43P software	
Number of simultaneously startable programs		8 programs	
Positioning	Method	PTP : Selection of absolute data method or incremental method Constant speed control : The absolute method and incremental method can be used together High-speed oscillation control : Absolute data method	
	Position commands	Commands can be selected for each axis. (see table below)	
	Speed command (command unit)	(see table below)	
Acceleration/ deceleration control	Automatic trapezoidal acceleration/ deceleration	(*1) (see table below)	
	S curve acceleration/ deceleration	S curve ratio setting: 0 to 100%	
Compensation	Backlash compensation	(0 to 65535) × position command unit (units converted to pulses: 0 to 65535 pulses)	
	Electronic gear	Compensation function for error in actual travel value with respect to command value	
Home position return function		When an absolute position system is not used : Selection of near-zero point dog type or count type When an absolute position system is used : Selection of data set type, near-zero point dog type or count type	
JOG operation function		Provided	

Position commands:

Control Unit	Command Unit	Address Setting Range	Travel Value Setting Range
mm	$\times 10^{-4}$ mm	−2147483648 to 2147483647	0 to ±2147483647
inch	$\times 10^{-5}$ inch		
degree	$\times 10^{-5}$ degree	0 to 35999999	

Speed command (command unit):

Control Unit	Speed Setting range
mm	0.01 to 6000000.00 (mm/min)
inch	0.001 to 600000.000 (inch/min)
degree	0.001 to 2147483.647 (degree/min)

Automatic trapezoidal acceleration/deceleration (*1):

Acceleration-fixed acceleration/deceleration	Time-fixed acceleration/deceleration
Acceleration time: 1 to 65535ms Deceleration time: 1 to 65535ms	Acceleration/deceleration time: 1 to 5000ms (Only constant speed control is possible.)

Table 2.2.1 PCPU Performance Specifications (Continued)

<table>
<tr><th colspan="2">Item</th><th>A172SHCPUN</th><th>A171SHCPUN</th></tr>
<tr><td colspan="2">Manual pulse generator operation function</td><td colspan="2">A maximum of one manual pulse generator can be connected.
A maximum of three manual pulse generators can be operated.
Setting of magnification: 1 to 10000. It is possible to set the smoothing magnification.</td></tr>
<tr><td colspan="2">M function</td><td colspan="2">M code output function provided
M code completion wait function provided</td></tr>
<tr><td colspan="2">Skip function</td><td colspan="2">Provided</td></tr>
<tr><td colspan="2" rowspan="2">Limit switch output function</td><td>Number of output points</td><td>8 point/axis</td></tr>
<tr><td>Number of ON/OFF setting points</td><td>10 points/axis</td></tr>
<tr><td colspan="2">Override ratio setting function</td><td colspan="2">Override ratio setting: 0 to 100%</td></tr>
<tr><td rowspan="2">High-speed reading of designated data</td><td>Number of input points</td><td colspan="2">Max. 9 points
(TREN input of A172SENC (1 point) + one motion slot PC input module (8 points))</td></tr>
<tr><td>Data latch timing</td><td colspan="2">At leading edge of the TREN input signal
Within 0.8ms of the signal leading edge for the PC input module</td></tr>
<tr><td colspan="2">Absolute position system</td><td colspan="2">Possible with a motor equipped with an absolute position detector.
(Possible to select the absolute data method or incremental method for each axis)</td></tr>
</table>

(*1) Acceleration-fixed acceleration/deceleration and time-fixed acceleration/deceleration are switched over as indicated below.

<table>
<tr><th>Acceleration-fixed acceleration/deceleration</th><th>Time-fixed acceleration/deceleration</th></tr>
<tr><td>During G100
G00 (without M code designation)
G28
G30
G53</td><td>During G100
G00 (with M code designation)
G01
G02
G03
G32</td></tr>
<tr><td>All move commands during G101</td><td>-</td></tr>
</table>

Table 2.2.2 PCPU Performance Specifications (A273UHCPU/A173UHCPU(S1))

<table>
<tr><th colspan="2">Item</th><th>A273UHCPU (32 axis feature)</th><th>A173UHCPU(S1)</th></tr>
<tr><td colspan="2">Number of control axes</td><td colspan="2">32 axes (simultaneous: 2 to 8 axes, independent: 32 axes)</td></tr>
<tr><td colspan="2">Interpolation functions</td><td colspan="2">Linear interpolation (4 axes max.), circular interpolation (2 axes)</td></tr>
<tr><td colspan="2">Control modes</td><td colspan="2">PTP(point to point), constant speed control, high-speed oscillation control</td></tr>
<tr><td colspan="2">Control units</td><td colspan="2">mm・inch・degree</td></tr>
<tr><td colspan="2">Programming language</td><td colspan="2">Dedicated instructions (NC language (EIA))</td></tr>
<tr><td rowspan="2">Motion program</td><td>Capacity</td><td colspan="2">126kbytes</td></tr>
<tr><td>Number of points for positioning</td><td colspan="2">Approx. 5400 points/axis
(These values vary depending on the programs. Positioning data can be designated indirectly.)</td></tr>
<tr><td colspan="2">Program setting method</td><td colspan="2">Setting with an IBM PC, running the GSV43P software</td></tr>
<tr><td colspan="2">Number of simultaneously startable programs</td><td colspan="2">8 programs</td></tr>
<tr><td rowspan="3">Positioning</td><td>Method</td><td colspan="2">PTP : Selection of absolute data method or incremental method
Constant speed control : The absolute method and incremental method can be used together
High-speed oscillation control : Absolute data method</td></tr>
<tr><td>Position commands</td><td colspan="2">Commands can be selected for each axis.
<table>
<tr><th>Control Unit</th><th>Command Unit</th><th>Address Setting Range</th><th>Travel Value Setting Range</th></tr>
<tr><td>mm</td><td>$\times 10^{-4}$ mm</td><td rowspan="2">–2147483648 to 2147483647</td><td rowspan="3">0 to ±2147483647</td></tr>
<tr><td>inch</td><td>$\times 10^{-5}$ inch</td></tr>
<tr><td>degree</td><td>$\times 10^{-5}$ degree</td><td>0 to 35999999</td></tr>
</table></td></tr>
<tr><td>Speed command (command unit)</td><td colspan="2">
<table>
<tr><th>Control Unit</th><th>Speed Setting range</th></tr>
<tr><td>mm</td><td>0.01 to 6000000.00 (mm/min)</td></tr>
<tr><td>inch</td><td>0.001 to 600000.000 (inch/min)</td></tr>
<tr><td>degree</td><td>0.001 to 2147483.647 (degree/min)</td></tr>
</table></td></tr>
<tr><td rowspan="2">Acceleration/ deceleration control</td><td>Automatic trapezoidal acceleration/ deceleration</td><td colspan="2">(*1)
<table>
<tr><th>Acceleration-fixed acceleration/deceleration</th><th>Time-fixed acceleration/deceleration</th></tr>
<tr><td>Acceleration time: 1 to 65535ms
Deceleration time: 1 to 65535ms</td><td>Acceleration/deceleration time: 1 to 5000ms
(Only constant speed control is possible.)</td></tr>
</table></td></tr>
<tr><td>S curve acceleration/ deceleration</td><td colspan="2">S curve ratio setting: 0 to 100%</td></tr>
<tr><td rowspan="2">Compensation</td><td>Backlash compensation</td><td colspan="2">(0 to 65535) × position command unit (units converted to pulses: 0 to 65535 pulses)</td></tr>
<tr><td>Electronic gear</td><td colspan="2">Compensation function for error in actual travel value with respect to command value</td></tr>
<tr><td colspan="2">Home position return function</td><td colspan="2">When an absolute position system is not used : Selection of near-zero point dog type or count type
When an absolute position system is used : Selection of data set type, near-zero point dog type or count type</td></tr>
<tr><td colspan="2">JOG operation function</td><td colspan="2">Provided</td></tr>
</table>

Table 2.2.2 PCPU Performance Specifications (Continued)

Item		A273UHCPU (32 axis feature)	A173UHCPU(S1)
Manual pulse generator operation function		Up to 3 manual pulse generators are connectable. Up to 3 axes can be operated simultaneously per manual pulse generator. Input magnification setting: 1 to 10000, with smoothing magnification setting	
M function		M code output function provided M code completion wait function provided	
Skip function		Provided	
Limit switch output function		Number of output points	8 point/axis
		Number of ON/OFF setting points	10 points/axis
Override ratio setting function		Override ratio setting: 0 to 100%	
High-speed reading of designated data	Number of input points	Max. 11 points (TREN input of A273EX (3 points) + one motion slot PC input module (8 points))	Max. 9 points (TREN input of A172SENC (1 point) + one motion slot PC input module (8 points))
	Data latch timing	At leading edge of the TREN input signal Within 0.8ms of the signal leading edge for the PC input module	
Absolute position system		Possible with a motor equipped with an absolute position detector. (Possible to select the absolute data method or incremental method for each axis)	

(*1) Acceleration-fixed acceleration/deceleration and time-fixed acceleration/deceleration are switched over as indicated below.

Acceleration-fixed acceleration/deceleration	Time-fixed acceleration/deceleration
During G100 G00 (without M code designation) G28 G30 G53	During G100 G00 (with M code designation) G01 G02 G03 G32
All move commands during G101	-

2.3 The Differences between A172SHCPUN/A171SHCPUN and A171S(S3) and the Differences between A273UHCPU (32 axis feature) and A173UHCPU(S1)

2.3.1 The differences between A172SHCPUN/A171SHCPUN and A171S(S3)

	Item		A172SHCPUN	A171SHCPUN	A171SCPU(S3)
Motion	Number of control axes		8 axes	4 axes	4 axes
Motion	Computing frequency		3.5ms/1 to 8 axes	3.5ms/1 to 4 axes	SV43: 3.5 ms/1 to 3 axes 7.1 ms/4 axes
PC	Sequencer CPU		Equivalent to reinforced I/O memory of A2SHCPU	Equivalent to A2SHCPU	Equivalent to A1SCPU
PC	Processing speed (µs) (Sequence instruction)	Direct method	0.25 to 1.9 µs/step		1.0 to 2.3 µs/step
PC		Refresh method	0.25 µs/step		1.0 µs/step
PC	No. of I/O		2048 points		–
PC	No. of actual I/O		1024 points	512 points	256 points
PC	Memory capacity (built-in RAM)		192 kbytes (Equivalent to A3NMCA24)	64 kbytes (Equivalent to A3NMCA8)	32 kbytes
PC	Program capacity (main sequence)		Max. 30 k step	Max. 14 k step	Max. 8 k step
PC	No. of file register (R)		Max. 8192 points		Max. 4096 points
PC	No. of expansion file register blocks (*1)		Max. 10 blocks	Max. 3 blocks	None
PC	MELSECNET/J		○ (Supported by special commands)		○ (By means of FROM/TO commands)
PC	Number of PC extension base unit		Max. three (*2)		Max. one
System configuration	Pulser synchronous encoder interface unit		A172SENC (Corresponding to external signal input 8-axes)		A171SENC (Corresponding to external signal input 4-axes)
System configuration	No. of SSCNET I/F		2CH. SSCNET1 .... For connection of servo amplifier SSCNET2 .... For personal computer link dedicated		A171S : 1CH. A171S-S3 : 2CH. (as given to the left)
System configuration	No. of available A271DVP		Unavailable		Max. two
Compatibility	Sequence program, parameter		After starting A172SH/A171SH and reading a file, those created by A171SCPU can be used as it is.		
Compatibility	Motion program				
Compatibility	Parameter				
Compatibility	System setting		By making sure of system setting screen after being started up by A172SH/A171SH and reading a file, changeover below is carried out: now the system is ready for operation. A171SCPU → A172SH/A171SHCPUN A171SENC → A172SENC		
Additional functions	• Support of high-resolution encoder (32768PLS/131072PLS)		○		×
Additional functions	• A torque limit value can be changed from a sequence program (CHGT instruction addition).		○		×
Additional functions	• Retracing during positioning		○		×

(*1) No. of expansion file register blocks varies depending on the setting of program capacity, No. of file registers, and No. of comments.

(*2) Up to one extension base for the MELSEC PC A2SHCPU-S1/A2SHCPU.

2.3.2 The differences between A273UHCPU and A173UHCPU(S1)

Item		A273UHCPU	A173UHCPU(S1)
Motion control	External input	A278LX, A273EX used	A172SENC used (up to 4 inputs usable)
	DOG/CHANGE signal	Near-zero point DOG signal and CHANGE signal are independent	Near-zero point DOG signal and CHANGE signal are shared
	Synchronous encoder	12 encoders usable	4 encoders usable
	Manual pulse generator	3 manual pulse generators usable: usable with one A273EX	3 manual pulse generators usable: one A172SENC needed per one manual pulse generator
	High-speed read (TREN input)	3 points	1 point
	External input clutch	12 points	4 points
	Usable servo amplifier	• MR-J2-□B/MR-H□B(N)/ MR-J□B • ADU (AC motor drive module)	• MR-J2-□B/MR-H□B(N)/ MR-J□B
	Motion extension base	Within 4 extension bases	None
	Cam data	256 lines of resolution × 256 pcs. (set by memory cassette)	A173UHCPU ...... 256 lines of resolution × 64 pcs. A173UHCPU-S1 ...... 256 lines of resolution × 256 pcs.
Sequence control	Key switch	2 key switches	1 key switch (equivalent to A172SHCPUN)
	LED indication	With segment indication	Without segment indication
	PC extension base	Within 7 extension bases	Within 1 extension base
Others	Peripheral software package	–	Usable from among A173UHCPU-compatible versions (Refer to section 1.3)

3. POSITIONING SIGNALS

The internal signals of the servo system CPU and the external signals sent to the servo system CPU are used as positioning signals.

(1) Internal signals
Of the devices available in the servo system CPU, the following four types are used for the internal signals of the servo system CPU.

- Internal relay (M) .... M1400 to M2047 (348 points)
  M2000 to M3839 (840 points)
  M4000 to M4719 (720 points)
- Special relay (SP.M) .... M9073 to M9079 (7 points)
  M9073 to M9079 (7 points)
- Data register (D) .... D500 to D1023 (524 points)
  D0 to D1689 (1690 points)
- Special register (SP.D) .... D9180 to D9199 (20 points)
  D1980 to D9199 (20 points)

(2) External signals
The external signals input to the servo system CPU are the upper and lower stroke end limit switch input signals, stop signals, near-zero point dog signal, speed/position switching signal, and manual pulse generator input signals.

- Upper and lower stroke end limit switch input signal .... Signals that control the upper limit and lower limit of the positioning range
- Stop signal .... Stop signal for speed control
- Near-zero point dog signal .... The ON/OFF signal from the near-zero point dog
- Speed/position switching signal .... Signal that switches control from speed to position control
- Manual pulse generator input .... Signal from the manual pulse generator

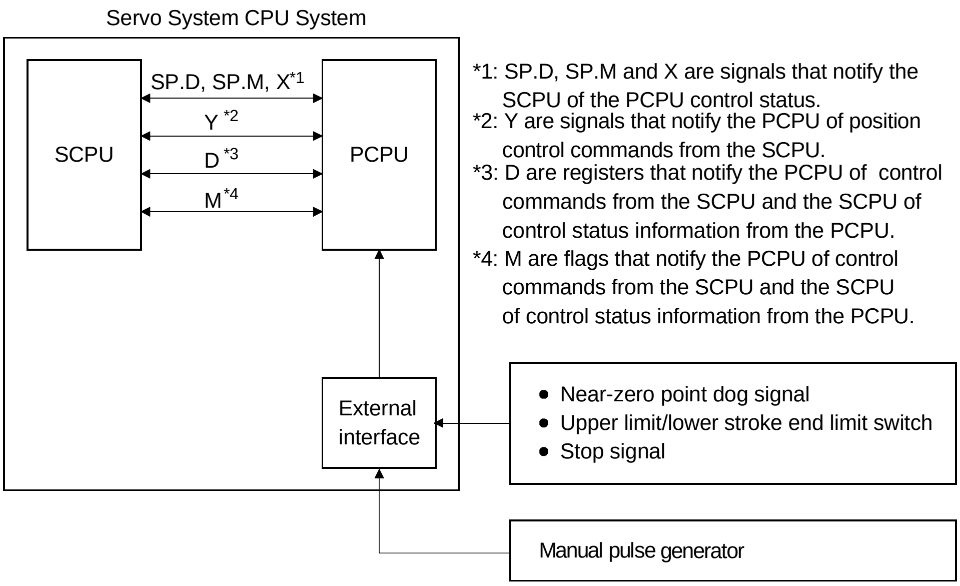


Fig.3.1 Flow of Positioning Signals

POINT
When the monitor data (machine values, actual present values, deviation counter, etc.) stored in the data registers (D) are used for magnitude comparison or four function arithmetic, they must be transferred to another device memory once and then processed. For transfer, refer to "Appendix-4.5".

The following section describes the positioning devices.
It indicates the device refresh cycles for signals with the positioning direction PCPU→SCPU and the device fetch cycles for those with the positioning direction SCPU→PCPU.

3.1 Internal Relays

(1) List of internal relays

A172SHCPUN

Device No.	Purpose
M0	User device (1400 points)
M1400	Axis status for SV43 (10 points × 8 axes)
M1480	Unusable (20 points)
M1500	Axis command signal for SV43 (10 points × 8 axes)
M1580	Unusable (20 points)
M1600	Axis status (20 points × 8 axes)
M1760	Unusable (40 points)
M1800	Axis command signal (20 points × 8 axes)
M1960	Common device (88 points)
M2000 M2047	

A171SHCPUN

Device No.	Purpose
M0	User device (1400 points)
M1400	Axis status for SV43 (10 points × 4 axes)
M1440	Unusable (60 points)
M1500	Axis command signal for SV43 (10 points × 4 axes)
M1540	Unusable (60 points)
M1600	Axis status (20 points × 4 axes)
M1680	Unusable (120 points)
M1800	Axis command signal (20 points × 4 axes)
M1880	Unusable (40 points)
M1960	Common device (88 points)
M2000 M2047	

A273UHCPU (32 axis feature)/
A173UHCPU(S1)

Device No.	Purpose
M0	User device (2000 points)
M2000	Common device (88 points)
M2320	Unusable (80 points)
M2400	Axis status (20 points × 32 axes)
M3040	Unusable (160 points)
M3200 M3839	Axis command signal (20 points × 32 axes)
M3840	User device (160 points)
M4000	Axis status for SV43 (10 points × 32 axes)
M4320	Unusable (80 points)
M4400	Axis command signal for SV43 (10 points × 32 axes)
M4720 M8191	User device (3472 points)

POINTS

- Total Number of User Device Points

A172SHCPUN	1400 points	A273UHCPU (32 axis feature)	5632 points
A171SHCPUN	1400 points	A173UHCPU(S1)	

(1) Internal relays for positioning control are not latched even inside the latch range.
In this manual, in order to indicate that internal relays for positioning control are not latched, the expression used in this text is "M1400 to M1999".

(2) Internal relays for positioning control are monitored from peripheral devices as shown below.
(a) When peripheral devices are started with GSV43P, positioning control internal relays within a latch range are indicated by L1400 to L1999.

(2) Axis status

• Axis status for SV43

Axis No.	A172SHCPUN Device Number	A171SHCPUN Device Number
1	M1400 to M1409	M1400 to M1409
2	M1410 to M1419	M1410 to M1419
3	M1420 to M1429	M1420 to M1429
4	M1430 to M1439	M1430 to M1439
5	M1440 to M1449	
6	M1450 to M1459	
7	M1460 to M1469	
8	M1470 to M1479	

Signal Name

	Signal Name	Fetch Cycle	Refresh Cycle	Signal Direction
0	Unusable		–	SCPU ← PCPU
1	Unusable			
2	Automatically operating		10ms	
3	Temporarily stopping			
4	Unusable		–	
5	Unusable			
6	Unusable			
7	Unusable			
8	Unusable			
9	Single block mode in progress (*1)		3.5ms	

(*1) The single block in progress is not an axis status. It is used with the first axis (M1409) only. The user cannot use it for other than the first axis.

• Axis status

Axis No.	A172SHCPUN Device Number	A171SHCPUN Device Number
1	M1600 to M1619	M1600 to M1619
2	M1620 to M1639	M1620 to M1639
3	M1640 to M1659	M1640 to M1659
4	M1660 to M1679	M1660 to M1679
5	M1680 to M1699	
6	M1700 to M1719	
7	M1720 to M1739	
8	M1740 to M1759	

Signal Name

	Signal Name	Fetch Cycle	Refresh Cycle	Signal Direction
0	Positioning start completed		3.5ms	SCPU ← PCPU
1	Positioning completed			
2	In-position			
3	Command in-position			
4	Unusable			
5	Unusable			
6	Zero pass			
7	Error detection		Immediately	
8	Servo error detection		3.5ms	
9	Home position return request		10ms	
10	Home position return completed		3.5ms	
11	External signal FLS		10ms	
12	External signal RLS			
13	External signal STOP			
14	External signal DOG/CHANGE			
15	Servo ON/OFF		3.5ms	
16	Torque control in progress			
17	(External signal DOG/CHANGE)		10ms	
18	Unusable		——	
19	M code output in progress		3.5ms	

• Axis status

Axis No.	A273UHCPU (32 axis feature) A173UHCPU (S1) Device No.
1	M2400 to M2419
2	M2420 to M2439
3	M2440 to M2459
4	M2460 to M2479
5	M2480 to M2499
6	M2500 to M2519
7	M2520 to M2539
8	M2540 to M2559
9	M2560 to M2579
10	M2580 to M2599
11	M2600 to M2619
12	M2620 to M2639
13	M2640 to M2659
14	M2660 to M2679
15	M2680 to M2699
16	M2700 to M2719
17	M2720 to M2739
18	M2740 to M2759
19	M2760 to M2779
20	M2780 to M2799
21	M2800 to M2819
22	M2820 to M2839
23	M2840 to M2859
24	M2860 to M2879
25	M2880 to M2899
26	M2900 to M2919
27	M2920 to M2939
28	M2940 to M2959
29	M2960 to M2979
30	M2980 to M2999
31	M3000 to M3019
32	M3020 to M3039

Single name

	Signal name	Refresh cycle			Fetch cycle			Signal direction
		Set number of axis			Set number of axis			
	SV43 A173UHCPU	1 to 12	13 to 24	25 to 32	1 to 12	13 to 24	25 to 32	
	SV43 A273UHCPU	1 to 8	9 to 18	19 to 32	1 to 8	9 to 18	19 to 32	
0	Positioning start completed	3.5ms	7.1ms	14.2ms				SCPU ← PCPU
1	Positioning completed							
2	In-position							
3	Command in-position							
4	Unusable							
5	Unusable							
6	Zero pass							
7	Error detection	Immediately						
8	Servo error detection	3.5ms	7.1ms	14.2ms				
9	Home position return request	10ms	20ms					
10	Home position return completed	3.5ms	7.1ms	14.2ms				
11	External signal FLS	10ms	20ms					
12	External signal RLS							
13	External signal STOP							
14	External signal DOG							
15	Servo ON/OFF	3.5ms	7.1ms	14.2ms				
16	Torque control in progress							
17	(External signal CHANGE)	10ms	20ms					
18	Unusable							
19	M code output in progress	3.5ms	7.1ms	14.2ms				

• Axis status for SV43

Axis No.	A273UHCPU (32 axis feature) A173UHCPU (S1) Device No.	Single name
1	M4000 to M4009	
2	M4010 to M4019	
3	M4020 to M4029	
4	M4030 to M4039	
5	M4040 to M4049	
6	M4050 to M4059	
7	M4060 to M4069	
8	M4070 to M4079	
9	M4080 to M4089	
10	M4090 to M4099	
11	M4100 to M4109	
12	M4110 to M4119	
13	M4120 to M4129	
14	M4130 to M4139	
15	M4140 to M4149	
16	M4150 to M4159	
17	M4160 to M4169	
18	M4170 to M4179	
19	M4180 to M4189	
20	M4190 to M4199	
21	M4200 to M4209	
22	M4210 to M4219	
23	M4220 to M4229	
24	M4230 to M4239	
25	M4240 to M4249	
26	M4250 to M4259	
27	M4260 to M4269	
28	M4270 to M4279	
29	M4280 to M4289	
30	M4290 to M4299	
31	M4300 to M4309	
32	M4310 to M4319	

	Signal name		Refresh cycle			Fetch cycle			Signal direction
			Set number of axis			Set number of axis			
	SV43	A173 UHCPU	1 to 12	13 to 24	25 to 32	1 to 12	13 to 24	25 to 32	
	SV43	A273 UHCPU	1 to 8	9 to 18	19 to 32	1 to 8	9 to 18	19 to 32	
0	Unusable		–						SCPU ← PCPU
1	Unusable								
2	Automatically operating		10ms	20ms					
3	Temporarily stopping								
4	Unusable		–						
5	Unusable								
6	Unusable								
7	Unusable								
8	Unusable								
9	Single block mode in progress (*1)		3.5ms	7.1ms	14.2ms				

(*1) The single block in progress is not an axis status. It is used with the first axis (M4009) only. The user cannot use it for other than the first axis.

(3) Axis command signals

• Axis command signals for SV43

Axis No.	A172SHCPUN Device Number	A171SHCPUN Device Number	Signal Name
1	M1500 to M1509	M1500 to M1509	
2	M1510 to M1519	M1510 to M1519	
3	M1520 to M1529	M1520 to M1529	
4	M1530 to M1539	M1530 to M1539	
5	M1540 to M1549		
6	M1550 to M1559		
7	M1560 to M1569		
8	M1570 to M1579		

	Signal Name	Fetch Cycle	Refresh Cycle	Signal Direction
0	Temporary stop command	3.5ms		SCPU → PCPU
1	Optional program stop	At start		
2	Optional block skip			
3	Single block			
4	Restart	3.5ms		
5	Override valid/invalid			
6	Unusable	–		
7	Unusable			
8	Single block mode (*1)			
9	Single block start (*1)			

(*1) The single block mode and single block start are not axis statuses. They are used with the first axis (M1508, M1509) only. The user cannot use them for other than the first axis.

• Axis command signals

Axis No.	A172SHCPUN Device Number	A171SHCPUN Device Number	Signal Name
1	M1800 to M1819	M1800 to M1819	
2	M1820 to M1839	M1820 to M1839	
3	M1840 to M1859	M1840 to M1859	
4	M1860 to M1879	M1860 to M1879	
5	M1880 to M1899		
6	M1900 to M1919		
7	M1920 to M1939		
8	M1940 to M1959		

	Signal Name	Fetch Cycle	Refresh Cycle	Signal Direction
0	Stop command	3.5ms		SCPU → PCPU
1	Rapid stop command			
2	Forward rotation JOG command	10ms		
3	Reverse rotation JOG command			
4	Completion signal OFF command			
5	Unusable	–		
6	Limit switch output enable	3.5ms		
7	Error reset	10ms		
8	Servo error reset			
9	Start-time stop input invalid	At start		
10	Unusable	–		
11	Unusable			
12	Unusable			
13	Unusable			
14	Unusable			
15	Servo OFF	3.5ms		
16	Unusable	–		
17	Unusable			
18	Unusable			
19	FIN signal	3.5ms		

• Axis command signals

Axis No.	A273UHCPU (32 axis feature) A173UHCPU (S1) Device No.	Single name
1	M3200 to M3219	
2	M3220 to M3239	
3	M3240 to M3259	
4	M3260 to M3279	
5	M3280 to M3299	
6	M3300 to M3319	
7	M3320 to M3339	
8	M3340 to M3359	
9	M3360 to M3379	
10	M3380 to M3399	
11	M3400 to M3419	
12	M3420 to M3439	
13	M3440 to M3459	
14	M3460 to M3479	
15	M3480 to M3499	
16	M3500 to M3519	
17	M3520 to M3539	
18	M3540 to M3559	
19	M3560 to M3579	
20	M3580 to M3599	
21	M3600 to M3619	
22	M3620 to M3639	
23	M3640 to M3659	
24	M3660 to M3679	
25	M3680 to M3699	
26	M3700 to M3719	
27	M3720 to M3739	
28	M3740 to M3759	
29	M3760 to M3779	
30	M3780 to M3799	
31	M3800 to M3819	
32	M3820 to M3839	

	Signal name	Refresh cycle Set number of axis			Fetch cycle Set number of axis			Signal direction
	SV43 A173UHCPU	1 to 12	13 to 24	25 to 32	1 to 12	13 to 24	25 to 32	
	SV43 A273UHCPU	1 to 8	9 to 18	19 to 32	1 to 8	9 to 18	19 to 32	
0	Stop command				3.5ms	7.1ms	14.2ms	SCPU → PCPU
1	Rapid stop command							
2	Forward rotation JOG command				10ms	20ms		
3	Reverse rotation JOG command							
4	Completion signal OFF command							
5	Unusable				3.5ms	7.1ms	14.2ms	
6	Limit switch output enable							
7	Error reset				10ms	20ms		
8	Servo error reset							
9	Start-time stop input invalid				At start			
10	Unusable				–			
11	Unusable							
12	Present feed value update request command				At start			
13	Unusable				–			
14	Unusable							
15	Servo OFF				3.5ms	7.1ms	14.2ms	
16	Unusable				–			
17	Unusable							
18	Unusable							
19	FIN signal				3.5ms	7.1ms	14.2ms	

• Axis command signals for SV43

Axis No.	A273UHCPU (32 axis feature) A173UHCPU (S1) Device No.
1	M4400 to M4409
2	M4410 to M4419
3	M4420 to M4429
4	M4430 to M4439
5	M4440 to M4449
6	M4450 to M4459
7	M4460 to M4469
8	M4470 to M4479
9	M4480 to M4489
10	M4490 to M4499
11	M4500 to M4509
12	M4510 to M4519
13	M4520 to M4529
14	M4530 to M4539
15	M4540 to M4549
16	M4550 to M4559
17	M4560 to M4569
18	M4570 to M4579
19	M4580 to M4589
20	M4590 to M4599
21	M4600 to M4609
22	M4610 to M4619
23	M4620 to M4629
24	M4630 to M4639
25	M4640 to M4649
26	M4650 to M4659
27	M4660 to M4669
28	M4670 to M4679
29	M4680 to M4689
30	M4690 to M4699
31	M4700 to M4709
32	M4710 to M4719

Single name

	Signal name	Refresh cycle			Fetch cycle			Signal direction
		Set number of axis			Set number of axis			
	SV43 A173UHCPU	1 to 12	13 to 24	25 to 32	1 to 12	13 to 24	25 to 32	
	SV43 A273UHCPU	1 to 8	9 to 18	19 to 32	1 to 8	9 to 18	19 to 32	
0	Temporary stop command	3.5ms	7.1ms	14.2ms				SCPU → PCPU
1	Optional program stop	At start						
2	Optional block skip							
3	Single block							
4	Restart	3.5ms	7.1ms	14.2ms				
5	Override valid/invalid							
6	Unusable	–						
7	Unusable							
8	Single block mode (*1)							
9	Single block start (*1)							

(*1) The single block mode and single block start are not axis statuses. They are used with the first axis (M4408, M4409) only. The user cannot use them for other than the first axis.

(4) Common devices

A172SHCPUN

Device Number	Signal Name		Fetch Cycle	Refresh Cycle	Signal Direction
M1960	Unusable (40 points)		–	–	–
M1961					
M1962					
M1963					
M1964					
M1965					
M1966					
M1967					
M1968					
M1969					
M1970					
M1971					
M1972					
M1973					
M1974					
M1975					
M1976					
M1977					
M1978					
M1979					
M1980					
M1981					
M1982					
M1983					
M1984					
M1985					
M1986					
M1987					
M1988					
M1989					
M1990					
M1991					
M1992					
M1993					
M1994					
M1995					
M1996					
M1997					
M1998					
M1999					
M2000	PC READY flag		10ms		SCPU→PCPU
M2001	Axis 1	START accept flag (8 points)		10ms	SCPU←PCPU
M2002	Axis 2				
M2003	Axis 3				
M2004	Axis 4				
M2005	Axis 5				
M2006	Axis 6				
M2007	Axis 7				
M2008	Axis 8				
M2009	All-axes servo ON accept flag				
M2010	Unusable (2 points)		–	–	–
M2011					
M2012	Manual pulse generator enable flag		10ms		SCPU→PCPU
M2013	Unusable (2 points)		–	–	–
M2014					
M2015	JOG simultaneous start command		10ms		SCPU→PCPU
M2016	Unusable (4 points)		–	–	–
M2017					
M2018					
M2019					
M2020	Start buffer full			END	SCPU←PCPU
M2021	Axis 1	Speed change flag (8 points)			
M2022	Axis 2				
M2023	Axis 3				
M2024	Axis 4				
M2025	Axis 5				
M2026	Axis 6				
M2027	Axis 7				
M2028	Axis 8				
M2029	Unusable (5 points)		–	–	–
M2030					
M2031					
M2032					
M2033					
M2034	PC link communication error flag			END	SCPU←PCPU
M2035	Unusable (6 points)		–	–	–
M2036					
M2037					
M2038					
M2039					
M2040					
M2041	System setting error flag			END	SCPU←PCPU
M2042	All-axes servo ON command		3.5ms		SCPU→PCPU
M2043	Unusable (4 points)		–	–	–
M2044					
M2045					
M2046					
M2047	Motion slot module error detection flag			END	SCPU←PCPU

A172SHCPUN

Device Number	Signal Name		Fetch Cycle	Refresh Cycle	Signal Direction
M1960	Unusable (40 points)		–	–	–
M1961					
M1962					
M1963					
M1964					
M1965					
M1966					
M1967					
M1968					
M1969					
M1970					
M1971					
M1972					
M1973					
M1974					
M1975					
M1976					
M1977					
M1978					
M1979					
M1980					
M1981					
M1982					
M1983					
M1984					
M1985					
M1986					
M1987					
M1988					
M1989					
M1990					
M1991					
M1992					
M1993					
M1994					
M1995					
M1996					
M1997					
M1998					
M1999					
M2000	PC READY flag		10ms		SCPU→PCPU
M2001	Axis 1	START accept flag (4 points)		10ms	SCPU←PCPU
M2002	Axis 2				
M2003	Axis 3				
M2004	Axis 4				
M2005	Unusable (4 points)		–	–	–
M2006					
M2007					
M2008					
M2009	All-axes servo ON accept flag			10ms	SCPU←PCPU
M2010	Unusable (2 points)		–	–	–
M2011					
M2012	Manual pulse generator enable flag		10ms		SCPU→PCPU
M2013	Unusable (2 points)		–	–	–
M2014					
M2015	JOG simultaneous start command		10ms		SCPU→PCPU
M2016	Unusable (4 points)		–	–	–
M2017					
M2018					
M2019					
M2020	Start buffer full			END	SCPU←PCPU
M2021	Axis 1	Speed change flag (4 points)			
M2022	Axis 2				
M2023	Axis 3				
M2024	Axis 4				
M2025	Unusable (9 points)		–	–	–
M2026					
M2027					
M2028					
M2029					
M2030					
M2031					
M2032					
M2033					
M2034	PC link communication error flag			END	SCPU←PCPU
M2035	Unusable (6 points)		–	–	–
M2036					
M2037					
M2038					
M2039					
M2040					
M2041	System setting error flag			END	SCPU←PCPU
M2042	All-axes servo ON command		3.5ms		SCPU→PCPU
M2043	Unusable (4 points)		–	–	–
M2044					
M2045					
M2046					
M2047	Motion slot module error detection flag			END	SCPU←PCPU

* The entry "END" in the Refresh Cycle column indicates 80ms or a longer sequence program scan time.

Device No.	Signal name (SV43)		Refresh cycle, Set No. of axis: A173UHCPU 1 to 12 / A273UHCPU 1 to 8	A173UHCPU 13 to 24 / A273UHCPU 9 to 18	A173UHCPU 25 to32 / A273UHCPU 19 to 32	Fetch cycle, Set No. of axis: A173UHCPU 1 to 12 / A273UHCPU 1 to 8	A173UHCPU 13 to 24 / A273UHCPU 9 to 18	A173UHCPU 25 to32 / A273UHCPU 19 to 32	Signal direction
M2000	PLC READY flag					10ms	20ms		SCPU → PCPU
M2001	Axis1								
M2002	Axis2								
M2003	Axis3								
M2004	Axis4								
M2005	Axis5								
M2006	Axis6								
M2007	Axis7								
M2008	Axis8								
M2009	Axis9								
M2010	Axis10								
M2011	Axis11								
M2012	Axis12								
M2013	Axis13								
M2014	Axis14								
M2015	Axis15								
M2016	Axis16	Start accept flag	10ms						SCPU ← PCPU
M2017	Axis17								
M2018	Axis18								
M2019	Axis19								
M2020	Axis20								
M2021	Axis21								
M2022	Axis22								
M2023	Axis23								
M2024	Axis24								
M2025	Axis25								
M2026	Axis26								
M2027	Axis27								
M2028	Axis28								
M2029	Axis29								
M2030	Axis30								
M2031	Axis31								
M2032	Axis32								
M2033	Unusable		–			–			–
M2034	PC link communication error flag		10ms						SCPU ← PCPU
M2035									
M2036									
M2037	Unusable (6 points)		–			–			–
M2038									
M2039									
M2040									
M2041	System setting error flag		10ms						SCPU ← PCPU
M2042	All axes servo ON command					3.5ms	7.1ms	14.2ms	SCPU → PCPU
M2043									
M2044	Unusable (4 points)		–			–			–
M2045									
M2046									
M2047	Motion slot module error detection flag		10ms						SCPU ← PCPU
M2048	JOG simultaneous start command					10ms	20ms		SCPU → PCPU
M2049	All axes servo ON accept flag		END						SCPU ← PCPU
M2050	Start buffer full								
M2051	Manual pulse generator 1 enable flag								
M2052	Manual pulse generator 2 enable flag					10ms	20ms		SCPU → PCPU
M2053	Manual pulse generator 3 enable flag								
M2054									
M2055									
M2056									
M2057	Unusable (7 points)		–			–			–
M2058									
M2059									
M2060									
M2061	Axis1								
M2062	Axis2								
M2063	Axis3								
M2064	Axis4								
M2065	Axis5								
M2066	Axis6								
M2067	Axis7								
M2068	Axis8								
M2069	Axis9								
M2070	Axis10	Speed change flag	END						SCPU ← PCPU
M2071	Axis11								
M2072	Axis12								
M2073	Axis13								
M2074	Axis14								
M2075	Axis15								
M2076	Axis16								
M2077	Axis17								
M2078	Axis18								
M2079	Axis19								
M2080	Axis20								
M2081	Axis21								
M2082	Axis22								
M2083	Axis23								
M2084	Axis24								
M2085	Axis25								
M2086	Axis26	Speed change flag	END						SCPU ← PCPU
M2087	Axis27								
M2088	Axis28								
M2089	Axis29								
M2090	Axis30								
M2091	Axis31								
M2092	Axis32								
M2093									
M2094									
M2095									
M2096									
M2097									
M2098									
M2099									
M2100									
M2101									
M2102									
M2103									
M2104									
M2105									
M2106									
M2107									
M2108									
M2109									
M2110	Unusable (35 points)		–			–			–
M2111									
M2112									
M2113									
M2114									
M2115									
M2116									
M2117									
M2118									
M2119									
M2120									
M2121									
M2122									
M2123									
M2124									
M2125									
M2126									
M2127									
M2128	Axis1								
M2129	Axis2								
M2130	Axis3								
M2131	Axis4								
M2132	Axis5								
M2133	Axis6								
M2134	Axis7								
M2135	Axis8								
M2136	Axis9								
M2137	Axis10								
M2138	Axis11								
M2139	Axis12								
M2140	Axis13								
M2141	Axis14								
M2142	Axis15								
M2143	Axis16	Automatically decelerating flag	3.5ms	7.1ms	14.2ms				SCPU ← PCPU
M2144	Axis17								
M2145	Axis18								
M2146	Axis19								
M2147	Axis20								
M2148	Axis21								
M2149	Axis22								
M2150	Axis23								
M2151	Axis24								
M2152	Axis25								
M2153	Axis26								
M2154	Axis27								
M2155	Axis28								
M2156	Axis29								
M2157	Axis30								
M2158	Axis31								
M2159	Axis32								

* The entry "END" in the Refresh Cycle column indicates 50ms or a longer sequence program scan time.

Device No.	Signal name	Refresh cycle			Fetch cycle			Signal direction
		Set No. of axis			Set No. of axis			
	SV43 A173UHCPU	1 to 12	13 to 24	25 to32	1 to 12	13 to 24	25 to32	
	SV43 A273UHCPU	1 to 8	9 to 18	19 to 32	1 to 8	9 to 18	19 to 32	
M2160								
M2161								
M2162								
M2163								
M2164								
M2165								
M2166								
M2167								
M2168								
M2169								
M2170								
M2171								
M2172								
M2173								
M2174								
M2175								
M2176								
M2177								
M2178								
M2179								
M2180								
M2181								
M2182								
M2183								
M2184								
M2185								
M2186								
M2187								
M2188								
M2189								
M2190								
M2191								
M2192								
M2193								
M2194								
M2195								
M2196								
M2197								
M2198								
M2199	Unusable	–			–			–
M2200	(80 points)							
M2201								
M2202								
M2203								
M2204								
M2205								
M2206								
M2207								
M2208								
M2209								
M2210								
M2211								
M2212								
M2213								
M2214								
M2215								
M2216								
M2217								
M2218								
M2219								
M2220								
M2221								
M2222								
M2223								
M2224								
M2225								
M2226								
M2227								
M2228								
M2229								
M2230								
M2231								
M2232								
M2233								
M2234								
M2235								
M2236								
M2237								
M2238								
M2239								

Device No.		Signal name	Refresh cycle			Fetch cycle			Signal direction
			Set No. of axis			Set No. of axis			
		SV43 A173UHCPU	1 to 12	13 to 24	25 to32	1 to 12	13 to 24	25 to32	
		SV43 A273UHCPU	1 to 8	9 to 18	19 to 32	1 to 8	9 to 18	19 to 32	
M2240	Axis1								
M2241	Axis2								
M2242	Axis3								
M2243	Axis4								
M2244	Axis5								
M2245	Axis6								
M2246	Axis7								
M2247	Axis8								
M2248	Axis9								
M2249	Axis10								
M2250	Axis11								
M2251	Axis12								
M2252	Axis13								
M2253	Axis14								
M2254	Axis15								
M2255	Axis16	Speed change accepting	3.5ms	7.1ms	14.2ms				SCPU ← PCPU
M2256	Axis17	flag "0"							
M2257	Axis18								
M2258	Axis19								
M2259	Axis20								
M2260	Axis21								
M2261	Axis22								
M2262	Axis23								
M2263	Axis24								
M2264	Axis25								
M2265	Axis26								
M2266	Axis27								
M2267	Axis28								
M2268	Axis29								
M2269	Axis30								
M2270	Axis31								
M2271	Axis32								
M2272									
M2273									
M2274									
M2275									
M2276									
M2277									
M2278									
M2279									
M2280									
M2281									
M2282									
M2283									
M2284									
M2285									
M2286									
M2287									
M2288									
M2289									
M2290									
M2291									
M2292									
M2293									
M2294									
M2295	Unusable		–			–			–
M2296	(48 points)								
M2297									
M2298									
M2299									
M2300									
M2301									
M2302									
M2303									
M2304									
M2305									
M2306									
M2307									
M2308									
M2309									
M2310									
M2311									
M2312									
M2313									
M2314									
M2315									
M2316									
M2317									
M2318									
M2319									

* The entry "END" in the Refresh Cycle column indicates 50ms or a longer sequence program scan time.

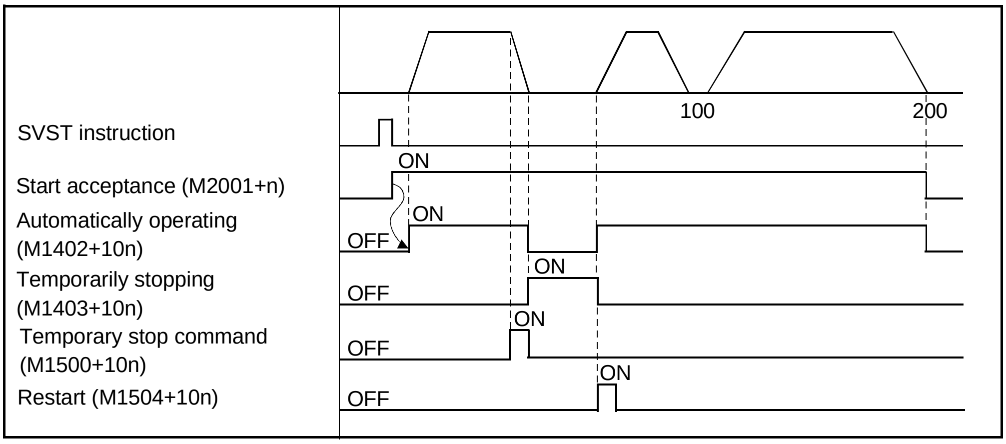
3.1.1 Axis status

(1) Automatically operating signal (M1402+10n/M4002+10n)
When the axis used is specified in the SVST instruction, this signal is ON while the block of the specified motion program is being executed. It turns OFF when:
- M02/M30 is executed;
- Temporary stop command turns ON (M1500+10n/M4400+10n);
- External STOP signal turns ON;
- Error reset is made;
- Emergency stop is made;
- Single block execution is ended by M0, M01 or single block;

or
- Stop or rapid stop command turns ON.

[Motion program example]

0001;	Program No.
G90 G00 X100.;	Absolute value command PTP positioning (X100.)
X200.;	PTP positioning (X200.)
M02;	Reset
%	



(2) Temporarily stopping signal (M1403+10n/M4003+10n)

(a) This signal turns ON if the temporary stop command is given when the automatically operating signal (M1402+10n/M4002+10n) is ON.
When the restart signal (M1504+10n/M4404+10n) is turned ON during a temporary stop, automatic operation is resumed from the block where it had stopped.
There is the following temporary stop command.
- Temporary stop command (M1500+10n/M4400+10n)

(b) The temporarily stopping signal turns OFF when:
- Restart signal (M1504+10n/M4404+10n) is turned ON;
- Error reset (M1807+20n/M3207+20n) is turned ON;
- Servo error reset (M1808+20n/M3208+20n) is turned ON;
- Error occurs;

or
- Emergency stop is made.

[Motion program example]

```
0001;              Program No.
G90 G00 X100.;     Absolute value command PTP positioning (X100.)
X200.;             PTP positioning (X200.)
M02;               Reset
%
```

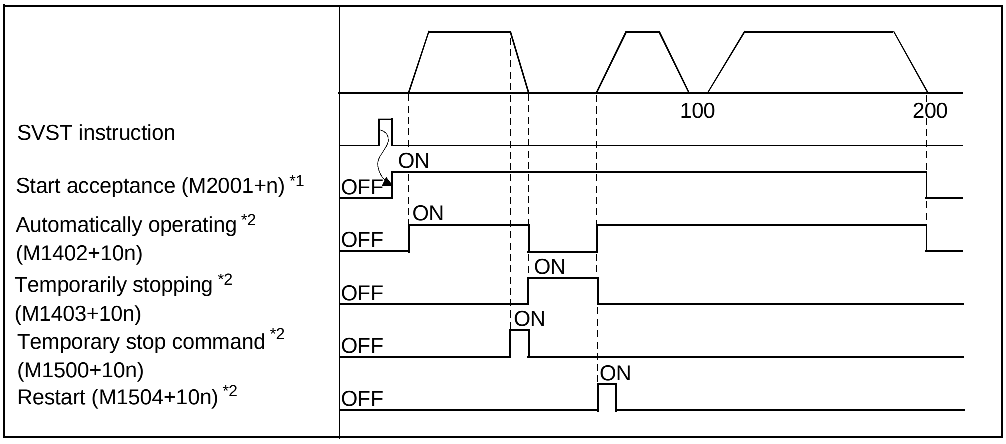


Fig. 3.2 Temporarily Stopping Signal ON/OFF Timing

REMARKS

*1: n in M2001+n indicates the value corresponding to the axis number.

*2: n indicates the value corresponding to the axis number as listed below.

<A172SHCPUN>

Axis No.	n
1	0
2	1
3	2
4	3
5	4
6	5
7	6
8	7

<A171SHCPUN>

Axis No.	n
1	0
2	1
3	2
4	3

<A273UHCPU (32 axis feature) / A173UHCPU>

Axis No.	n	Axis No.	n	Axis No.	n	Axis No.	n
1	0	9	8	17	16	25	24
2	1	10	9	18	17	26	25
3	2	11	10	19	18	27	26
4	3	12	11	20	19	28	27
5	4	13	12	21	20	29	28
6	5	14	13	22	21	30	29
7	6	15	14	23	22	31	30
8	7	16	15	24	23	32	31

(3) Single block in progress signal (M1409/M4009)

(a) The single block is available in two modes: a mode where a single block is specified before a program start; and a mode where a single block is executed at any point during program execution.
The single block in progress signal indicates that a single block can be executed in the mode where a single block is executed at any point during program execution.

(b) A single block is executed when the single block in progress signal is ON. When the single block in progress signal is OFF, make an SVST start or turn single block start from OFF to ON to perform continuous operation.

(c) The single block in progress signal turns ON when:
• The single block mode signal (M1508/M4408) is turned ON.

(d) The single block in progress signal turns OFF when:
• The single block start signal (M1509/M4409) is turned from OFF to ON after the single block mode signal (M1508/M4408) is turned OFF.

[Motion program example]

```
0001;                          Program No.
N1 G90 G01 X100. F1000.;       Absolute value command PTP positioning (X100.)
N2 X200.;                      CP positioning (X200.)
N3 X300.;                      CP positioning (X300.)
N4 X400.;                      CP positioning (X400.)
M02;                           Reset
%
```

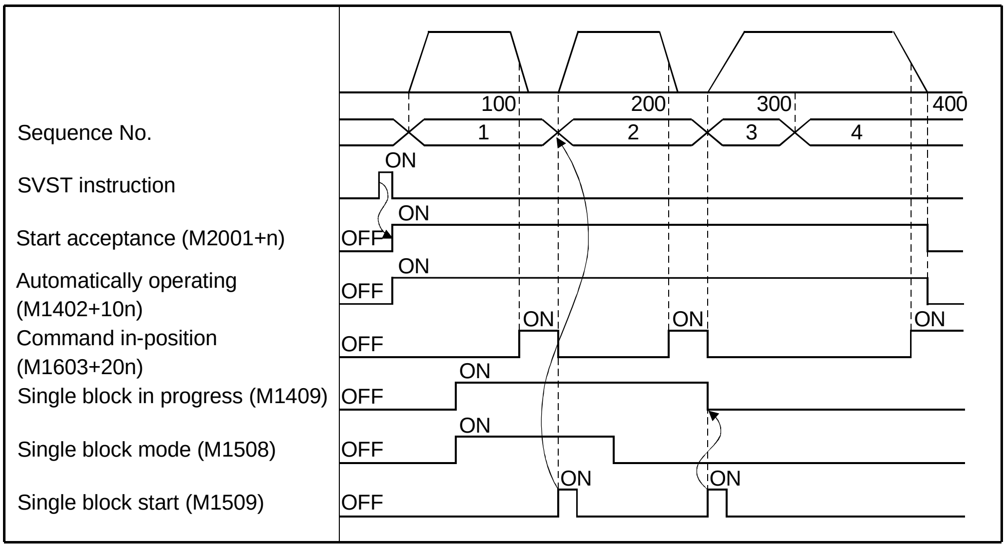


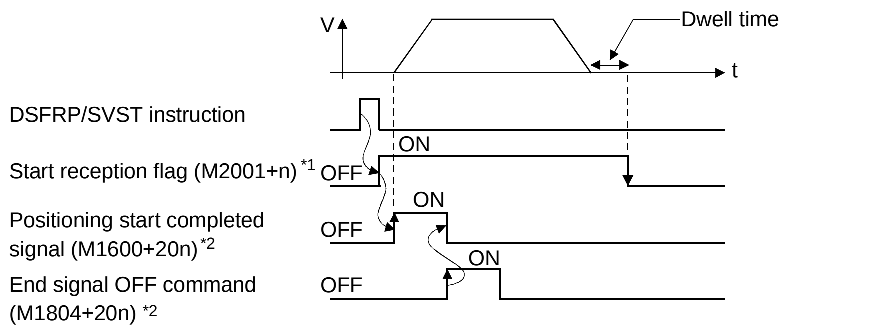
Fig. 3.3 Single Block Signal Timings

(4) Positioning start completed signal (M1600+20n/M2400+20n)

(a) This signal comes ON when starting of positioning control of the axis designated by the DSFRP/SVST instruction in the sequence program is completed.
It does not come ON when positioning control starts due to a home position return, JOG operation or manual pulse generator operation.

(b) The positioning start completed signal goes OFF at the leading edge (OFF→ON) of the end signal OFF command (M1804+20n) or when positioning is completed.

At the leading edge (OFF → ON) of the end signal OFF command (M1804 + 20n)



When positioning is completed

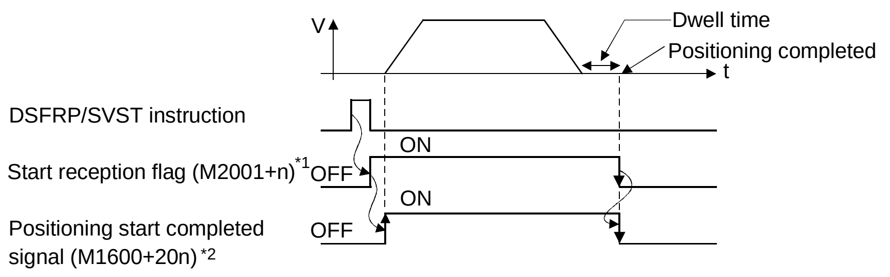


Fig. 3.4 Positioning Start Completed Signal ON/OFF Timing

REMARKS

*1: n in M2001+n indicates the value corresponding to the axis number.
*2: n indicates the value corresponding to the axis number as listed below.

<A172SHCPUN>

Axis No.	n
1	0
2	1
3	2
4	3
5	4
6	5
7	6
8	7

<A171SHCPUN>

Axis No.	n
1	0
2	1
3	2
4	3

<A273UHCPU (32 axis feature) / A173UHCPU>

Axis No.	n	Axis No.	n	Axis No.	n	Axis No.	n
1	0	9	8	17	16	25	24
2	1	10	9	18	17	26	25
3	2	11	10	19	18	27	26
4	3	12	11	20	19	28	27
5	4	13	12	21	20	29	28
6	5	14	13	22	21	30	29
7	6	15	14	23	22	31	30
8	7	16	15	24	23	32	31

(5) Positioning completed signal (M1601+20n/M2401+20n)

(a) This signal comes ON when positioning control of the axis designated by the DSFRP/SVST instruction in the sequence program is completed.
It does not come ON when positioning control is started, or stopped part way through, due to a home position return, JOG operation, manual pulse generator operation, or speed control.
It does not come ON when positioning is stopped part way through.

(b) The positioning completed signal goes OFF at the leading edge (OFF→ON) of the end signal OFF command (M1804+20n), or when a positioning control start is completed.

[Motion program example]

```
0001;                        Program No.
N1 G90 G01 X100. F1000.;     Absolute value command PTP positioning (X100.)
X200.;                       PTP positioning (X200.)
G00 X300. G04 P500;          PTP positioning (X300.), dwell (500ms)
M02;                         Reset
%
```

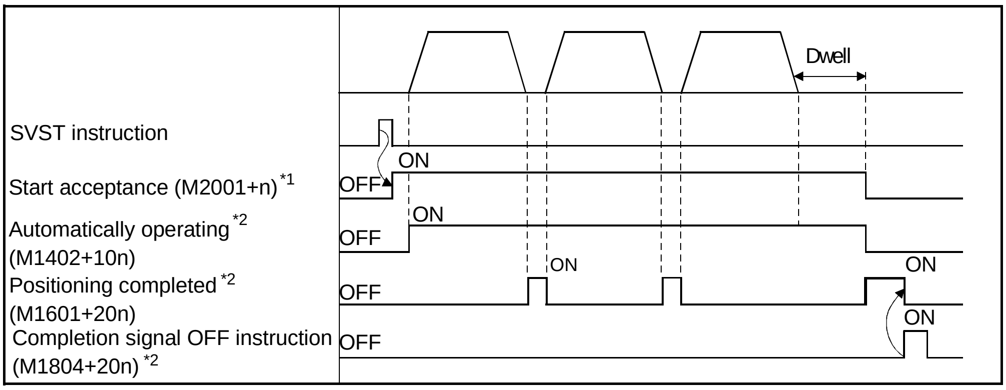


Fig. 3.5 Positioning Completed Signal ON/OFF Timing

REMARKS

*1: n in M2001+n indicates the value corresponding to the axis number.
*2: n indicates the value corresponding to the axis number as listed below.

<A172SHCPUN>

Axis No.	n
1	0
2	1
3	2
4	3
5	4
6	5
7	6
8	7

<A171SHCPUN>

Axis No.	n
1	0
2	1
3	2
4	3

<A273UHCPU (32 axis feature) / A173UHCPU>

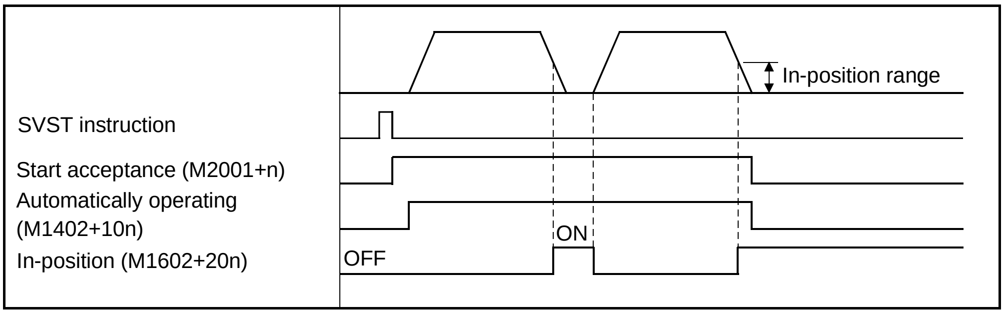
Axis No.	n	Axis No.	n	Axis No.	n	Axis No.	n
1	0	9	8	17	16	25	24
2	1	10	9	18	17	26	25
3	2	11	10	19	18	27	26
4	3	12	11	20	19	28	27
5	4	13	12	21	20	29	28
6	5	14	13	22	21	30	29
7	6	15	14	23	22	31	30
8	7	16	15	24	23	32	31

(6) In-position signal (M1602+20n/M2402+20n)

(a) The in-position signal comes ON when the number of droop pulses in the deviation counter enters the "in-position range" set in the servo parameters. It goes OFF when axis motion starts.

[Motion program example]

0001;	Program No.
G90 G00 X100.;	Absolute value command PTP positioning (X100.)
X200.;	PTP positioning (X200.)
M02;	Reset
%	



(b) An in-position check is performed in the following cases.

- When the servo power supply is switched on
- After automatic acceleration/deceleration is started during positioning control
- After deceleration is started as a result of the JOG start signal going OFF
- When manual pulse generator operation is in progress
- After the near-zero point dog comes ON during a home position return
- After deceleration is started as a result of a stop command
- When a speed change to a speed of "0" is executed
- After deceleration is started under temporary stop command

(7) Command in-position signal (M1603+20n/M2403+20n)

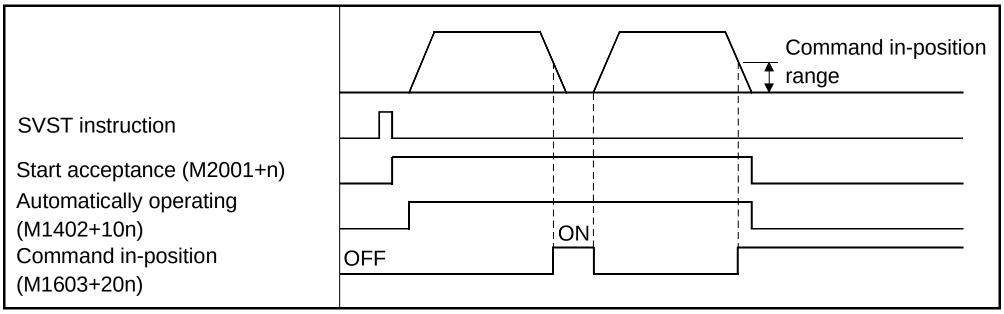
(a) The command in-position signal comes ON when the absolute value of the difference between the command position and the feed present value enters the "command in-position range" set in the fixed parameters.
It goes OFF in the following cases.

- When positioning control starts
- When a home position return is executed
- When speed control is executed
- When JOG operation is performed
- When manual pulse generator operation is performed

(b) Command in-position checks are continually performed during positioning control.

[Motion program example]

```
0001;             Program No.
G90 G00 X100.;    Absolute value command PTP positioning (X100.)
X200.;            PTP positioning (X200.)
M02;              Reset
%
```

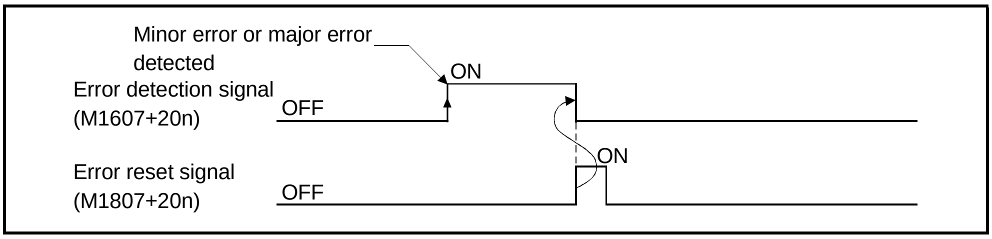


(8) Zero pass signal (M1606+20n/M2406+20n)
This signal comes ON when the zero point is passed after the power to the servo amplifier has been switched ON.
Once the zero point has been passed, the signal remains ON until the CPU has been reset.

(9) Error detection signal (M1607+20n/M2407+20n)

(a) The error detection signal comes ON when a minor error or major error is detected and is used to determine whether or not errors have occurred.
When a minor error is detected, the corresponding error code*1 is stored in the minor error code storage area. (Refer to section 3.2.1.)
When a major error is detected, the corresponding error code*2 is stored in the major error code storage area. (Refer to section 3.2.1.)

(b) When the error reset signal (M1807+20n/M3207+20n) comes ON, the error detection signal goes OFF.



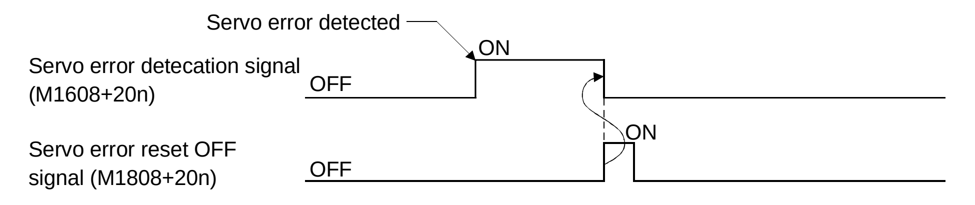
REMARKS

*1: For details on the error codes when minor errors occur, see Appendix 2.2.
*2: For details on the error codes when major errors occur, see Appendix 2.3.

(10) Servo error detection signal (M1608+20n/M2408+20n)

(a) The servo error detection signal comes ON when an error occurs at the servo amplifier side (excluding errors that cause alarms, and emergency stops)*1, and is used to determine whether or not servo errors have occurred.
When an error is detected at the servo amplifier side, the corresponding error code*1 is stored in the servo error code storage area.

(b) The servo error detection signal goes OFF when the servo error reset signal (M1808+20n/M3208+20n) comes ON, or when the servo power supply is switched back on.



REMARK

*1: For details on the error codes of errors detected at the servo amplifier side, see Appendix 2.4.

(11) Home position return request signal (M1609+20n/M2409+20n)
This signal comes ON when it is necessary to confirm the home position address when the power is switched on or during positioning control.
(a) When not using an absolute value system
1) The home position return request signal comes ON in the following cases:
- When the power is switched on, or the servo system CPU is reset.
- During a home position return operation.
2) The home position return request signal goes OFF when the home position return operation is completed.

(b) When using an absolute value system
1) The home position return request signal comes ON in the following cases:
- During a home position return operation.
- When a backup data (reference value) sum check error occurs (when the power is switched on).
2) The home position return request signal goes OFF when the home position return operation is completed.
Operation in G28 of the motion program changes with the ON/OFF of the home position return request signal.

When home position return request signal is OFF	The axis starts from the present position, passes through the specified mid point, and returns to the home position at rapid feedrate.
When home position return request signal is ON	Dog, count or data setting type home position return is performed in accordance with the home position return data.

(12) Home position return completed signal (M1610+20n/M2410+20n)
(a) The home position return completed signal turns ON when a home position return started by the DSFLP/CHGA instruction is completed properly.

(b) This signal turns OFF at positioning start, JOG operation start or manual pulse generator operation start.

(c) If near-zero point dog type home position return is started by the DSFLP/CHGA instruction while the home position return completed signal is ON, "continuous home position return start error" occurs and a home position return start cannot be made.

(13) FLS signal (M1611+20n/M2410+20n)

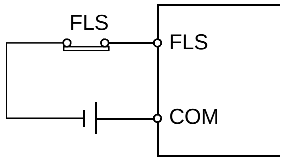
(a) FLS signal is controlled by the ON/OFF status of the upper stroke end limit switch input (FLS) to the A172SENC, A171SENC or A278LX from an external source.

- Upper stroke end limit switch input OFF ...... FLS signal: ON
- Upper stroke end limit switch input ON .... FLS signal: OFF

(b) The status of the upper stroke end limit switch input (FLS) when the FLS signal is ON/OFF is indicated in the figure below.

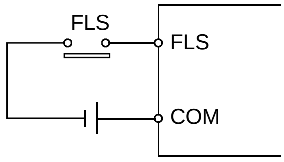
FLS signal: ON

A172SENC, A171SENC or A278LX



FLS signal: OFF

A172SENC, A171SENC or A278LX



(14) RLS signal (M1612+20n/M2412+20n)

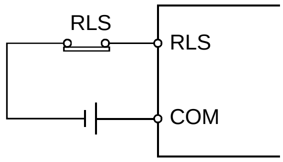
(a)The RLS signal is controlled by the ON/OFF status of the lower stroke end limit switch input (FLS) to the A172SENC, A171SENC or A278LX from an external source.

- Lower stroke end limit switch input OFF ...... RLS signal: ON
- Lower stroke end limit switch input ON .... RLS signal: OFF

(b) The status of the lower stroke end limit switch input (RLS) when the RLS signal is ON/OFF is indicated in the figure below.

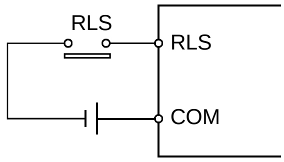
RLS signal: ON

A172SENC, A171SENC or A278LX



RLS signal: OFF

A172SENC, A171SENC or A278LX



(15) STOP signal (M1613+20n/A2413+20n)

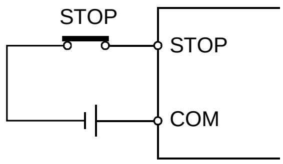
(a) The STOP signal is controlled by the ON/OFF status of the stop signal (STOP) sent to the A172SENC, A171SENC or A278LX from an external source.

- Stop signal OFF ..... STOP signal: OFF
- Stop signal ON .... STOP signal: ON

(b) The status of the external stop switch (STOP) when the STOP signal is ON/OFF is indicated in the figure below.

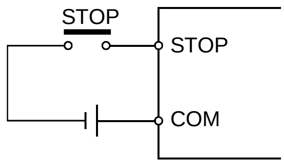
STOP signal: ON

A172SENC, A171SENC or A278LX



STOP signal: OFF

A172SENC, A171SENC or A278LX



(16) DOG/CHANGE signal (M1614+20n) (for A172SHCPUN/A171SHCPUN)

(a) The DOG/CHANGE signal is controlled by the ON/OFF of the external near-zero point dog input or speed/position control switching input (DOG/CHANGE) provided to the A172SENC or A171SENC.

(b) Independently of whether the "Leading edge valid" or "Trailing edge valid" setting has been made in the system settings, the DOG/CHANGE signal turns ON and the near-zero point dog or CHANGE signal turns OFF when the near-zero point dog or CHANGE signal turns ON.

(c) When the "Leading edge valid" setting is made in the system settings, a near-zero point dog or CHANGE input is provided when the near-zero point dog or CHANGE signal turns ON. When the "Trailing edge valid" setting is made, a near-zero point dog or CHANGE input is provided when the near-zero point dog or CHANGE signal turns OFF.

(17) DOG signal (M2414+20n) (for A273UHCPU (32 axis feature)/A173UHCPU(S1))

(a) The DOG signal is controlled by the ON/OFF of the external near-zero point dog (DOG) input provided to the A278LX.

(b) Independently of whether the "A contact input" or "B contact input" setting has been made in the system settings, the near-zero point dog signal turns ON when the near-zero point dog turns ON, and the near-zero point dog signal turns OFF when the near-zero point dog turns OFF.

(c) When the "A contact input" setting is made in the system settings, a near-zero point dog input is provided when the near-zero point dog turns ON, and when the "B contact input" setting is made, a near-zero point dog input is provided when the near-zero point dog turns OFF.

(18) Servo READY signal (M1615+20n/M2415+20n)

(a) The servo READY signal comes ON when the servo amplifiers connected to each axis are in the READY status.

(b) The signal goes OFF in the following cases.
- When M2042 is OFF
- When no servo amplifier is installed
- When the servo parameters have not been set
- When the power supply module has received an emergency stop input from an external source
- When the M1815+20n signal comes ON and establishes the servo OFF status
- When a servo error occurs
  For details, see Appendix 2.4 "Servo Errors"

POINT

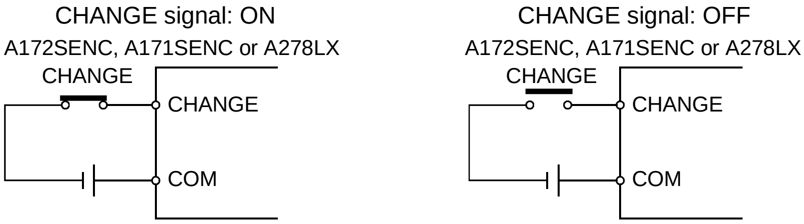
When an axis driven by an MR-□-B becomes subject to a servo error, the affected axis only goes into the servo OFF status.

(19) Torque control in progress signal (M1616+20n/M2416+20n)
Signals for axes whose torque is being controlled are ON.

(20) CHANGE signal (M2417+20n) (for A273UHCPU (32 axis feature)/A173UHCPU(S1))

(a) The CHANGE signal is controlled by the ON/OFF of the external speed/position control switching input (CHANGE) provided to the A278LX.
- Speed/position switching input is OFF ..... CHANGE signal: OFF
- Speed/position switching input is ON .... CHANGE signal: ON

(b) The following diagrams show the positions of the speed select switch (CHANGE) when the CHANGE signal is ON and OFF.



(21) M code output signal (M1619+20n/M2419+20n)

(a) This signal turns ON when M** in the motion program is executed.
This signal turns OFF when the FIN signal (M1819+20n/M3219+20n) turns ON.
Read the M code when the M code outputting signal is ON.

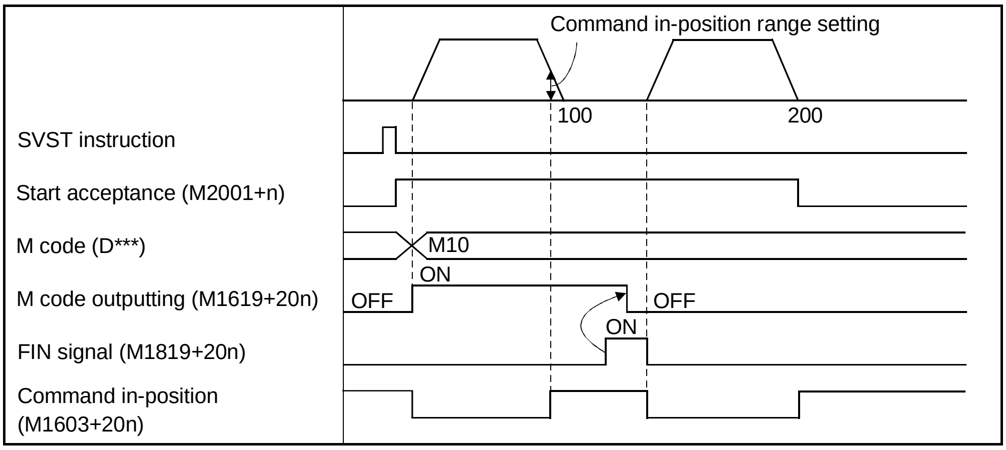
(b) If the G and M codes are described in the same block, the M code output signal turns ON at the start of G code processing.

(c) When you want to execute the miscellaneous function M after completion of position control, describe the M code independently.

(d) For M00, M01, M02, M30, M98, M99 and M100, the M code output signal does not turn ON. (Internal processing only)

[Motion program example]

0001;	Program No.
G90 G00 X100. M10.;	Absolute value command PTP positioning (X100.) M10
X200.;	PTP positioning (X200.)
M02;	Reset
%	



3.1.2 Axis command signals

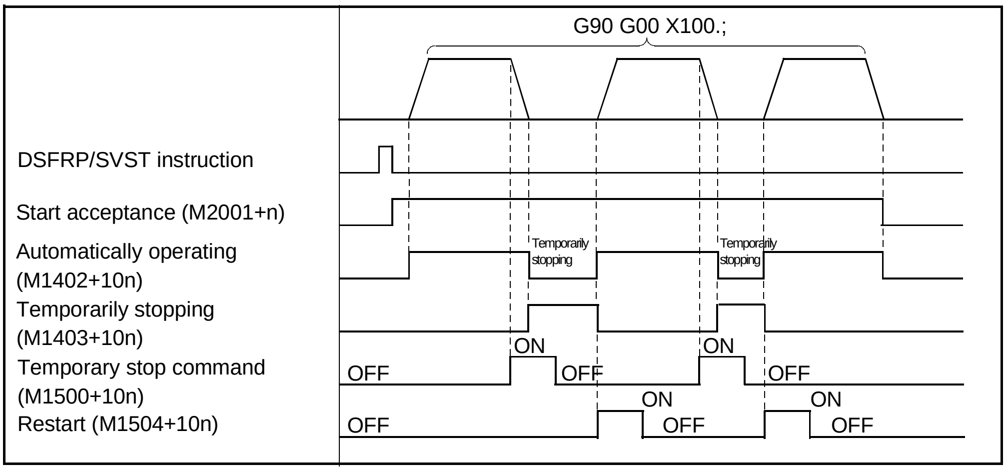
(1) Temporary stop command (M1500+10n/M4400+10n)

(a) The motion program which is making a positioning start (G00, G01, etc.) under the DSFRP/SVST instruction is stopped temporarily by the temporary stop command.
(The motion program stops temporarily if any of the temporary stop commands for the axis names specified in the SVST instruction turns ON.)

(b) To restart, turn ON M1504+10n/M4404+10n.

[Motion program example]

01;	Program No.
G90 G00 X100.;	Absolute value command PTP positioning (X100.)
M02;	Reset
%	



(c) Among the positioning start instructions, the following instructions must be noted.

1) A dog, count or data setting type home position return under G28 is stopped and ended by the temporary stop command. After that, restart (M1504+10n) is invalid.
When you want to execute G28 again, start the motion program using the SVST instruction.

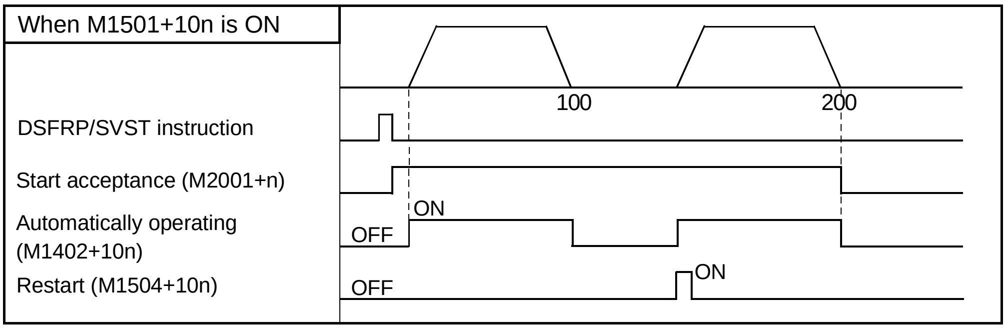
2) The axis executing G25 (high-speed oscillation) ignores the temporary stop.

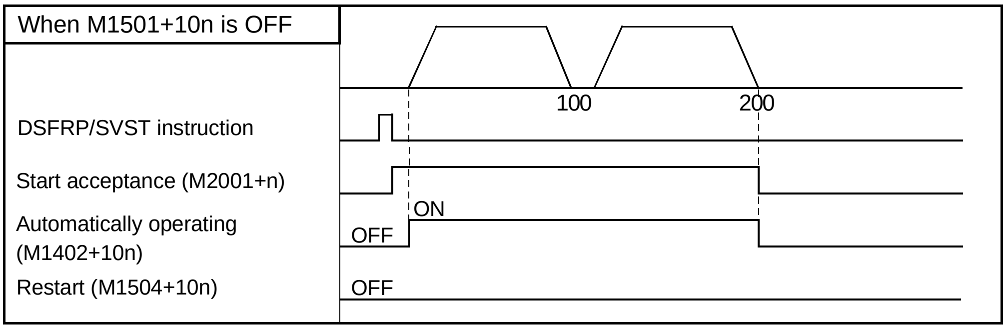
POINT
(1) During a home position return made by JOG operation, manual pulse generator, DSFLP/CHGA instruction or the like, the temporary stop command is ignored.

(2) Optional program stop command (M1501+10n/M4401+10n)
This signal is used to select whether a block stop is made in a block where "M01" exists.
- ON ...... A block stop is made at the end of that block.
- OFF ..... Execution shifts to the next block.

[Motion program example]

0001;	Program No.
G90 G00 X100.;	Absolute value command PTP positioning (X100.)
M01;	Optional program stop command
X200.;	PTP positioning (X200.)
M02;	Reset
%	

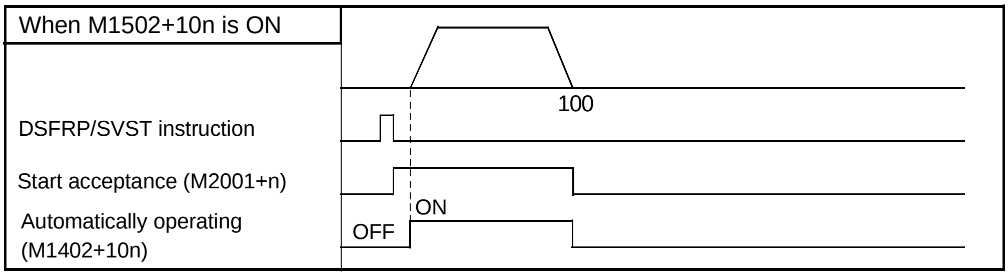


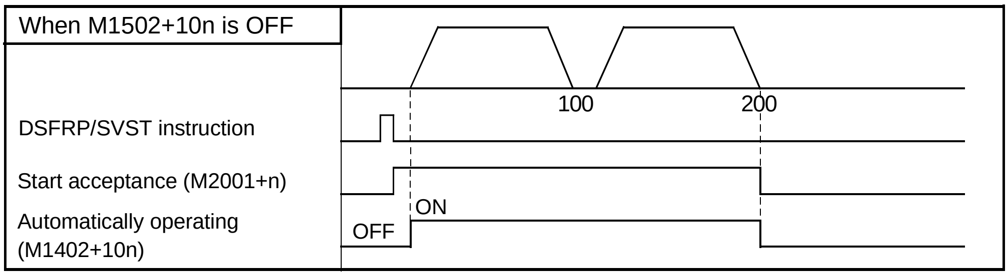


(3) Optional block skip command (M1502+10n/M4402+10n)
This signal is used to select whether a block headed by "/" is to be executed or not.
- ON ...... That block is not executed and execution shifts to the next block.
- OFF .... That block is executed.

[Motion program example]

0001;	Program No.
G90 G00 X100.;	Absolute value command PTP positioning (X100.)
/X200.;	PTP positioning (X200.)
M02;	Reset
%	





(4) Single block command (M1503+10n/M4403+10n)

This single block is the mode where a single block is specified before a program start. For the mode where a single block is executed at any point during program run, refer to the single block mode signal (M1508/M4408).

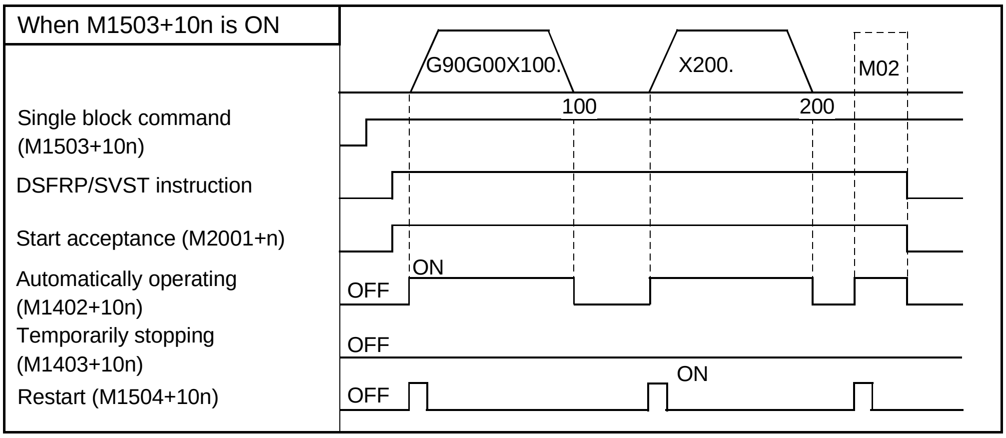
By turning ON the single block command before a program start, commands in program operation can be executed block by block.

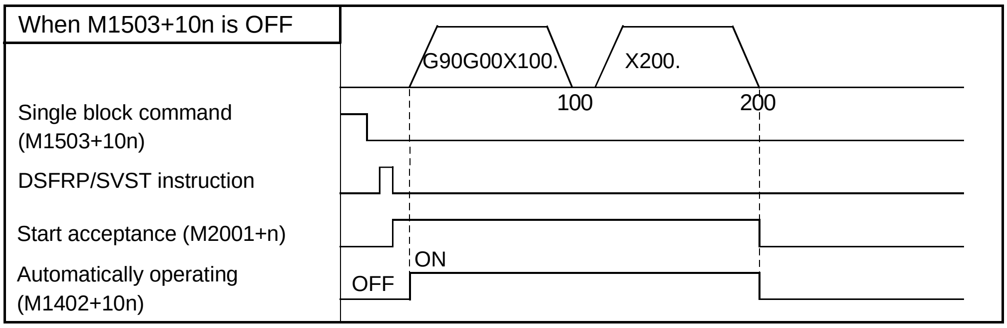
The single block signal is checked only at a motion program start and is not checked during operation. Therefore, the single block signal is not made valid if it is turned ON during operation.

- ON .... Program is executed block by block.
  The first start is made by turning ON the restart command (M1504+10n) after execution of the DSFRP/SVST instruction.
  After that, a start is made by turning ON the restart command (M1504+10n/M4404+10n).
- OFF .... All blocks are executed continuously by the DSFRP/SVST instruction.

[Motion program example]

0001;	Program No.
G90 G00 X100.;	Absolute value command PTP positioning (X100.)
X200.;	PTP positioning (X200.)
M02;	Reset
%	

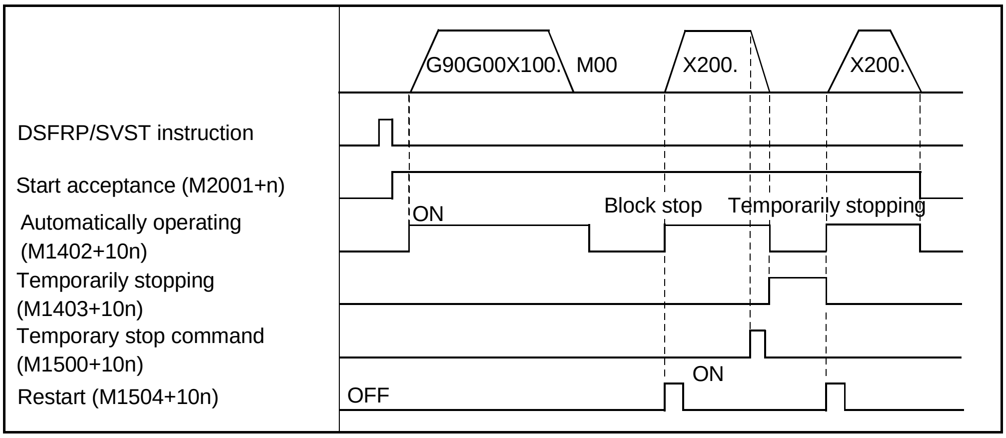




(5) Restart command (M1504+10n/M4404+10n)
This signal resumes bock execution when it is turned ON during a block stop under the M00, M01 or single block command or during a temporary stop under the temporary stop command. (This signal is valid for the motion program only. It is invalid for a home position return, etc.)

[Motion program example]

0001;	Program No.
G90 G00 X100.;	Absolute value command PTP positioning (X100.)
M00;	Block stop
X200.;	PTP positioning (X200.)
M02;	Reset
%	



(6) Override ratio valid/invalid (M1505+10n/M4405+10n)
This signal is used to set whether the override ratio is valid or invalid.
- ON .... Valid: Turning ON M1505+10n/M4405+10n during motion program run starts positioning at the specified speed multiplied by the value (%) stored in the override ratio setting register.*1
- OFF .... Invalid: Positioning is controlled at the override ratio of 100%.

REMARK

*1: Under G25 (high-speed oscillation) or G28 (dog, count, data setting) in the motion program or during a home position return made by JOG operation, manual pulse generator, DSFLP/CHGA instruction or the like, positioning is controlled at the override ratio of 100%. (The override ratio is made invalid.)

(7) Single block mode signal (M1508/M4408)
(a) The single block mode signal makes a single block valid in the mode where a single block is executed at any point during program execution.

(b) Turning ON the single block mode turns ON the single block in progress (M1409).

(8) Single block start signal (M1509/M4409)

(a) The single block start signal restarts a single block in the mode where a single block is executed at any point during program execution.

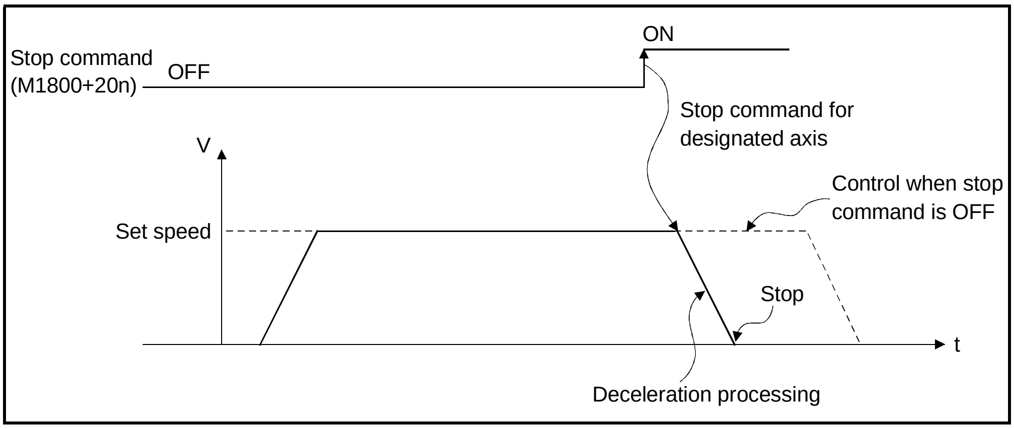
(b) The single block start is made valid by turning it from OFF to ON. Note that it is not accepted during axis movement.

(c) When the single block in progress (M1409/M4409) is ON and the single block mode (M1508/M4408) is ON, making a single block start continues single block operation.

(d) When the single block in progress (M1409/M4409) is ON and the single block mode (M1508/M4408) is OFF, making a single block start stops single block operation and starts continuous operation. At this time, the single block in progress (M1409/M4409) turns OFF.

(9) Stop command (M1800+20n/M3200+20n)

(a) The stop command is a signal used to stop an axis that is currently being driven and becomes effective at its leading edge (OFF→ON). (An axis for which the stop command is ON cannot be started.)



(b) During automatic operation started by the DSFRP/SVST instruction, the program is ended by the stop command. (The motion program is stopped if any of the stop commands for the axis names specified in the DSFRP/SVST instruction turns ON.)

(c) M1504+10n/M4404+10n (restart) is valid only after M1500+10n/M4400+10n (temporary stop).

(d) The following stop processing is performed when the stop command is turned ON.

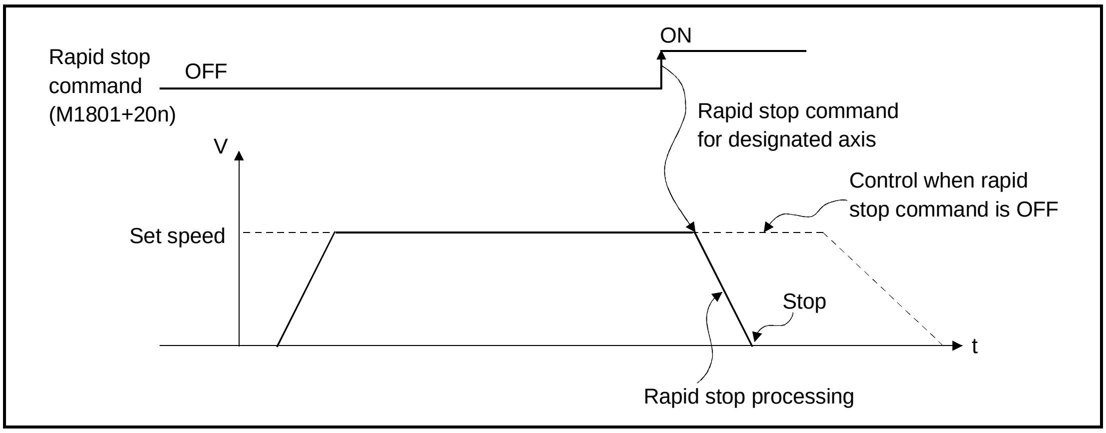
Control Being Executed	Processing when the Stop Command Comes ON	
	If Control is Being Executed	If Deceleration Stop Processing is Being Executed
Position control during motion program run	The axis decelerates to a stop in the deceleration time set in the parameter block or servo program. (Note 1)	The stop command is ignored and deceleration stop processing continues. (Note 1)
JOG operation	The axis decelerates to a stop in the deceleration time set in the parameter block or servo program. (Note 1)	The stop command is ignored and deceleration stop processing continues. (Note 1)
Manual pulse generator operation	An immediate stop is executed, with no deceleration processing.	–
Home position return	(1) The axis decelerates to a stop in the deceleration time set in the parameter block. (2) A "stop during home position return" error occurs and the error code (202) is stored in the minor error storage area for each axis.	

(Note 1) The deceleration time under G00 including M code, G01, G02, G03 or G32 is equivalent to the acceleration time set in the parameter block.

POINT
If a home position return being made is stopped by turning ON the stop command (M1800+20n/M3200+20n), make a home position return again. If the stop command is turned ON after the near-zero point dog has turned ON in the near-zero point dog type home position return, make a home position return after performing JOG operation, positioning or the like to move the axis to a position before the near-zero point dog is turned ON.

(10) Rapid stop command (M1801+20n/M3201+20n)

(a) The rapid stop command is a signal used to rapidly stop an axis that is currently being driven and becomes effective at its leading edge (OFF→ON). (An axis for which the rapid stop command is ON cannot be started.)



(b) During automatic operation started by the DSFRP/SVST instruction, the program is ended by the rapid stop command.
(The motion program is stopped if any of the rapid stop commands for the axis names specified in the DSFRP/SVST instruction turns ON.)

(c) M1504+10n/M4404+10n (restart) is valid only after M1500+10n/M4400+10n (temporary stop).

(d) The following stop processing is performed when the rapid stop command is turned ON.

Control Being Executed	Processing when the Rapid Stop Command Comes ON	
	If Control is Being Executed	If Deceleration Stop Processing is Being Executed
Position control during motion program run	The axis decelerates to a stop in the deceleration time set in the parameter block or servo program. (Note 1)	Deceleration processing is canceled and rapid stop processing executed instead. (Note 1)
JOG operation		
Manual pulse generator operation	An immediate stop is executed, with no deceleration processing.	—
Home position return	(1) The axis decelerates to a stop in the rapid stop deceleration time set in the parameter block. (2) A "stop during home position return" error occurs and the error code (203) is stored in the minor error storage area for each axis.	

(Note 1) The deceleration-to-rapid-stop time under G00 including M code, G01, G02, G03 or G32 is equivalent to the acceleration time set in the parameter block.

POINT

If a home position return being made is stopped by turning ON the rapid stop command (M1801+20n/M3201+20n), make a home position return again.
If the rapid stop command is turned ON after the near-zero point dog has turned ON in the near-zero point dog type home position return, make a home position return after performing JOG operation, positioning or the like to move the axis to a position before the near-zero point dog is turned ON.

(11) Forward JOG start command (M1802+20n/M3202+20n)/Reverse JOG start command (M1803+20n/M3203+20n)

(a) While the sequence program keeps M1802+20n/M3203+20n ON, JOG operation is executed in the direction in which address numbers increase. When M1802+20n/M3202+20n is turned OFF, a deceleration stop is executed in the deceleration time set in the parameter block.

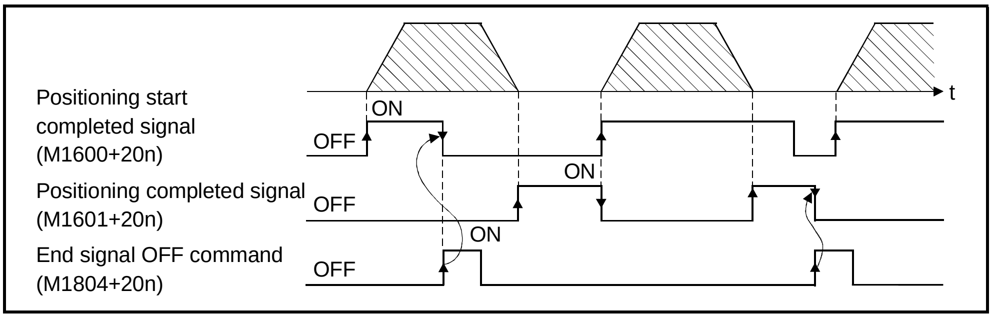
(b) While the sequence program keeps M1803+20n/M3203+20n ON, JOG operation is executed in the direction in which address numbers decrease. When M1803+20n/M3203+20n is turned OFF, a deceleration stop is executed in the deceleration time set in the parameter block.

POINT

Establish an interlock in the sequence program to make it impossible for the forward JOG start command (M1802+20n/M3202+20n) and the reverse JOG start command (M1803+20n/M3203+20n) to be ON at the same time.

(12) End signal OFF command (M1804+20n/M3204)

(a) The end signal OFF command is used to turn off the positioning start completed signal (M1600+20n/M2400+20n) and the positioning completed signal (M1601+20n/M2401+20n) by using the sequence program.



POINT

Do not turn the end signal OFF command ON with a PLS command.
If it is turned ON with a PLS command, it will not be possible to turn OFF the positioning start completed signal (M1600+20n/M2400+20n) or the positioning completed signal (M1601+20n/M2401+20n).

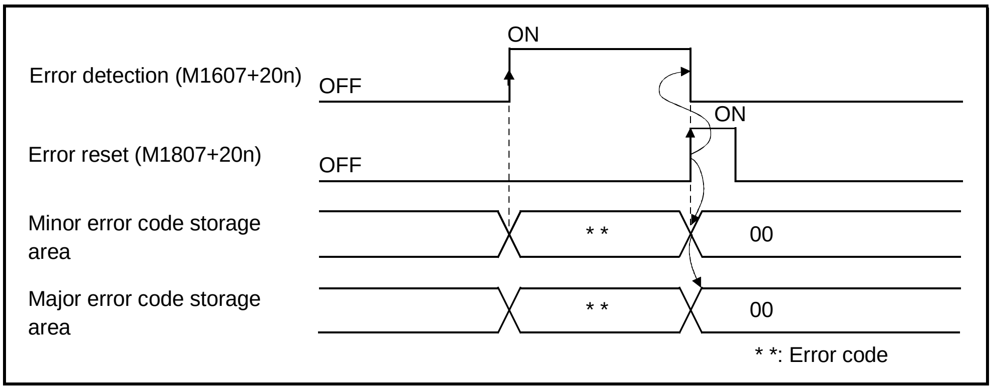
(13) Limit switch output enable command (M1806+20n/M3208+20n)

The limit switch output enable command is used to enable limit switch output.

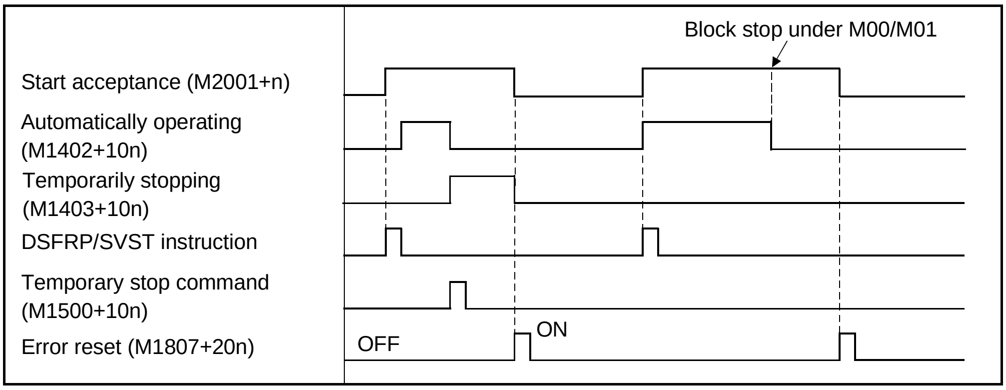
- ON ....The limit switch output ON/OFF pattern can be output.
- OFF ....Limit switch output goes OFF.

(14) Error reset command (M1807+20n/M3207+20n)

(a) The error reset command is used to clear the minor error code or major error code storage area of an axis for which the error detection signal has come ON (M1607+20n/M3207+20n: ON), and reset the error detection signal (M1607+20n/M3207+20n).



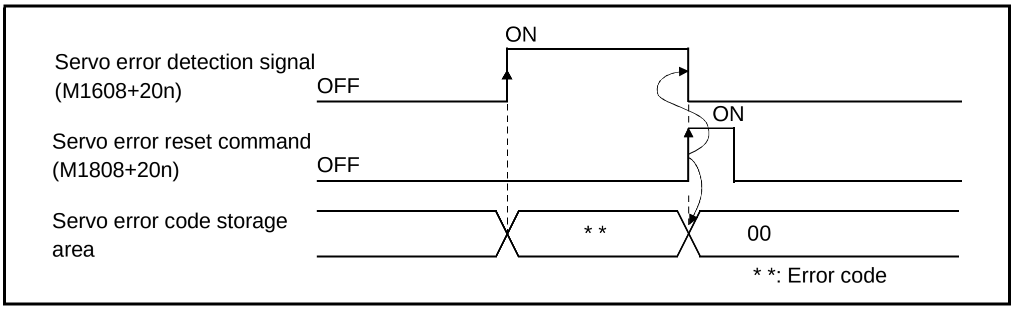
(b) If an error reset is made during the temporary stop (M1403+10n/M4003+10n) under the stop command (M1800+20n/M3200+20n) during automatic operation or if an error reset is made during a block stop under M00/M01, the motion program running status is reset.
When a next start is made, the DSFRP/SVST instruction must be executed. (Restart cannot be made.)



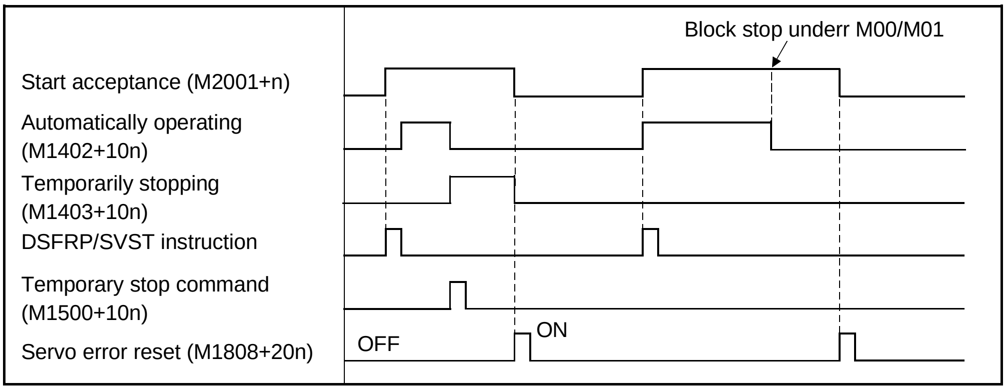
(c) When the error reset command is turned ON during automatic operation (M1402+10n/M4002+10n ON), the above reset processing is performed after the stop processing is carried out under the temporary stop command (M1500+10n/M4400+10n).

(15) Servo error reset command (M1808+20n/M3208+20n)

(a) The servo error reset command is used to clear the servo error code storage area of an axis for which the servo error detection signal has come ON (M1608+20n/M2408+20n): ON), and reset the servo error detection signal (M1608+20n/M2408+20n).



(b) If an error reset is made during the temporary stop (M1403+10n/M4003+10n) under the stop command (M1800+20n/M2400+20n) during automatic operation or if an error reset is made during a block stop under M00/M01, the motion program running status is reset.
When a next start is made, the DSFRP/SVST instruction must be executed. (Restart cannot be made.)



(c) When the error reset command is turned ON during automatic operation (M1402+10n/M4002+10n ON), the above reset processing is performed after the stop processing is carried out under the temporary stop command (M1500+10n/M4400+10n).

POINT

*: Do not turn the error reset command (M1807+20n/M3207+20n) or servo error reset command (M1808+20n/M3208+20n) ON with a PLS command.
If a PLS command is used, it will not be possible to reset the error or servo error.

REMARK

For details on minor error code, major error code, and servo error code storage areas, see Appendix 2.

(16) External STOP input/invalid when starting command (M1809+20n/M3209+20n)
This signal is used to make external STOP signal input valid or invalid.
- ON.........External STOP input is set as invalid, and even axes for which STOP input is currently ON can be started.
- OFF.......External STOP input is set as valid, and axes for which STOP input is currently ON cannot be started.

POINTS
(1) To stop an axis by external STOP input after it has been started with the M1809+20n/M3209+20n command ON, switch the STOP input from OFF to ON
(if STOP input is ON when the axis is started, switch it from ON to OFF to ON).
(2) External STOP input causes a block stop during automatic operation (M1402+10n/M4002+10n ON).

(17) Servo OFF command (M1815+20n/M3215+20n)
The servo OFF command is used to establish the servo OFF status (free run status).
- M1815+20n/M3215+20n : OFF ....Servo ON
- M1815+20n/M3215+20n : ON...........Servo OFF (free run status)

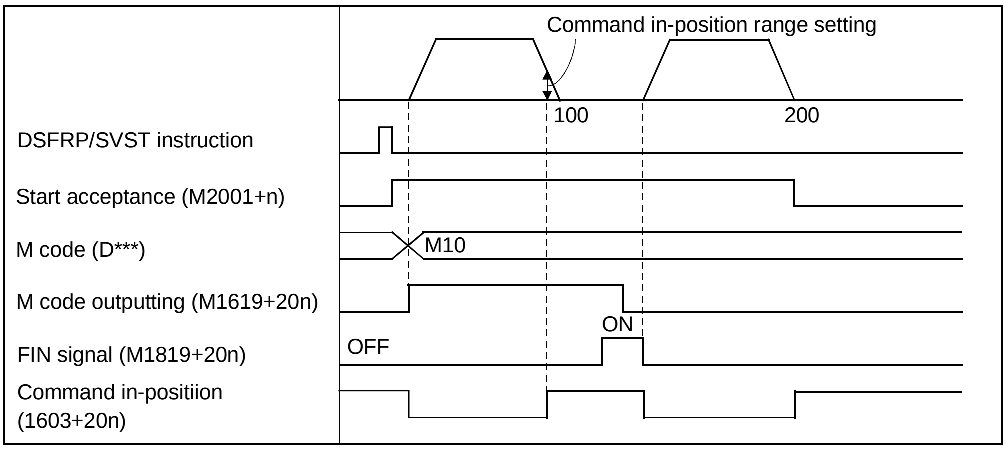
This command is not effective during positioning and should therefore be executed on completion of positioning.

CAUTION
Turn the power supply at the servo side OFF before turning a servomotor by hand.

(18) FIN signal (M1819+20n/M3219+20n)
When an M code is set in a point during positioning, travel to the next block does not take place until the FIN signal state changes as follows:
OFF→ON→OFF
Positioning to the next block begins after the FIN signal state changes as above.

[Motion program example]

0001;	Program No.
G90 G00 X100. M10;	Absolute value command PTP positioning (X100.) M10
X200.;	PTP positioning (X200.)
M02;	Reset
%	



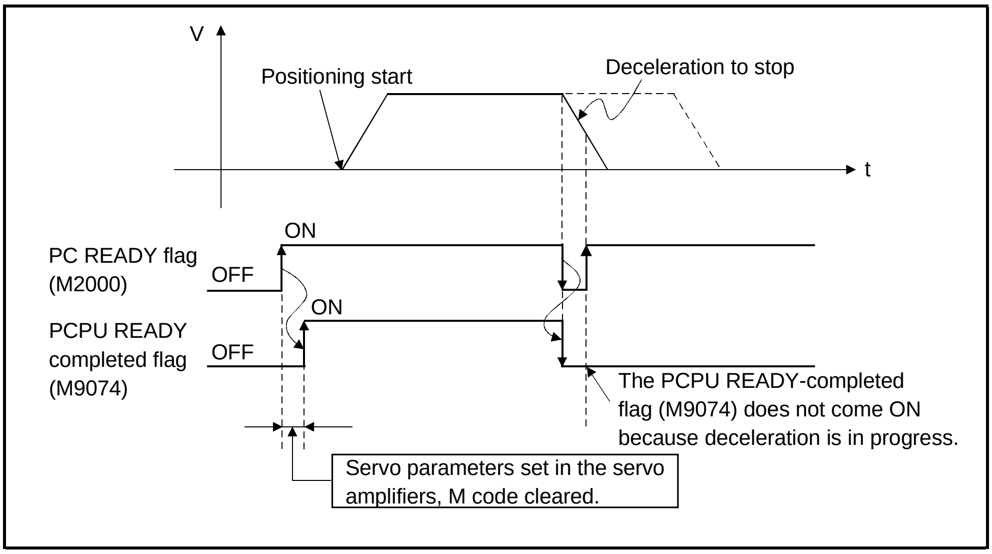
3.1.3 Common devices

POINTS
(1) Internal relays for positioning control are not latched even inside the latch range. In this manual, in order to indicate that internal relays for positioning control are not latched, the expression used in this text is "M2000 to M2047". (2) The range of devices allocated as internal relays for positioning control cannot be used by the user even if their applications have not been set.

(1) PC READY flag (M2000)...........Signal sent from SCPU to PCPU
  (a) This signal serves to notify the PCPU that the SCPU is normal. It is switched ON and OFF by the sequence program.
    1) While M2000 is ON, the positioning control or home position return specified by the motion program, or the JOG operation or manual pulse generator operation specified by the sequence program, can be executed.
    2) Control in above (1) is not exercised if M2000 is turned ON while M2000 is OFF or in the test mode using peripheral device [while the test mode in progress flag (M9075) is ON].

  (b) The fixed parameters, servo parameters, and limit switch output parameters can only be changed using a peripheral device when M2000 is OFF. If an attempt is made to change this data while M2000 is ON, an error will occur.

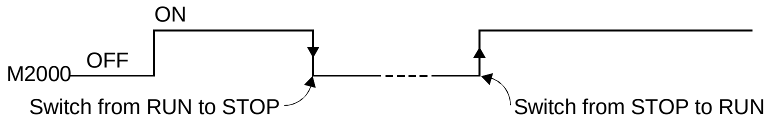
  (c) When M2000 is switched from OFF to ON, the following processing occurs.
    1) Processing details
      - The servo parameters are transferred to the servo amplifier.
      - The M code storage area for all axes is cleared.
      - The default value of 300% is set in the torque limit value storage area. (See Section 4.6.)
      - The PCPU READY-completed flag (M9074) is turned ON.
    2) If there is an axis currently being driven, an error occurs, and the processing in (c) 1) above is not executed.
    3) While the test mode is in effect, the processing in (c) 1) above is not executed. When the test mode is cancelled, the processing in (c) 1) above is executed if M2000 is ON.



(d) When M2000 is switched from ON to OFF, the following processing is executed.
1) Processing details
- The PCPU READY-completed flag (M9074) is turned OFF.
- The axis being driven is decelerated to a stop.

POINT

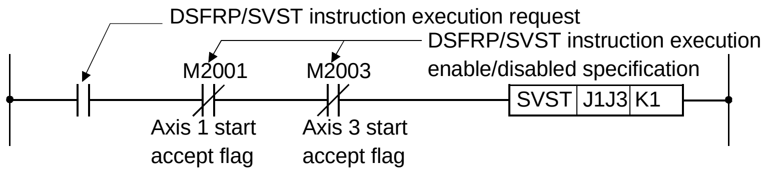
The PC READY flag (M2000) goes OFF when the servo system CPU is in the STOP status. When the RUN status is re-established, the status is the same as before the STOP was executed.



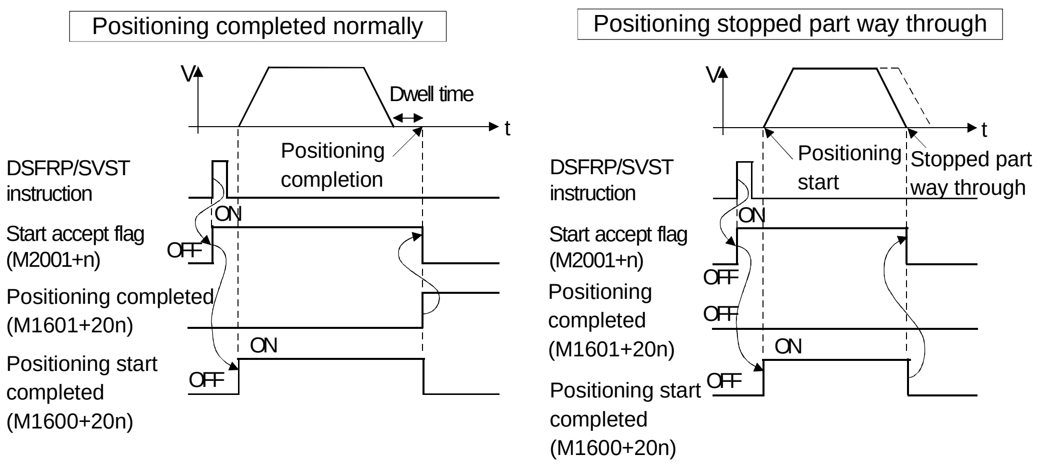
(2) Start accept flag (M2001+n)...........Signal sent from PCPU to SCPU
(a) The start accept flag comes ON when the positioning start (DSFRP/ SVST) instruction is executed in the sequence program: use it as an interlock to enable or disable execution of the DSFRP/SVST instruction.

Example

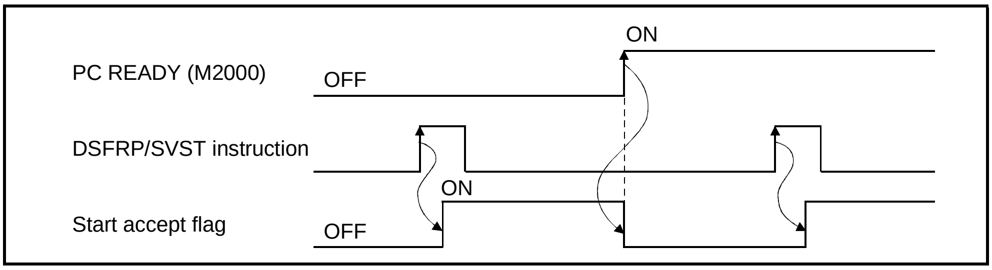
When requesting execution of the servo programs for positioning on axis 1 and axis 3, use the start accept flags in the way shown below.



(b) The start accept flag ON/OFF processing takes the following form.
1) The start accept flag for the designated axis comes ON in response to a DSFRP/SVST instruction, and goes OFF on completion of positioning. The start accept flag will also go OFF if positioning is stopped part way through.
(However, if positioning is stopped part way through by a speed change to speed 0, the start accept flag will remain ON.)



2) When positioning control is executed by turning ON the JOG operation command (M1802+20n/M3202+20n or M1803+20n/M3203+20n), the start accept flag goes OFF when positioning is stopped by turning the JOG operation command OFF.
3) The start accept flag is ON while the manual pulse generator enable flag (M2012/M2051: ON) is ON.
The start accept flag is OFF while the manual pulse generator enable flag (M2012/M2051: OFF) is OFF.
4) When M2000 is OFF, execution of a DSFRP/SVST instruction causes the start accept flag to come ON; the flag goes OFF when M2000 comes ON.



⚠ CAUTION

The user must not turn start accept flags ON/OFF.

- If a start accept flag that is ON is switched OFF with the sequence program or a peripheral device, no error will occur but the positioning operation will not be reliable. Depending on the type of machine, it might operate in an unanticipated manner.
- If a start accept flag that is OFF is switched ON with the sequence program or a peripheral device, no error will occur at that time, but the next time an attempt is made to start the axis an error will occur during a start accept flag being ON and the axis will not start.

REMARK

A numerical value corresponding to an axis number is entered for "n".

<A172SHCPUN>

Axis No.	n
1	0
2	1
3	2
4	3
5	4
6	5
7	6
8	7

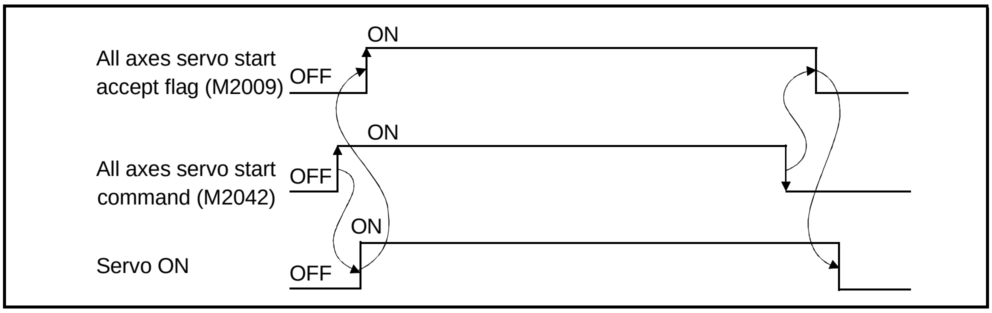
<A171SHCPUN>

Axis No.	n
1	0
2	1
3	2
4	3

<A273UHCPU (32 axes feature) / A173UHCPU>

Axis No.	n	Axis No.	n	Axis No.	n	Axis No.	n
1	0	9	8	17	16	25	24
2	1	10	9	18	17	26	25
3	2	11	10	19	18	27	26
4	3	12	11	20	19	28	27
5	4	13	12	21	20	29	28
6	5	14	13	22	21	30	29
7	6	15	14	23	22	31	30
8	7	16	15	24	23	32	31

(3) All axis servo start accept flag (M2009/M2049)
....Signal sent from PCPU to SCPU
The all axis servo start accept flag serves to notify that servo operation is possible.
- ON ....The servomotor can be driven.
- OFF ....The servomotor cannot be driven.



(4) Manual pulse generator enable flag (M2012/M2051 to M2053)
....Signal sent from SCPU to PCPU
The manual pulse generator enable flags set the enabled or disabled status for positioning with the pulse input from the manual pulse generators connected to P1 of the A273EX/A172SENC/A171SENC.
- ON ....Positioning control is executed in accordance with the input from the manual pulse generators.
- OFF ....Positioning with the manual pulse generators is not possible because the input from the manual pulse generators is ignored.

REMARK

*: For details on the P1 connector of the A273EX/A172SENC/A171SENC, refer to the (A172SHCPUN/A171SHCPUN/A273UHCPU/A173UHCPU(S1)) Motion Controller User's Manual.

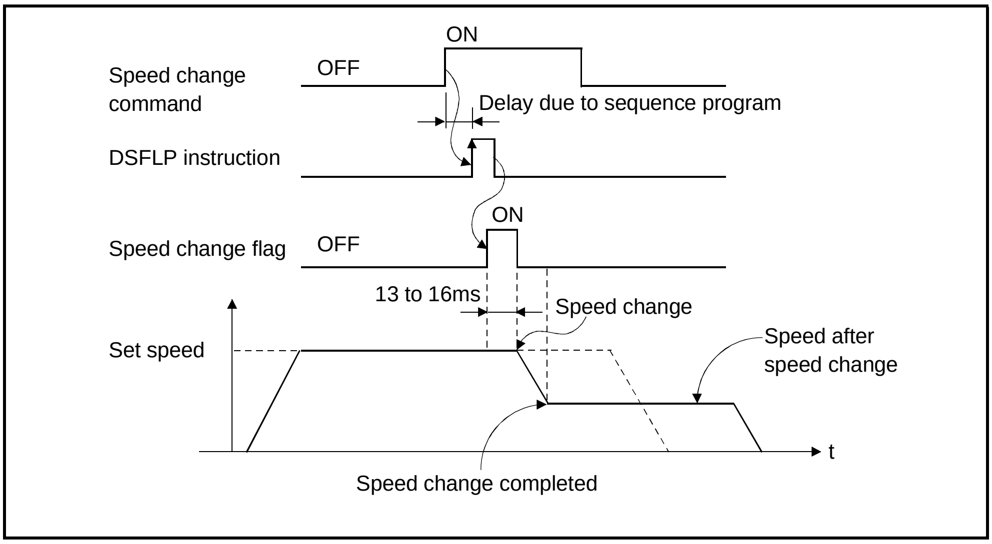
(5) JOG simultaneous start command (M2015/M2048)
....Signal sent from SCPU to PCPU
(a) When M2015/M2048 is turned ON, JOG operation is simultaneously started on the axes for which JOG operation is to be executed (of axes 1 to 4) as set in the JOG operation simultaneous start axis setting register (D1015).

(b) When M2015/M2048 is turned OFF, motion on the axis currently executing JOG operation decelerates to a stop.

(6) Start buffer full (M2020/M2050) ....Signal sent from PCPU to SCPU
(a) This signal comes ON when 16 or more requests have been issued simultaneously to the PCPU by means of position start (DSFRP/SVST) instructions and/or control change (DSFLP) instructions in the sequence program.

(b) Reset M2020/M2050 by using the sequence program.

(7) Speed change flags (M2021 to M2028/M2061+n)
....Signal from PCPU to SCPU
The speed change flags come ON when a speed change is executed in response to a control change (DSFLP/CHGV) instruction in the sequence program: use them for interlocks in speed change programs.



(8) System setting error flag (M2041)................. Signal sent from PCPU to SCPU
When the power is switched ON, or when the servo system CPU is reset, the system setting data set with a peripheral device is input, and a check is performed to determine if the set data matches the module mounting status (of the main base unit and extension base units).
- ON .... Error
- OFF .... Normal

(a) When an error occurs, the ERROR LED at the front of the CPU comes ON. Also, the error log can be known from the peripheral devices started by GSV43P.

(b) When M2041 is ON, positioning cannot be started. You must eliminate the cause of the error and switch the power back ON, or reset the servo system CPU.

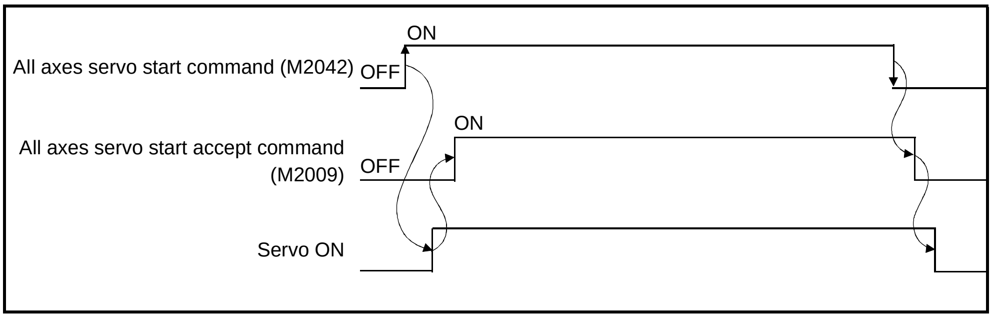
REMARK

Even if a module is loaded at a slot set as "NO USE"
in the system setting data set with a peripheral device, that slot will be regarded as not used.

(9) All axes servo start command (M2042) .... Signal from SCPU to PCPU
The all axes servo start command is used to enable servo operation.

(a) Servo operation enabled· · · · · · M2042 is turned ON while the servo OFF signal (M1815+20n) is OFF and there is no servo error.

(b) Servo operation disable· · · · · · · ·
- M2042 is OFF
- The servo OFF signal (M1815+20n) is ON
- Servo error

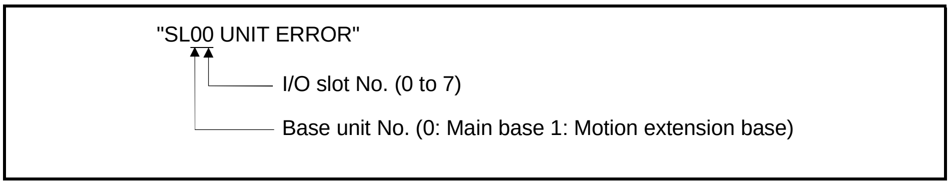


POINT
M2042 has been turned ON, it will not go OFF even if the CPU is set in the STOP status.

(10) Motion slot module fault detection flag (M2047)
.... Signal from PCPU to SCPU
This flag is used to determine whether the modules loaded in the motion slots of the main base unit are "normal" or "abnormal".
- ON ...... Loaded module is abnormal
- OFF .... Loaded module is normal

The module information at power-on and the module information after power-on are always checked to detect abnormality.

(a) When M2047 turns ON, the ERROR LED of the A172SHCPUN/A171SHCPUN/A173UHCPU(S1) is lit.
The following message appears on the LED display of the A273UHCPU.



(b) Use the sequence program to perform appropriate processing (e.g. stop the operating axis or switch servo OFF) at detection of a fault.

3.2 Data Registers

(1) Data registers

A172SHCPUN

Device No.	Purpose
D0	User device (500 points)
D500	Control change register for SV43 (6 points × 8 axes)
D560	Tool length offset data (40 points)
D600	Axis monitor device for SV43 (20 points × 8 axes)
D760	Unusable (40 points)
D800	Axis monitor device (20 points × 8 axes)
D960	Control change register (6 points × 8 axes)
D1008 D1023	Common device (16 points)

A171SHCPUN

Device No.	Purpose
D0	User device (500 points)
D500	Control change register for SV43 (6 points × 4 axes)
D560	Tool length offset data (40 points)
D600	Axis monitor device for SV43 (20 points × 4 axes)
D680	Unusable (120 points)
D800	Axis monitor device (20 points × 4 axes)
D880	Unusable (80 points)
D960	Control change register (6 points × 4 axes)
D984	Unusable (24 points)
D1008 D1023	Common device (16 points)

A273UHCPU (32 axis feature) / A173UHCPU (S1)

Device No.	Purpose
D0	Axis monitor device (20 points × 32 axes)
D640	Control change register (2 points × 32 axes)
D704 D799	Common device (96 points)
D800	Axis monitor device for SV43 (20 points × 32 axes)
D1440	Control change register for SV43 (6 points × 32 axes)
D1632	Unusable (18 points)
D1650	Tool length offset data (40 points)
D1690 D8191	User device (6502 points)

POINT

- Total number of user device points

A172SHCPUN	800 points	A273UHCPU (32 axis feature)	6502 points
A171SHCPUN	800 points	A173UHCPU (S1)	

(2) Axis monitor devices

• Axis monitor devices for SV43

Axis No.	A172SHCPUN Device No.	A171SHCPUN Device No.
1	D600 to D619	D600 to D619
2	D620 to D639	D620 to D639
3	D640 to D659	D640 to D659
4	D660 to D679	D660 to D679
5	D680 to D699	
6	D700 to D719	
7	D720 to D739	
8	D740 to D759	

Signal name

	Signal name	Refresh cycle	Fetch cycle	Unit	Signal direction
0 1	Current value	END		Command unit	SCPU←PCPU
2	Execution sequence No. (main)			–	
3	Execution block No. (main)			–	
4	Execution program No. (sub)			–	
5	Execution sequence No. (sub)			–	
6	Execution block No. (sub)			–	
7	Unusable	–		–	
8	G43/44 command	END		–	
9	Tool length offset data No.			–	
10 11	Tool length offset			Command unit	
12	Unusable	–		–	
13	Unusable			–	
14	Unusable			–	
15	Unusable			–	
16	Unusable			–	
17	Unusable			–	
18	Unusable			–	
19	Unusable			–	

• Axis monitor devices

Axis No.	A172SHCPUN Device No.	A171SHCPUN Device No.
1	D800 to D819	D800 to D819
2	D820 to D839	D820 to D839
3	D840 to D859	D840 to D859
4	D860 to D879	D860 to D879
5	D880 to D899	
6	D900 to D919	
7	D920 to D939	
8	D940 to D959	

Signal name

	Signal name	Refresh cycle	Fetch cycle	Unit	Signal direction
0 1	Machine value	3.5ms		Command unit	SCPU←PCPU
2 3	Actual current value			Command unit	
4 5	Deviation counter value			PLS	
6	Minor error code	Immedi-ately		–	
7	Major error code			–	
8	Servo error code	10ms		–	
9 10	Travel after DOG/CHANGE ON	END		Command unit	
11	Home position return second travel	3.5ms		PLS	
12	Execution program No.			–	
13	M code			–	
14	Torque limit value			%	
15	Unusable	–		–	
16	Unusable			–	
17 18	Actual present value at STOP input	END		Command unit	
19	Unusable	–		–	

* The entry "END" in the Refresh Cycle column indicates 80ms or a longer sequence program scan time.

(2) Axis monitor device

• Axis monitor device

Axis No.	A273UHCPU (32 axis feature)/ A173UHCPU(S1) Device No.
1	D0 to D19
2	D20 to D39
3	D40 to D59
4	D60 to D79
5	D80 to D99
6	D100 to D119
7	D120 to D139
8	D140 to D159
9	D160 to D179
10	D180 to D199
11	D200 to D219
12	D220 to D239
13	D240 to D259
14	D260 to D279
15	D280 to D299
16	D300 to D319
17	D320 to D339
18	D340 to D359
19	D360 to D379
20	D380 to D399
21	D400 to D419
22	D420 to D439
23	D440 to D459
24	D460 to D479
25	D480 to D499
26	D500 to D519
27	D520 to D539
28	D540 to D559
29	D560 to D579
30	D580 to D599
31	D600 to D619
32	D620 to D639

Signal name

	Signal name	Refresh cycle			Fetch cycle			Unit	Signal direction
		Set No. of axis			Set No. of axis				
	SV43 A173UHCPU	1 to 12	13 to 24	25 to32	1 to 12	13 to 24	25 to32		
	SV43 A273UHCPU	1 to 8	9 to 18	19 to 32	1 to 8	9 to 18	19 to 32		
0 1	Machine value	3.5ms	7.1ms	14.2ms				Command Unit	SCPU ← PCPU
2 3	Actual current value							Command Unit	
4 5	Deviation counter value							PLS	
6	Minor error code	Immediately						–	
7	Major error code							–	
8	Servo error code	10ms	20ms					–	
9	Home position return second Travel	3.5ms	7.1ms	14.2ms				PLS	
10 11	Travel after DOG/CHANGE ON	END						Command unit	
12	Execution program No.	At start						–	
13	M code	3.5ms	7.1ms	14.2ms				–	
14	Torque limit value							%	
15	Unusable	–						–	
16	Unusable							–	
17	Unusable							–	
18 19	Actual present value at stop input	END						Command unit	

*"END" in Refresh Cycle indicates a longer one of "50ms" and "sequence program scan time".

(2) Axis monitor device

• Axis monitor device for SV43

Axis No.	A273UHCPU (32 axis feature)/ A173UHCPU(S1) Device No.
1	D800 to D819
2	D820 to D839
3	D840 to D859
4	D860 to D879
5	D880 to D899
6	D900 to D919
7	D920 to D939
8	D940 to D959
9	D960 to D979
10	D980 to D999
11	D1000 to D1019
12	D1020 to D1039
13	D1040 to D1059
14	D1060 to D1079
15	D1080 to D1099
16	D1100 to D1119
17	D1120 to D1139
18	D1140 to D1159
19	D1160 to D1179
20	D1180 to D1199
21	D1200 to D1219
22	D1220 to D1239
23	D1240 to D1259
24	D1260 to D1279
25	D1280 to D1299
26	D1300 to D1319
27	D1320 to D1339
28	D1340 to D1359
29	D1360 to D1379
30	D1380 to D1399
31	D1400 to D1419
32	D1420 to D1439

Signal name

	Signal name		Refresh cycle			Fetch cycle			Unit	Signal direction
			Set No. of axis			Set No. of axis				
	SV43	A173UHCPU	1 to 12	13 to 24	25 to32	1 to 12	13 to 24	25 to32		
		A273UHCPU	1 to 8	9 to 18	19 to 32	1 to 8	9 to 18	19 to 32		
0 1	Current value		END						Command Unit	SCPU ← PCPU
2	Execution sequence No. (main)								–	
3	Execution block No. (main)								–	
4	Execution program No. (sub)								–	
5	Execution sequence No. (sub)								–	
6	Execution block No. (sub)								–	
7	Unusable		–						–	
8	G43/G44 command		END						–	
9	Tool length offset data No.								–	
10 11	Tool length offset								Command unit	
12	Unusable		–						–	
13	Unusable								–	
14	Unusable								–	
15	Unusable								–	
16	Unusable								–	
17	Unusable								–	
18	Unusable								–	
19	Unusable								–	

*"END" in Refresh Cycle indicates a longer one of "50ms" and "sequence program scan time".

(3) Control change register

• Control change register for SV43

Axis No.	A172SHCPUN Device No.	A171SHCPUN Device No.	Signal name
1	D500 to D505	D500 to D505	
2	D506 to D511	D506 to D511	
3	D512 to D517	D512 to D517	
4	D518 to D523	D518 to D523	
5	D524 to D529		
6	D530 to D535		
7	D536 to D541		
8	D542 to D547		
	D548 to D559	D524 to D559	Unusable

	Signal name	Refresh cycle	Fetch cycle	Unit	Signal direction
0	Override ratio setting register		3.5ms	%	SCPU → PCPU
1	Unusable		–	–	
2	Unusable			–	
3	Unusable			–	
4	Unusable			–	
5	Unusable			–	

• Control change register

Axis No.	A172SHCPUN Device No.	A171SHCPUN Device No.	Signal name
1	D960 to D965	D960 to D965	
2	D966 to D971	D966 to D971	
3	D972 To D977	D972 To D977	
4	D78 to D983	D78 to D983	
5	D984 to D989		
6	D990 to D995		
7	D996 to D1001		
8	D1002 to D1007		

	Signal name	Refresh cycle	Fetch cycle	Unit	Signal direction
0	Unusable			–	SCPU → PCPU
1	Unusable			–	
2, 3	Speed change flag		At DSFLP execution	Command unit	
4, 5	JOG speed setting register *1		At start	Command unit	

(*1) □ indicates the backup register.

(3) Control change register

• Control change register

Axis No.	A273UHCPU (32 axis feature)/ A173UHCPU(S1) 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			Fetch cycle			Unit	Signal direction
				Set No. of axis			Set No. of axis				
		SV43	A173UHCPU	1 to 12	13 to 24	25 to32	1 to 12	13 to 24	25 to32		
			A273UHCPU	1 to 8	9 to 18	19 to 32	1 to 8	9 to 18	19 to 32		
0 1	JOG speed setting register						At start			Command unit	SCPU → PCPU

(3) Control change register

• Control change register for SV43

Axis No.	A273UHCPU (32 axis feature)/ A173UHCPU(S1) Device No.	Signal name
1	D1440 to D1445	
2	D1446 to D1451	
3	D1452 to D1457	
4	D1458 to D1463	
5	D1464 to D1469	
6	D1470 to D1475	
7	D1476 to D1481	
8	D1482 to D1487	
9	D1488 to D1493	
10	D1494 to D1499	
11	D1500 to D1505	
12	D1506 to D1511	
13	D1512 to D1517	
14	D1518 to D1523	
15	D1524 to D1529	
16	D1530 to D1535	
17	D1536 to D1541	
18	D1542 to D1547	
19	D1548 to D1553	
20	D1554 to D1559	
21	D1560 to D1565	
22	D1566 to D1571	
23	D1572 to D1577	
24	D1578 to D1583	
25	D1584 to D1589	
26	D1590 to D1595	
27	D1596 to D1601	
28	D1602 to D1607	
29	D1608 to D1613	
30	D1614 to D1619	
31	D1620 to D1625	
32	D1626 to D1631	

	Signal name		Refresh cycle			Fetch cycle			Unit	Signal direction
			Set No. of axis			Set No. of axis				
	SV43	A173UHCPU	1 to 12	13 to 24	25 to32	1 to 12	13 to 24	25 to32		
		A273UHCPU	1 to 8	9 to 18	19 to 32	1 to 8	9 to 18	19 to 32		
0	Override ratio setting register					3.5ms	7.1ms	14.2ms	%	SCPU → PCPU
1	Unusable					–			–	
2	Unusable					–			–	
3	Unusable					–			–	
4	Unusable					–			–	
5	Unusable					–			–	

(4) Common devices

A172SHCPUN

Device No.	Signal Name		Fetch Cycle	Refresh Cycle	Signal Direction
D1008	Limit switch output disable setting register (4 points)		3.5ms		SCPU →PCPU
D1009					
D1010					
D1011					
D1012	Setting Register for a axis number controlled with manual pulse generator 1		Manual pulse generator operation enabled		
D1013	Unusable (2 points)		–	–	–
D1014					
D1015	JOG operation simultaneous start axis setting register		At driving		SCPU →PCPU
D1016	Axis 1	1 pulse input modification setting register for manual pulse generators (8 points)	Manual pulse generator operation enabled		
D1017	Axis 2				
D1018	Axis 3				
D1019	Axis 4				
D1020	Axis 5				
D1021	Axis 6				
D1022	Axis 7				
D1023	Axis 8				

A171SHCPUN

Device No.	Signal Name		Fetch Cycle	Refresh Cycle	Signal Direction
D1008	Limit switch output disable setting register (4 points)		3.5ms		SCPU →PCPU
D1009					
D1010					
D1011					
D1012	Setting Register for a axis number controlled with manual pulse generator 1		Manual pulse generator operation enabled		
D1013	Unusable (2 points)		–	–	–
D1014					
D1015	JOG operation simultaneous start axis setting register		At driving		SCPU →PCPU
D1016	Axis 1	1 pulse input modification setting register for manual pulse generator (4 points)	Manual pulse generator operation enabled		
D1017	Axis 2				
D1018	Axis 3				
D1019	Axis 4				
D1020	Unusable (4 points)		–	–	–
D1021					
D1022					
D1023					

(4) Common devices

A273UHCPU (32 axis feature) / A173UHCPU (S1)

Device No.	Signal name		Refresh cycle			Fetch cycle			Signal direction
			Set No. of axis			Set No. of axis			
	SV43	A173UHCPU	1 to 12	13 to 24	25 to 32	1 to 12	13 to 24	25 to 32	
		A273UHCPU	1 to 8	9 to 18	19 to 32	1 to 8	9 to 18	19 to 32	
D704	Unusable (6 points)		–			–			–
D705									
D706									
D707									
D708									
D709									
D710	JOG simultaneous start axis setting register					At start			SCPU → PCPU
D711									
D712									
D713									
D714	Manual pulse generator 1 axis No. setting register					When manual pulse generator enable			
D715									
D716	Manual pulse generator 2 axis No. setting register								
D717									
D718	Manual pulse generator 3 axis No. setting register								
D719									
D720	Axis 1	Manual pulse generator 1-pulse input magnification setting register							
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								
D752	Manual pulse generator 1 smoothing magnification setting register								
D753	Manual pulse generator 2 smoothing magnification setting register								
D754	Manual pulse generator 3 smoothing magnification setting register								
D755	Unusable (5 points)		–			–			–
D756									
D757									
D758									
D759									
D760	Limit switch output disable setting register					3.5ms	7.1ms	14.2ms	SCPU → PCPU
D761									
D762									
D763									
D764									
D765									
D766									
D767									
D768									
D769									
D770									
D771									
D772									
D773									
D774									
D775									
D776	Limit switch output status storage register								
D777									
D778									
D779									
D780									
D781									
D782									
D783									
D784									
D785									
D786									
D787									
D788									
D789									
D790									
D791									
D792	Servo amplifier type		At power ON						
D793									
D794									
D795									
D796									
D797									
D798									
D799									

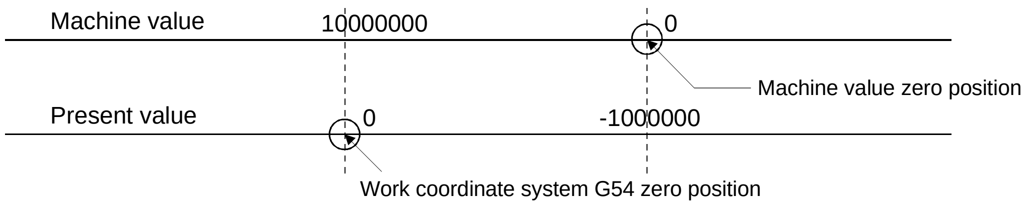
3.2.1 Axis monitor devices

(1) Monitor data areas (D600 to D759, D800 to D959, D800 to D1439, D0 to D639)
....Data from PCPU to SCPU

The monitor data areas are used by the PCPU to store data such as the present value, actual machine value and deviation counter's droop pulse value during positioning control.
They can be used to check the positioning control status in the sequence program.
The user cannot write data into the monitor data areas.
For the delay time from when a positioning device (input, internal relay, special relay) turns ON/OFF until data is stored into the monitor data area, refer to Appendix 6 Processing Time List.

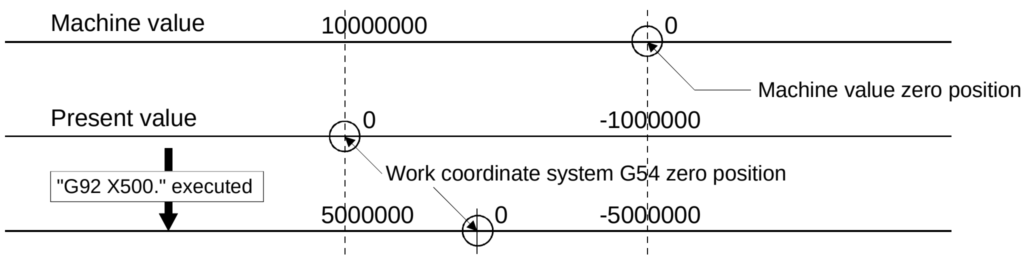
(a) Present value...Data from PCPU to SCPU

1) This register stores the address in the work coordinate system (G54 to G59) specified in the motion program.
This value is stored on the assumption that 0.0001mm is equal to 1. (1mm = 10000)
The following assumes that the setting from the peripheral device is G54=1000.



At the 10000000 position of the machine value, the present value is 0.

2) The present value shifts depending on the work coordinate system selection (G54 to G59) and G92 (coordinate system setting).
When "G90 G00 X0.;" (G54 selected) and "G92 X500." are executed in the above status, the present value is as follows.



The 0 position of the present value is re-set to 500., which results in the present value of 5000000.

(b) Execution sequence No. (main) storage register ...Data from PCPU to SCPU
This register stores the N No. (sequence No.) of the main sequence being executed.
This number changes to zero at a motion program start.
The following data are the changes of the execution motion program No., execution sequence No. and execution block No.

Program		Execution motion program No.	Execution sequence No.	Execution block No.
0001;		1	0	0
	G00 X100.;	1	0	1
	X200.;	1	0	2
N100	Y100.;	1	100	0
	Z100.;	1	100	1
	X300.;	1	100	2
N200	G01 X350. F100.;	1	200	0
	Y200. Z200;	1	200	1
	M10;	1	200	2
	M02;	1	200	3
	%	1	200	3

(c) Execution block No. (main) storage register...........Data from PCPU to SCPU
This register stores the block No. being executed.
This number changes to zero when the motion program is started by the DSFRP/SVST instruction.
This number changes to zero when the sequence No. (N****) described in the motion program is executed, and is incremented every time a single block is executed. (Be careful when executing the IF-THEN-ELSE-END or WHILE-DO instruction. For details, refer to Sections 6.11.2 and 6.11.3.)

(d) Execution program No. (sub) storage register ....Data from PCPU to SCPU
1) This register stores the 0 No. of the subprogram started by "M98" (subprogram call).
2) When a subprogram is called from a subprogram, this number changes to the 0 No. of the subprogram called.
When the subprogram is ended by "M99", this number changes to the 0 No. of the subprogram which called.
3) This number changes to 0 when the motion program is started by the DSFRP/SVST instruction.

(e) Execution sequence No. (sub) storage register .....Data from PCPU to SCPU
1) This register stores the 0 No. of the subprogram started by "M98" (subprogram call).
2) When a subprogram is called from a subprogram, this number changes to the 0 No. of the subprogram called.
When the subprogram is ended by "M99", this number changes to the 0 No. of the subprogram which called.
3) This number changes to 0 when the motion program is started by the DSFRP/SVST instruction.

(f) Execution block No. (sub) storage register ....Data from PCPU to SCPU
1) This register stores the block No. of the subprogram started by "M98" (subprogram call).
2) When a subprogram is called from a subprogram, this number changes to the block No. of the subprogram called.
When the subprogram is ended by "M99", this number changes to the block No. of the subprogram which called.
3) This number changes to 0 when the motion program is started by the DSFRP/SVST instruction.

(g) G43/G44 instruction storage register......................Data from PCPU to SCPU

1) Any of the following values is stored when the tool length offset (G43, G44) or tool length offset cancel (G49) set in the motion program is executed.
   - For G43 43
   - For G44 44
   - For G49 0
2) This value defaults to 0.

(h) Tool length offset data No ....Data from PCPU to SCPU

1) When the tool length offset (G43, G44) command is given, this register stores the preset tool length offset data No.

   [Example] When the X axis is assigned to axis 3

"G43 X100. H20;" is executed.

   ↓

   20 is stored into D649.

2) This value defaults to 0.

(i) Tool length offset

1) This register stores the offset value specified in the tool length offset data No.
2) When the tool length offset (G43, G44) command is given, the contents of the corresponding data registers (D560 to D599: offset value) are stored into the tool length offset area according to the preset tool length offset data No.

   [Example] When the X axis is assigned to axis 3

   D560 = 50000 (H1 = 5.0000mm)

"G43 X50. H1;" is executed.

   ↓

   50000 is stored into D610 and D611.

"G49 X50.;" is executed.

   ↓

   0 is stored into D610 and D611.

(j) Machine value storage register...............................Data from PCPU to SCPU

The machine value represents the address in the mechanical coordinate system determined by a home position return.
This value remains unchanged if "G92" and work coordinate system (G54 to G59) are executed.
This value is used to process the stroke limit range and limit switch output.

(k) Actual machine value..Data from PCPU to SCPU

1) This register stores the actual motor position (machine value - deviation counter value).
2) In a stop status, the machine value is equal to the actual machine value. (At a motor stop, the servo lock force of the motor causes the actual machine value to vary slightly.)

(l) Deviation counter value (droop pulses) ....Data from PCPU to SCPU

This register stores the difference between the machine value and actual machine value.

(m) Minor error code ....Data from PCPU to SCPU
1) This register stores the corresponding error code at occurrence of a minor error.
If another minor error occurs after the storage of the error code, the old error code is overwritten by a new error code.
2) Use the error reset (M1807+20n) to clear the minor error code.

(n) Major error code ....Data from PCPU to SCPU
1) This register stores the corresponding error code at occurrence of a major error.
If another major error occurs after the storage of the error code, the old error code is overwritten by a new error code.
2) Use the error reset (M1807+20n) to clear the major error code.

(o) Servo error code...Data from PCPU to SCPU
1) This register stores the corresponding error code at occurrence of a servo error.
If another servo error occurs after the storage of the error code, the old error code is overwritten by a new error code.
2) Use the servo error reset (M1808+20n) to clear the servo error code.

(p) After near-zero point dog ON travel storage register
....Data from PCPU to SCPU
This register stores the distance (unsigned) traveled from when the near-zero point dog turns ON after start of home position return until completion of home position return.

(q) Home position return second travel storage register
....Data from PCPU to SCPU
If the position where the axis has stopped as specified in the travel setting after near-zero point dog ON by the peripheral device is not the zero point, the axis is moved to the zero point in the second travel.
At this time, this register stores the distance (signed) traveled by the axis up to the zero point in the second travel.
(In the data setting type, the data remains unchanged from the previous value.)

(r) Execution program No. (main) storage register ......Data from PCPU to SCPU
1) When the SVST instruction is executed, this register stores the 0 No. (motion program No.) of the main program being run.
The 0 No. of the subprogram started by "M98" (subprogram call) is stored into another register.
2) When JOG operation, manual pulse generator operation or home position return operation is performed, the corresponding value is stored as follows.
   - JOG operation .... FFFFH
   - Manual pulse generator operation......... FFFEH
   - Home position return operation............. FFFCH
   - At power-on FF00H
3) FFFDH is stored while the following items are executed in the test mode using peripheral device.
   - Home position return is made.
   - Position loop gain or position control gain 1 check is executed in servo diagnostics.

(s) M code storage register..Data from PCPU to SCPU
1) The M code set in the motion program is stored at the start of executing that block.
This value is "0" if the M code is not set in the motion program.
2) The preceding value remains until the M code is executed next.

(t) Torque limit value storage register ....Data from PCPU to SCPU
This register stores the torque limit value commanded to the servo.
300% is stored at power-on of the servo or on the leading edge of PC ready (M2000).

(u) STOP input-time actual machine value storage register
....Data from PCPU to SCPU
This area stores the actual machine value at input of the external "STOP" signal.

3.2.2 Control change registers

(1) Control changing data storage areas (D500 to D559, D960 to D1007, D1440 to D1631, D640 to D703) ....Data from SCPU to PCPU
The control changing data storage areas are used to store the override ratio setting data, speed change data and JOG operation speed data.

(a) Override ratio setting register
1) This register is used to set the override ratio of 0 to 100% in 1% increments to the command speed in the motion program.
2) The actual feed rate is the result of multiplying the command speed in the motion program by the override ratio of 0 to 100% in 1% increments.
3) Refer to Section 7.10 for details of override ratio setting.

(b) Speed change register
1) When the speed of the operating axis is changed, this register stores a new speed.
2) The ranges of setting made to the speed change register are indicated below.

Item \ Unit	mm		inch		degree	
	Setting range	Unit	Setting range	Unit	Setting range	Unit
New speed value	0 to 600000000	$\times 10^{-2}$mm/min	0 to 600000000	$\times 10^{-3}$inch/min	0 to 2147483647	$\times 10^{-3}$degree/min

3) Execution of the positioning control change instruction (DSFLP) causes the value set in the speed change register to be used as the positioning speed.
4) Refer to Section 7.7 for details of speed changing.

(c) JOG speed setting register
1) This register stores the JOG speed for JOG operation.
2) The setting ranges of the JOG speed are indicated below.

Item \ Unit	mm		inch		degree	
	Setting range	Unit	Setting range	Unit	Setting range	Unit
JOG speed	1 to 600000000	$\times 10^{-2}$mm/min	0 to 600000000	$\times 10^{-3}$inch/min	0 to 2147483647	$\times 10^{-3}$degree/min

3) The JOG speed is the value stored in the JOG speed setting register on the leading edge (OFF to ON) of the JOG start signal.
The JOG speed cannot be changed if the data is changed during JOG operation.
4) Refer to Section 7.8 for details of JOG operation.

3.2.3 Tool length offset data

(1) Tool length offset data setting registers (D560 to D599/D1650 to D1689)
....Data from SCPU to PCPU

(a) These registers are used to set the tool length offset values.

(b) The tool length offset data No. can be set within the range H1 to H20.
Tool length offset data setting registers

Tool Length Offset Data No.	Corresponding Registers A172SHCPUN/A171SHCPUN		Corresponding Registers A273UHCPU (32 axis feature) / A173UHCPU (S1)	
	Upper	Lower	Upper	Lower
H1	D561	D560	D1651	D1650
H2	D563	D562	D1653	D1652
H3	D565	D564	D1655	D1654
H4	D567	D566	D1657	D1656
H5	D569	D568	D1659	D1658
H6	D571	D570	D1661	D1660
H7	D573	D572	D1663	D1662
H8	D575	D574	D1665	D1664
H9	D577	D576	D1667	D1666
H10	D579	D578	D1669	D1668
H11	D581	D580	D1671	D1670
H12	D583	D582	D1673	D1672
H13	D585	D584	D1675	D1674
H14	D587	D586	D1677	D1676
H15	D589	D588	D1679	D1678
H16	D591	D590	D1681	D1680
H17	D593	D592	D1683	D1682
H18	D595	D594	D1685	D1684
H19	D597	D596	D1687	D1686
H20	D599	D598	D1689	D1688

(c) The setting ranges of the tool length offset data are indicated below.

Item / Unit	mm		degree	
	Setting range	Unit	Setting range	Unit
Tool compensation (H1 to H20)	-999.9999 to 999.9999	mm	-359.99999 to 359.99999	degree

(d) Refer to Sections 6.8.16 and 6.8.17 for the tool length offset details.

3.2.4 Common device

3.2.4.1 A172SHCPUN/A171SHCPUN

(1) Limit switch output disable setting register (D1008 to D1011).......... Data from SCPU to PCPU
   (a) This is a register for disabling the external output of limit switch output in 1 point units. If a bit is set to "1", the output of the corresponding limit switch is disabled, then the external output goes OFF.

<A172SHCPUN>

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
D1008	LY0F	LY0E	LY0D	LY0C	LY0B	LY0A	LY09	LY08	LY07	LY06	LY05	LY04	LY03	LY02	LY01	LY00
	For axis 2								For axis 1							
D1009	LY1F	LY1E	LY1D	LY1C	LY1B	LY1A	LY19	LY18	LY17	LY16	LY15	LY14	LY13	LY12	LY11	LY10
	For axis 4								For axis 3							
D1010	LY2F	LY2E	LY2D	LY2C	LY2B	LY2A	LY29	LY28	LY27	LY26	LY25	LY24	LY23	LY22	LY21	LY20
	For axis 6								For axis 5							
D1011	LY3F	LY3E	LY3D	LY3C	LY3B	LY3A	LY39	LY38	LY37	LY36	LY35	LY34	LY33	LY32	LY31	LY30
	For axis 8								For axis 7							

1) "1" or "0" is stored for each bit.
   1: Disable ••••• The limit switch output status is OFF.
   0: Enable ••••• The limit switch output comes ON and goes OFF in accordance with the set data.
2) LY of LY00 to LY3F shows the limit switch output.

<A171SHCPUN>

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
D1008	LY0F	LY0E	LY0D	LY0C	LY0B	LY0A	LY09	LY08	LY07	LY06	LY05	LY04	LY03	LY02	LY01	LY00
	For axis 2								For axis 1							
D1009	LY1F	LY1E	LY1D	LY1C	LY1B	LY1A	LY19	LY18	LY17	LY16	LY15	LY14	LY13	LY12	LY11	LY10
	For axis 4								For axis 3							

1) "1" or "0" is stored for each bit.
   1: Disable ••••• The limit switch output status is OFF.
   0: Enable ••••• The limit switch output comes ON and goes OFF in accordance with the set data.
2) LY of LY00 to LY1F shows the limit switch output.

(2) Registers for setting axis numbers controlled by manual pulse generators (D1012) ......Data from SCPU to PCPU

(a) These registers store the axis numbers controlled by manual pulse generators.

	b15 – b12	b11 – b8	b7 – b4	b3 – b0
P1 D1012				
		3 digits	2 digits	1 digit

With a maximum of 3 decimal digits, set the controlled axes for each digit.
A172SHCPUN Axis 1 to 8
A171SHCPUN Axis 1 to 4

(b) For details on manual pulse generator operation, see Section 7.9.

(3) JOG operation simultaneous start axis setting register (D1015) ....Data from SCPU to PCPU

(a) This register is used to set the axis numbers of axes on which JOG operation is to be executed, and the direction of motion.

<A172SHCPUN>

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
D1015	Axis 8	Axis 7	Axis 6	Axis 5	Axis 4	Axis 3	Axis 2	Axis 1	Axis 8	Axis 7	Axis 6	Axis 5	Axis 4	Axis 3	Axis 2	Axis 1
	Axes started in reverse JOG operation								Axes started in forward JOG operation							

*The possible settings for each axis moved in a simultaneous start JOG operation are "1" to "0".
1: Simultaneous start executed
0: Simultaneous start not executed

<A171SHCPUN>

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
D1015					Axis 4	Axis 3	Axis 2	Axis 1					Axis 4	Axis 3	Axis 2	Axis 1
	Axes started in reverse JOG operation								Axes started in forward JOG operation							

*The possible settings for each axis moved in a simultaneous start JOG operation are "1" to "0".
1: Simultaneous start executed
0: Simultaneous start not executed

(b) For details on simultaneous starting in JOG operation, see Section 7.8.2.

(4) 1 pulse input magnification setting registers for manual pulse generators (D1016 to D1023)...Data from SCPU to PCPU

(a) This register is used to set the magnification (from 1 to 100) per pulse for the number of input pulses from a manual pulse generator in manual pulse generator operation.

<A172SHCPUN>

1-pulse Input Magnification Setting Register	Corresponding Axis No.	Setting Range
D1016	Axis 1	1 to 100
D1017	Axis 2	
D1018	Axis 3	
D1019	Axis 4	
D1020	Axis 5	
D1021	Axis 6	
D1022	Axis 7	
D1023	Axis 8	

<A171SHCPUN>

1-pulse Input Magnification Setting Register	Corresponding Axis No.	Setting Range
D1016	Axis 1	1 to 100
D1017	Axis 2	
D1018	Axis 3	
D1019	Axis 4	

(b) For details on manual pulse generator operation, see Section 7.9.

3.2.4.2 A273UHCPU (32 axis feature)/A173UHCPU(S1)

(1) Jog operation simultaneous start axis setting registers (D710 to D713)
....Data from SCPU to PCPU

(a) These registers are used to set the axis numbers and directions of the axes which are simultaneously started for JOG operation.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	
D710	Axis 16	Axis 15	Axis 14	Axis 13	Axis 12	Axis 11	Axis 10	Axis 9	Axis 8	Axis 7	Axis 6	Axis 5	Axis 4	Axis 3	Axis 2	Axis 1	Forward JOG operation
D711	Axis 32	Axis 31	Axis 30	Axis 29	Axis 28	Axis 27	Axis 26	Axis 25	Axis 24	Axis 23	Axis 22	Axis 21	Axis 20	Axis 19	Axis 18	Axis 17	
D712	Axis 16	Axis 15	Axis 14	Axis 13	Axis 12	Axis 11	Axis 10	Axis 9	Axis 8	Axis 7	Axis 6	Axis 5	Axis 4	Axis 3	Axis 2	Axis 1	Reverse JOG operation
D713	Axis 32	Axis 31	Axis 30	Axis 29	Axis 28	Axis 27	Axis 26	Axis 25	Axis 24	Axis 23	Axis 22	Axis 21	Axis 20	Axis 19	Axis 18	Axis 17	

*The possible settings for each axis moved in a simultaneous start JOG operation are "1" to "0".
1: Simultaneous start executed
0: Simultaneous start not executed

(b) Refer to Section 7.19.3 for details of simultaneous start of JOG operation.

(2) Manual pulse generator-controlled axis No. setting registers (D714 to D719)
....Data from SCPU to PCPU

(a) These registers are used to store the axis numbers controlled by the manual pulse generators.

		b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
P1	D714	Axis 16	Axis 15	Axis 14	Axis 13	Axis 12	Axis 11	Axis 10	Axis 9	Axis 8	Axis 7	Axis 6	Axis 5	Axis 4	Axis 3	Axis 2	Axis 1
	D715	Axis 32	Axis 31	Axis 30	Axis 29	Axis 28	Axis 27	Axis 26	Axis 25	Axis 24	Axis 23	Axis 22	Axis 21	Axis 20	Axis 19	Axis 18	Axis 17
P2	D716	Axis 16	Axis 15	Axis 14	Axis 13	Axis 12	Axis 11	Axis 10	Axis 9	Axis 8	Axis 7	Axis 6	Axis 5	Axis 4	Axis 3	Axis 2	Axis 1
	D717	Axis 32	Axis 31	Axis 30	Axis 29	Axis 28	Axis 27	Axis 26	Axis 25	Axis 24	Axis 23	Axis 22	Axis 21	Axis 20	Axis 19	Axis 18	Axis 17
P3	D718	Axis 16	Axis 15	Axis 14	Axis 13	Axis 12	Axis 11	Axis 10	Axis 9	Axis 8	Axis 7	Axis 6	Axis 5	Axis 4	Axis 3	Axis 2	Axis 1
	D719	Axis 32	Axis 31	Axis 30	Axis 29	Axis 28	Axis 27	Axis 26	Axis 25	Axis 24	Axis 23	Axis 22	Axis 21	Axis 20	Axis 19	Axis 18	Axis 17

*The possible settings for each axis moved in a simultaneous start JOG operation are "1" to "0".
1: Simultaneous start executed
0: Simultaneous start not executed

(b) Refer to Section 7.20 for details of manual pulse generator operation.

(3) 1 pulse input magnification setting registers for manual pulse generators (D720 to D751) ....Data from SCPU to PCPU

(a) This register is used to set the magnification (from 1 to 100) per pulse for the number of input pulses from a manual pulse generator in manual pulse generator operation.

1-pulse Input Magnification Setting Register	Corresponding Axis No.	Setting Range	1-pulse Input Magnification Setting Register	Corresponding Axis No.	Setting Range
D720	Axis 1	1 to 100	D736	Axis 17	1 to 100
D721	Axis 2		D737	Axis 18	
D722	Axis 3		D738	Axis 19	
D723	Axis 4		D739	Axis 20	
D724	Axis 5		D740	Axis 21	
D725	Axis 6		D741	Axis 22	
D726	Axis 7		D742	Axis 23	
D727	Axis 8		D743	Axis 24	
D728	Axis 9		D744	Axis 24	
D729	Axis 10		D745	Axis 26	
D730	Axis 11		D746	Axis 27	
D731	Axis 12		D747	Axis 28	
D732	Axis 13		D748	Axis 29	
D733	Axis 14		D749	Axis 30	
D734	Axis 15		D750	Axis 31	
D735	Axis 16		D751	Axis 32	

(b) For details on manual pulse generator operation, see Section 7.9.

(4) Manual pulse generator smoothing magnification setting area (D752 to D754) ....Data from SCPU to PCPU

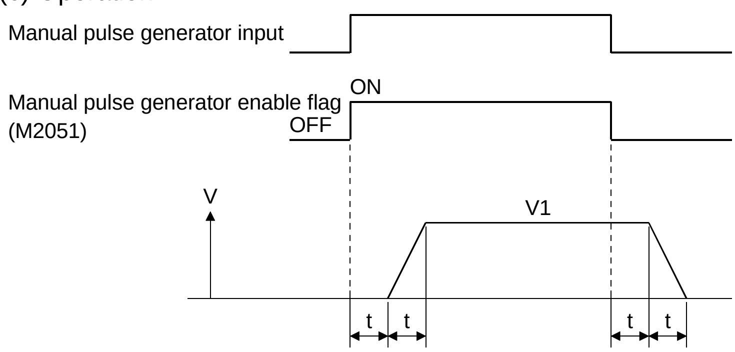
(a) These devices are used to set the smoothing time constants of the manual pulse generators.

Manual Pulse Generator Smoothing Magnification Setting Register	Setting Range
Manual pulse generator 1 (P1): D752	0 to 59
Manual pulse generator 2 (P2): D753	
Manual pulse generator 3 (P3): D754	

(b) By setting the smoothing magnification, the smoothing time constant is as indicated by the following equation.

Smoothing time constant (t) = (smoothing magnification + 1) $\times$ 56.8 [ms]

(c) Operation



Output speed (V1) = (number of input pulses/ms) $\times$ (manual pulse generator 1-pulse input magnification setting)
Travel (L) = (travel per pulse) $\times$ (number of input pulses) $\times$ (manual pulse generator 1-pulse input magnification setting)

REMARKS

1) The travel per pulse of the manual pulse generator is as follows.

• Setting unit
- mm :0.1μm
- inch :0.00001inch
- degree :0.00001degree
- PULSE :1pulse

2) The smoothing time constant is 56.8ms to 3408ms.

(5) Limit switch output disable setting registers (D760 to D775)
....Data from SCPU to PCPU

(a) These registers are used to disable the external outputs of the limit switch outputs on a point by point basis. Set the corresponding bit to 1 to disable the limit switch output and turn OFF the external output.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
D760	LY0F	LY0E	LY0D	LY0C	LY0B	LY0A	LY09	LY08	LY07	LY06	LY05	LY04	LY03	LY02	LY01	LY00
	For axis 2								For axis 1							
D761	LY1F	LY1E	LY1D	LY1C	LY1B	LY1A	LY19	LY18	LY17	LY16	LY15	LY14	LY13	LY12	LY11	LY10
	For axis 4								For axis 3							
D762	LY2F	LY2E	LY2D	LY2C	LY2B	LY2A	LY29	LY28	LY27	LY26	LY25	LY24	LY23	LY22	LY21	LY20
	For axis 6								For axis 5							
D763	LY3F	LY3E	LY3D	LY3C	LY3B	LY3A	LY39	LY38	LY37	LY36	LY35	LY34	LY33	LY32	LY31	LY30
	For axis 8								For axis 7							
D764	LY4F	LY4E	LY4D	LY4C	LY4B	LY4A	LY49	LY48	LY47	LY46	LY45	LY44	LY43	LY42	LY41	LY40
	For axis 10								For axis 9							
D765	LY5F	LY5E	LY5D	LY5C	LY5B	LY5A	LY59	LY58	LY57	LY56	LY55	LY54	LY53	LY52	LY51	LY50
	For axis 12								For axis 11							
D766	LY6F	LY6E	LY6D	LY6C	LY6B	LY6A	LY69	LY68	LY67	LY66	LY65	LY64	LY63	LY62	LY61	LY60
	For axis 14								For axis 13							
D767	LY7F	LY7E	LY7D	LY7C	LY7B	LY7A	LY79	LY78	LY77	LY76	LY75	LY74	LY73	LY72	LY71	LY70
	For axis 16								For axis 15							
D768	LY8F	LY8E	LY8D	LY8C	LY8B	LY8A	LY89	LY88	LY87	LY86	LY85	LY84	LY83	LY82	LY81	LY80
	For axis 18								For axis 17							
D769	LY9F	LY9E	LY9D	LY9C	LY9B	LY9A	LY99	LY98	LY97	LY96	LY95	LY94	LY93	LY92	LY91	LY90
	For axis 20								For axis 19							
D770	LYAF	LYAE	LYAD	LYAC	LYAB	LYAA	LYA9	LYA8	LYA7	LYA6	LYA5	LYA4	LYA3	LYA2	LYA1	LYA0
	For axis 22								For axis 21							
D771	LYBF	LYBE	LYBD	LYBC	LYBB	LYBA	LYB9	LYB8	LYB7	LYB6	LYB5	LYB4	LYB3	LYB2	LYB1	LYB0
	For axis 24								For axis 23							
D772	LYCF	LYCE	LYCD	LYCC	LYCB	LYCA	LYC9	LYC8	LYC7	LYC6	LYC5	LYC4	LYC3	LYC2	LYC1	LYC0
	For axis 26								For axis 25							
D773	LYDF	LYDE	LYDD	LYDC	LYDB	LYDA	LYD9	LYD8	LYD7	LYD6	LYD5	LYD4	LYD3	LYD2	LYD1	LYD0
	For axis 28								For axis 27							
D774	LYEF	LYEE	LYED	LYEC	LYEB	LYEA	LYE9	LYE8	LYE7	LYE6	LYE5	LYE4	LYE3	LYE2	LYE1	LYE0
	For axis 30								For axis 29							
D775	LYFF	LYFE	LYFD	LYFC	LYFB	LYFA	LYF9	LYF8	LYF7	LYF6	LYF5	LYF4	LYF3	LYF2	LYF1	LYF0
	For axis 32								For axis 31							

1) Specify 1 or 0 to set each bit.
   1: Disable ..... Limit switch output remains OFF.
   0: Enable ...... Limit switch output turns ON/OFF based on set data.
2) "LY" in LY00 to LYFF indicates limit switch output.

(6) Limit switch output status storage registers (D776 to D791)
....Data from PCPU to SCPU

(a) The output states (ON/OFF) of the limit switch outputs set on the peripheral device and output to the AY42 are stored in terms of 1 and 0.
- ON 1
- OFF.......................................0

(b) These registers can be used to export the limit switch output data in the sequence program, for example.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
D776	LY0F	LY0E	LY0D	LY0C	LY0B	LY0A	LY09	LY08	LY07	LY06	LY05	LY04	LY03	LY02	LY01	LY00
	Axis 2								Axis 1							
D777	LY1F	LY1E	LY1D	LY1C	LY1B	LY1A	LY19	LY18	LY17	LY16	LY15	LY14	LY13	LY12	LY11	LY10
	Axis 4								Axis 3							
D778	LY2F	LY2E	LY2D	LY2C	LY2B	LY2A	LY29	LY28	LY27	LY26	LY25	LY24	LY23	LY22	LY21	LY20
	Axis 6								Axis 5							
D779	LY3F	LY3E	LY3D	LY3C	LY3B	LY3A	LY39	LY38	LY37	LY36	LY35	LY34	LY33	LY32	LY31	LY30
	Axis 8								Axis 7							
D780	LY4F	LY4E	LY4D	LY4C	LY4B	LY4A	LY49	LY48	LY47	LY46	LY45	LY44	LY43	LY42	LY41	LY40
	Axis 10								Axis 9							
D781	LY5F	LY5E	LY5D	LY5C	LY5B	LY5A	LY59	LY58	LY57	LY56	LY55	LY54	LY53	LY52	LY51	LY50
	Axis 12								Axis 11							
D782	LY6F	LY6E	LY6D	LY6C	LY6B	LY6A	LY69	LY68	LY67	LY66	LY65	LY64	LY63	LY62	LY61	LY60
	Axis 14								Axis 13							
D783	LY7F	LY7E	LY7D	LY7C	LY7B	LY7A	LY79	LY78	LY77	LY76	LY75	LY74	LY73	LY72	LY71	LY70
	Axis 16								Axis 15							
D784	LY8F	LY8E	LY8D	LY8C	LY8B	LY8A	LY89	LY88	LY87	LY86	LY85	LY84	LY83	LY82	LY81	LY80
	Axis 18								Axis 17							
D785	LY9F	LY9E	LY9D	LY9C	LY9B	LY9A	LY99	LY98	LY97	LY96	LY95	LY94	LY93	LY92	LY91	LY90
	Axis 20								Axis 19							
D786	LYAF	LYAE	LYAD	LYAC	LYAB	LYAA	LYA9	LYA8	LYA7	LYA6	LYA5	LYA4	LYA3	LYA2	LYA1	LYA0
	Axis 22								Axis 21							
D787	LYBF	LYBE	LYBD	LYBC	LYBB	LYBA	LYB9	LYB8	LYB7	LYB6	LYB5	LYB4	LYB3	LYB2	LYB1	LYB0
	Axis 24								Axis 23							
D788	LYCF	LYCE	LYCD	LYCC	LYCB	LYCA	LYC9	LYC8	LYC7	LYC6	LYC5	LYC4	LYC3	LYC2	LYC1	LYC0
	Axis 26								Axis 25							
D789	LYDF	LYDE	LYDD	LYDC	LYDB	LYDA	LYD9	LYD8	LYD7	LYD6	LYD5	LYD4	LYD3	LYD2	LYD1	LYD0
	Axis 28								Axis 27							
D790	LYEF	LYEE	LYED	LYEC	LYEB	LYEA	LYE9	LYE8	LYE7	LYE6	LYE5	LYE4	LYE3	LYE2	LYE1	LYE0
	Axis 30								Axis 29							
D791	LYFF	LYFE	LYFD	LYFC	LYFB	LYFA	LYF9	LYF8	LYF7	LYF6	LYF5	LYF4	LYF3	LYF2	LYF1	LYF0
	Axis 32								Axis 31							

"1" or "0" is set at each bit of D776 to D791.
・ON ・・・1
・OFF・・・0

REMARK

LY in LY□□ of D776 to D791 indicates limit switch output.

(7) Servo amplifier type (D792 to D799) .... Data from PCPU to SCPU
The servo amplifier types set in system settings are stored when the servo system CPU control power supply (A6□P) is switched on or reset.

	b15 to b12	b11 to b8	b7 to b4	b3 to b0
D792	Axis 4	Axis 3	Axis 2	Axis 1
D793	Axis 8	Axis 7	Axis 6	Axis 5
D794	Axis 12	Axis 11	Axis 10	Axis 9
D795	Axis 16	Axis 15	Axis 14	Axis 13
D796	Axis 20	Axis 19	Axis 18	Axis 17
D797	Axis 24	Axis 23	Axis 22	Axis 21
D798	Axis 28	Axis 27	Axis 26	Axis 25
D799	Axis 32	Axis 31	Axis 30	Axis 29

→ Servo amplifier type
・0・・・Unused axis
・1・・・ADU (Main base)
・2・・・MR-□-B
・3・・・ADU (Motion extension base)

3.3 Special Relays (SP.M)

The servo system CPU has 256 special relay points from M9000 to M9255.
Of there, the 7 points from M9073 to M9079 are used for positioning control, and their applications are indicated in Table 3.1.

Table 3.1 Special Relays

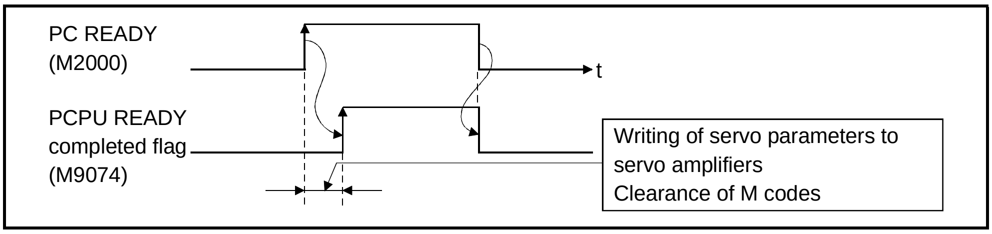
Device No.	Signal Name	Fetch Cycle	Refresh Cycle	Signal Direction
M9073	PCPU WDT error flag		END	PCPU → SCPU
M9074	PCPU REDAY-completed flag			
M9075	In-test-mode flag			
M9076	External emergency stop input flag			
M9077	Manual pulse generator axis setting error flag			
M9078	Test mode request error flag			
M9079	Servo program setting error flag			

*"END" in Refresh Cycle indicates a longer one of "80ms" and "sequence program scan time".

(1) PCPU WDT error flag (M9073).....................Signal sent from PCPU to SCPU
This flag comes ON when a "watchdog timer error" is detected by the PCPU's self-diagnosis function.
When the PCPU detects a WDT error, it executes an immediate stop without deceleration on the driven axis.
When the WDT error flag has come ON, reset the servo system CPU with the key switch.
If M9073 remains ON after resetting, there is a fault at the PCPU side.
The error cause is stored in the PCPU error cause storage area (D9184) (see Section 3.4 (2)).

(2) PCPU REDAY-completed flag (M9074)...... Signal sent from PCPU to SCPU
This flag is used to determine whether the PCPU is normal or abnormal from the sequence program.
(a) When the PC READY flag (M2000) turns from OFF to ON, the fixed parameters, servo parameters, limit switch output data, etc., are checked, and if no error is detected the PCPU READY-completed flag comes ON.
The servo parameters are written to the servo amplifiers and the M codes are cleared.

(b) When the PC READY flag (M2000) goes OFF, the PCPU READY-completed flag also goes OFF



(3) In-test-mode(M9075) ......Signal from PCPU to SCPU
   (a) This flag is used to determine whether or not a test mode established from a peripheral device is currently effective. Use it, for example, for an interlock effective when starting a servo program with a DSFRP/SVST instruction in the sequence program.
      - OFF ....When the test mode is not in effect
      - ON ....When the test mode is in effect

   (b) If a test mode request is issued from a peripheral device but the test mode is not established, the test mode request error flag (M9078) comes ON.

(4) External emergency stop input flag (M9076) Signal from PCPU to SCPU
   This flag is used to check the ON or OFF status of external emergency stop signal input at the EMG terminal.
   - OFF......External emergency stop input is ON
   - ON........External emergency stop input is OFF

(5) Manual pulse generator axis setting error flag (M9077) ....Signal sent from PCPU to SCPU
   (a) This flag is used to determine whether the setting in the manual pulse generator axis setting register (D1012/D714 to D719) is normal or abnormal.
      - OFF ....When D1012/D714 to D719 is normal
      - ON ....When D1012/D714 to D719 is abnormal

   (b) When M9077 comes ON, the error contents are stored in the manual pulse generator axis setting error register (D9187).

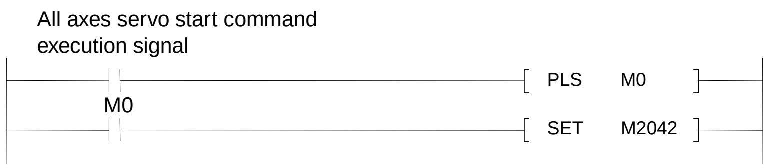
(6) Test mode request error flag (M9078) ......Signal sent from PCPU to SCPU
 (a) This flag comes ON if the test mode is not established when a test mode request is sent from a peripheral device
 (b) When M9078 comes ON, the error contents are stored in the test mode request error register (D9188/D9182, D9183).

POINTS

(1) When an emergency stop signal (EMG) is input during positioning, the feed present value is advanced within the rapid stop deceleration time set in the parameter block. At the same time, the servo OFF status is established because the all axes servo start command (M2042) goes OFF. When the rapid stop deceleration time has elapsed after input of the emergency stop signal, the feed present value returns to the value at the point when the emergency stop was initiated.

(2) If the emergency stop is reset before the emergency stop deceleration time has elapsed, a servo error occurs.

(3) If you do not want to establish the servo ON status immediately after an emergency stop has been reset, include the following section in the sequence program.



(7) Motion program setting error flag (M9079) ...Signal from PCPU to SCPU
This flag is used to determine whether the positioning data of the motion program designated by a DSFRP/SVST instruction is normal or abnormal.
- OFF...... Normal
- ON........ Abnormal

3.4 Special Registers (SP.D)

3.4.1 A172SHCPUN/A171SHCPUN

A servo system CPU has 256 special register points from D9000 to D9255.
Of these, the 20 points from D9180 to D9199 are used for positioning control.
The special registers used for positioning are shown in the table below (for the applications of special registers other than D9180 to D9199, see Appendix 3.2.)

Table 3.2 Special Registers

A172SH CPUN/ A171SH CPUN Device Number	Signal Name	Refresh Cycle	Fetch Cycle	Signal Direction
D9180 D9181 D9182 D9183	Limit switch output status	3.5ms		SCPU←PCPU
D9184	PCPU WDT error cause	At PCPU WDT error occurrence		
D9185 D9186	Servo amplifier type	Power ON		
D9187	Manual pulse generator axis setting error information	Manual pulse generator operation enabled		
D9188	Test mode request error information	Test mode request		
D9189	Error program number	At driving		
D9190	Error item information			
D9191	Servo amplifier loading information	Power ON, 10 ms		
D9192	Manual pulse generator 1 smoothing magnification setting register		Manual pulse generator operation enabled	SCPU→PCPU
D9193 D9194 D9195	Unusable	–	–	–
D9196	PC link communication error code	3.5ms		SCPU←PCPU
D9197 D9198 D9199	Unusable	–	–	–

(1) Limit switch output status storage register (D9180 to D9183) ...... Data from PCPU to SCPU

(a) This register stores the output status (ON/OFF) for limit switch output to AY42 with a peripheral device as "1" or "0".

- ON 1
- OFF 0

(b) This register can be used for purposes such as outputting limit switch output data to external destinations by using the sequence program.

<A172SHCPUN>

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
D9180	LY0F	LY0E	LY0D	LY0C	LY0B	LY0A	LY09	LY08	LY07	LY06	LY05	LY04	LY03	LY02	LY01	LY00
	For axis 2								For axis 1							
D9181	LY1F	LY1E	LY1D	LY1C	LY1B	LY1A	LY19	LY18	LY17	LY16	LY15	LY14	LY13	LY12	LY11	LY10
	For axis 4								For axis 3							
D9182	LY2F	LY2E	LY2D	LY2C	LY2B	LY2A	LY29	LY28	LY27	LY26	LY25	LY24	LY23	LY22	LY21	LY20
	For axis 6								For axis 5							
D9183	LY3F	LY3E	LY3D	LY3C	LY3B	LY3A	LY39	LY38	LY37	LY36	LY35	LY34	LY33	LY32	LY31	LY30
	For axis 8								For axis 7							

*"1" or "0" is set at each bit of D9180 to D9183.
• ON • • • • 1
• OFF • • • 0

REMARK

"LY" in LY□□ of D9180 to D9181 indicates a limit switch output.

<A171SHCPUN>

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
D9180	LY0F	LY0E	LY0D	LY0C	LY0B	LY0A	LY09	LY08	LY07	LY06	LY05	LY04	LY03	LY02	LY01	LY00
	For axis 2								For axis 1							
D9181	LY1F	LY1E	LY1D	LY1C	LY1B	LY1A	LY19	LY18	LY17	LY16	LY15	LY14	LY13	LY12	LY11	LY10
	For axis 4								For axis 3							

*"1" or "0" is set at each bit of D9180 to D9181.
• ON • • • • 1
• OFF • • • 0

REMARK

"LY" in LY□□ of D9180 to D9181 indicates a limit switch output.

(2) PCPU error cause(D9184).......Data from PCPU to SCPU
This register is used to identify the nature of errors occurring in the PCPU part of
the servo system.

Error Code	Error Cause	Operation when Error Occurs	Action to Take
1	PCPU software fault 1	All axes stop immediately, after which operation cannot be started.	Reset with the reset key.
2	PCPU operation synchronization time over		
3	PCPU software fault 2		
30	PCPU/SCPU hardware fault		
200 201	Hardware fault of module loaded on motion main base unit or extension base unit. 2 0 0 Indicates the slot number (0,1) where the module with the fault is loaded. Indicates the stage number of the base on which the module with the fault is loaded. 0: Main base		Reset with the reset key. If the error reoccurs after resetting, the relevant module or the relevant slot (base unit) is probably faulty: replace the module/base unit.
250 251	SSCNET interface hardware fault 2 5 0 Faulty SSCNET No. 0: SSCNET 1 (Amplifier interface) 1: SSCNET 2 (PC link interface)		Exchange the CPU unit.
300	PCPU software fault 3		Reset with the reset key.
302	Data stored in flash ROM is not normal when CPU power is switched on in "ROM operation mode" setting (registered code is unauthorized).	Data in flash ROM is not loaded into built-in SRAM and "ROM operation mode" is not established. After that, a STOP status is set up and a start is not made.	After checking the program parameter of the built-in SRAM, perform "ROM write → ROM operation mode" operation again. If the error recurs, the flash ROM has reached the end of its life. Perform operation in "RAM operation mode" or change the CPU module.

(3) Servo amplifier classification (D9185 to D9186).......Data from PCPU to SCPU
On switching on the power to the servo system CPU or resetting, the servo amplifier type set in the system settings is set in these devices.
(a) A172SHCPUN

	b15 to b12	b11 to b8	b7 to b4	b3 to b0
D9185	Axis 4	Axis 3	Axis 2	Axis 1
D9186	Axis 8	Axis 7	Axis 6	Axis 5

Servo amplifier type
•0 •••• Unused axis
•2 •••• MR-□-B

(b) A171SHCPUN

	b15 to b12	b11 to b8	b7 to b4	b3 to b0
D9185	Axis 4	Axis 3	Axis 2	Axis 1
D9186	0			

Servo amplifier type
•0 •••• Unused axis
•2 •••• MR-□-B

(4) Manual pulse generator axis setting error (D9187).......Data from PCPU to SCPU

When the manual pulse generator axis setting error flag (M9077) turns ON, the definition of the manual pulse generator axis setting error is stored into this register.

(a) A172SHCPUN

	b15	b14	b13	b12	b11	b10	b9	b8		b3		b0
D9187	Axis 8	Axis 7	Axis 6	Axis 5	Axis 4	Axis 3	Axis 2	Axis 1	0	P1	0	P1

- 1 pulse input magnification setting error
  - 0: Normal
  - 1: Setting error (Outside the range 1 to 10000)
- Manual pulse generator axis setting error
  - 0: Normal
  - 1: Setting error (When the axis setting for each digit is outside the range 1 to 8)
- Manual pulse generator smoothing magnification setting error
  - 0: Normal
  - 1: Setting error (Outside the range 0 to 59)

(b) A171SHCPUN

	b15	b11	b10	b9	b8		b3		b0
D9187	0	Axis 4	Axis 3	Axis 2	Axis 1	0	P1	0	P1

- 1 pulse input magnification setting error
  - 0: Normal
  - 1: Setting error (Outside the range 1 to 100)
- Manual pulse generator axis setting error
  - 0: Normal
  - 1: Setting error (When the axis setting for each digit is outside the range 1 to 4)
- Manual pulse generator smoothing magnification setting error
  - 0: Normal
  - 1: Setting error (Outside the range 0 to 59)

(5) Test mode request error (D9188) ......Data from PCPU to SCPU
When the test mode request error flag (M9078) turns ON, the data of the operating axes are stored into this register.

(a) A172SHCPUN

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
D9188	0	0	0	0	0	0	0	0	Axis 8	Axis 7	Axis 6	Axis 5	Axis 4	Axis 3	Axis 2	Axis 1

b7 to b0: Stores the operating/stopped status of each axis.
- 0: Stopped
- 1: Operating

b15 to b8: All set to "0"

(b) A171SHCPUN

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
D9188	0	0	0	0	0	0	0	0	0	0	0	0	Axis 4	Axis 3	Axis 2	Axis 1

b3 to b0: Stores the operating/stopped status of each axis.
- 0: Stopped
- 1: Operating

b15 to b4: All set to "0"

(6) Error program No. (D9189).......Data from PCPU to SCPU
(a) When the motion program setting error flag (M9079) turns on, the motion program No. (1 to 256) in error is stored into this register.

(b) When an error program No. has been stored and an error occurs in another motion program, the new error program No. is stored.

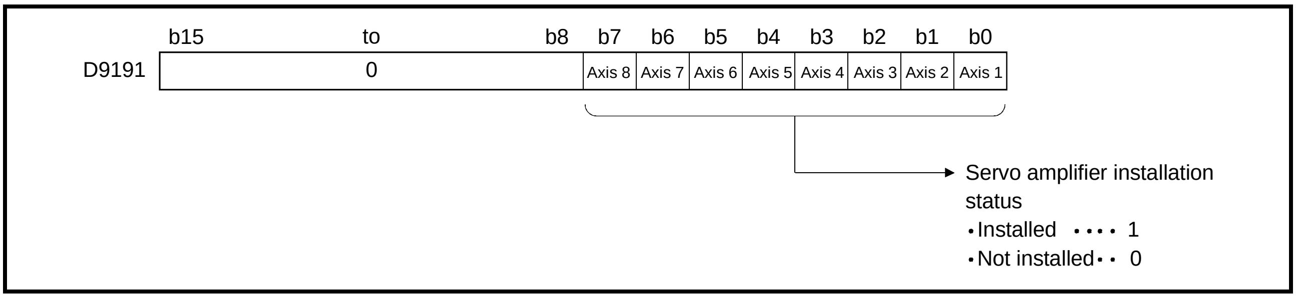
(7) Error item information (D9190) ....Data from PCPU to SCPU
When the motion program setting error flag (M9079) turns on, the error code corresponding to the setting item in error is stored into this register.
The error code No. list is given in Appendix 2.1.

(8) Servo amplifier installation information (D9191) ....Data from PCPU to SCPU
On switching on the control power supply to the servo system CPU or resetting, the servo amplifier installation status is checked and the result is set in this device.

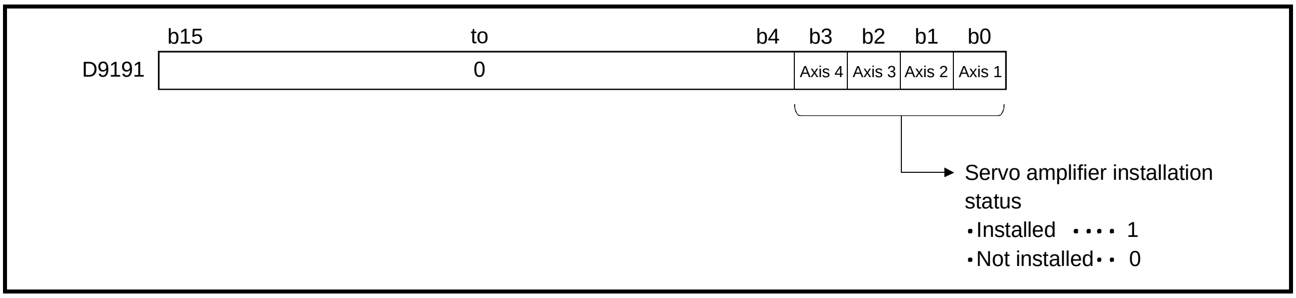
Lower 8 bits ...... Servo amplifier installation status (A172SHCPUN)
Lower 4 bits ...... Servo amplifier installation status (A171SHCPUN)

The "installed" status will be stored for axes for which an amplifier is installed after the power is switched on. However, if the amplifier for an axis is removed, the "installed" status will not change to "not installed".

<A172SHCPUN>



<A171SHCPUN>



(a) Servo amplifier installation status
1) Installed/not installed status
- "installed" status............... The MR-□-B is normal (i.e. communication with the servo amplifier is normal)
- "not installed" status......... No servo amplifier is installed.
The servo amplifier power is OFF.
Normal communication with the servo amplifier is not possible due, for example, to a connecting cable fault.

2) The system settings and servo amplifier installation statuses are indicated below.

System Setting	MR-□-B	
	Installed	Not Installed
Used (axis number setting)	"1" is stored	"0" is stored
Unused	"0" is stored	"0" is stored

(9) Area for setting the smoothing magnification for the manual pulse generator (D9192) ...... Data from SCPU to PCPU

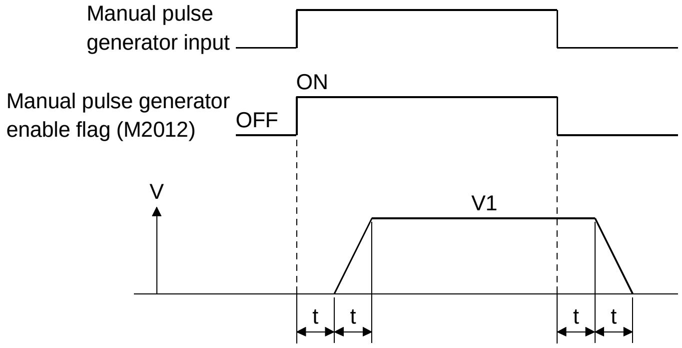
(a) This device stores the manual pulse generator smoothing time constant.

Manual Pulse Generator Smoothing Magnification Setting Register	Setting Range
D9192	0 to 59

(b) When the smoothing magnification is set, the smoothing time constant is determined by the formula given below.

Smoothing time constant (t) = (smoothing magnification + 1) $\times$ 56.8 [ms]

(c) Operation



Output speed (V1) = (Number of input pulses/ms) $\times$ (1 manual pulse generator pulse input magnification setting)

Travel value (L) = (Travel value per pulse) $\times$ Number of input pulses $\times$ (1 manual pulse generator pulse input magnification setting)

REMARKS

1) The travel value per manual pulse generator pulse is set in one of the following units.

• Setting unit
- mm :0.0001mm
- inch :0.00001inch
- degree :0.00001degree

2) The range for the smoothing time constant is 56.8 ms to 3408 ms.

3.4.2 A273UHCPU (32 axis feature)/A173UHCPU(S1)

A servo system CPU has 256 points of special registers from D9000 to D9255. Among these, the 20 points of D9180 to D9199 are used for positioning control. The special registers used for positioning control are listed below. (Refer to Appendix 3.2 for the applications of special registers other than D9180 to D9199.)

Table 3.3 Special Register List

Device No.	Signal name		Refresh cycle			Fetch cycle			Signal direction
			Set number of axes			Set number of axes			
	SV43	A173UHCPU	1 to 12	13 to 24	25 to 32	1 to 12	13 to 24	25 to 32	
		A273UHCPU	1 to 8	9 to 18	19 to 32	1 to 8	9 to 18	19 to 32	
D9180 D9181	Unusable		–			–			–
D9182 D9183	Test mode request error information		When test mode is requested						SCPU ← PCPU
D9184	PCPU WDT error cause		When PCPU WDT error occurs						
D9185 D9186 D9187	Manual pulse generator axis setting error information		When manual pulse generator operation is enabled						
D9188	Unusable		–			–			–
D9189	Error program No.		At start						SCPU ← PCPU
D9190	Error item information								
D9191 D9192	Servo amplifier loading information		At power-on and 10ms	20ms					
D9193 D9194 D9195	Unusable		–			–			–
D9186	Personal computer link communication error code		3.5ms	7.1ms	14.2ms				SCPU ← PCPU
D9187 D9198 D9199	Unusable		–			–			–

(1) Test mode request error information (D9182 to D9183) .... Data from PCPU to SCPU

If there are axes operating at the peripheral device's request for test mode, a test mode request error occurs, the error flag (M9078) turns ON, and the operating/stopping information of each axis is stored.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
D9182	Axis16	Axis15	Axis14	Axis13	Axis12	Axis11	Axis10	Axis9	Axis8	Axis7	Axis6	Axis5	Axis4	Axis3	Axis2	Axis1
D9183	Axis32	Axis31	Axis30	Axis29	Axis28	Axis27	Axis26	Axis25	Axis24	Axis23	Axis22	Axis21	Axis20	Axis19	Axis18	Axis17

Stores the operating/stopped status of each axis
- 0: Stopped
- 1: Operating

(2) PCPU error cause (D9184) ....Data from PCPU to SCPU
This register is used to identify the faults of the PCPU section in the sequence program.

<table>
<tr><th>Error Code</th><th>Error Cause</th><th>Operation when Error Occurs</th><th>Action to Take</th></tr>
<tr><td>1</td><td>PCPU software fault 1</td><td rowspan="4">All axes stop immediately, after which operation cannot be started.</td><td rowspan="4">Reset with the reset key.</td></tr>
<tr><td>2</td><td>PCPU operation synchronization time over</td></tr>
<tr><td>3</td><td>PCPU software fault 2</td></tr>
<tr><td>30</td><td>PCPU/SCPU hardware fault</td></tr>
<tr><td>100 to 107
110 to 117
120 to 127
130 to 137
140 to 147</td><td>AC servo motor drive module CPU fault
100
Indicates the slot No.(0 to 7) where the AC motor drive module with the fault is loaded.
Indicates the stage No. of the base on which the AC motor drive module with the fault is loaded.
0: Main base
1: Extension base 1st stage
2: Extension base 2nd stage
3: Extension base 3rd stage
4: Extension base 4th stage</td><td>Servo error detection flag (M2408+20n) of the corresponding axis turns on, resulting in servo OFF status. After that, processing follows the "ADU servo error-time processing setting" in system settings.</td><td>Reset with the reset key. If the error recurs after reset, change the ADU module as it may be faulty.</td></tr>
<tr><td>200 to 207
210 to 217
220 to 227
230 to 237
240 to 247</td><td>Motion main/extension base-loaded module hardware fault
200
Indicates the slot No.(0 to 7) where the module with the fault is loaded.
Indicates the stage No. of the base on which the module with the fault is loaded.
0: Main base
1: Extension base 1st stage
2: Extension base 2nd stage
3: Extension base 3rd stage
4: Extension base 4th stage</td><td rowspan="4">All axes stop immediately, after which operation cannot be started.</td><td rowspan="2">Reset with the reset key. If the error recurs after reset, change the corresponding module or slot (base) as it may be faulty.</td></tr>
<tr><td>250 to 253</td><td>Separated servo amplifier (MR-□-B) interface hardware fault
250
Faulty SSCNET No.
0: SSCNET 1
1: SSCNET 2
2: SSCNET 3
3: SSCNET 4</td></tr>
<tr><td>300</td><td>PCPU software fault 3</td><td>Reset with the reset key.</td></tr>
<tr><td>301</td><td>CPSTART instructions of 8 or more points were given in excess of the number of simultaneously startable programs.<table><tr><th></th><th>Number of Simultaneously Startable Programs</th></tr><tr><td>Conventional function version</td><td>20</td></tr><tr><td>Added function version</td><td>14</td></tr></table></td><td>Reset with the reset key. Reduce the CPSTART instructions of 8 or more points to less than the number of simultaneously startable programs.</td></tr>
</table>

(3) Manual pulse generator axis setting error information (D9185 to D9187) ....Data from PCPU to SCPU

If an error is found after checking of the set data on the leading edge of the manual pulse generator enable signal, the following error information is stored into D9185 to D9187 and the manual pulse generator axis setting error flag (M9077) turns ON.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
D9185	0	0	0	0	0	0	0	0	0	0	P3	P2	P1	P3	P2	P1

b2 to b0: Store the axis setting errors of the manual pulse generators connected to P1 to P3 of A273EX.
- 0: Normal
- 1: Setting error

(Axis setting in each digit is other than 1 to 32)

b5 to b3: Store the smoothing magnification setting errors of the manual pulse generators connected to P1 to P3 of A273EX.
- 0: Normal
- 1: Setting error

(Axis setting in each digit is other than 0 to 59)

b15 to b6: All turn to 0.

D9186	Axis16	Axis15	Axis14	Axis13	Axis12	Axis11	Axis10	Axis9	Axis8	Axis7	Axis6	Axis5	Axis4	Axis3	Axis2	Axis1
D9187	Axis32	Axis31	Axis30	Axis29	Axis28	Axis27	Axis26	Axis25	Axis24	Axis23	Axis22	Axis21	Axis20	Axis19	Axis18	Axis17

Store the 1-pulse input magnification setting errors of the axes.
- 0: Stopping
- 1: Operating

(Input magnification of each axis is other than 1 to 100)

(4) Error program No. (D9189) Data from ....Data from PCPU to SCPU

(a) When an error occurs in the servo program at a servo program start (SVST instruction), the servo program setting error flag (M9079) turns ON and the faulty servo program No. (0 to 4095) is stored into this register.

(b) When an error program No. has been stored and an error occurs in another servo program, the new error program No. is stored.

(5) Error item information (D9190)....................................Data from PCPU to SCPU

When an error occurs in the servo program at a servo program start (SVST instruction), the servo program setting error flag (M9079) turns on and the error code corresponding to the setting item in error is stored into this register.
For details of the servo program setting errors, refer to Appendix 2.1.

(6) Servo amplifier loading information (D9191 to D9192)
....Data from PCPU to SCPU
When the servo system CPU control power supply (A6□P) is switched on or reset, the servo amplifier and option slot loading states are checked and its results are stored.
The axis which turned from non-loading to loading status after power-on is handled as loaded. However, the axis which turned from loading to non-loading status remains handled as loaded.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
D9191	Axis16	Axis15	Axis14	Axis13	Axis12	Axis11	Axis10	Axis9	Axis8	Axis7	Axis6	Axis5	Axis4	Axis3	Axis2	Axis1
D9192	Axis32	Axis31	Axis30	Axis29	Axis28	Axis27	Axis26	Axis25	Axis24	Axis23	Axis22	Axis21	Axis20	Axis19	Axis18	Axis17

Sarvo amplifier loading status
• Loaded••••••• 1
• Non-loaded••• 0

(a) Servo amplifier loading status
1) Loading/non-loading status

- Loading status.................................The ADU or MR-□-B is normal (communication with the servo amplifier can be made properly).
- Non-loading status...........................The servo amplifier is not loaded.
  Servo amplifier power is OFF.
  Due to connection cable fault or the like, communication with the servo amplifier cannot be made properly.

2) The system setting and servo amplifier loading status are listed below.

System Setting	ADU		MR-□-B	
	Loaded	Non-loaded	Loaded	Non-loaded
Used (Axis No. setting)	1 is stored	Major error	1 is stored	0 is stored
Not used	0 is stored	0 is stored	0 is stored	0 is stored

(7) PC link communication error code (D9196)
When an error occurs during PC link communication, the error code that corresponds to the error is stored in this device.

PC Communication Error Code Storage Register	Contents
D9196	00: No error 01: Receiving timing error 02: CRC error 03: Communication response code error 04: Receiving flame error 05: Communication task start error (Each error code is reset to 00 when normal communication is restarted.)

For details of PC link communication errors, see Appendix 2.5.

4. PARAMETERS FOR POSITIONING CONTROL

There are the following eight different parameters for positioning control.

(1) System settings
The system settings are used to set the used modules, axis numbers and others.
For details, refer to Section 4.1.

(2) Fixed parameters
The fixed parameters are set for each axis and their data are determined in accordance with the mechanical system or other factors.
They are used for command position calculation, etc. when exercising positioning control.
For details, refer to Section 4.2.

(3) Servo parameters
The servo parameters are set for each axis and their data are determined by the servo motor connected, e.g. servo model and motor type.
They are used to control the servo motor when exercising positioning control.
For details, refer to Section 4.3.

(4) Home position return data
The home position return data are set for each axis and they are such data as the home position return direction, method and speed.
They are used when making a home position return.
For details, refer to Section 4.4.

(5) JOG operation data
The JOG operation data are set for each axis and they are JOG speed limit value and parameter block No. data.
They are used when exercising positioning control by JOG operation.
For details, refer to Section 4.5.

(6) Parameter blocks
The parameter blocks are data such as acceleration and declaration times and speed limit value, and you can set 16 blocks.
The parameter blocks are specified in the sequence program, JOG operation data or home position return data to facilitate acceleration/deceleration processing (acceleration/declaration time, speed limit value) and other changes.
For details, refer to Section 4.6.

(7) Limit switch output data
The limit switch output data is set for the axis used and it is the ON/OFF pattern data output when the limit switch output setting is "Used" in the fixed parameter.
The axis where the limit switch output data is set outputs the ON/OFF pattern set for positioning control.
For details, refer to Section 7.1.

(8) Work coordinate data
The work coordinate data are used to set the work coordinates and you can set six different work coordinates (G54 to G59) per axis.
For the work coordinate system, specify the position with the offset from the mechanical coordinate system. Set the offset value with the distance from the mechanical coordinate system home position (0).
For details, refer to Section 4.7.

4.1 System Settings

(1) System settings such as base unit selection, unit allocation, axis number setting in programs, servo motor setting (model name), and servo amplifier setting (model name) are made according to the actual system.
(No settings are required when the unit is used as a PC extension base.)

(2) Data settings and modifications can be made interactively for some peripheral devices.

4.2 Fixed Parameters

(1) The fixed parameters are set for each axis and their data is fixed in accordance with the mechanical system or other factors.

(2) The fixed parameters are set with a peripheral device.

(3) The fixed parameters to be set are shown in Table 4.1.

Table 4.1 Fixed Parameters

No.	Item		Setting Range						Default		Remarks	Explanatory Section
			mm		inch		degree					
			Setting Range	Units	Setting Range	Units	Setting Range	Units	Initial Value	Units		
1	Unit setting		0	–	1	–	2	–	0	–	• Set the command unit for positioning control per axis.	–
2	Travel value per pulse (A)	Number of pulses per revolution (A_P)	1 to 65535 PLS						20000	PLS	• Set the number of feedback pulses per motor revolution determined by the mechanical system.	4.2.1
3		Travel value per revolution (A_L)	0.0001 to 6.5535	mm	0.00001 to 0.65535	inch	0.00001 to 0.65535	degree	2.0000	mm	• Set the travel per motor revolution determined by the mechanical system.	
4		Unit magnification (A_M)	1: ×1, 10: ×10, 100: ×100, 1000: ×1000						–	–	• Set to change the magnification for travel per pulse.	
5	Backlash compensation amount		0 to 6.5535	mm	0 to 0.65535	inch	0 to 0.65535	degree	0	mm	• Set the amount of backlash in the machine. • Backlash compensation is made every time the positioning direction changes during positioning.	7.2
6	Upper stroke limit		–214748.3648 to 214748.3647	mm	–21474.83648 to 214748.3647	inch	0 to 359.99999	degree	214748.3647	mm	• Set the upper limit value of the machine moving range.	4.2.2
7	Lower stroke limit		–214748.3648 to 214748.3647	mm	–21474.83648 to 21474.83647	inch	0 to 359.99999	degree	0	mm	• Set the lower limit value of the machine moving range.	
8	Command in-position range		0.0001 to 214748.3647	mm	0.00001 to 21474.83647	inch	0.00001 to 359.99999	degree	0.0100	mm	• Set the position where the command in-position signal (M1603+20n) is turned ON [(positioning address)-(present value)].	4.2.3
9	Limit switch output used/not used		0: Not used 1: used						0	–	• Set whether the limit switch output function is used or not for each axis.	7.1
10	Rapid feedrate		0.01 to 6000000.00	mm/min	0.001 to 600000.000	inch/min	0.001 to 2147483.647	degree/min	2000.00	mm/min	• Set the positioning speed under G00. • Set the home position return speed under G28.	4.2.4

4.2.1 Setting the number of pulses per revolution / travel value per revolution / unit magnification

This section explains how to set the number of pulses per revolution, the travel value per revolution, and the unit magnification.

(1) Setting method 1

(a) Finding the smallest position resolution ($\Delta 1$).
The smallest position resolution ($\Delta 1$) is determined by the travel value per revolution (ΔS) and the number of encoder feedback pulses (P_f).

$$\triangle 1 = \frac{\triangle S}{Pf}$$

(b) Finding the unit magnification (A_M)
Find the unit magnification on the basis of $\Delta 1$ determined as described in (a) above. However, make sure that the smallest command unit is not smaller than $\Delta 1$.

$\Delta 1$ found in (a) [mm]	Smallest Command Unit [mm]	Unit Magnification (A_M)
$0.00001 < \Delta 1 \leq 0.0001$	0.0001	1
$0.0001 < \Delta 1 \leq 0.001$	0.001	10
$0.001 < \Delta 1 \leq 0.01$	0.01	100
$0.01 < \Delta 1 \leq 0.1$	0.1	1000

[Example] Assuming that the travel value per revolution (ΔS) is 10 [mm] and the number of encoder feedback pulses (P_f) is 12000 [pulse/rev]:

$$\triangle 1 = \frac{10[\text{mm}]}{12000[\text{pulse/rev}]} = 0.00083 \rightarrow 0.0001 < 0.00083 \leq 0.001$$

This means that the smallest command unit is 0.001 [mm] and the unit magnification (A_M) is 10.
Therefore, 0.001 [mm] units can be specified in commands.

(c) Finding the travel value per revolution (AL).
If the unit magnification (AM) is 1, the travel value per revolution is the value of AL as it is. However, if the unit magnification (AM) is not 1, the travel value per revolution is the product of AL and AM.

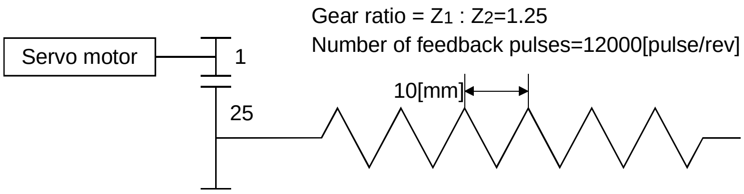
[Example] Assume that the travel value per revolution is 10[mm] and the unit magnification is 10:

$$A_L = \frac{10.0000[\text{mm}]}{10} = 1.0000[\text{mm}]$$

Accordingly, set the travel value per revolution (AL) to 1000.0[µm].

(d) Number of pulses per revolution (A_P)
Set the number of feedback pulses per revolution of the encoder.

(e) The number of pulses per revolution, travel value per revolution, and unit magnification for the example configuration shown here are calculated below.



1) Travel value per feedback pulse

$$\triangle S=10\times\frac{Z_1}{Z_2}=10\times\frac{1}{25}$$

$$\triangle 1=\frac{\triangle S}{Pf}=\frac{10}{25\times 12000}=0.000033.... \rightarrow \triangle 1=0.0001$$

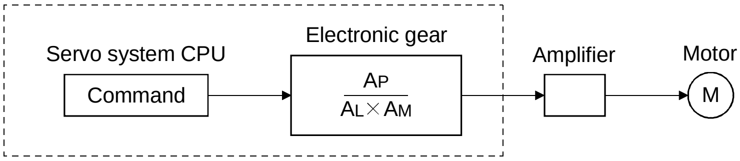
2) Unit magnification (A_M)
Since $\Delta 1$ is 0.0001, the unit magnification (A_M) is "1".

3) Travel distance per revolution (A_L)

$$A_L=\frac{10[mm]}{25}=0.4[mm]=400.0[\mu m]$$

4) Number of pulses per revolution (A_P)
A_P = 12000 [pulse/rev] ... fixed according to the encoder model.

(2) Setting method 2
If A_L cannot be set by using setting method 1, calculate the numerator and denominator of the electronic gear, and set A_P as the numerator and $A_L \times A_M$ as the denominator.



Example: With the example configuration shown above, and under the following conditions;

$$\begin{cases} \text{Gear ratio}=Z_1 : Z_2=1 : 39 \\ \text{Ball screw pitch}=25.4[mm] \end{cases}$$

$$A_L=\frac{25.4[mm]}{29}=0.65128205[mm] =651.28205[\mu m]$$

and A_L cannot be set, calculate as follows....

Elecronic gear

$$=\frac{Pf}{\triangle S}\times\frac{12000[pulse]}{25.4[mm]\times 1000\times\frac{1}{39}}=\frac{468000}{25400}=\frac{2340 \cdots\cdots A_P}{127 \cdots\cdots A_L\times A_M}$$

$$\begin{cases} A_P=2340[pulse] \\ A_L{}^*=12.7[\mu m] \text{ .... and set the following values} \\ A_M=1 \end{cases}$$

* : When actually setting A_L, calculate it as indicated in the table below.

Unit	Set Value for A (when A_M is "1")
mm	Denominator $\times 10^{-1}$ [μm]
inch	Denominator $\times 10^{-5}$ [inches]
degree	Denominator $\times 10^{-5}$ [degrees]

4.2.2 Upper stroke limit value/lower stroke limit value

These are the settings for the upper limit value and lower limit value in the travel range of the mechanical system.
Use the values in the mechanical coordinate system to set the upper and lower stroke limit values. The mechanical coordinate system is determined by a home position return.

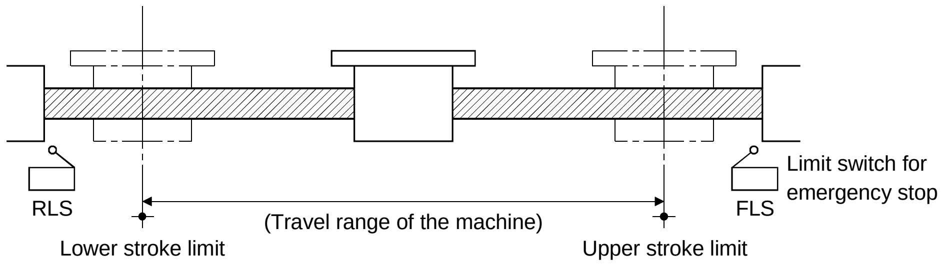


Fig. 4.1 Travel Range When Setting the Upper Stroke Limit Value and Lower Stroke Limit Value

(1) Stroke limit range check
The stroke limit range check is made at start or during progress of any of the following operations after home position return completion (M1610+20n ON).

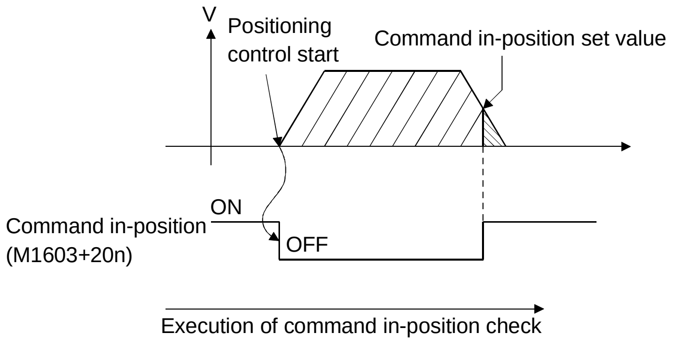
Operation Started	Check Executed/ Not Executed	Remarks
Positioning control (PTP, CP)	Executed	• When positioning is started, whether the positioning address is within the stroke limit range or not is checked. If it is outside the range, an error (error code: 580) occurs and positioning is not executed. • If the interpolation path goes out of the stroke limit range during circular interpolation, an error (error code: 207, 208) occurs and the axis decelerates to a stop.
JOG operation	Executed	• The axis stops if the present value goes out of the stroke limit range. (Error code: 207) The axis can move in the direction of returning to within the stroke.
Manual pulse generator operation	Executed	• The axis stops if the present value goes out of the stroke limit range. (Error code: 207) The axis can move in the direction of returning to within the stroke.

POINTS

(1) Besides setting the stroke limit upper limit value/lower limit value in the fixed parameters, the stroke limit range can also be set by using the external limit signals (FLS, RLS).
(2) When the external limit signal goes OFF, a deceleration stop is executed. The time taken to decelerate to a stop can be set by setting the "deceleration time" and "rapid stop deceleration time" in the parameter block.
(3) The stroke limit range check for positioning control (PTP, CP) is made after completion of a home position return. If a home position return is not yet completed, an error (error code: 162) occurs and the check cannot be made.
Always perform a home position return after power-on.
(4) Positioning cannot be started from outside the stroke limit range. Start positioning control after returning the axis to within the stroke by JOG or manual pulse generator operation.

4.2.3 Command in-position range

The command in-position is the difference between the positioning address (command position) and feed present value.
Once the value for the command in-position has been set, the command in-position signal (M1603 + 20n) will come ON when the difference between the command position and the feed present value enters the set range [(command position – feed present value) ≤ (command in-position range)].
The command in-position range check is executed continuously during positioning control.



4.2.4 Rapid feedrate setting

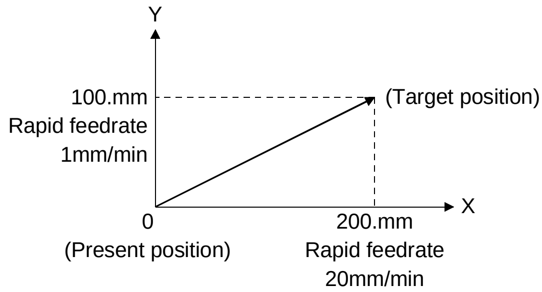
The rapid feedrate is the positioning speed used to perform positioning under G00 or to make a home position return under G28, and this data is needed to execute G00 or G28.
When exercising interpolation control under G00, change the speed of each axis on the basis of the axis whose time to reach the target position is the longer, and find the composite speed.
The following is a rapid feedrate setting example for interpolation control under G00.

[Example] When exercising interpolation control from the present position (X=0, Y=0) to the target position (X=200, Y=100)

High feedrate setting	X axis	20(mm/min)
	Y axis	1(mm/min)

G00 X200. Y100. : (Interpolation control executed)
Find the composite travel.

$\sqrt{100\text{mm}^2+200\text{mm}^2} \fallingdotseq 223.6067$ (mm)



After the above program is run, the target position reaching time of each axis is as follows.
X axis: 200.(mm)/20(mm/min) = 10(min)
Y axis: 100.(mm)/1(mm/min) = 100(min)
Since the reaching time of the Y axis is longer, use the Y axis as the reference axis for the feed rate and find the composite speed.

$$\underset{\text{(Reference axis feedrate)}}{1\text{mm/min}} \times \frac{\overset{\text{(Composite travel)}}{223.6067\text{mm}}}{\underset{\text{(Reference axis travel)}}{100\text{mm}}} \fallingdotseq \underset{\text{(Composite speed)}}{2.23\text{mm/min}}$$

POINTS

(1) The rapid feedrate of each axis is clamped at the speed limit value of the parameter block. The clamped value is also used to determine the axis whose time to reach the target position is the longest.
(2) In the above calculation, the travels and feed rates used are values without units. Care must be taken when their units differ.
(Example) 10000 for the travel of 1mm, 100000 for 1 degree, 100000 for 1 inch
100 for the feed rate of 1mm/min, 1000 for 1 degree/min, 1000 for 1 inch/min

4.3 Servo Parameters

(1) The servo parameters are parameters set for each axis: their settings are data fixed by the specifications of the controlled motors and data required to execute servo control.

(2) The servo parameters are set with a peripheral device.

⚠ CAUTION
⚠ After setting the servo parameters at a peripheral device, execute a "RELATIVE CHECK" and execute positioning control in the "NO ERROR" status. If there is an error, check the relevant points indicated in this manual and reset it.

4.3.1 MR-□-B servo parameters

The servo parameters to be set are indicated in Tables 4.2 through 4.4.

(1) Basic parameters

Table 4.2 Servo Parameters (Basic Parameters)

No.	Item	Setting Range						Default		Remarks	Explanatory Section
		mm		inch		degree		Initial Value	Units		
		Setting Range	Units	Setting Range	Units	Setting Range	Units				
*1	Amplifier setting	Set automatically in accordance with the system settings.									4.1
*2	Regenerative resistor										
*3	External dynamic brake										
*4	Motor type										
*5	Motor capacity										
6	Motor rpm (R)										
7	Number of feedback pulses (N)										APP. 5
8	Direction of rotation	0: Forward rotation (CCW) when the positioning address increases. 1: Reverse rotation (CW) when the positioning address decreases.						0	—	• Set the direction of rotation as seen from the load side. Forward rotation: reverse rotation:	—
9	Automatic tuning	0: Speed only 1: Position/speed 2: Not executed						1*1	—	• Set the gain (speed/position, speed) for executing automatic setting.	4.3.8
10	Servo responsive-ness	1 to 12						1	—	• Set in order to increase servo responsiveness.	4.3.9

*1: For MR-J-B, the default is "2".

> **POINT**
>
> After changing any of the items marked "*" in the table above, turn the servo power supply on after resetting the servo system CPU with the key switch or turning the PC READY signal (M2000) ON.

(2) Adjustment parameters

Table 4.3 Servo Parameter List (Adjustment Parameters)

No.	Item	Setting Range: mm Setting Range	Setting Range: mm Units	Setting Range: inch Setting Range	Setting Range: inch Units	Setting Range: degree Setting Range	Setting Range: degree Units	Default Initial Value	Default Units	Remarks	Expla-natory Section
1	Load inertia ratio	0.0 to 100.0						3.0[*1]	—	• Set the ratio of moment of load inertia for the motor.	4.3.7
2	Position control gain 1	Valid range 4 to 1000 rad/sec Setting range 1 to 9999 rad/sec						70	rad/sec	• Set to increase the follow-up with respect to the position command.	4.3.2
3	Speed control gain 1	Valid range 20 to 5000 rad/sec Setting range 1 to 9999 rad/sec						1200	rad/sec	• Set to increase the follow-up with respect to the speed command.	4.3.3
4	Position control gain 2	Valid range 10 to 500 rad/sec Setting range 1 to 9999 rad/sec						25	rad/sec	• Set to increase the position response with respect to load disturbance.	4.3.2
5	Speed control gain 2	Valid range 20 to 5000 rad/sec Setting range 1 to 9999 rad/sec						600[*2]	rad/sec	• Set when vibration is generated, for example in machines with a large backlash.	4.3.3
6	Speed integral compensation	Valid range 1 to 1000 rms Setting range 1 to 9999 rad/sec						20	ms	• Set the time constant for integral compensation.	4.3.4
7	Notch filter	0: Not used 1: 1125 2: 750 3: 562 4: 450 5: 375 6: 321 7: 281						0	Hz	• Set the frequency for the notch filter.	4.3.10
8	Feed forward gain	0 to 100% 0: Feed forward control is not executed.						0	%	• Set the feed forward coefficient used in positioning control.	4.3.6
9	In-position range[*3]	0.0001 to 214748.3647	mm	0.00001 to 21474.83647	inch	0.00001 to 359.99999	degree	0.0100	mm	• Sets the quantity of droop pulses in the deviation counter. • The in-position signal is ON when the number of droop pulses is within the set range. The expression below shows the setting range. 1 ≤ (in-position range) × AP/AL・AM ≤ 32767	4.3.5
10	Electromagnetic brake sequence[*4]	0 to 1000 ms						100	ms	• Set the time delay between actuation of the electromagnetic brake and base disconnection.	4.3.11
11	Monitor output mode (monitor 1)	(MR-H-B/MR-J-B) 0: Speed (±) 1: Torque (±) 2: Speed (+) 3: Torque (+) 4: Current command output 5: Command FΔT 6: Droop pulse 1/1 7: Droop pulse 1/4 8: Droop pulse 1/16 9: Droop pulse 1/32 (MR-J2-B) 0: Speed (±) 1: Torque (±) 2: Speed (+) 3: Torque (+) 4: Current command output 5: Command FΔT 6: Droop pulse 1/1 7: Droop pulse 1/16 8: Droop pulse 1/64 9: Droop pulse 1/256 10: Droop pulse 1/1024						0	—	• Set the monitor items output as analog outputs in real time.	4.3.12
12	Monitor output mode (monitor 2)[*4]							1	—		

*1: For MR-J2-B, the default is "7.0".
*2: For MR-J-B, the default is "500".
*3: The display of the possible setting range differs according to the electronic gear value.
*4: Setting not possible for MR-J-B.

Table 4.3 Servo Parameter List (Adjustment Parameters) (Continued)

No.	Item	Setting Range						Default		Remarks	Expla-natory Section
		mm		inch		degree					
		Setting Range	Units	Setting Range	Units	Setting Range	Units	Initial Value	Units		
13	Optional function 1 (carrier frequency selection)	0: 2.25 kHz (non low-noise operation) 3: 9 kHz (low-noise operation)						0	kHz	• Set "low noise" to improve the sound of the frequencies generated from the motor.	4.3.13
14	Optional function 1 (Encoder type)*4	0: 2-wire type 1: 4-wire type						0	—	• Set the type of encoder cable.	4.3.13
15	Optional function 2 (selection of no-motor operation)*6	0: Invalid 1: Valid						0	—	• To check the status without connecting a motor, set "valid".	4.3.14
16	Optional function 1 (external emergency stop signal)*5	0: Used 1: Not used						0	—	• To invalidate the external emergency stop signal (EMG) set "not used".	4.3.13
17	Optional function 2 (electro-magnetic brake interlock output timing)*6	0: Regardless of the rotational speed of the servo motor, output occurs under any of the following conditions. • Servo OFF • Occurrence of an alarm • Emergency stop input OFF (valid) 1: Output occurs under any of the above conditions provided that the servo motor rotational speed is zero (expansion parameters).						0	—	• Set the interlock timing for the electromagnetic brake interlock signal.	4.3.14
18	Optional function 2 (selection of microvibration suppression function)*5	0: Valid 1: Invalid						0	—	• Set "valid" to suppress vibration on stopping.	4.3.14
19	Optional function 2 (motor lock operation)*5	0: Valid 1: Invalid						0	—	• To carry out test operation without rotating the motor, set "valid".	4.3.14

*4: Setting not possible for MR-J-B.
*5: Cannot be set with MR-H-B/MR-J-B
*6: Cannot be set with MR-J2-B

(3) Expansion parameters

Table 4.4 Servo Parameters (Expansion Parameters)

No.	Item	Setting Range (mm / inch / degree: Setting Range, Units)	Default Initial Value	Default Units	Remarks	Explanatory Section
1	Motion output 1 offset	(MR-H-B/MR-J-B) –9999 to 9999 mv; (MR-J2-B) –999 to 999 mv	0	mv	• Set the offset value for motion output 1.	4.3.15
2	Motion output 2 offset*1	(MR-H-B/MR-J-B) –9999 to 9999 mv; (MR-J2-B) –999 to 999 mv	0*3	mv	• Set the offset value for motion output 2.	4.3.15
3	Pre-alarm data selection (sampling time selection)*1	0: 1.77 1: 3.55 2: 7.11 3: 14.2 4: 28.4	0	ms	• Set the analog data output when an alarm occurs.	4.3.16
4	Pre-alarm data selection (data selection 1)*1	0: Speed (±) 1: Torque (±) 2: Speed (+) 3: Torque (+) 4: Current command output 5: Command FΔT 6: Droop pulse 1/1 7: Droop pulse 1/4 8: Droop pulse 1/16 9: Droop pulse 1/32	0	—	• Set the analog data output when an alarm occurs.	4.3.16
5	Pre-alarm data selection (data selection 2)*1	(same as No. 4)	0	—	• Set the analog data output when an alarm occurs.	4.3.16
6	Zero speed	0 to 10000 r/min	10000	r/min	• Set the speed at which the motor speed is judged to be "0".	4.3.17
7	Excessive error alarm level	1 to 1000kPLS	80	kPLS	• Set the value at which an excessive droop pulses alarm is output.	4.3.18
8	Close encoder rotation direction	Unusable				
9	Home position return reference encoder	Unusable				
10	Optional function 5 (PI-PID control switching)	0: Invalid 1: Switching in accordance with droop during position control valid 2: Speed amplifier proportional control valid	0	—	• Set the conditions for PI-PID control switching.	4.3.19
11	Optional function 5 (Servo readout characters)*1	0: Japanese 1: English	0	—	• Set the display format for the parameter unit.	4.3.19
12	PI-PID switching position droop*1	0 to 50000 PLS	0	PLS	• Set the amount of position droop at the switch to PI-PID control when position control is executed.	4.3.20
13	Torque control compensation factor*1*2	–19 to 9979	0	—	• Set to expand the torque control range up to the speed limit value in torque control.	4.3.21
14	Speed differential compensation	0 to 1000	980	—	• Set the differential compensation value for the actual speed loop.	4.3.22

*1: Cannot be set when using MR-J-B.
*2: Cannot be set when using MR-J2-B.
*3: For MR-J2-B, the default is "1".

Table 4.4 Servo Parameters (Expansion Parameters) (Continued)

<table>
<tr><th rowspan="3">No.</th><th rowspan="3">Item</th><th colspan="6">Setting Range</th><th colspan="2">Default</th><th rowspan="3">Remarks</th><th rowspan="3">Expla-natory Section</th></tr>
<tr><th colspan="2">mm</th><th colspan="2">inch</th><th colspan="2">degree</th><th rowspan="2">Initial Value</th><th rowspan="2">Units</th></tr>
<tr><th>Setting Range</th><th>Units</th><th>Setting Range</th><th>Units</th><th>Setting Range</th><th>Units</th></tr>
<tr><td>15</td><td>Number of gear teeth at motor side</td><td colspan="10" rowspan="3">Unusable</td></tr>
<tr><td>16</td><td>Number of gear teeth at machine side</td></tr>
<tr><td>17</td><td>Number of closed encoder pulses</td></tr>
</table>

*1: Cannot be set when using MR-J-B.
*2: Cannot be set when using MR-J2-B.
*3: For MR-J2-B, the default is "1".

POINT

(1) The "setting range" for position control gain 1 and 2, speed control gain 1 and 2, and speed integral compensation can be set from a peripheral device, but if a setting outside the "valid range" is set, the following servo errors will occur when the power to the servo system CPU is turned ON, when the CPU is reset, and at the leading edge of the PC ready signal (M2000).

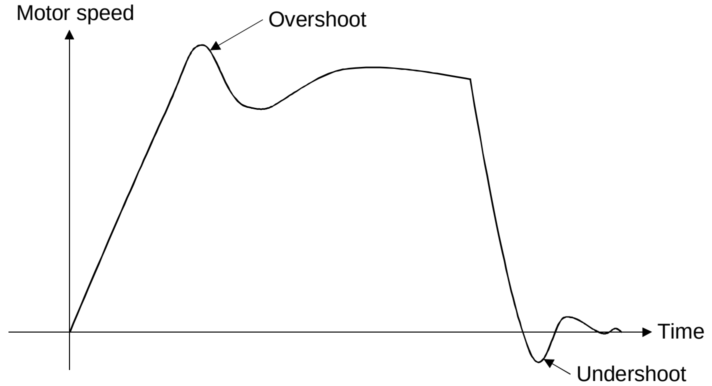
<table>
<tr><th>Servo Error Code</th><th>Error Contents</th><th>Processing</th></tr>
<tr><td>2613</td><td>Initial parameter error (position control gain 1)</td><td rowspan="5">Correct the setting for the relevant parameter so that it is within the "valid range", turn M2000 from OFF to ON, or reset with the reset key.</td></tr>
<tr><td>2614</td><td>Initial parameter error (speed control gain 1)</td></tr>
<tr><td>2615</td><td>Initial parameter error (position control gain 2)</td></tr>
<tr><td>2616</td><td>Initial parameter error (speed control gain 2)</td></tr>
<tr><td>2617</td><td>Initial parameter error (speed integral compensation)</td></tr>
</table>

4.3.2 Position control gain 1, 2

(1) Position control gain 1

(a) Position control gain 1 is set in order to make the stabilization time shorter.

(b) If the position control gain 1 is too high, it could cause overshoot and the value must therefore be adjusted so that it will not cause overshoot or undershoot.



(2) Position control gain 2

(a) Position control gain 2 is set in order to increase position response with respect to load disturbance.

(b) Calculate the position control gain 2 value to be set from the load inertia ratio and the speed control gain 2.

$$\text{Position control gain 2} = \frac{\text{Speed control gain 2}}{1 + \text{load inertia ratio}} \times \frac{1}{10}$$

POINTS
(1) If the position control gain 1 setting is too low, the number of droop pulses will increase and a servo error (excessive error) will occur at high speed. (2) The position control gain 1 setting can be checked from a peripheral device. (For the method used to execute this check, refer to the operating manual for the peripheral device used.)

4.3.3 Speed control gain 1, 2

(1) Speed control gain 1
  (a) In the speed control mode
    Normally, no change is necessary.
  (b) In the position control mode
    Set to increase the follow-up with respect to commands.

(2) Speed control gain 2
  (a) Speed control gain 2 is set when vibration occurs, for example in low-rigidity machines or machines with a large backlash.
    When the speed control gain 2 setting is increased, responsiveness is improved but vibration (abnormal motor noise) becomes more likely.
  (b) A guide to setting speed gain 2 is presented in Table 4.5 below.

Table 4.5 Guide to Speed Control Gain 2 Setting

Load Inertia Ratio (GD_L^2/GD_M^2)	1	3	5	10	20	30 or Greater	Remarks
Set value (ms)	800	1000	1500	2000	2000	2000	Setting possible within the range 1 to 9999 (valid range: 20 to 5000)

POINTS

(1) When the setting for speed control gain 1 is increased, the overshoot becomes greater and vibration (abnormal motor noise) occurs on stopping.
(2) The speed control gain 1 setting can be checked from a peripheral device.
(For the method used to execute this check, refer to the operating manual for the peripheral device used.)

4.3.4 Speed integral compensation

(1) This parameter is used to increase frequency response in speed control and improve transient characteristics.

(2) If the overshoot in acceleration/deceleration cannot be made smaller by adjusting speed loop gain or speed control gain, increasing the setting for the speed integral compensation value will be effective.

(3) A guide to setting the speed integral compensation is presented in Table 4.6 below.

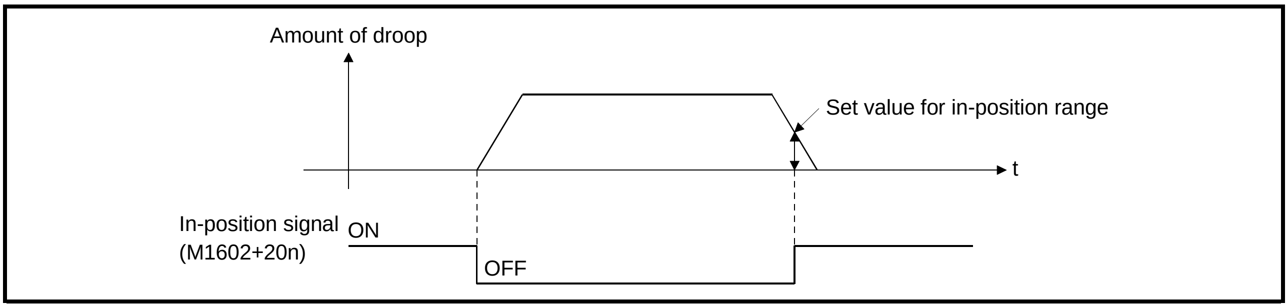
Table 4.6 Guide to Speed Integral Compensation Setting

Load Inertia Ratio (GD_L^2/GD_M^2)	1	3	5	10	20	30 or Greater	Remarks
Set value (ms)	20	30	40	60	100	200	Setting possible within the range 1 to 9999 (valid range: 1 to 1000)

4.3.5 In-position range

(1) The "in-position" refers to the quantity of droop pulses in the deviation counter.

(2) If an in-position value is set, the in-position signal (M1602 + 20n) will come ON when the difference between the position command and position feedback from the servomotor enters the set range.



4.3.6 Feed forward gain

This parameter is used to improve the follow-up of the servo system.
The setting range is as follows:

When using an MR-□-B.................0 to 100 (%)

4.3.7 Load inertia ratio

(1) This parameter sets the ratio of moment of load inertia for the servomotor.
The ratio of moment of load inertia is calculated using the equation below:

$$\text{Ratio of moment of load inertia} = \frac{\text{Moment of load inertia}}{\text{Motor's moment of inertia}}$$

(2) If automatic tuning is used, the result of automatic tuning is automatically set.

4.3.8 Automatic tuning

This is a function whereby the moment of inertia of the load is automatically calculated, and the most suitable gain is automatically set, by sensing the current and speed when motion starts.

POINT
When performing automatic tuning with MB-J-B, set the zero speed in the expansion parameters to at least 50rpm.

4.3.9 Servo responsiveness setting

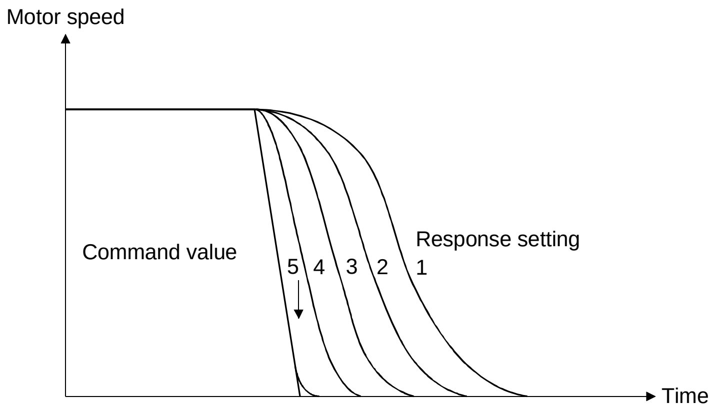
(1) This parameter setting is used to increase servo responsiveness.
Changing the set value to a higher value in the sequence 1, 2..., 5 improves servo responsiveness.
For machines with high friction, use the set values in the range 8 through C.

Response settings
1: Low-speed response
2:
3:
4:
5: High-speed response
Normal machine (MR-□-B usable) — Standard mode

8: Low-speed response
9:
A:
B:
C: High-speed response
Machines with high friction (only MR-H-B usable) — High frictional load mode

(2) Increase the response setting step by step starting from the low-speed response setting, observing the vibration and stop stabilization of the motor and machine immediately before stopping as you do so. If the machine resonates, decrease the set value.
If the load inertia is 5 times the motor inertia, make the set value 1 or greater.

(3) The figure below shows how the motor's response changes according to the servo responsiveness setting.



Change in motor response in accordance with response setting
(at the time of position control)

(4) Change the servo responsiveness setting while the motor is stopped.

4.3.10 Notch filter

This parameter sets the notch frequency for the notch filter.

Set Value	Notch Frequency (Hz)
0	Not used
1	1125
2	750
3	562
4	450
5	375
6	321
7	281

4.3.11 Electromagnetic brake sequence

This parameter sets the time delay between actuation of the electromagnetic brake and base disconnection.
(applies only when using MR-H-B/MR-J2-B.)

4.3.12 Monitor output mode

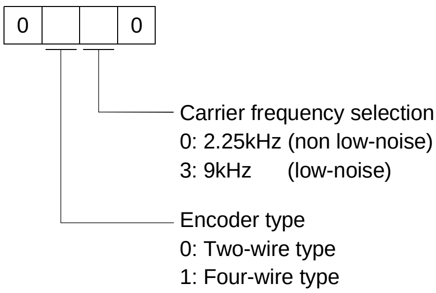
This parameter is set to output the operation status of the servo amplifier in real time as analog data.
This analog output makes it possible to check the operation status.
Note that the number of monitored items that can be set depends on the servo amplifier used, as indicated below:

When using an MR-H-B/MR-J2-B.......... 2 types
When using an MR-J-B........................... 1 type

4.3.13 Optional function 1

(1) Selection of carrier frequency
When low noise is set, the amount of electromagnetic noise of audible frequencies emitted from the motor can be reduced.

(2) Encoder type (applies only when using MR-H-B/MR-J2-B)
Set the type of encoder cable used.



POINT
(1) Optional function 1 (carrier frequency selection) When low-noise is set, the continuous output capacity of the motor is reduced.

(3) External emergency stop signal (applies only when using MR-J2-B)
The external emergency stop signal (EMG) can be made invalid.
0: External emergency stop signal is valid.
1: External emergency stop signal is invalid (automatically turned ON internally).
Since the emergency stop signal at the MR-J2-B cannot be used, do not set "0".

4.3.14 Optional function 2

(1) Selection of no-motor operation (applies when using MR-H-B/MR-J-B only)
0: Invalid
1: Valid
If no-motor operation is selected, the output signals that would be output if the motor were actually running can be output, and statuses indicated, without connecting the motor.
This makes it possible to check the sequence program of the servo system CPU without connecting a motor.

(2) Electromagnetic brake interlock output timing (applies only when using MR-H-B/MR-J2-B)
Select the output timing for the electromagnetic brake interlock signal from among the following.
0: Regardless of the rotational speed of the servo motor, output occurs under any of the following conditions.
- Servo OFF
- Occurrence of an servo alarm
- Emergency stop input

1: Output occurs under any of the above conditions provided that the servo motor rotational speed is zero (expansion parameters).

(3) Selection of microvibration suppression function (applies to MR-J2-B)
Set to suppress vibration specific to the servo system on stopping.
0: Microvibration suppression control is invalid.
1: Microvibration suppression control is valid.

(4) Motor lock operation (applies only when using MR-J2-B)
Allows test operation with the motor connected but without rotating the motor.
The operation is the same as no-motor operation with MR-H-B/MR-J-B.
0: Motor lock operation is invalid.
1: Motor lock operation is valid.
When motor lock operation is made valid, operation is possible without connecting the motor. However, since when MR-J2-B is used the connected motor is automatically identified before operation is started, if no motor is connected the connected motor type may be regarded as a default, depending on the type of amplifier. If this default motor type differs from the setting made in the system settings, the controller will detect minor error 900 (motor type in system settings differs from actually mounted motor), but this will not interfere with operation.

POINT

(1) Optional function 2 (no-motor operation selection)
No-motor operation differs from operation in which an actual motor is run in that, in response to signals input in no-motor operation, motor operation is simulated and output signals and status display data are created under the condition that the load torque zero and moment of load inertia are the same as the motor's moment of inertia. Accordingly, the acceleration/
deceleration time and effective torque, and the peak load display value and the regenerative load ratio is always 0, which is not the case when an actual motor is run.

4.3.15 Monitor output 1, 2 offset

Set the offset value for the monitored items set when setting monitor outputs 1 and 2.

4.3.16 Pre-alarm data selection

Used to output from the servo amplifier in analog form the data status when an alarm occurs.
(applies only when using MR-H-B/MR-J2-B)

(1) Sampling time selection
Set the intervals in which the data status data when an alarm occurs is recorded in the servo amplifier.

(2) Data selection
Set the data output in analog form from the servo amplifier.
Two types of data can be set.

0			

Data selection 2 / Data selection 1
0: Speed (±)
1: Torque (±)
2: Speed (+)
3: Torque (+)
4: Current command output
5: Command F△T
6: Droop pulse 1/1
7: Droop pulse 1/4
8: Droop pulse 1/16
9: Droop pulse 1/32

Sampling time selection
0: 1.77[ms]
1: 3.55[ms]
2: 7.11[ms]
3: 14.2[ms]
4: 28.4[ms]

4.3.17 Zero speed

This parameter sets the speed at which the motor speed is judged to be zero.

4.3.18 Excessive error alarm level

This parameter sets the range in which the alarm for excessive droop pulses is output.

4.3.19 Optional function 5

(1) PI-PID control switching
This parameter sets the condition under which switching from PI to PID control, or from PID control to PI control, is valid.

(2) Servo readout characters (applies only when using MR-H-B/MR-J2-B)
When the optional parameter unit is connected, set whether the screen display on the parameter unit will be in Japanese or English.

4.3.20 PI-PID switching position droop

This parameter sets the amount of position droop on switching to PI-PID control during position control. (applies only when using MR-H-B/MR-J2-B.)
The setting becomes effective when switching in accordance with the droop during position control is made valid by the setting for PI-PID control switching made using optional function 5.

4.3.21 Torque control compensation factor

This parameter is used to expand the torque control range up to the speed control value during torque control. (applies only when using MR-H-B.)
If a large value is set, the speed limit value may be exceeded and the motor may rotate.

4.3.22 Speed differential compensation

This parameter sets the differential compensation value for the actual speed loop.
In PI (proportional integration) control, if the value for speed differential compensation is set at 1000, the range for normal P (proportional) control is effective; if it is set to a value less than 1000, the range for P (proportional) control is expanded.

4.4 Home Position Return Data

The home position return data are data used to make a home position return.
Set them on the peripheral device.
For details of the setting, refer to Section 7.6.

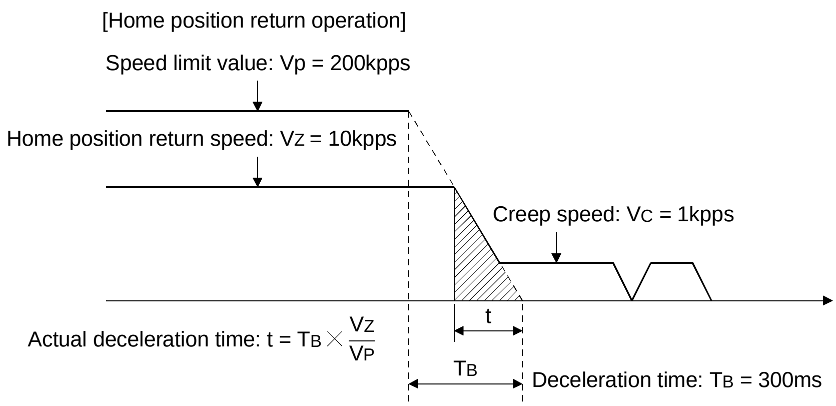
Table 4.7 Home Position Return Data List

No.	Item	Setting Range						Default	Remarks	Expla-natory Section
		mm		inch		degree				
		Setting Range	Units	Setting Range	Units	Setting Range	Units	Initial Value		
1	Home position return direction	0: Reverse direction (address decreasing direction) 0: Forward direction (address increasing direction)						0	• Set the direction in which a home position return will be made. • Starting a home position return moves the axis in the specified direction.	–
2	Home position return method	0: Near-zero point dog type 1: Count type 2: Data setting type						0	• Set the home position return method. • It is recommended to use the near-zero point dog or count type for the servo amplifier which is not absolute value-compatible, and the data setting type for the servo amplifier which is absolute value-compatible.	–
3	Home position address	-2147483648 to 2147483647	$\times 10^{-4}$mm	-2147483648 to 2147483647	$\times 10^{-5}$inch	0 to 35999999	$\times 10^{-5}$degree	0	• Set the present value of the home position on completion of home position return. • It is recommended to define the home position address at either of the upper or lower limit value of the stroke limit.	–
4	Second home position address	-2147483648 to 2147483647	$\times 10^{-4}$mm	-2147483648 to 2147483647	$\times 10^{-5}$inch	0 to 35999999	$\times 10^{-5}$degree	0	• Set the present value of the second home position on completion of the second home position return. • It is recommended to define the second home position address at either of the upper or lower limit value of the stroke limit.	–
5	Home position return speed	0.01 to 6000000.00	mm/min	0.001 to 600000.000	inch/min	0.001 to 2147483.647	degree/min	0.01	• Set the speed for home position return.	–
6	Creep speed	0.01 to 6000000.00	mm/min	0.001 to 600000.000	inch/min	0.001 to 2147483.647	degree/min	0.01	• Set the creep speed after near-zero point dog ON (low speed immediately before a stop which is made after deceleration from the home position return speed).	–
7	Setting of travel after near-zero point dog	0 to 214748.3647	mm	0 to 21474.83647	inch	0 to 21474.83647	degree	–	• For the count type, set the travel after near-zero point dog ON. • Set the value not less than the distance of deceleration made from the home position return speed.	4.4 (1)
8	Parameter block designation	1 to 16						1	• Set the parameter block (refer to Section 4.6) number used for home position return.	–

(1) Setting of travel after near-zero point dog ON
   (a) This data is the travel after near-zero point dog ON and is set when the count type home position return is made.
   (b) The first zero point after the movement of the preset travel after near-zero point dog ON is the home position.
   (c) The setting of the travel after near-zero point dog ON should be not less than the distance of deceleration made from the home position return speed.

Example

The following example gives how to calculate the deceleration distance when the speed limit value, home position return speed, creep speed and deceleration time are set as follows.



[Deceleration distance (Shaded area in the chart)]

$$= \frac{1}{2} \times \frac{V_Z}{1000} \times t$$

Converted into speed per 1ms

$$= \frac{V_Z}{2000} \times \frac{T_B \times V_Z}{V_P}$$

$$= \frac{10 \times 10^3}{2000} \times \frac{300 \times 10 \times 10^3}{200 \times 10}$$

= 75 ········ Set 75 or more.

4.5 JOG Operation Data

The JOG operation data is used to perform JOG operation.
Set this data on the peripheral device.

Table 4.8 JOG Operation Data List

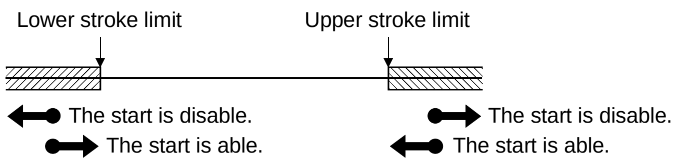
No.	Item	Setting Range						Default		Remarks	Expla-natory Section
		mm		inch		degree		Initial Value	Units		
		Setting Range	Units	Setting Range	Units	Setting Range	Units				
1	JOG speed limit value	0.01 to 6000000.00	mm/min	0.001 to 600000.000	inch/min	0.001 to 2147483.647	degree/min	200.00	mm/min	• Set the maximum speed for JOG operation. • If the JOG speed setting is higher than the JOG speed limit value, it is controlled at the JOG speed limit value.	–
2	Parameter block designation	1 to 16						1	–	• Set the parameter block number used for JOG operation.	4.6

(1) Checking the JOG operation data
A relative check is made on the preset JOG operation data at any of the following timings:
- At power-on
- On leading edge (OFF to ON) of PC ready (M2000)
- When test mode is selected.

(2) Processing at data error
- When a relative check is made, only the data where an error has been detected is controlled at the default value.
- The error code corresponding to each data of the faulty axis is stored into the data register.

POINT

(1) During JOG operation, the axis cannot be started toward the outside of the stroke limit range in the fixed parameter. However, when the axis is outside the stroke limit range, JOG operation can be performed in the rotation direction toward the stroke limit range.



4.6 Parameter Block

(1) The parameter blocks serve to make setting changes easy by allowing data such as the acceleration/deceleration control to be set for each positioning processing.

(2) A maximum of 16 blocks can be set as parameter blocks.

(3) Parameter blocks can be set at a peripheral device.

(4) The parameter block settings to be made are shown in Table 4.9.

Table 4.9 Parameter Block Settings

No.	Item	Setting Range: mm Setting Range	mm Units	inch Setting Range	inch Units	degree Setting Range	degree Units	Default: Initial Value	Default: Units	Remarks	Explanatory Section
1	Interpolation control unit	0	–	1	–	2	–	0	—	• Set the units for compensation control. • Can also be used as the units for the command speed and allowable error range for circular interpolation set in the motion program.	6.6.5
2	Speed limit value	0.01 to 6000000.00	mm/min	0.001 to 600000.000	inch/min	0.001 to 2147483.647	degree/min	2000.00	mm/min	• Set the maximum speed for positioning/home position return. • If the positioning speed or home position return speed setting exceeds the speed limit value, control is executed at the speed limit value.	4.6.1
3	Acceleration time	Acceleration-fixed acceleration/deceleration mode			1 to 65535ms			1000	ms	• Set the time from start of operation until the speed limit value is reached.	
		Time-fixed acceleration/deceleration mode			1 to 5000ms					• The acceleration/ deceleration time is always as preset.	
4	Deceleration time	Acceleration-fixed acceleration/deceleration mode			1 to 65535ms			1000	ms	• Set the time from the speed limit value until a stop is made.	
		Time-fixed acceleration/deceleration mode			Invalid					• The setting is ignored.	
5	Rapid stop deceleration time	Acceleration-fixed acceleration/deceleration mode			1 to 65535ms			1000	ms	• For a rapid stop, set the time from the speed limit value until a stop is made.	
		Time-fixed acceleration/deceleration mode			Invalid					• The setting is ignored.	
6	S curve ratio	Acceleration-fixed acceleration/deceleration mode			0 to 100%			0	%	• Set the S curve ratio for S-pattern acceleration/deceleration processing. • Trapezoidal acceleration/deceleration processing is performed at the S curve ratio of 0%.	4.6.2
		Time-fixed acceleration/deceleration mode			Invalid					• Always set 0%.	
7	Torque limit value	1 to 500%						300	%	• Set the torque limit value in the servo program.	—
8	Deceleration processing on STOP input	0: Deceleration stop executed based on the deceleration time. 1: Deceleration stop executed based on the rapid stop deceleration time.						0	—	• Set the deceleration processing when external signals (STOP, FLS, RLS) are input.	—
9	Allowable error range for circular interpolation	0 to 10.0000	mm	0 to 1.00000	inch	0 to 1.00000	degree	0.0100	mm	• Set the permissible range for the locus of the arc and the set end point coordinates.	4.6.3

POINTS

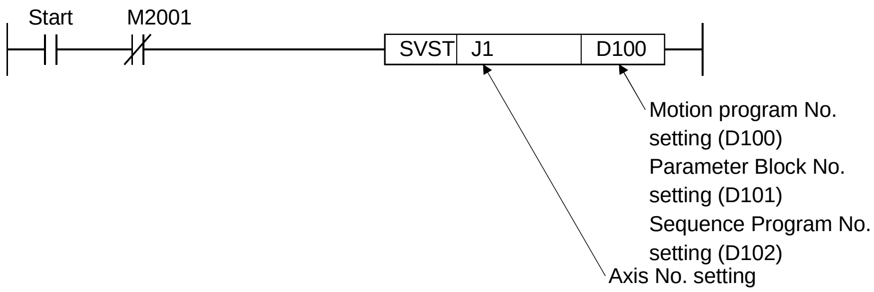
(1) Parameter blocks are designated in the home position return data, JOG operation data, or sequence program.
(2) The speed limit value is the feed rate setting range of the feed rate (F) set in the motion program.

POINT

(1) The data set in the parameter block are used for positioning control, home position return and JOG operation.

(a) The parameter block No. used in positioning control is set by indirect designation of the SVST instruction in the sequence program from the peripheral device.
For indirect designation, specify the motion program No. (0 No.) and parameter block No.
When the parameter block No. setting is 0 (no setting) or 17 or more, control is exercised with the data of parameter block No. 1.

[Sequence program]



(b) The parameter block No. used for home position return is set when setting the "home position return data" with a peripheral device.

[Home position return data setting screen]

[HOME POSITION RETURN DATA]

X AXIS <mm>	SETTING DATA	SETTING RANGE
A DIRECTION	0	0: REVERSE 1: FORWARD
B METHOD	0	0: DOG 1: COUNT 2: DATA SET
C ADDRESS	0.0000	-214748.3648 - 214748.3647 (mm)
D 2ND ADDRESS	0.0000	-214748.3648 - 214748.3647 (mm)
E SPEED	0.01	0.01 - 6000000.00 (mm/min)
F CREEP SPEED	0.01	0.01 - 6000000.00 (mm/min)
G MOVEMENT AFTER DOG	——	——
H P.B. NO.	1	1 - 16

End: SET Esc: STOP

1 2 3 4 5 6 7 8 9 0

Parameter block No. setting

(c) The parameter block No. used for JOG operation is set when setting the "JOG operation data" with a peripheral device.

[JOG operation data setting screen]

[JOG OPERATION DATA]

X AXIS <mm>	SET DATA	SETTING RANGE
1 SPEED LIMIT	2000.00	0.01 - 6000000.00 (mm/min)
2 P.B NO.	1	1 - 16

End: SET Esc: STOP

1 2 3 4 5 6 7 8 9 0

Parameter block No. setting

4.6.1 Relationships among the speed limit value, acceleration time, deceleration time, and rapid stop deceleration time

According to the G code instructions, there are two different acceleration/deceleration modes, acceleration-fixed acceleration/deceleration and time-fixed acceleration/deceleration.

(1) Acceleration-fixed acceleration/deceleration

(a) G01, G02, G03 or G32 during G101 execution
The acceleration/deceleration mode is acceleration-fixed acceleration/deceleration.
The actual acceleration time, deceleration time and rapid stop deceleration time are shorter than their settings as the positioning speed is lower than the speed limit value.
The setting ranges of the acceleration time, deceleration time and rapid stop deceleration time used are 1 to 65535ms.

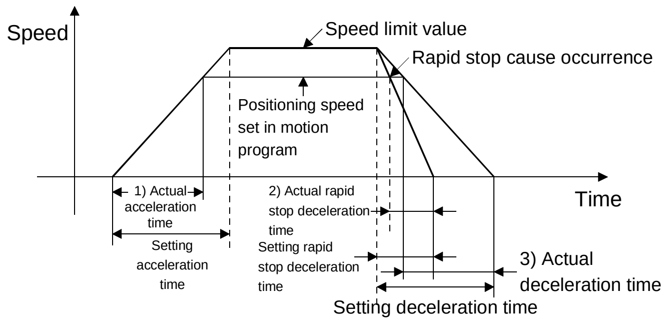
(b) G00 (without M code), G28 (high-speed home position return), G30, G53 or G00 including M code during G101 execution
The acceleration/deceleration mode is acceleration-fixed acceleration/deceleration.
The calculation of acceleration for acceleration/deceleration is based on the lower speed of the feedrate from the rapid feedrate in the fixed parameter (refer to Section 4.2.4) and the speed limit value in the parameter block.
At the override of 100%, the actual acceleration time, actual rapid stop deceleration time and actual deceleration time are equal to their settings.
The setting ranges of the acceleration time, deceleration time and rapid stop deceleration time used are 1 to 65535ms.

(2) Time-fixed acceleration/deceleration

(a) G00 including M code during G100 execution (default), G01, G02, G03 or G32
The acceleration/deceleration mode is time-fixed acceleration/deceleration.
The preset acceleration time is used to perform acceleration, deceleration or rapid stop deceleration processing.
The setting range of the acceleration time used is 1 to 5000ms.
If the setting exceeds 5000ms, the acceleration time is clamped at 5000ms.
At this time, an error does not occur.

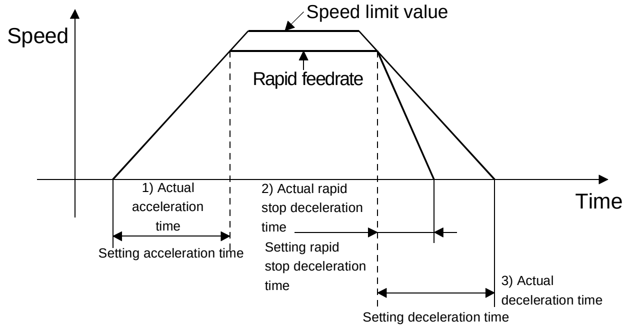
(1) Acceleration-fixed acceleration/deceleration

(a) G01, G02, G03 or G32 during G101 execution



1) Actual acceleration time
Time until the positioning speed set in the motion program is reached
2) Actual rapid stop deceleration time
Time from the positioning speed set in the motion program to a rapid stop
3) Actual deceleration time
Time from the positioning speed set in the motion program to a stop

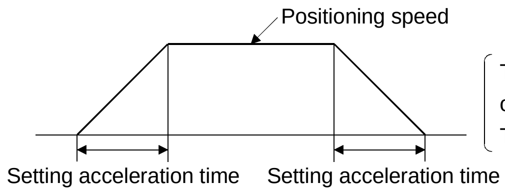
(b) G00 (without M code), G28 (high-speed home position return), G30, G53 or G00 including M code during G101 execution



1) Actual acceleration time
Equal to the preset acceleration time at the override of 100%.
2) Actual rapid stop deceleration time
Equal to the preset rapid stop deceleration time at the override of 100%.
3) Actual deceleration time
Equal to the preset deceleration time at the override of 100%.

(2) Time-fixed acceleration/deceleration

(a) G00 including M code during G100 execution (default), G01, G02, G03 or G32

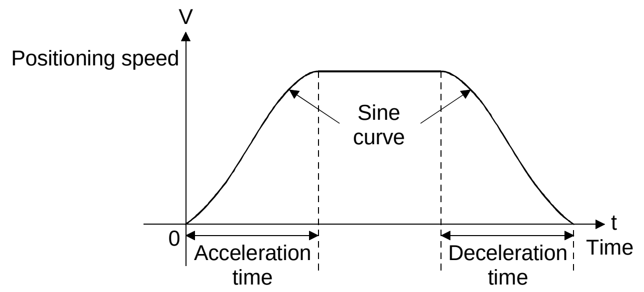


The acceleration/deceleration time is fixed independently of the positioning speed (always acceleration time).
The deceleration time and rapid stop time are ignored.

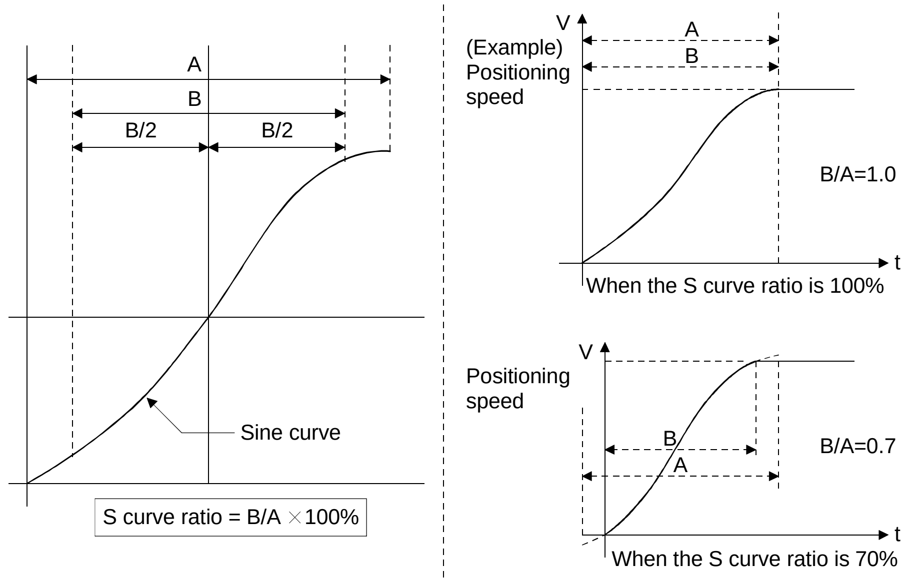
Fig. 4.2 Relationships among the Speed Limit Value, Acceleration Time, Deceleration Time, and Rapid Stop Deceleration Time

4.6.2 S curve ratio

The S curve ratio used when S pattern processing is used as the acceleration and deceleration processing method can be set.
The setting range for the S curve ratio is 0 to 100 (%).
If a setting that is outside the applicable range is made, an error occurs on starting, and control is executed with the S curve ratio set at 100%.
Errors are set in the servo program setting error area (D9190).
Setting an S curve ratio enables acceleration and deceleration processing to be executed gently.
The S curve ratio is set by the parameter block. (Refer to section 4.6.)
The graph for S pattern processing is a sine curve, as shown below.



As shown below, the S curve ratio setting serves to select the part of the sine curve to be used as the acceleration and deceleration curve.



Note: Under G00 including M code, G01, G02, G03 or G32, the S curve ratio is ignored and operation is always performed at the ratio of 0%.

4.6.3 Allowable error range for circular interpolation

In control with the center point designated, the locus of the arc calculated from the start point address and center point address may not coincide with the set end point address.
The allowable error range for circular interpolation sets the allowable range for the error between the locus of the arc determined by calculation and the end point address.
If the error is within the allowable range, circular interpolation to the set end point address is executed while also executing error compensation by means of spiral interpolation.
If the setting range is exceeded, an error occurs and positioning does not start.
When such an error occurs, the relevant axis is set in the minor error code area.

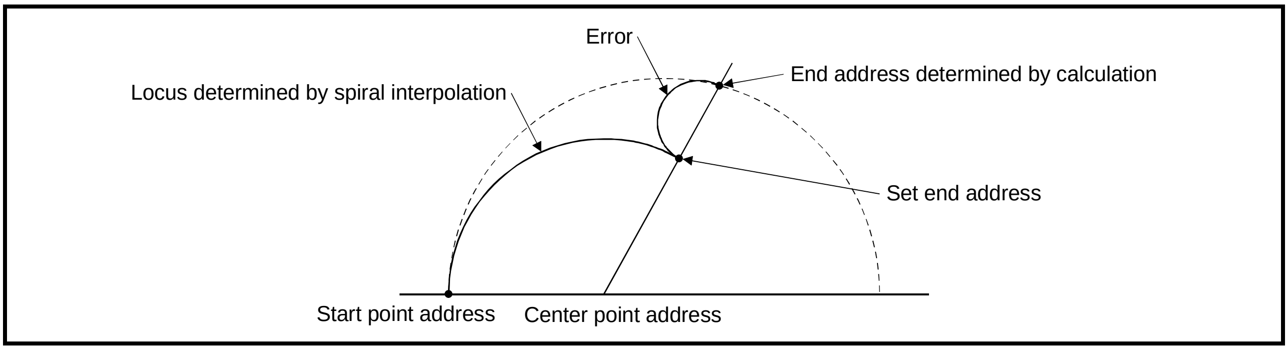


Fig. 4.3 Spiral Interpolation

4.7 Work Coordinate Data

(1) The work coordinate data are used to set the work coordinates and you can set six different work coordinates (G54 to G59) per axis. (For details, refer to Section 4.7.)

(2) For the work coordinate system, specify the position with the offset from the mechanical coordinate system home position. The offset setting is the distance from the mechanical coordinate system home position (0).

(3) Set the work coordinate data on the peripheral device.

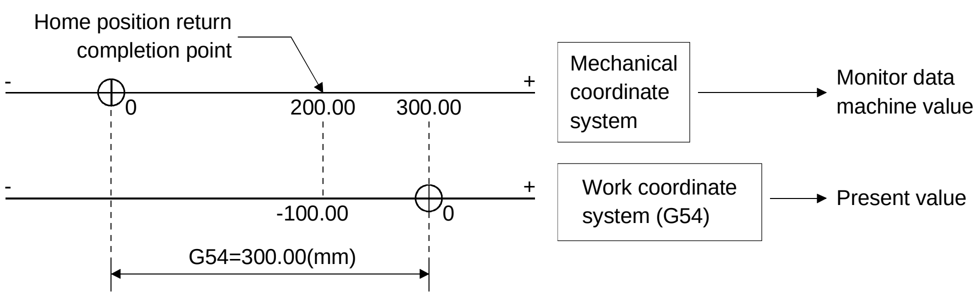
(4) The work coordinate data to be set are listed in Table 4.10.

Table 4.10 Work Coordinate Data List

No.	Item	Setting range						Default		Remark	Section For details
		mm		inch		degree		Initial value	Unit		
		Setting range	Unit	Setting range	Unit	Setting range	Unit				
1	G54	-214748.3648 to 214748.3647	mm	-21474.83648 to 21474.83647	inch	-359.99999 to 359.99999	degree	0	mm	Set the work coordinate systems 1 to 6.	6.7
2	G55	-214748.3648 to 214748.3647	mm	-21474.83648 to 21474.83647	inch	-359.99999 to 359.99999	degree	0	mm		
3	G56	-214748.3648 to 214748.3647	mm	-21474.83648 to 21474.83647	inch	-359.99999 to 359.99999	degree	0	mm		
4	G57	-214748.3648 to 214748.3647	mm	-21474.83648 to 21474.83647	inch	-359.99999 to 359.99999	degree	0	mm		
5	G58	-214748.3648 to 214748.3647	mm	-21474.83648 to 21474.83647	inch	-359.99999 to 359.99999	degree	0	mm		
6	G59	-214748.3648 to 214748.3647	mm	-21474.83648 to 21474.83647	inch	-359.99999 to 359.99999	degree	0	mm		

(5) When a home position return is made on the basis of the home position return setting data, the mechanical coordinate system and work coordinate system are as shown below.

[Example] The X-axis home position address of the home position return data is set to 200.00(mm) and the X axis: G54 of the work coordinate data is set to 300.00(mm) to make a home position return.



On completion of a home position return, the machine value is equal to 200.00(mm) and the present value to -100.00(mm).
When the work coordinate data is set to 0, the present value is equal to the machine value.

5. SEQUENCE PROGRAMS AND SFC PROGRAMS

This section explains how to start a motion program using a sequence program or SFC program for positioning control, and gives other information.

5.1 Cautions on Creating a Sequence Program or SFC Program

The following cautions should be observed when creating a sequence program or SFC program.

(1) Positioning control instructions
The motion program start request instruction (DSFRP)/(SVST) (see Section 5.2) and the home position return instructions (DSFLP)/(CHGA) (See section 5.3) speed change instructions (see Section 5.4) are used as positioning instructions.

(2) Unusable instructions
It is not possible to use the DSFL (word data 1 word shift to left) or DSFR (word data 1 word shift to right) instruction.
If a DSFL instruction of DSFR instruction is executed, an operation error occurs and the following happens:
(a) Operation error flag (M9010, M9011) is turned ON.
(b) 50(OPERATION ERROR) is stored in the self-diagnosis error code register (D9008)
(c) The step in which the DSFR or DSFL instruction was executed is stored in the error step register (D9010, D9011).
In order to shift word data, use the BMOV instruction (see Appendix 4).

(3) Dedicated devices for the PCPU
Of the servo system CPU devices, those shown in Table 5.1 are exclusively for use with the PCPU.
Check the applications of devices before using them in the sequence program (for details, see Chapter 3).

Table 5.1 Dedicated Devices for the PCPU

Device Name	Device No.
Internal relays	M1400 to M2047
Data registers	D500 to D1023
Special relays	M9073 to M9079
Special registers	D9180 to D9199

Note that internal relays (M1400 to M2047) and data registers (D500 to D1023) will not be latched even if a latch range setting is made for them. (The device symbols for M1400 to M2047 are displayed as M, L, and S by the GPP device in accordance with the M, L, and S settings in the parameters.)

(4) SFC programs
Refer to the manuals below for details on the SFC programming method.
MELSAP II Programming Manual (IB-66361)

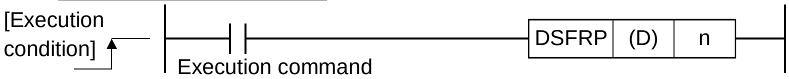
5.2 Motion Program Start Request Instruction (DSFRP/SVST)

There are two motion program start request instructions: the DSFRP instruction and the SVST instruction.
When executing positioning control, up to 3 axes can be controlled with the DSFRP instruction and up to 4 axes can be controlled with the SVST instruction. When the A273UHCPU (32 axis feature)/A173UHCPU(S1) is used, the DSFRP instruction cannot be used as a servo program start request instruction. It may be used only as a word data shift instruction.

5.2.1 Start request instruction for 1 to 3 axes (DSFRP): when using A172SHCPUN/A171SHCPUN

	Usable Devices																						Digit Designation	Number of Steps	Subset	Index	Carry Flag	Error Flag	
	Bit Devices							Word (16 Bit) Devices									Constants		Pointers		Level							
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K	H	P	I	N					M9012	M9010	M9011
(D)										○													7		×		○	○
n																	○	○										

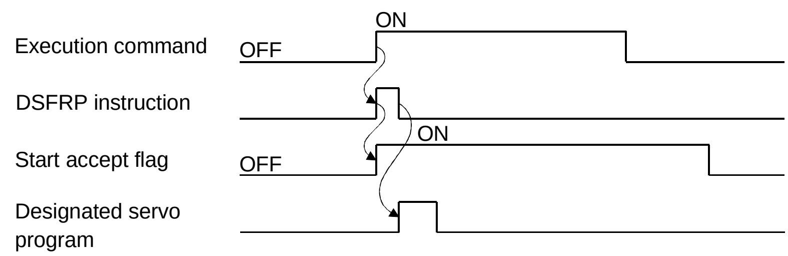
SEQUENCE PROGRAM



	Setting data	Setting range		
(D)	No. of axis to be started	D1 to D8 (A172SHCPUN) D1 to D4 (A171SHCPUN)		
n	No. of servo program to be executed	Direct designation	1 to 256	
		Indirect designation	Decimal	K30000 to K30497
			Hexadecimal	H7530 to H7721

The following processing is executed at the leading edge (OFF→ON) of the DSFRP instruction:

- The start accept flag (M2001+n) designated in (D) is turned ON (see Section 3.1.3 (2)).
- A start request is issued for the servo program designated by "n".



[Data Settings]

(1) Setting the axes to be started
The axes to be started are set in (D) in the way shown below.

D □□□

→ Designate digits from 1 to 3.
- 1 axis to started ··················· Make the setting for 1 axis (1 digit)
- 2 axis to interpolation to be started ··· Make the setting for 2 axes (2 digits)
- 3 axis to interpolation to be started ··· Make the setting for 3 axes (3 digits)

Designate started axis numbers 1 to 4 for an A171SHCPUN, or 1 to 8 for an A172SHCPUN.

→ Device symbol (only "D" can be used)

Example

The axes to be started are designated as follows.
- Axis 1 ....D1
- Axis 1 and axis 2......................D12
- Axis 1, axis 2, and axis3 ....D123

(2) Motion program No. setting
There are two types of motion program number setting: direct and indirect.
(a) In direct setting, the motion program number is designated directly as the number itself (1 to 256).

Example

Motion program No.50 would be set as follows.
- When designated with a K device........... K50

(b) In indirect setting, the motion program number, the parameter block No. and the sequence program No. are set as a value in a data register.
The data registers that can be used are D0 to D497, and they are set as follows.
1) K 3 0 □□□

→ Designation of the data register number (000 to 497)
••• 3 digits must be set.
Example: For 50, set 050.

→ Date register disignation

Set the data register values as indicated below.
Data register of specified number ....Motion program No.
Data register of specified number + 1 ....Parameter block No.
Data register of specified number + 2 .... Sequence program No.

2) It is also possible to designate a hexadecimal number (H7530 to H7721) converted from a decimal (K) number.

Example

Make the following setting when specifying the motion program number, parameter block number and sequence program number to be started as the data register (D50, D51, D52) data.

- When designated with a K device • • • • K30050 • • • • Specifies D50, D51[*1], D52[*2].

*1: When the parameter block number setting (D51) is outside the range 1 to 16, control is exercised with the parameter block No. 1.
*2: When the sequence number setting (D52) is outside the range 1 to 9999, a start is made at the beginning of the motion program.

POINTS

(1) (1) In (D), specify all axes described in the motion program.
(2) In (D), "D" is used as the device symbol but the present values of the data register numbers used in the sequence program are ignored.

[Error Details]

In the following cases, an operation error occurs and the DSFRP instruction is not executed.,
- When the setting for (D) comprises 4 or more digits.
- When the axis number given in any digit of (D) is a number other than 1 to 8 (A172SHCPUN).
- When the axis number given in any digit of (D) is a number other than 1 to 4 (A171SHCPUN).
- When the same axis number is set twice in (D).
- When n is a value outside the range 1 to 256.
- When the settings for (D) or n are made by indirect setting with an index register (Z, V).

POINT

- For indirect designation, do not specify the last data register (D499) and its preceding register (D498).

5.2.2 Start request instruction for 1 to 8/1 to 4/1 to 32 axes (SVST)

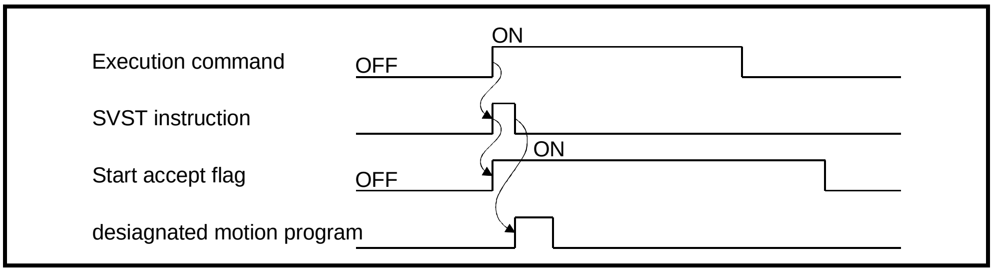
	Usable Devices																					Digit Designation	Number of Steps	Subset	Index	Carry Flag	Error Flag	
	Bit Devices							Word (16 Bit) Devices									Constants		Pointers		Level							
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K	H	P	I	N					M9012	M9010	M9011
(D)																							13		*1 ○		○	○
n										○	○						○	○										

*1: Possible with indirect setting only

	Setting data	Setting range	
(D)	J+No. of axis to be started	J1 to J8 (A172SHCPUN) J1 to J4 (A171SHCPUN) J1 to J32 (A273UHCPU (32 axis feature)/ A173UHCPU (S1))	
n	No. of servo program to be executed	Direct designation	1 to 256
		Indirect designation (Indirect designation device uses 3 words.)	D0 to D497 W0 to W3FE

The following processing is executed at the leading edge (OFF → ON) of the SVST instruction.

- The start accept flag (M2001+n) corresponding to the axis designated in (D) is turned ON (see Section 3.1.3 (2)).
- A start request is issued for the motion program designated by "n".



[Data Settings]

(1) Setting the axes to be started
The axes to be started are set in (D) in the way shown below.

○ Setting for 1 to 8 axes (A172SHCPUN)
○ Setting for 1 to 4 axes (A171SHCPUN)
•1 axis to be started • • • • • • • • • • • • Make the setting for 1 axis (J**)
•2 axes interpolation to be started • • • Make the setting for 2 axis (J**J**)
•3 axes interpolation to be started • • • Make the setting for 3 axis (J**J**J**)
•4 axes interpolation to be started • • • Make the setting for 4 axis (J**J**J**J**)
○ Designate J+started axis number 1 to 8 for an A172SHCPUN
○ Designate J+started axis number 1 to 4 for an A171SHCPUN
○ Designate J+started axis number 1 to 32 for an A273UHCPU (32 axis feature) / A173UHCPU(S1)
The number of digits in the axis number display is fixed at 3 including J (i.e. "J**")

---- Example ----

The axes to be started are designated as follows.
- Axis 1 .... J1
- Axis 1 and axis 2.................................... J1J2
- Axis 1, axis 2, and axis3 .... J1J2J3
- Axis 1, axis 2, axis3, and axis4............... J1J2J3J4

(2) Motion program No. setting
There are two types of servo program number setting: direct and indirect.
(a) In direct setting, the motion program number is designated directly as the number itself (1 to 256).

---- Example ----

Motion program No.50 would be set as follows.
- When designated with a K device........... K50

(b) In indirect setting, the motion program number, parameter block number and sequence program number are set as word device values.
The word device values are set as follows.
Specified word device .... Motion program No.
Word device next to specified one .... Parameter block No.
Word device second next to specified one .... Sequence program No.

POINT

(1) In (D), specify all axes described in the motion program.

1) The word devices that can be used are indicated in the table below.

Word Device	Usable Devices	
	A172SHCPUN/ A171SHCPUN	A273UHCPU (32 axis feature)/ A173UHCPU (S1)
D	0 to 497	1690 to 8199
W	0 to 3FD	0 to 1FFD

Example

Make the following setting when specifying the motion program number, parameter block number and sequence program number to be started as the data register (D50, D51, D52) data.

- When word device is used to specify —— SVST | J1J2J3 | D50

D50: Motion program No.

D51: Parameter block No.*1

D52: Sequence program No.*2

*1: When the parameter block number setting (D51) is outside the range 1 to 16, control is exercised with the parameter block No. 1.

*2: When the sequence number setting (D52) is outside the range 1 to 9999, a start is made at the beginning of the motion program.

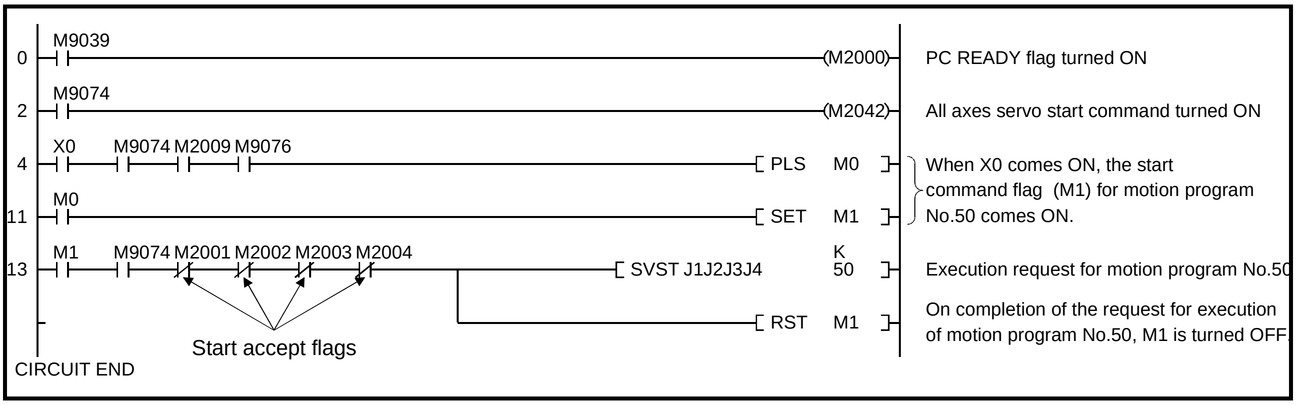
2) An index register (Z, V) can be used for index designation of the indirectly set word device.
   - For details on index registers (Z, V), see the ACPU Programming Manual (Fundamentals) (IB-66249).

[Error Details]

In the following cases, an operation error occurs and the SVST instruction is not executed.

- When the setting for (D) is for 9 or more axes (A172SHCPUN/A273UHCPU (32 axis feature)/A173UHCPU (S1)).
- When the setting for (D) is for 5 or more axes (A171SHCPUN).
- When the axis number given in any digit of (D) is a number other than J1 to J4 (A171SHCPUN).
- When the axis number given in any digit of (D) is a number other than J1 to J8 (A172SHCPUN).
- When the axis number given in any digit of (D) is a number other than J1 to J32 (A273UHCPU (32 axis feature)/A173UHCPU (S1)).
- When the same axis number is set twice in (D).
- When the setting for n is outside the applicable range.

[Program example]



5.3 Home Position Return Instructions (DSFLP/CHGA)

These instructions are used to make a home position return of the axis at a stop.

5.3.1 DSFLP instruction: when using A172SHCPUN/A171SHCPUN

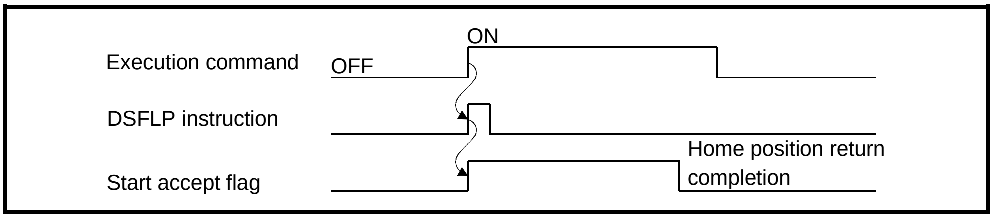
	Usable Devices																					Digit Designation	Number of Steps	Subset	Index	Carry Flag	Error Flag	
	Bit Devices							Word (16 Bit) Devices									Constants		Pointers		Level							
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K	H	P	I	N					M9012	M9010	M9011
(D)										○													7		○		○	○
n																	○	○										

SEQUENCE PROGRAM

	Setting data	Setting range
(D)	Axis No. which will be returned to home position	D1 to D8 (A172SHCPUN) D1 to D4 (A171SHCPUN)
n	Designation of home position return	K2 or H2

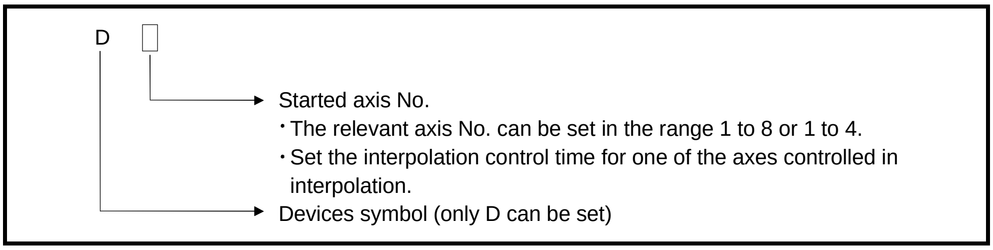
(1) The following processing is performed on the leading edge (OFF to ON) of the DSFLP instruction execution command.
   1) The start acceptance flag (M2001 to M2008/M2001 to M2004) corresponding to the axis specified in (D) is turned ON.
   2) The axis specified in (D) is returned to the home position in accordance with the home position return data specified in the parameters.
   3) The start acceptance is turned OFF on completion of the home position return.

[Operation Timing]



[Data Settings]

(1) Setting of the axis which will be returned to home position
   In (D), set the axis which will be returned to the home position as follows.



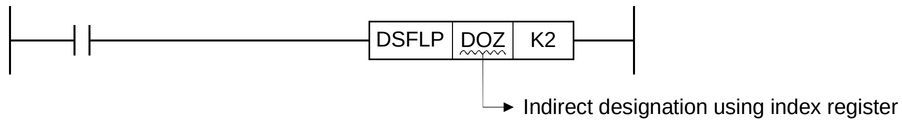
Example

The axes to be started are designated as follows.
- Axis 1 .... D1

(2) Home position return
Set a home position return as indicated below.
• Home position return ....Set K2 or H2.

POINT

For the DSFLP instruction, indirect setting cannot be made in (D) and n using the index register.



If indirect setting is made using the index register, an operation error occurs and the DSFLP instruction is not executed.

[Error Details]

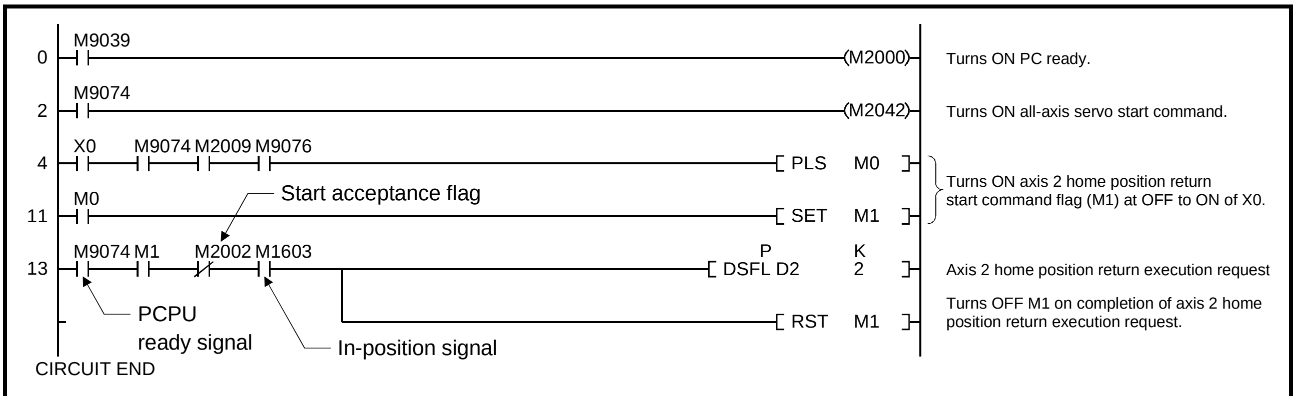
(1) In the following cases, an operation error occurs and the DSFLP instruction is not executed.
- Setting in (D) is other than 1 to 8/1 to 4.
- Setting in n is other than 1 or 2.
- Setting in (D) or n has been made by indirect setting using the index register (Z, V).

(2) In the following case, a minor error (error at control change) occurs and a home position return is not made.
At this time, the error detection flag (M1607+20n) is turned ON and the error code is stored into the minor error code area of the corresponding axis.
- When the axis specified in (D) for home position return is operating

[Program Example]

(1) The following program is designed to make a home position return of axis 2.
(a) Conditions
1) Home position return command.......... Leading edge (OFF to ON) of X0
2) Home position return execution flag.... M1
3) Axis 2 start acceptance (axis 2 stopping/operating confirmation) flag
.... M2002 (axis 2 start acceptance flag)

(b) Program example



POINT

When making a home position return, provide M9074 and in-position signal as interlock conditions.

5.3.2 CHGA instruction

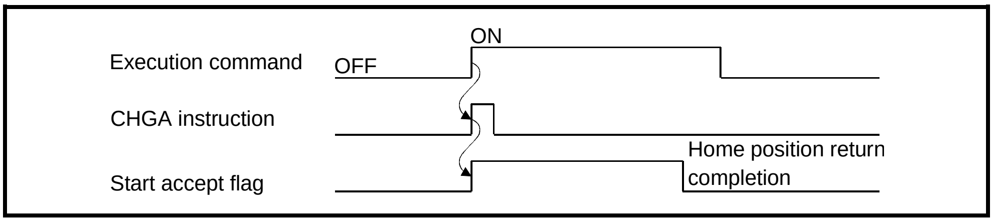
	Usable Devices																				Digit Designation	Number of Steps	Subset	Index	Carry Flag	Error Flag	
	Bit Devices							Word (16 Bit) Devices									Constants		Pointers		Level							
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K	H	P	I	N					M9012	M9010	M9011
(D)																							7		○		○	○
n										○	○	○					○	○										

SEQUENCE PROGRAM

	Setting data	Setting range
(D)	J + axis No. which will be returned to home position	J1 to J8 (A172SHCPUN) J1 to J4 (A171SHCPUN) J1 to J32 (A273UHCPU (32 axis feature)/A173UHCPU (S1))
n	Dummy	

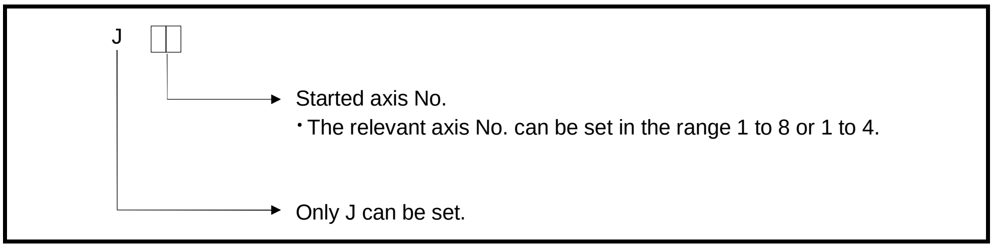
(1) The following processing is performed on the leading edge (OFF to ON) of the CHGA instruction execution command.
   1) The start acceptance flag (M2001 to M2008/M2001 to M2004) corresponding to the axis specified in (D) is turned ON.
   2) The axis specified in (D) is returned to the home position in accordance with the home position return data specified in the parameters.
   3) The start acceptance is turned OFF on completion of the home position return.

[Operation Timing]



[Data Settings]

(1) Setting of the axis which will be returned to home position
   In (D), set the axis which will be returned to the home position as follows.



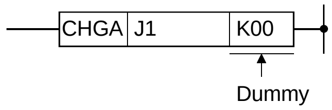
Example

The axes to be started are designated as follows.
• Axis 1 .... J1

(2) Home position return setting
Set a dummy for a home position return.

Example

• Set a dummy.



[Error Details]

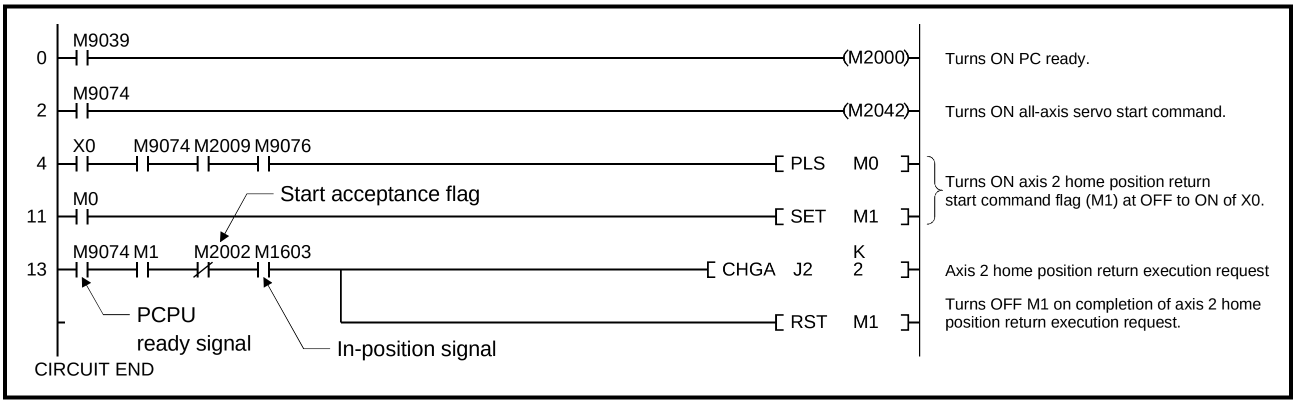
(1) In the following case, an operation error occurs and the CHGA instruction is not executed.
  • Setting in (D) is other than J1 to J8/J1 to J4.

(2) In the following case, a minor error (error at control change) occurs and a home position return is not made.
At this time, the error detection flag (M1607+20n) is turned ON and the error code is stored into the minor error code area of the corresponding axis.
  • When the axis specified in (D) for home position return is operating

[Program Example]

(1) The following program is designed to make a home position return of axis 2.
  (a) Conditions
    1) Home position return command.......... Leading edge (OFF to ON) of X0
    2) Home position return execution flag.... M1
    3) Axis 2 start acceptance (axis 2 stopping/operating confirmation) flag
       .... M2002 (axis 2 start acceptance flag)

  (b) Program example



(2) The following program is designed to change the positioning speed of axis 2.
  (a) Condition
    1) Speed change command .... Leading edge (OFF to ON) of X000

  (b) Program example

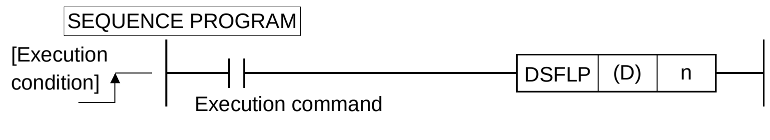
POINT
When override is valid, the speed change using DSFLP/CHGV is ignored for the axes operating automatically.

5.4 Speed Change Instructions (DSFLP/CHGV)

This instruction is used to change the speed of an axis during positioning or JOG operation.

5.4.1 DSFLP instruction (When using A172SHCPUN/A171SHCPUN)

	Usable Devices																					Digit Designation	Number of Steps	Subset	Index	Carry Flag	Error Flag	
	Bit Devices							Word (16 Bit) Devices									Constants		Pointers		Level							
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K	H	P	I	N					M9012	M9010	M9011
(D)										○													7		○		○	○
n																	○	○										



	Setting data	Setting range
(D)	No. of speed change axis	D1 to D8 (A172SHCPUN) D1 to D4 (A171SHCPUN)
n	Speed change designation	K1 or H1

(1) The following processing is executed at the leading edge (OFF → ON) of the DSFLP instruction:

(a) Present value change

1) The speed change in progress (M2021 to M2028/M2021 to M2024) corresponding to the axis designated in (D) is turned ON.
2) A command to change the currently effective positioning speed to the speed stored in the speed change register for the axis designated in (D) is issued.
3) The speed change in progress flag is turned OFF.

(2) The numbers of registers used for present value change and speed change operations are indicated in the table below. (For details, see Section 3.2.2.)

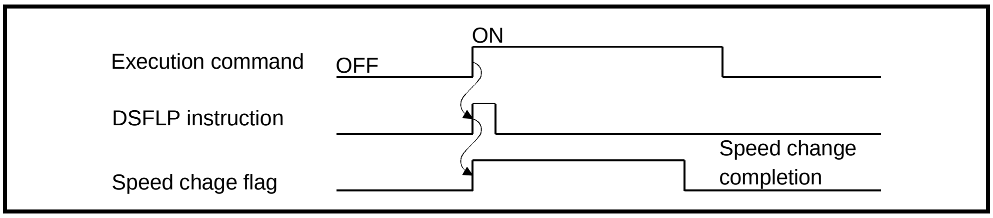
<A172SHCPUN>

Axis No.	Speed Change Registers	
	Upper	Lower
Axis 1	D963	D962
Axis 2	D969	D968
Axis 3	D975	D974
Axis 4	D981	D980
Axis 5	D987	D986
Axis 6	D993	D992
Axis 7	D999	D998
Axis 8	D1005	D1004

<A171SHCPUN>

Axis No.	Speed Change Registers	
	Upper	Lower
Axis 1	D963	D962
Axis 2	D969	D968
Axis 3	D975	D974
Axis 4	D981	D980

[Operation Timing]



[Data Settings]

(1) Setting the axis for which the speed change is to be executed
The axis for which the speed change set in (D) is executed is set as follows.

D □

- □ → Started axis No.
  - The relevant axis No. can be set in the range 1 to 4 or 1 to 8.
  - Set the interpolation control time for one of the axes controlled in interpolation.
- D → Devices symbol (only D can be set)

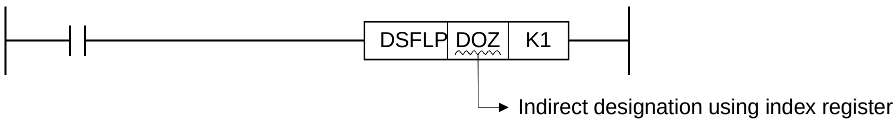
Example

The started axis is designated as follows.
- Axis 1 ....D1
- Interpolation control with axis 1 and axis 2 ....D1 or D2

(2) Speed change
The setting for a present value change/speed change is as follows.
- Speed change........................Set K1 or H1.

POINT

When using a DSFLP instruction, it is not possible to indirectly designate (D) or n using index registers (Z, V).



If an indirect designation with an index register is made, an operation error occurs, and the DSFLP instruction is not executed.

[Error Details]

(1) In the following cases an operation error occurs and the DSFLP instruction is not executed.
- When the setting for (D) is other than 1 to 8/1 to 4.
- When the setting for n is a value other than 1 and 2.
- When the setting for (D) or n has been indirectly designated using an index register (Z, V).

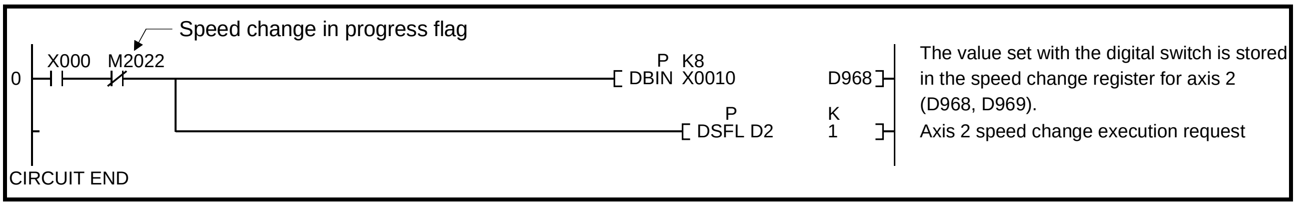
(2) In the following cases, a minor error (error on control change) occurs and the speed change is not executed.
When this happens, the error detection flag (M1607+20n) is turned ON and the error code is stored in the minor error code area for the relevant axis.
- When the axis designated in (D) is executing a home position return when the speed change is made.
- When the axis designated in (D) is decelerating when the speed change is made.
- When the absolute value of speed designated in n exceeds the speed limit value when the speed change is made.

[Program Example]

The program shown below changes the positioning speed of axis 2 to the value set with an 8-digit digital switch.

(1) Conditions
1) Numbers of inputs for the digital switch......... X010 to X02F
2) Speed Change command.............................. Leading edge (OFF→ON) of X000

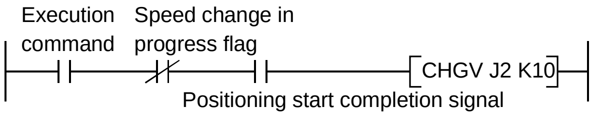
(2) Program example

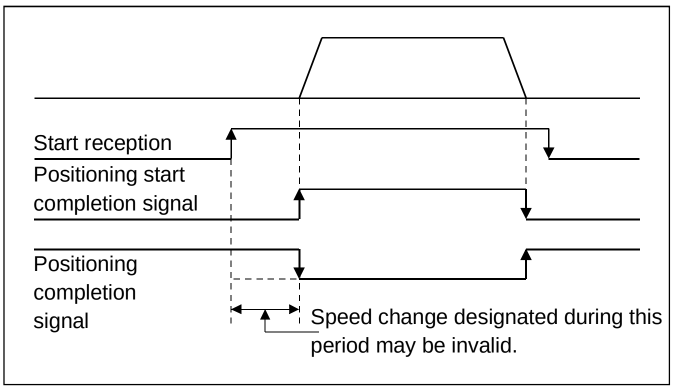


POINT

- Points to note when a speed change is performed
  - If a speed change instruction (CHGV) is executed in the period between execution of the servo program start request instruction (SVST/DSFRP) and the point where the "positioning start completion signal" comes ON, the speed change may be invalid. To perform speed changes in approximately the same timing as a start, be sure to enter the positioning start completion signal ON status as an interlock for execution of the speed change instruction.

Example)

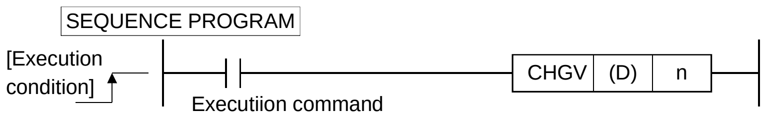




5.4.2 CHGV instruction

	Usable Devices																				Digit Designation	Number of Steps	Subset	Index	Carry Flag	Error Flag	
	Bit Devices							Word (16 Bit) Devices									Constants		Pointers		Level						
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K	H	P	I	N					M9012	M9010	M9011
(D)																							7		○		○	○
n										○	○	○					○	○										

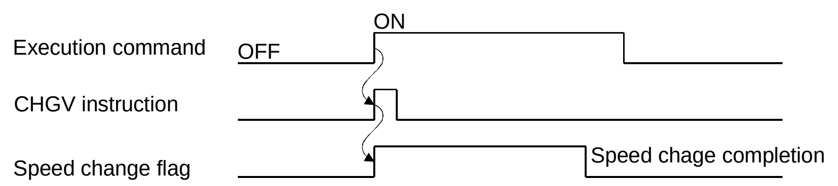
SEQUENCE PROGRAM



	Setting data	Setting range	
(D)	J + No. of speed change axis	J1 to J8 (A172SHCPUN) J1 to J4 (A171SHCPUN) J1 to J32 (A273UHCPU (32 axis feature)/A173UHCPU (S1))	
n	Setting of speed to be changed (Indirect designation device uses 2 words)	Direct designation	mm : –600000000 to 600000000 $\times10^{-2}$mm/min inch : –600000000 to 600000000 $\times10^{-3}$inch/min deg : –2147483648 to 2147483647 $\times10^{-3}$deg/min
		Indirect designation	D0 to D498 W0 to W3FE R0 to R8190

(1) The following processing is executed at the leading edge (OFF→ON) of the CHGV intruction:
   1) The speed change flag (M2021 to M2028/M2021 to M2024/M2061 to M2092) corresponding to the axis designated in (D) is turned ON.
   2) The speed of the axis designated in (D) is changed to the present value designated in n.
   3) The speed change in progress flag is turned OFF.

[Operation Timing]



[Data Settings]

(1) Setting the axis for which a speed change is to be executed
The axis with respect to which the speed change set in (D) is to be executed is set as follows.

J □

- □ → Started axis No.
  - Set the relevant axis No. in the range 1 to 4 / 1 to 8.
  - Set the interpolation control time for one of the axes involved in the interpolation.
- J → Only J can be used.

---- Example ----

Axes to be started are designated as shown below.
- Axis 1 .... J1

(2) Setting the speed change
There are two types of setting for speed changes: direct setting and indirect setting.

(a) In direct setting, the speed to be changed to is specified directly as a numerical value. (For the setting range, refer to Section 3.2.2.).

---- Example ----

If the speed to be changed "10", the setting as follows.
- When designated with a K device................ K10

(b) The word devices that can be used are indicated in the table below.

1) The word devices that can be used are indicated in the table below.

Word Device	Usable Devices	
	A172SHCPUN/ A171SHCPUN	A273UHCPU (32 axis feature)/ A173UHCPU (S1)
D	0 to 498	1690 to 8190
W	0 to 3FE	0 to 1FFF
R	0 to 8190	0 to 8190

---- Example ----

Make the following setting to designate the present value to be changed to with the data stored in data register D50:
- Designated with a word device [CHGV | J11 | D50]

2) An index register (Z, V) can be used for index designation of the indirectly set word device.

[Error Details]

(1) In the following cases an operation error occurs and the CHGV instruction is not executed.
- When the setting for (D) is other than J1 to J8/J1 to J4. (A172SHCPUN/A171SHCPUN)
- When the setting for (D) is other than J1 to J32. (A273UHCPU (32 axis feature)/A173UHCPU (S1))

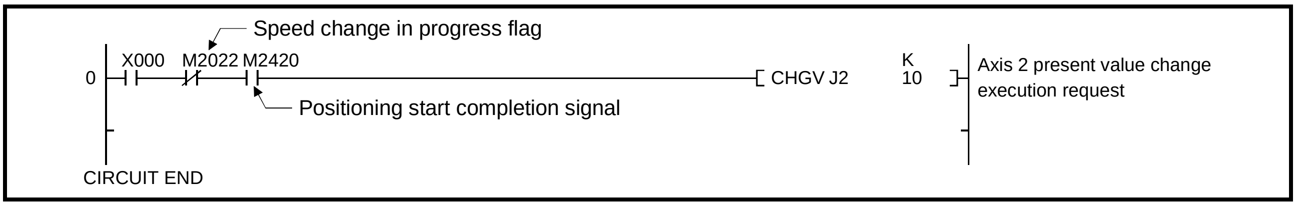
(2) In the following cases, a minor error (error on control change) occurs and the speed change is not executed.
When this happens, the error detection flag (M1607+20n/M2407+20n) is turned ON and the error code is stored in the minor error code area for the relevant axis.
- When the axis designated in (D) is executing a home position return when the speed change is made.
- When the axis designated in (D) is decelerating when the speed change is made.
- When the speed designated by n is outside the range of 0 to the speed limit value when the speed change is made.

[Program Example]

The program shown below changes the present value for axis 2.

(1) Conditions
1) Speed change command.............................. Leading edge (OFF→ON) of X000

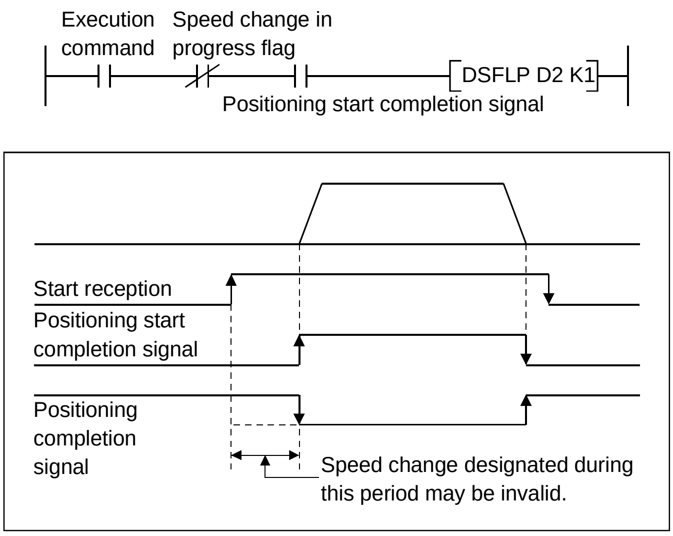
(2) Program example



POINT

- <u>Points to note when a speed change is performed</u>
  - If a speed change instruction (DSFLP) is executed in the period between execution of the servo program start request instruction (SVST/DSFRP) and the point where the "positioning start completion signal" comes ON, the speed change may be invalid. To perform speed changes in approximately the same timing as a start, be sure to enter the positioning start completion signal ON status as an interlock for execution of the speed change instruction.

Example)



5.5 Moving Backward during Positioning

When a speed change is made to a negative speed by the CHGV instruction, the travel direction can be changed to the direction opposite to the intended positioning direction.
Operation for each instruction is as follows.

<table>
<tr><th>G Code Instruction</th><th colspan="2">Operation</th></tr>
<tr><td>G00
G28 (high-speed home position return)
G30
G53</td><td colspan="2" rowspan="2">The axis is reversed in travel direction, returns to the positioning start point at the specified speed, and stops (stands by) there.</td></tr>
<tr><td>G02
G03</td></tr>
<tr><td>G01
G32</td><td colspan="2">The axis is reversed in travel direction, returns to the preceding point at the specified speed, and stops (waits) there.</td></tr>
<tr><td>G25</td><td rowspan="2">Speed change cannot be made.</td><td>Minor error 310 occurs.</td></tr>
<tr><td>G28 (dog, count type home position return)</td><td>Minor error 301 occurs.</td></tr>
<tr><td>JOG operation</td><td>Speed change to negative speed is not made.
Speed is controlled at speed limit value.</td><td>Minor error 305 occurs.</td></tr>
</table>

(Reference) Minor error 301: Speed change was made during home position return.
Minor error 305: Preset speed is outside the range of 0 to speed limit value.
Minor error 310: Speed change was made during high-speed oscillation.

[Control Details]

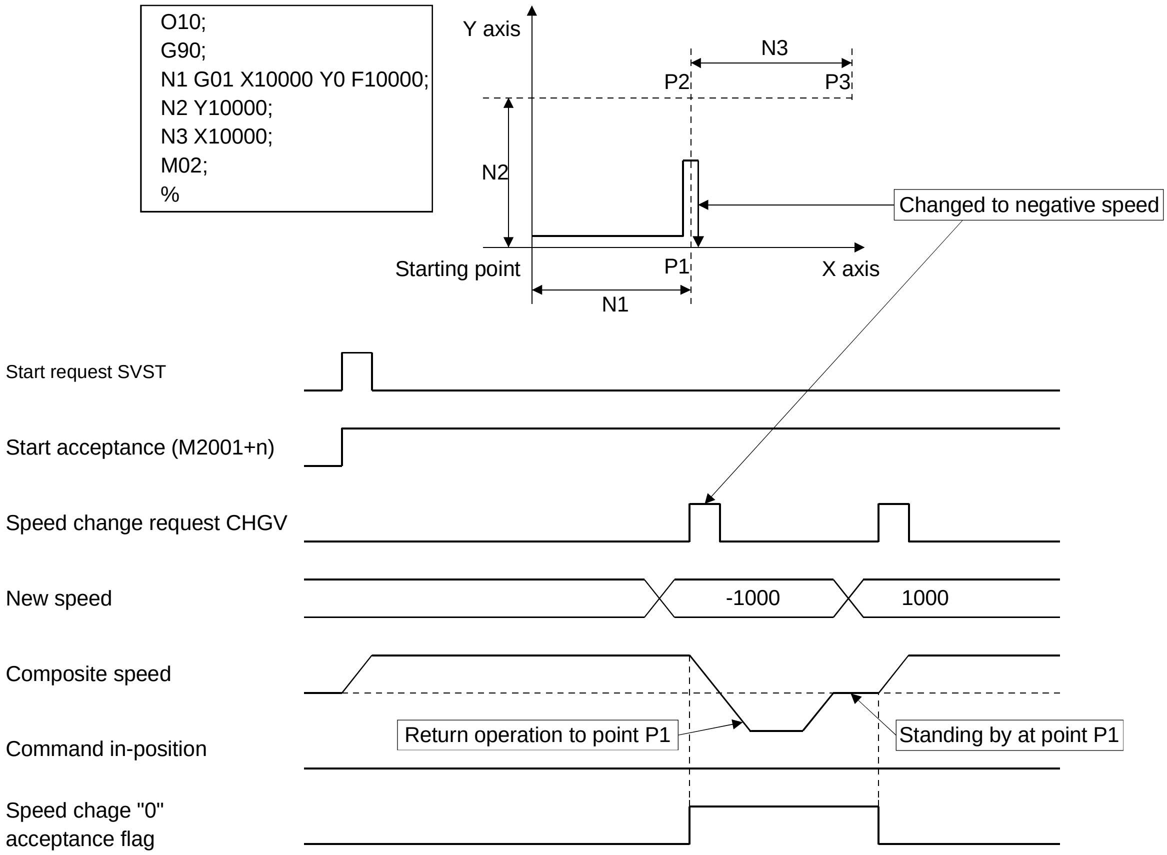
(1) When a speed change is made to negative speed, speed is controlled as listed above according to the G code in execution.

(2) The backing command speed is the absolute value of the new speed. If it exceeds the speed limit value, minor error 305 occurs and the speed is controlled at the speed limit value.

(3) When the axis is standing by at the return position
(a) Signal states
- Start acceptance (M2001+20n) ON (Remains unchanged from before execution of CHGV)
- Positioning start completion (M1600+20n/M2400+20n) ON (Remains unchanged from before execution of CHGV)
- Positioning completion (M1601+20n/M2401+20n) OFF
- In-position (M1602+20n/M2402+20n) OFF
- Command in-position (M1603+20n/M2403+20n) OFF

(b) When making a restart, make a speed change to positive speed.
(c) When terminating positioning, turn ON the stop command.
(d) A speed change made to negative speed again will be ignored.

[Operation Example under G01]



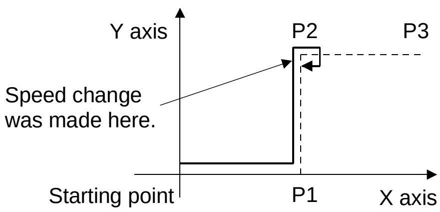
When a speed change is made to negative speed during positioning to P2 in the N2 block as shown above, the axis returns to P1 along the track specified in the program and stands by at P1.

(1) While the axis is standing by after returning to P1, a speed change to negative speed is invalid (ignored) if it is made again.

(2) While the axis is standing by at P1, the start acceptance (M2001+n) remains ON. To terminate positioning at this point, turn ON the stop command.

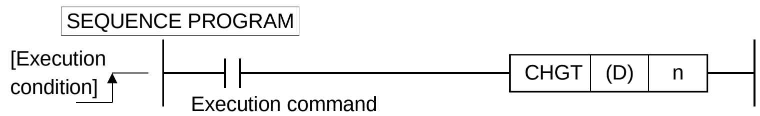
(3) A speed change to negative speed is ignored if it is made while the axis is waiting for FIN during a stop using the M code FIN waiting function under constant-speed control.

(4) In the above example, the axis returns to P2 if the axis passes through P2 during a speed change made to negative speed immediately before P2.



5.6 CHGT Instruction

	Usable Devices																					Digit Designation	Number of Steps	Subset	Index	Carry Flag	Error Flag	
	Bit Devices							Word (16 Bit) Devices									Constants		Pointers		Level							
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K	H	P	I	N					M9012	M9010	M9011
(D)																							7		○		○	○
n										○	○	○					○	○										

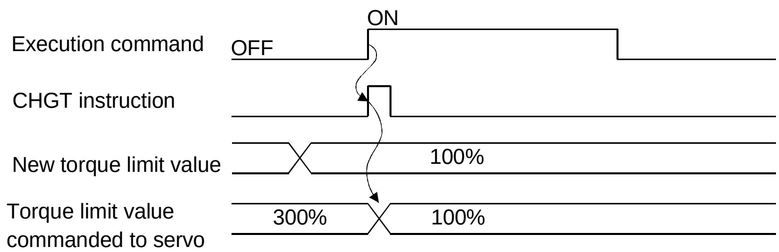


	Setting data	Setting range
(D)	J + axis No. which will be changed in torque limit value	J1 to J8 (A172SHCPUN) J1 to J4 (A171SHCPUN) J1 to J32 (A273UHCPU (32 axis feature)/A173UHCPU (S1))
n	New torque limit value (%) (Indirect designation device uses 1 word)	1 to 500 (%)

This instruction changes the torque limit value on the leading edge (OFF to ON) of the CHGT instruction execution command in the sequence program.

[Operation Timing]

Any axis that has completed starting may be changed in torque limit value in any of the operating, stopping, servo ON and servo OFF statuses.



[Operation Details]

If any torque limit value has been set in the motion program, the torque limit value cannot be changed to the value higher than the new torque limit value specified in the CHGT instruction. (The torque limit value can changed to the value lower than the new torque limit value specified in the CHGT instruction.)

(1) If the torque limit value is changed by the CHGT instruction before a motion program start or JOG operation start, the torque limit value is clamped at the torque limit value specified in the CHGT instruction when the torque limit value set in the motion program to be started is higher than that limit value.

(2) During interpolation operation, the above clamp processing of the torque limit value is performed only for the axis whose torque limit value has been changed by the CHGT instruction.

(3) When the torque limit value is set at a mid point under constant-speed control, the torque limit value cannot be changed to a value higher than the torque limit value specified in the CHGT instruction.

(4) While the motion program is running the CHGT instruction also allows the torque limit value to be changed to a value higher than the torque limit value set in that motion program.

[Error Details]

(1) The setting range is 1 to 500(%).
If the setting is outside this range, the minor error 311 occurs and a torque limit value change is not made.

(2) When the CHGT instruction is executed for any axis that has not yet been started, the minor error 312 occurs and a torque limit value change is

5.7 SFC Programs

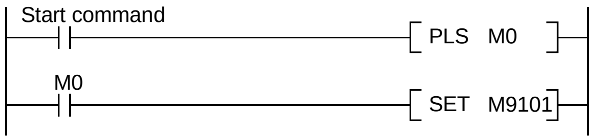
This section explains how to start motion programs using SFC programs.

5.7.1 Starting and stopping SFC programs

SFC programs are started and stopped from the main sequence program. The methods for starting and stopping SFC programs are described below.

(1) Starting SFC programs

(a) An SFC program is started by turning M9101 (SFC program start/stop) ON in the main sequence program.

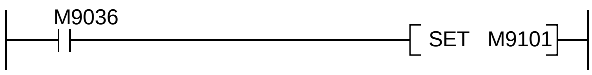


(b) There are two types of SFC program start, as indicated below, and the one that is effective is determined by the ON/OFF status of special relay M9102 (SFC program start status selection).

1) SFC program initial start
By turning special relay M9101 ON while special relay M9102 is OFF, the SFC program is started from the initial step of block 0.

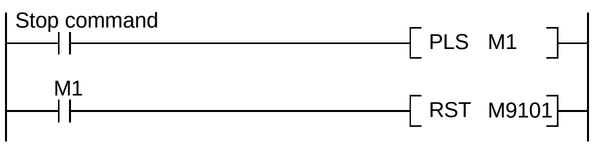
2) SFC program resumptive start
By turning special relay M9101 ON while special relay M9102 is ON, the SFC program is started from the block and step that was being executed immediately before operation was stopped.

(c) On creation of an SFC program, if no main sequence program has been created (applies only when step 0 is an END instruction), the circuit shown below is automatically created in the main sequence program area by the peripheral device.



(2) Stopping SFC programs.

(a) An SFC program is stopped by turning M9101 (SFC program start/stop) OFF in the main sequence program.



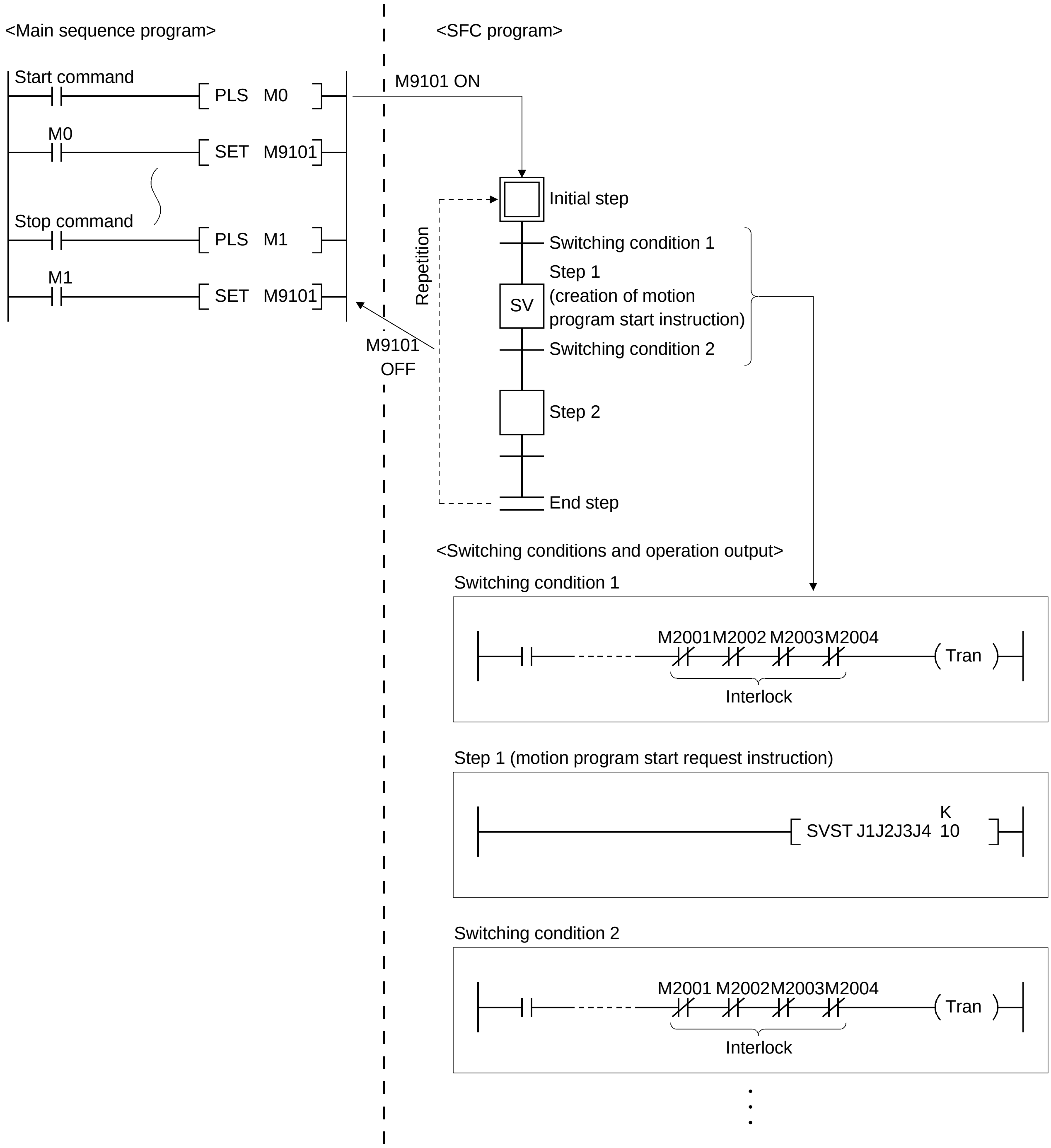
(b) When an SFC program is stopped, all the operation outputs in the step being executed are turned OFF.

POINT
Write during run in the SFC mode is not possible with respect to the motion controller.

5.7.2 Motion program start request

A motion program can be started in one of two ways: by using the program start-up symbol intended for this purpose ([SV]), or by inputting a motion program start request instruction in the internal circuit of a normal step.(☐)

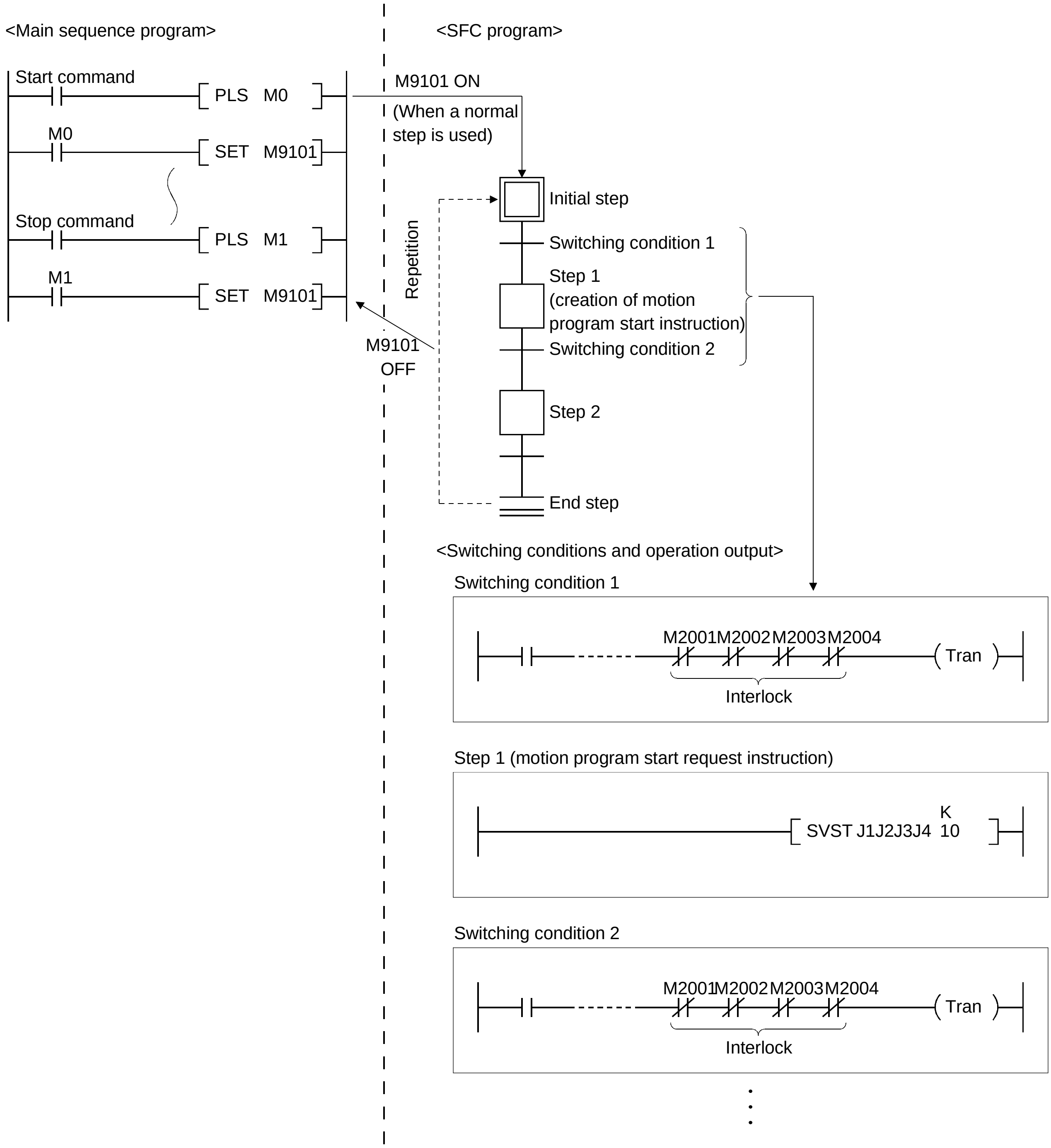
(1) When an [SV] step is created.



POINT

(1) When an [SV] step is created, the motion program start request ladder block (├──[SVST ***]┤) is mandatorily inserted in the sequence program.
(2) When a DSFRP instruction is used, input it directly into the sequence program at a normal step (□).
(3) If an SVST instruction is edited and converted, a start accept bit (M2001 to M2008/M2001 to M2004) is automatically inserted into the switching conditions before and after the relevant SFC step to act as an interlock. However, if the order of steps has been changed by addition or insertion, this interlock may not be automatically added/deleted in the switching conditions. Therefore, if a step has been added or inserted, always display the switching conditions using ZOOM display and check the interlock.
(4) Only the sequence (├──[SVST ***]┤) can be set at an [SV] step. If any additional instructions are to be set, either set them in a normal step (□) or set another sequence instruction section executed in parallel as a normal step (□).

(2) When a motion program start instruction is input inside a normal step (□)



POINTS

(1) When a DSFRP or DSFLP instruction is used, input it directly into the internal circuit of a normal step (□).

(2) If an SVST/DSFRP instruction is edited and converted, a start accept bit (M2001+n) is automatically inserted into the switching conditions before and after the relevant SFC step to act as an interlock.

(3) If a DSFLP instruction is edited and converted, a speed change in progress flag (M2021 to M2028/M2021 to M2024) is automatically inserted into the switching conditions before and after the relevant SFC step to act as an interlock.

(4) Set commands such as speed change commands and stop commands, which are executed in an arbitrary timing, in the main sequence program.

6. MOTION PROGRAMS FOR POSITIONING CONTROL

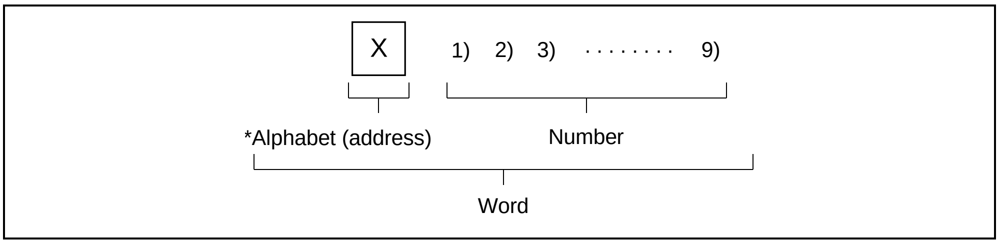
The motion controller (SV43) uses a motion program in the NC language (EIA) format as a programming language.
A motion program is used to specify the positioning control type and positioning data required for the servo system CPU to exercise positioning control. The makeup and specifying method of a motion program will be described.

6.1 Motion Program Makeup

This section provides the format and makeup of a motion program.
A motion program is called a word address format, which consists of a single alphabet (address) and numerals.

(1) Word and address
A word is a collection of characters arranged in given order and this is used as a unit to process that information to perform a specific operation.
In the motion controller (SV43), a word is made up of a single alphabet (address) and a subsequent several-digit number. (The number may be headed by a "+" or "-" sign.)

<Word makeup>



* The alphabet at the beginning of a word is called an address and defines the meaning of the subsequent numeric information.

(2) Block

A block is a collection of several words. It includes information necessary to perform a single specific operation of a machine and acts as a complete command on a block basis.

A block is ended by the EOB (End Of Block) code to indicate separation.

<Block makeup>

$$\underbrace{\underbrace{\overset{1)}{\text{N100}}}_{\text{Word}}\quad \underbrace{\overset{2)}{\text{G01}}}_{\text{Word}}\quad \underbrace{\overset{3)}{\text{X250.}}}_{\text{Word}}\quad \underbrace{\overset{4)}{\text{Y-123.4}}}_{\text{Word}}\quad \underbrace{\overset{5)}{\text{F1500.}}}_{\text{Word}}\quad \underbrace{\overset{6)}{;}}_{\text{E0B}}}_{\text{Block}}$$

1) N100 .... Sequence number : Used to identify a program block and represented by a number (max. 4 digits) after alphabet N.

2) G01 .... Preparatory code : Denotes the basic instruction which commands the motion of motion control. (G code)

3) X250. .... Coordinate position data* : Indicates the command for the coordinate position of the X axis. This word commands 250mm of the X axis.

4) Y-123.4 .... Coordinate position data* : Indicates the command for the coordinate position of the Y axis. This word commands -123.4mm of the Y axis.

5) F1500.......... Feedrate : Represents the command of feedrate in linear or circular interpolation. (F code) This word indicates the speed of 1500mm per minute.

6) ;................... EOB (End Of Block) : Denotes the end (separation) of a program block.

* The coordinate position data has the following two modes.

Incremental value command .... Mode in which a command of the next target position is given on the basis of the present position (G91)

Absolute value command .... Mode in which the axis moves to the specified coordinate position independently of the present position (G90)

(3) Motion program
A motion program is a collection of several blocks and commands a series of operations.

<Motion program makeup>

```
00001 O100;  ------------------------------------- 1) Motion program number
00002 N10 G91 G00;
00003     G28 X0. Y0.;
00004     X250.;
00005 N20 M20;
00006     X-50. Y120.;  ---------------------------- 2) Program block
00007 N30 G01 X25. F500.;
  ⋮         ⋮
00020 N80 M21;
00021     M02;
00022     %  ---------------------------------------- Indicates a program end.
  └--------------------------------------------- 3) Line number
```

1) Motion program number ....Number specified in a sequence program. You can set alphabet O (oh) and any number of 1 to 256.

2) Program block ....Consists of multiple program blocks necessary for motion operations in control order.

3) Line number............................Automatically displayed in serial number when a motion program is created on the peripheral device.

POINT
The motion controller (SV43) can store up to 256 motion programs in memory. These motion programs are managed using motion program numbers.

6.2 Instructions for Creating Motion Programs

(1) A motion program cannot be rewritten during its execution.
Write a program after making sure that the PC ready flag (M2000) is OFF.

(2) Calling of a subprogram from another subprogram (nesting) is allowed up to eight levels.

(3) In one block, one G code can be selected from each modal group. Up to two G codes can be commanded.
For G code combinations, refer to Table 6.1.

Table 6.1 G Code Combination List

First G Codes \ Second G Codes	G00	G01	G02	G03	G04	G09	G28	G43	G44	G49	G53	G54	G55	G56	G57	G58	G59	G61	G64	G90	G91	G92
G00					○			○	○	○												
G01					○			○	○	○												
G02					○																	
G03					○																	
G04																						
G09		○	○	○																		
G23																						
G24																						
G25																						
G26																						
G28											○											
G30											○											
G32																						
G43																						
G44																						
G49							○															
G53							○															
G54	○	○	○	○																		○
G55	○	○	○	○																		○
G56	○	○	○	○																		○
G57	○	○	○	○																		○
G58	○	○	○	○																		○
G59	○	○	○	○																		○
G61	○	○	○	○																		
G64	○	○	○	○																		
G90	○	○	○	○																		
G91	○	○	○	○																		
G92																						

How to use the above table
(a) When G09 is specified as the first G code, G01, G02 or G03 may be specified as the second code.

IMPORTANT

If motion programs are specified for the same axis, they cannot be run concurrently.
If they are run, we cannot guarantee their operations.

(b) When G90 is specified as the first G code, G00, G01, G02 or G03 may be specified as the second code.
G90 G61; and G90 G64; result in a format error.

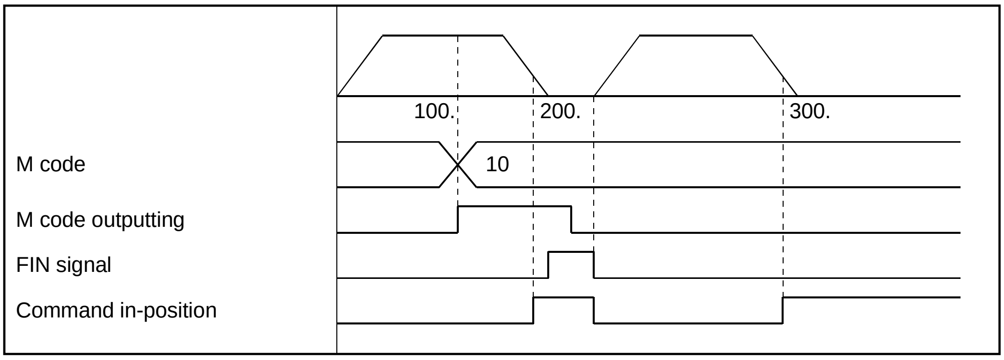
(4) With the exception of M00, M01, M02, M30, M98, M99 and M100, the M code may be specified with another command in the same block. However, if it is specified together with the move command (G00 to G03), operation is performed as follows.
- The M function is executed simultaneously with the move command (G00 to G03, G32).

(5) With the exception of M00, M01, M02, M30, M98, M99 and M100, multiple M codes may be specified in one block but only the last one is valid.

(6) When there is the miscellaneous function (M) at any point in continuous G01 blocks
If the M code is set at any point in continuous G01 blocks, operation is performed in either of the following two ways.

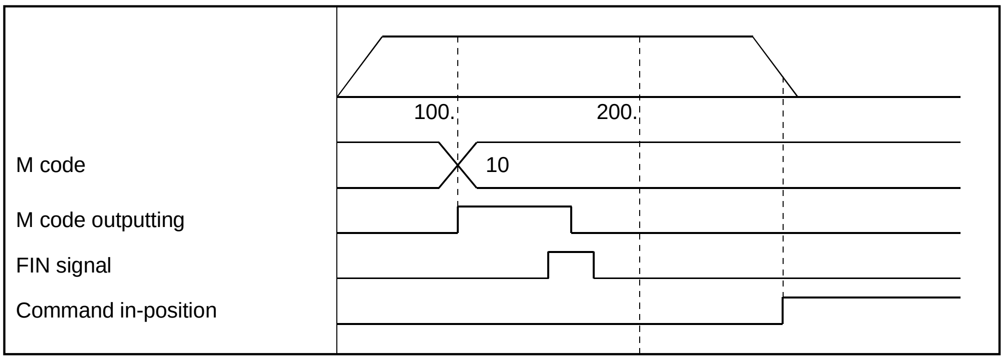
O100;	
1) G90 G01 X100. F1000.;	CP positioning of X
2) X200. M10;	CP positioning of X, M code
3) X300.;	CP positioning of X

(a)



When the FIN signal is not turned from OFF to ON to OFF during positioning in block 2), the axis decelerates to a stop once in the block of the M code.

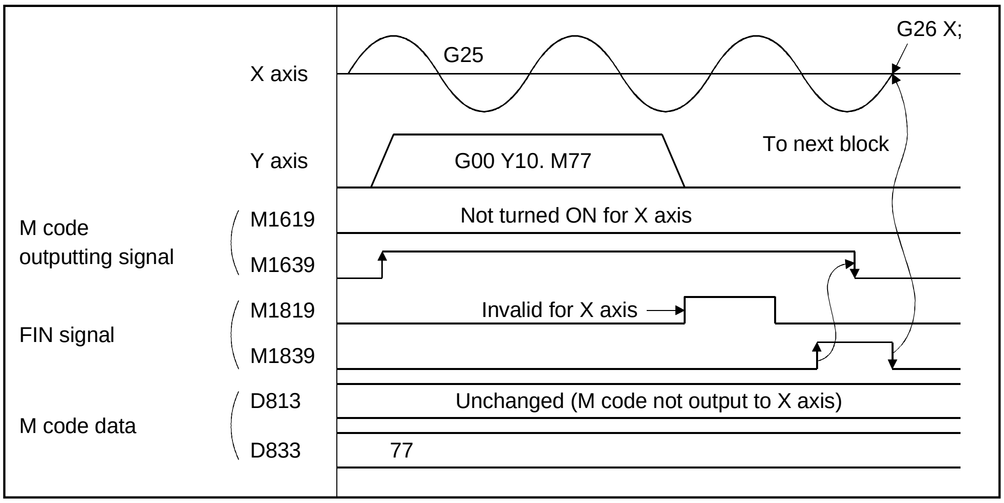
(b)



When the FIN signal is turned from OFF to ON to OFF during positioning in block 2), the axis performs CP operation without decelerating to a stop in the block of the M code.

(7) With the exception of M00, M01, M02, M30, M98, M99 and M100, the M code is output to the data registers (D813, D833, ...) and axis input signals (M code outputting signals: M1619+20n) of all axes specified in the SVST instruction. However, the data register data and axis input signals are not output to the axis in execution of high-speed oscillation. Also, the FIN signal (M1819+20n) entered into the axis in execution of high-speed oscillation is invalid.
(Program No. 1 is started with X (axis 1) and Y (axis 2) specified
-[SVST J1J2 K1]-)

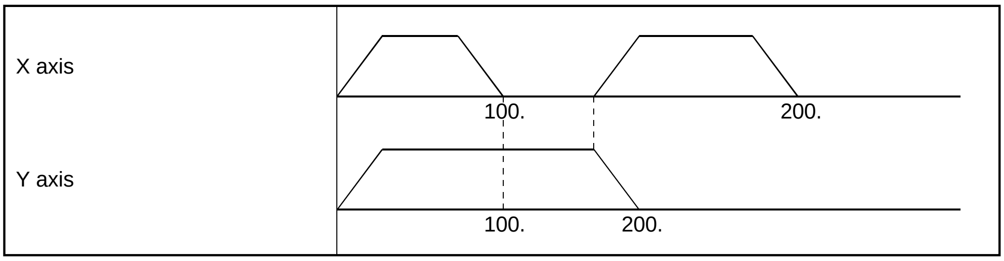
01;	
N1 G25 X START90. STRK10. F30;	X-axis high-speed oscillation start
N2 G00 Y10. M77;	PTP positioning of Y axis
N3 G26 X;	X-axis high-speed oscillation stop
M02;	
%	



(8) Acceleration/deceleration processing of G01

G91 G01 X100. Y100. F100.;	CP positioning of X, Y.........Block 1
Y100.;	CP positioning of Y ....Block 2
X100.;	CP positioning of X ....Block 3

When the above program is run, the acceleration/deceleration processings of the X and Y axes are as follows.



- Both the acceleration and deceleration times are equal to the acceleration time of the parameter block.
- When the M code is commanded in G00, the acceleration and deceleration times are also equal to the acceleration time of the parameter block as in G01. (Example: G00 X□ M□;)
- In G02, G03 and G32, the acceleration and deceleration times are also equal to the acceleration time of the parameter block as in G01.

(9) Operation of G09 (exact stop check)
Since a shift cannot be made by the command in-position, execution shifts to the next block after the command is given.

(10) G28 (home position return) operation
The axis whose home position return request signal (M1609+20n) is ON makes a dog, count or data setting type home position return.
The axis whose home position return request signal (M1609+20n) is OFF makes a high-speed feed home position return.

(11) Checking the used axes at program start
(a) If there is an axis used in the already started program and an attempt is made to start that axis in another program, that program cannot be run because an error (error code: 101) occurs at execution of the SVST instruction.

(b) If the axis not specified in the axis number setting of the SVST instruction in the program waiting to be started is described in the motion program, the corresponding axis in the program stops due to an error (error code: 594) when its positioning processing is started.

(12) Variable prereading
Variables in up to eight blocks including the one currently executed are preread. Where possible, set variables before starting the program.

(13) About the motion program including high-speed oscillation
Note the following when high-speed oscillation (G25) is to be performed for all axes specified in SVST.
(Program No. 1 is started with X (axis 1) and Y (axis 2) specified
-[SVST J1J2 K1]-)

```
01 ;
N1 G25 X START90.  STRK10. F30; @@X-axis high-speed oscillation start
N2 G25 Y START90.  STRK20. F10; @@Y-axis high-speed oscillation start
N3 ←———————— Be careful when programming N3 and later.
          ⋮
```

(a) The G code instructions other than G26 (high-speed oscillation stop) and G04 (dwell) should not be executed.

(b) The M codes other than M00, M01, M02, M30, M98 and M99 should not be executed.

6.3 G Code List

Table 6.2 indicates the G codes used in motion programs.

Table 6.2 G Code List

G Code	Group*	Function
G00*	01	PTP positioning at rapid feedrate
G01		CP positioning at speed specified in F
G02		Circular interpolation (CW)
G03		Circular interpolation (CCW)
G04	00	Dwell (standby)
G09	00	Exact stop check When G01 blocks continue, a stop is made at each block before execution of the next block.
G23*	02	Cancel, cancel start invalid
G24		Cancel, cancel start
G25	00	High-speed oscillation
G26	00	High-speed oscillation stop
G28	00	Home position return (positioning to home position address at rapid feedrate at the second time and later)
G30	00	Second home position return (positioning to second home position address at rapid feedrate)
G32	00	Skip
G43	08	Tool length offset (+)
G44		Tool length offset (-)
G49*		Tool length offset cancel
G53	00	Machine coordinate system selection
G54*	12	Work coordinate system 1 selection
G55		Work coordinate system 2 selection
G56		Work coordinate system 3 selection
G57		Work coordinate system 4 selection
G58		Work coordinate system 5 selection
G59		Work coordinate system 6 selection
G61	13	Exact stop check mode (stopped when G01 continues)
G64*		Cutting mode (not stopped when G01 continues)
G90*	03	Absolute value command
G91		Incremental value command
G92	00	Coordinate system setting Work coordinate system is shifted by setting virtual mechanical coordinate system.
G100	——	Time-fixed acceleration/deceleration switch-over instruction
G101	——	Acceleration-fixed acceleration/deceleration switch-over instruction

* indicates the G code selected at power-on.

*The above groups will be described.

Class	Description
Modal G codes (Groups 01, 02, 03, 08, 12, 13)	Once any G code is commanded, it is valid until another G code in the same group is commanded. Initial status (at power-on) is as follows. Group 01..........G00 (PTP positioning at rapid feedrate) Group 02..........G23 (Cancel, cancel start invalid) Group 03..........G90 (Absolute value command) Group 08..........G49 (Tool length offset cancel) Group 12..........G54 (Word coordinate system 1 selection) Group 13..........G64 (Cutting mode)
Unmodal G codes (Group 00)	Valid only for the block in which any G code has been commanded.

6.4 Special M Code List

Table 6.3 indicates the special M codes used in motion programs.

Table 6.3 Special M Code List

M Code	Function	Remarks
M00	Program stop	Executing this code stops the program at the end of that block.
M01	Optional program stop	Has the same function as M00 if M1501+10n is ON. Invalid if it is OFF.
M02	Program end	Specify M02/M30 at program end.
M30	Program end	Specify M02/M30 at program end.
M98	Subprogram call	
M99	Subprogram end	
M100	Preread inhibit	

• Special M codes are not output to the PC.

6.5 Instruction Symbol/Character List

Table 6.4 indicates the instruction symbols/characters used in motion programs.

Table 6.4 Instruction Symbol/Character List

Symbol/Character	Function	Description
A	Coordinate position data	Symbols used to specify the axes to be moved when commanding positioning. Set the axis numbers and axis names in system settings.
B	Coordinate position data	
C	Coordinate position data	
U	Coordinate position data	
V	Coordinate position data	
W	Coordinate position data	
X	Coordinate position data	
Y	Coordinate position data	
Z	Coordinate position data	
I	Circular arc center coordinate 1	Used in G02 or G03 (arc center coordinate designation).
J	Circular arc center coordinate 2	
R	Radius of R point-designated circular arc	Used in G02 or G03 (R designation).
F	Interpolation feed composite speed	Used in G01, G02 or G03.
G	Preparatory function (G code)	Refer to Section 6.3 G Code List.
H	Subprogram call sequence number	Used in M98.
	Tool length offset data number	Used in G43 or G44.
L	Subprogram repeat count	Used in M98.
M	Miscellaneous function (M code)	Refer to Section 6.4 Special M Code List and Section 6.9.
N	Sequence number	Indicates a sequence number.
O	Program number	Indicates a motion program number.
P	Dwell timer	Used in G04.
	Start program No.	Used in G24.
	Subprogram call number	Used in M98.
PB	Parameter block No.	Changes the parameter block.
TL	Torque limit value	Changes the torque limit value.
+	Addition	Used in arithmetic operation commands.
-	Subtraction	
*	Multiplication	
/	Division	
	Optional block skip	Optional block skip is specified for a block which is headed by this symbol. (Refer to Section 3.1.29.)
MOD	Remainder	Used in arithmetic operation commands.
(,)	Comment	Gives comment in the inside of parentheses.
[,]	Brackets	Used in conditional expressions.
#	Variable	Symbols used for indirect designation.
	Device designation	
%	Program end code	Indicates the end of a program.
;	Block separation	Indicates separation of blocks.
IF	Condition	Used in conditional branch instructions.
THEN		
ELSE		
GOTO	Jump	
WHILE	Repeat	
DO		
END		

- Multiple operators cannot be used in one block.
- For the instruction symbol setting ranges, refer to Section 6.6.4.

Table 6.4 Instruction Symbol/Character List (Continued)

Symbol/Character	Function	Description
EQ	Comparison instruction (=)	Used in comparison instructions.
NE	Comparison instruction (!=)	
GT	Comparison instruction (>)	
LT	Comparison instruction (<)	
GE	Comparison instruction (>=)	
LE	Comparison instruction (<=)	
OR	Logical operation instruction (OR)	Used in arithmetic operation commands.
XOR	Logical operation instruction (exclusive OR)	
AND	Logical operation instruction (AND)	
SIN	Trigonometric function (sine)	
COS	Trigonometric function (cosine)	
TAN	Trigonometric function (tangent)	
ASIN	Trigonometric function (arcsine)	
ACOS	Trigonometric function (arccosine)	
ATAIN	Trigonometric function (arctangent)	
INT	Numerical conversion (real number to integer)	
FLT	Numerical conversion (integer to real number)	
SET	Specified device ON	Used in extended control instructions.
RST	Specified device OFF	
CAN	Cancel device designation	Used in G24.
START	Starting angle designation	Used in G25.
STRK	Amplitude designation	
SKIP	Skip device designation	Used in G32.

- Multiple operators cannot be used in one block.
- For the instruction symbol setting ranges, refer to Section 6.6.4.

6.6 Method for Setting Positioning Data

This section explains how to set the positioning data (addresses, speeds, operational expressions) used in motion programs.
There are the following two ways to set the positioning data.
• Direct designation (entering numerical values for data setting)
....Refer to Section 6.6.1.
• Indirect designation (using variable: #**** or device: #W*** for data setting)
.... Refer to Section 6.6.2.
"Direct designation" and "indirect designation" can be used together in one motion program.

6.6.1 Direct designation (numerical value)

Direct designation is a way to set each positioning data with a numerical value, and these data are fixed data. Data setting and correction may be made on the peripheral device only.

<Example of setting positioning data by direct designation>

```
O200;
N99 G90 G00 X100. Y110.;
     G01 X200. Y202. F204.;
     G91 G00 Z300.;
M02;
%
```

Values specified as positioning data

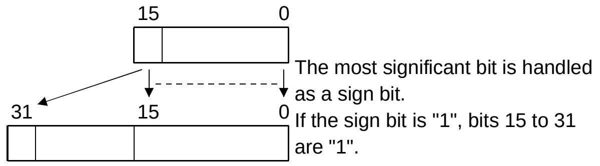
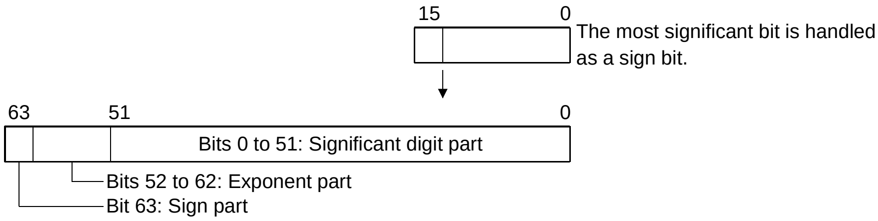
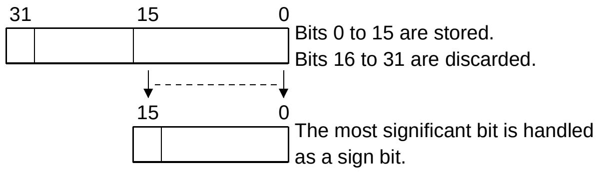
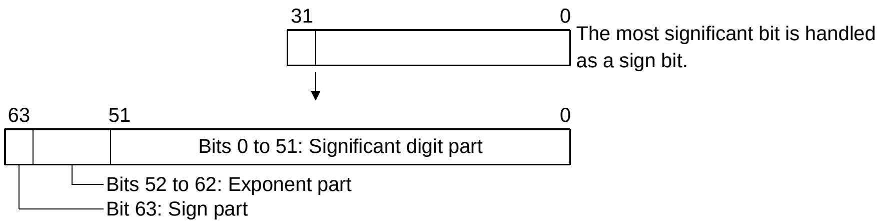
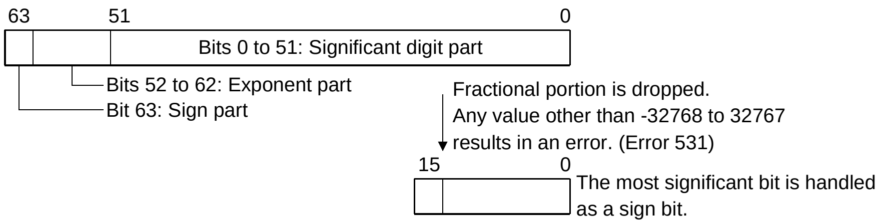
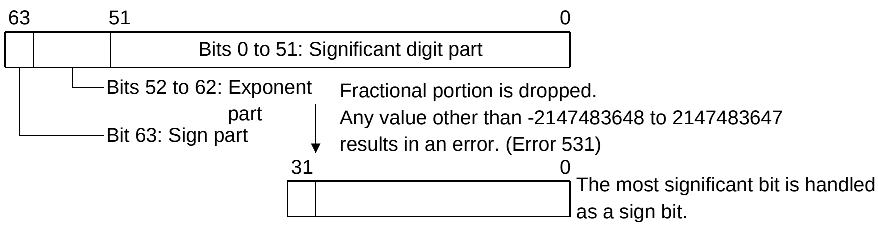
6.6.2 Indirect designation (variable: #****)

Indirect designation is a way to use variables (#****) or devices (#W****) to specify values used in the addresses, speeds and operational expressions in a motion program.
By using variables or devices to set values, multiple positioning controls can be exercised in one motion program.

(1) About variable representation
The 16-bit integer type, 32-bit integer type and 64-bit double precision real number can be handled as variables.
When handled, these variables are described as follows.

	Variable (D register)	Device (W register)
16-bit integer type	#n, #Dn, #DnS, #n: S, #Dn: S	#Wn: S
32-bit integer type	#nL, #DnL, #n: L, #Dn: L	#Wn: L
64-bit double precision real number	#nF, #DnF, #n: F, #Dn: F	#Wn: F

(2) About variable conversion
When variables of different types are used for operation, the types are matched by internal operation.
Type conversion is made by internal operation as follows.

<table>
<tr><th>Conversion Format</th><th>Description</th></tr>
<tr><td>16 bit to 32 bit</td><td>The 16-bit integer type is extended to the 32-bit integer type.

</td></tr>
<tr><td>16 bit to 64 bit</td><td>The 16-bit integer type is converted to the 64-bit double precision real number.

</td></tr>
<tr><td>32 bit to 16 bit</td><td>The 32-bit integer type is converted to the 16-bit integer type.
Note that any value other than -32768 to 32767 results in an error. (Error 531)

</td></tr>
<tr><td>32 bit to 64 bit</td><td>The 32-bit integer type is converted to the 64-bit double precision real number.

</td></tr>
<tr><td>64 bit to 16 bit</td><td>The 64-bit double precision real number is converted to the 16-bit integer type.
Note that any value other than -32768 to 32767 results in an error. (Error 531)

</td></tr>
<tr><td>64 bit to 32 bit</td><td>The 64-bit double precision real number is converted to the 32-bit integer type.
Note that any value other than -2147483648 to 2147483647 results in an error. (Error 531)

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

(3) Variable designation (#n n = integer)

(a) How to handle variable as 16-bit integer

When a #n variable is followed by "S" or ": S", it is handled as a 16-bit integer. (-32768 to 32767)

[Example]

#0	: [D0]
#1S	: [D1]
#2: S	: [D2]

- Odd numbers may be used as 16-bit designated variables.

(b) How to handle variable as 32-bit integer

Variables are handled as 32 bits. (-2147483648 to 2147483647)

[Example]

	Upper	Lower		Upper	Lower
#100: L	: [D101,	D100]	#102: L	: [D103,	D102]

- When a variable is specified as 2 words (32 bits), only an even number may be used. The data size of a variable is 4 bytes.

<Example of setting positioning data by variable designation>

```
O200;
N99 G90 G00 X#100 Y#110;
    G01 X#200 Y#202 F#204;
    #300 = #302 - #304;
    G91 G00 Z300.;
    IF [#310 EQ 1000] GOTO99;
M02;
%
```

Motion program No. (0) cannot be set indirectly.

Indirect designation (address, speed, operational expression)

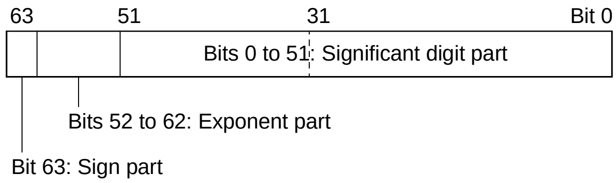
Direct designation

(c) How to handle variable as 64-bit double precision real number (#n:F)
By handling a variable as a 64-bit double precision real number, arithmetic operation spanning multiple blocks can be performed without reduction in precision.
Describe an upper-case ":F" after a #n variable.
#nF: Four variables of #n to #n+3 are used and handled as a 64-bit double precision real number.

63 31 Bit 0

#n+3	#n+2	#n+1	#n

The data format of a 64-bit double precision real number conforms to the binary floating-point type double precision (64 bits) of IEEE Standard.



[Example]

#10: F=#20: L/#22: L
The division result of 32-bit integers, [#21, #20] and [#23, #22], is stored into a 64-bit real number, [#13, #12, #11, #10].

#10: F=#20: L
A 32-bit integer, [#21, #20], is expanded in sign to a 64-bit real number, [#13, #12, #11, #10].

#40: L=#30: F
A 64-bit real number, [#33, #32, #31, #30], is expanded in sign to a 32-bit integer, [#41, #40]

<Restrictions>
Functions INT and FLT cannot use 64-bit double precision real numbers.

(4) About assignment of variable
When a decimal point is added for assignment of a value to a variable, the value is assigned as indicated below.
#10: L= 1.; → 10000 enters #10, #11.
#10: F=1.; → 10000 (64-bit double precision real number) enters #10, #11, #12, #13.

"1." is converted into a value of four decimal places.
(Converted into a value of four decimal places independently of the unit (mm, inch, degree).)

[Example]

<For command address 1>

G91;
#10: L= 1.;
G0 X#10: L ← The travel of the X axis is any of the following values.

mm	inch	degree
1mm	0.1 inch	0.1 degree

<For command address 2>

G91;
#10: L= 1.;
G0 X#10: F ← The travel of the X axis is equivalent to any of the following values if it is "#10F=1.;" (64-bit double precision real number).

mm	inch	degree
1mm	0.1 inch	0.1 degree

<For feedrate (F) 1>

G91;
#10: L= 1.;
G01 X10. F#10: L ← The feedrate (F) of X is any of the following values.

mm	inch	degree
100mm/min	10 inch/min	10 degree/min

<For feedrate (F) 2>

G91;
#10: F= 1.;
G01 X10. F#10: F ← The feedrate (F) of X is equivalent to any of the following values if it is "#10F=1.;" (64-bit double precision real number).

mm	inch	degree
100mm/min	10 inch/min	10 degree/min

(5) Device designation (#Xx, Xx is device)
The word device (D, W) or bit device (X, Y, M, TC, TT, CC, CT, B, F) of the sequence control section can be referred to by device designation.
The four fundamental operations of bit devices cannot be performed.

[Example]

#X180: X180
#M2000: M2000
#D100: L: [D101, D100] ([upper, lower])

(6) About usable device ranges
PC devices can be used to indirectly specify all the positioning addresses, command speeds, M codes and others set in a motion program.
(a) Word devices

CPU	Device	Reference Range	Writable Range
A172SHCPUN A171SHCPUN	D	0 to 799	0 to 499
	W	000 to 3FF	000 to 3FF
A273UHCPU (32-axis feature) A173UHCPU (S1)	D	0 to 8191	1690 to 8191
	W	0000 to 1FFF	0000 to 1FFF

POINT
- For two-word designation, always specify an even-numbered device.
  Also, when setting data to that device in a sequence program, always use the "DMOV(P)" instruction.

(b) Bit devices

CPU	Device	Reference Range	SET/RST Enabled Range (*1)
A172SHCPUN A171SHCPUN	X	000 to 7FF	——
	Y	000 to 7FF	000 to 7FF
	M/L	0 to 2047	0 to 1399
	M	9000 to 9255	——
	B	0 to 3FF	——
	F	0 to 255	——
	TT (timer contact)	0 to 255	——
	TC (timer coil)	0 to 255	——
	CT (counter contact)	0 to 255	——
	CC (counter coil)	0 to 255	——
A273UHCPU (32-axis feature)	X	000 to 1FFF	——
	Y	000 to 1FFF	000 to 1FFF
	M/L	0 to 8191	0 to 1999 4720 to 8191
	M	9000 to 9255	——
	B	000 to 1FFF	——
	F	0 to 2047	——
	TT (timer contact)	0 to 2047	——
	TC (timer coil)	0 to 2047	——
	CT (counter contact)	0 to 1023	——
	CC (counter coil)	0 to 1023	——

(*1) Even outside the SET/RST enabled range, an error will not occur if the bit device is within the reference range.

Conditions of SET/RST-enabled bit devices
1) Write (SET/RST) cannot be performed from both programs of sequence ladder and motion program to the same bit device (in increments of 16 points). (Write operation will not be guaranteed.)
   Therefore, the user should manage the side where write is performed. The minimum increments are 16 points.
2) When the I/O control system is the "direct mode" (A172SHCPUN/A171SHCPUN), output will not be provided to the output card of the PC slot if write to device Y is performed. To provide PC output, use the "refresh mode".

(7) Device data import
The data of the indirectly designated devices are imported by the PCPU during motion program run.
Therefore, when making indirect designation, inhibit preread of M100.The following table indicates the device data setting procedures and instructions on a starting method basis.

Starting Method	Setting procedure	Instructors
At start using SVST instruction (Indirect designation in SVST instruction)	Set data to the indirectly designated devices ↓ Start is made by SVST.	Do not change the indirectly designated devices until the "positioning start completion signal" of the started axis turns ON.
At automatic start by cancel start (Indirect designation of start program)	Set data to the indirectly designated devices set in the start program. ↓ Turn ON the cancel command device.	
After program start (Indirect designation in program)	Set command data to the indirectly designated devices. ↓ Execute M100 preread inhibit. ↓ Refer to the values set to the indirectly designated devices until M100 is executed.	Example 010; N1 G00 X0 F1000. ; N2 M100; N3 G01 X100. F1500. ; N4 G01 X#D0L F1500; N2; % Set "D0, D1" before execution of N2. They may not be reflected after execution of N2.

POINTS

(1) The motion program No. (0) cannot be set indirectly.
(2) Provide interlocks using the start acceptance signals (M2001 to M2008) to ensure that the data of the devices specified for indirect setting from being changed until the specified axes accept a start.
If the data is changed before the acceptance of a start, positioning control may not be exercised with proper values.
(3) Set a variable latch on the peripheral device.
(4) Variable designated #**** is the same in value as device-designated #D**** which uses data registers.
Example) #0=1;
#D0=2; ← The value of #0 is also 2.
(5) In variable designation or device designation using word devices, the PCPU imports the data of the specified devices (2-word or 4-word) when it runs a motion program.
When performing positioning control, therefore, a motion program start request must be made after data have been set to the indirect setting devices.

6.6.3 About operational data

(1) Four fundamental operations (+, -, *, /, MOD)
The following table indicates the data type combinations and conversion methods for four fundamental operations (+, -, *, /, MOD).

Operation result = [data 1] operator [data 2]

Operator denotes +, -, *, / or MOD.

Internal operation is performed after conversion into the type of the operation result.
If there is no operation result such as a conditional expression, internal operation is performed with 32-bit data.
For MOD, however, if the operation result type is 64-bit data with floating point, internal operation is performed with 32-bit data, which is then converted into the operation result type and stored.

n: Indicates variable number or device number.

No.	Operation Result	Data 1	Data 2
1	#n (16 bit) No conversion Error occurs if operation result exceeds 16 bit range. (Error 531)	#n (16 bit) No conversion	#n (16 bit) No conversion
2			#nL, #n: L (32 bit) 32-bit data is converted into 16-bit data. Error occurs if conversion result exceeds 16 bit range. (Error 531)
3			#nF, #n: F (64 bit) 64-bit data is converted into 16-bit data. Fractional portion is dropped during conversion. Error occurs if conversion result exceeds 16 bit range. (Error 531)
4		#nL, #n: L (32 bit) 32-bit data is converted into 16-bit data. Error occurs if conversion result exceeds 16 bit range. (Error 531)	#n (16 bit) No conversion
5			#nL, #n: L (32 bit) 32-bit data is converted into 16-bit data. Error occurs if conversion result exceeds 16 bit range. (Error 531)
6			#nF, #n: F (64 bit) 64-bit data is converted into 16-bit data. Fractional portion is dropped during conversion. Error occurs if conversion result exceeds 16 bit range. (Error 531)
7		#nF, #n: F (64 bit) 64-bit data is converted into 16-bit data. Fractional portion is dropped during conversion. Error occurs if conversion result exceeds 16 bit range. (Error 531)	#n (16 bit) No conversion
8			#nL, #n: L (32 bit) 32-bit data is converted into 16-bit data. Error occurs if conversion result exceeds 16 bit range. (Error 531)
9			#nF, #n: F (64 bit) 64-bit data is converted into 16-bit data. Fractional portion is dropped during conversion. Error occurs if conversion result exceeds 16 bit range. (Error 531)

n: Indicates variable number or device number.

<table>
<tr><th>No.</th><th>Operation Result</th><th>Data 1</th><th>Data 2</th></tr>
<tr><td>10</td><td rowspan="9">#nL, #n: L (32 bit)
(32 bit)
No conversion
Error occurs if operation result exceeds 32 bit range. (Error 531)</td><td rowspan="3">#n (16 bit)
16-bit data is converted into 32-bit data.</td><td>#n (16 bit)
16-bit data is converted into 32-bit data.</td></tr>
<tr><td>11</td><td>#nL, #n: L (32 bit)
No conversion</td></tr>
<tr><td>12</td><td>#nF, #n: F (64 bit)
64-bit data is converted into 32-bit data.
Fractional portion is dropped during conversion.
Error occurs if conversion result exceeds 32 bit range. (Error 531)</td></tr>
<tr><td>13</td><td rowspan="3">#nL, #n: L (32 bit)
No conversion</td><td>#n (16 bit)
16-bit data is converted into 32-bit data.</td></tr>
<tr><td>14</td><td>#nL, #n: L (32 bit)
No conversion</td></tr>
<tr><td>15</td><td>#nF, #n: F (64 bit)
64-bit data is converted into 32-bit data.
Fractional portion is dropped during conversion.
Error occurs if conversion result exceeds 32 bit range. (Error 531)</td></tr>
<tr><td>16</td><td rowspan="3">#nF, #n: F (64 bit)
64-bit data is converted into 32-bit data.
Fractional portion is dropped during conversion.
Error occurs if conversion result exceeds 32 bit range. (Error 531)</td><td>#n (16 bit)
16-bit data is converted into 32-bit data.</td></tr>
<tr><td>17</td><td>#nL, #n: L (32 bit)
No conversion</td></tr>
<tr><td>18</td><td>#nF, #n: F (64 bit)
64-bit data is converted into 32-bit data.
Fractional portion is dropped during conversion.
Error occurs if conversion result exceeds 32 bit range. (Error 531)</td></tr>
</table>

- For +, -, *, / (except MOD)

n: Indicates variable number or device number.

<table>
<tr><th>No.</th><th>Operation Result</th><th>Data 1</th><th>Data 2</th></tr>
<tr><td>19</td><td rowspan="9">#nF, #n: F (64 bit)
(64 bit)
No conversion</td><td rowspan="3">#n (16 bit)
16-bit data is converted into 64-bit data.</td><td>#n (16 bit)
16-bit data is converted into 64-bit data.</td></tr>
<tr><td>20</td><td>#nL, #n: L (32 bit)
32-bit data is converted into 64-bit data.</td></tr>
<tr><td>21</td><td>#nF, #n: F (64 bit)
No conversion</td></tr>
<tr><td>22</td><td rowspan="3">#nL, #n: L (32 bit)
32-bit data is converted into 64-bit data.</td><td>#n (16 bit)
16-bit data is converted into 64-bit data.</td></tr>
<tr><td>23</td><td>#nL, #n: L (32 bit)
32-bit data is converted into 64-bit data.</td></tr>
<tr><td>24</td><td>#nF, #n: F (64 bit)
No conversion</td></tr>
<tr><td>25</td><td rowspan="3">#nF, #n: F (64 bit)
No conversion</td><td>#n (16 bit)
16-bit data is converted into 64-bit data.</td></tr>
<tr><td>26</td><td>#nL, #n: L (32 bit)
32-bit data is converted into 64-bit data.</td></tr>
<tr><td>27</td><td>#nF, #n: F (64 bit)
No conversion</td></tr>
</table>

• For MOD

n: Indicates variable number or device number.

<table>
<tr><th>No.</th><th>Operation Result</th><th>Data 1</th><th>Data 2</th></tr>
<tr><td>28</td><td rowspan="9">#nF, #n: F (64 bit)
(64 bit)
Internal operation result (32 bit) is converted into 64-bit data.</td><td rowspan="3">#n (16 bit)
16-bit data is converted into 32-bit data.</td><td>#n (16 bit)
16-bit data is converted into 32-bit data.</td></tr>
<tr><td>29</td><td>#nL, #n: L (32 bit)
No conversion</td></tr>
<tr><td>30</td><td>#nF, #n: F (64 bit)
64-bit data is converted into 32-bit data.
Fractional portion is dropped during conversion.
Error occurs if conversion result exceeds 32 bit range. (Error 531)</td></tr>
<tr><td>31</td><td rowspan="3">#nL, #n: L (32 bit)
No conversion</td><td>#n (16 bit)
16-bit data is converted into 32-bit data.</td></tr>
<tr><td>32</td><td>#nL, #n: L (32 bit)
No conversion</td></tr>
<tr><td>33</td><td>#nF, #n: F (64 bit)
64-bit data is converted into 32-bit data.
Fractional portion is dropped during conversion.
Error occurs if conversion result exceeds 32 bit range. (Error 531)</td></tr>
<tr><td>34</td><td rowspan="3">#nF, #n: F (64 bit)
64-bit data is converted into 32-bit data.
Fractional portion is dropped during conversion.
Error occurs if conversion result exceeds 32 bit range. (Error 531)</td><td>#n (16 bit)
16-bit data is converted into 32-bit data.</td></tr>
<tr><td>35</td><td>#nL, #n: L (32 bit)
No conversion</td></tr>
<tr><td>36</td><td>#nF, #n: F (64 bit)
64-bit data is converted into 32-bit data.
Fractional portion is dropped during conversion.
Error occurs if conversion result exceeds 32 bit range. (Error 531)</td></tr>
</table>

(2) Logical operations (AND, OR, XOR, NOT), shift operators (<<, >>)

- For AND, OR, XOR, <<, >>

  The following table indicates the data type combinations and conversion methods for logical operations (AND, OR, XOR) and shift operators (<<, >>).

  Operation result = [data 1] operator [data 2]

  Operator denotes AND, OR, XOR, << or >>.

  For logical and shift operations, operation including the 64-bit floating-point type cannot be performed. (Error 560: format error)

n: Indicates variable number or device number.

No.	Operation Result	Data 1	Data 2	Remarks
1	#n (16 bit) No conversion	#n (16 bit) No conversion	#n (16 bit) No conversion	
2			#nL, #n: L (32 bit) 32-bit data is converted into 16-bit data. Error occurs if conversion result exceeds 16 bit range. (Error 531)	
3			#nF, #n: F (64 bit) Operation cannot be performed.	Operation disabled
4		#nL, #n: L (32 bit) 32-bit data is converted into 16-bit data. Error occurs if conversion result exceeds 16 bit range. (Error 531)	#n (16 bit) No conversion	
5			#nL, #n: L (32 bit) 32-bit data is converted into 16-bit data. Error occurs if conversion result exceeds 16 bit range. (Error 531)	
6			#nF, #n: F (64 bit) Operation cannot be performed.	Operation disabled
7		#nF, #n: F (64 bit) Operation cannot be performed.	#n (16 bit) Operation cannot be performed.	Operation disabled
8			#nL, #n: L (32 bit) Operation cannot be performed.	Operation disabled
9			#nF, #n: F (64 bit) Operation cannot be performed.	Operation disabled
10	#nL, #n: L (32 bit) (32 bit) No conversion	#n (16 bit) 16-bit data is converted into 32-bit data.	#n (16 bit) 16-bit data is converted into 32-bit data.	
11			#nL, #n: L (32 bit) No conversion	
12			#nF, #n: F (64 bit) Operation cannot be performed.	Operation disabled
13		#nL, #n: L (32 bit) No conversion	#n (16 bit) 16-bit data is converted into 32-bit data.	
14			#nL, #n: L (32 bit) No conversion	
15			#nF, #n: F (64 bit) Operation cannot be performed.	Operation disabled
16		#nF, #n: F (64 bit) Operation cannot be performed.	#n (16 bit) Operation cannot be performed.	Operation disabled
17			#nL, #n: L (32 bit) Operation cannot be performed.	Operation disabled
18			#nF, #n: F (64 bit) Operation cannot be performed.	Operation disabled

- For NOT

  The following table indicates the data type combinations and conversion methods for NOT.

  Operation result = operator [data 1]

  Operator denotes NOT.

  For logical and shift operations, operation including the 64-bit floating-point type cannot be performed. (Error 560: format error)

n: Indicates variable number or device number.

<table>
<tr><th>No.</th><th>Operation Result</th><th>Data 1</th><th>Remarks</th></tr>
<tr><td>1</td><td rowspan="3">#n (16 bit)
No conversion</td><td>#n (16 bit)
No conversion</td><td></td></tr>
<tr><td>2</td><td>#nL, #n: L (32 bit)
32-bit data is converted into 16-bit data.
Error occurs if conversion result exceeds 16 bit range. (Error 531)</td><td></td></tr>
<tr><td>3</td><td>#nF, #n: F (64 bit)
Operation cannot be performed.</td><td>Operation disabled</td></tr>
<tr><td>4</td><td rowspan="3">#nL, #n: L (32 bit)
(32 bit)
No conversion</td><td>#n (16 bit)
16-bit data is converted into 32-bit data.</td><td></td></tr>
<tr><td>5</td><td>#nL, #n: L (32 bit)
No conversion</td><td></td></tr>
<tr><td>6</td><td>#nF, #n: F (64 bit)
Operation cannot be performed.</td><td>Operation disabled</td></tr>
</table>

(3) Trigonometric functions (SIN, COS, TAN, ASIN, ACOS, ATAN)
The following table indicates the data type combinations and conversion methods for trigonometric functions (SIN, COS, TAN, ASIN, ACOS, ATAN).

Operation result = trigonometric function [data 1]

Trigonometric function denotes SIN, COS, TAN, ASIN, ACOS or ATAN.

Internal operation is performed with the 64-bit floating-point type.
When there is operation in data 1, operation is performed after conversion into 64-bit data.

n: Indicates variable number or device number.

<table>
<tr><th>No.</th><th>Operation Result</th><th>Data 1</th><th>Remarks</th></tr>
<tr><td>1</td><td rowspan="3">#n (16 bit)
Internal operation result (64 bit) is multiplied by 10000 and result of multiplication is converted into 16-bit data.
Fractional portion is dropped during conversion.
Error occurs if operation result exceeds 16 bit range. (Error 531)</td><td>#n (16 bit)
16-bit data is converted into 64-bit data.
Data is divided by 10000 during conversion.</td><td></td></tr>
<tr><td>2</td><td>#nL, #n: L (32 bit)
32-bit data is converted into 64-bit data.
Data is divided by 10000 during conversion.</td><td></td></tr>
<tr><td>3</td><td>#nF, #n: F (64 bit)
Data is divided by 10000 during conversion.</td><td></td></tr>
<tr><td>4</td><td rowspan="3">#nL, #n: L (32 bit)
Internal operation result (64 bit) is multiplied by 10000 and result of multiplication is converted into 32-bit data.
Fractional portion is dropped during conversion.
Error occurs if operation result exceeds 32 bit range. (Error 531)</td><td>#n (16 bit)
16-bit data is converted into 64-bit data.
Data is divided by 10000 during conversion.</td><td></td></tr>
<tr><td>5</td><td>#nL, #n: L (32 bit)
32-bit data is converted into 64-bit data.
Data is divided by 10000 during conversion.</td><td></td></tr>
<tr><td>6</td><td>#nF, #n: F (64 bit)
Data is divided by 10000 during conversion.</td><td></td></tr>
<tr><td>7</td><td rowspan="3">#nF, #n: F (64 bit)
Internal operation result (64 bit) is stored as it is.</td><td>#n (16 bit)
16-bit data is converted into 64-bit data.</td><td>Different from current one in usage.</td></tr>
<tr><td>8</td><td>#nL, #n: L (32 bit)
32-bit data is converted into 64-bit data.</td><td>Different from current one in usage.</td></tr>
<tr><td>9</td><td>#nF, #n: F (64 bit)
No conversion</td><td>Different from current one in usage.</td></tr>
</table>

(4) Floating-point type real number processing instructions (INT, FLT)
The following table indicates the data type combination and conversion method for floating-point type real number processing instructions (INT, FLT).

Operation result = function [data 1]

Function denotes INT or FLT.

The floating-point type real number processing instructions (INT, FLT) can operate the 32-bit type only.
The floating-point type real number processing instructions cannot operate data other than the 32-bit type. (Error 560: Format error)
INT And FLT cannot be used with other operations.

n: Indicates variable number or device number.

No.	Operation Result	Data 1
1	#nL, #n: L (32 bit) <INT> 32-bit floating-point type is converted into 32-bit type. Fractional portion is dropped during conversion. Error occurs if operation result exceeds 32 bit range. (Error 531) <FLT> 32-bit type is converted into 32-bit floating-point type.	#nL, #n: L (32 bit) No conversion

(5) Functions (SQRT, ABS, LN, EXP)

The following table indicates the data type combinations and conversion methods for functions (SQRT, ABS, LN, EXP).

Operation result = function [data 1]

Function denotes SQRT, ABS, LN or EXP.

Internal operation of SQRT LN or EXP is performed with the 64-bit floating-point type.

Internal operation of ABS is performed by making conversion into the operation result type.

When there is operation in data 1 for SQRT, operation is performed after conversion into 64-bit data.

• For SQRT, LN, EXP

n: Indicates variable number or device number.

No.	Operation Result	Data 1
1	#n (16 bit) Internal operation result (64 bit) is converted into 16-bit data. Fractional portion is dropped during conversion. Error occurs if operation result exceeds 16 bit range. (Error 531)	#n (16 bit) 16-bit data is converted into 64-bit data.
2		#nL, #n: L (32 bit) 32-bit data is converted into 64-bit data.
3		#nF, #n: F (64 bit) No conversion
4	#nL, #n: L (32 bit) Internal operation result (64 bit) is converted into 32-bit data. Fractional portion is dropped during conversion. Error occurs if operation result exceeds 32 bit range. (Error 531)	#n (16 bit) 16-bit data is converted into 64-bit data.
5		#nL, #n: L (32 bit) 32-bit data is converted into 64-bit data.
6		#nF, #n: F (64 bit) No conversion
7	#nF, #n: F (64 bit) No conversion	#n (16 bit) 16-bit data is converted into 64-bit data.
8		#nL, #n: L (32 bit) 32-bit data is converted into 64-bit data.
9		#nF, #n: F (64 bit) No conversion

• For ABS

n: Indicates variable number or device number.

No.	Operation Result	Data 1
1	#n (16 bit) No conversion	#n (16 bit) No conversion
2		#nL, #n: L (32 bit) 32-bit data is converted into 16-bit data.
3		#nF, #n: F (64 bit) 64-bit data is converted into 16-bit data.
4	#nL, #n: L (32 bit) No conversion	#n (16 bit) 16-bit data is converted into 32-bit data.
5		#nL, #n: L (32 bit) No conversion
6		#nF, #n: F (64 bit) 64-bit data is converted into 323-bit data.
7	#nF, #n: F (64 bit) No conversion	#n (16 bit) 16-bit data is converted into 64-bit data.
8		#nL, #n: L (32 bit) 32-bit data is converted into 64-bit data.
9		#nF, #n: F (64 bit) No conversion

(6) Functions (BIN, BCD)
The following table indicates the data type combinations and conversion methods for functions (BIN, BCD).

Operation result = function [data 1]

Function denotes BIN or BCD.

Internal operation is performed by making conversion into the 32-bit type.
Operation including the 64-bit floating-point type cannot be performed. (Error 560: format error)
BIN and BCD cannot be used with other operations.

n: Indicates variable number or device number.

No.	Operation Result	Data 1
1	#n (16 bit) Internal operation result (64 bit) is converted into 16-bit data. Error occurs if operation result exceeds 16 bit range. (Error 531)	#n (16 bit) 16-bit data is converted into 32-bit data.
2		#nL, #n: L (32 bit) No conversion
3		#nF, #n: F (64 bit) Operation cannot be performed.
4	#nL, #n: L (32 bit) No conversion	#n (16 bit) 16-bit data is converted into 32-bit data.
5		#nL, #n: L (32 bit) No conversion
6		#nF, #n: F (64 bit) Operation cannot be performed.

(7) Functions (round-off (RND), round-down (FIX), round-up (FUP))
The following table indicates the data type combinations and conversion methods for round-off (RND), round-down (FIX) and round-up (FUP).

Operation result = function [data 1]

Function denotes RND, FIX or FUP.

Round-off (RND), round-down (FIX) and round-up (FUP) cannot perform operation of other than the 64-bit floating-point type.
(Error 560: format error)

n: Indicates variable number or device number.

No.	Operation Result	Data 1
1	#nF, #n: F (64 bit) No type conversion <RND> Rounds off data 1 to one decimal place. <FIX> Rounds down data 1 to the units. <FUP> Rounds up data 1 to the units.	#nF, #n: F (64 bit) No type conversion

6.6.4 Instruction symbol setting range list

Table 6.5 lists the setting ranges of the instruction symbols used in motion programs.

Table 6.5 Instruction Symbol Setting Range List

<table>
<tr><th rowspan="2"></th><th rowspan="2">Symbol</th><th rowspan="2">Function</th><th colspan="2">Setting Range</th></tr>
<tr><th>Motion program description</th><th>Variable (D register setting)</th></tr>
<tr><td rowspan="12">Address</td><td>A</td><td>Coordinate position data</td><td rowspan="11">-214748.3648 to 214748.3647 (mm)
-21474.83648 to 21474.83647 (inch)
0 to 359.99999 (degree)</td><td rowspan="11">-2147483648 to 2147483647
0 to 35999999</td></tr>
<tr><td>B</td><td>Coordinate position data</td></tr>
<tr><td>C</td><td>Coordinate position data</td></tr>
<tr><td>U</td><td>Coordinate position data</td></tr>
<tr><td>V</td><td>Coordinate position data</td></tr>
<tr><td>W</td><td>Coordinate position data</td></tr>
<tr><td>X</td><td>Coordinate position data</td></tr>
<tr><td>Y</td><td>Coordinate position data</td></tr>
<tr><td>Z</td><td>Coordinate position data</td></tr>
<tr><td>I</td><td>Circular arc center coordinate 1</td></tr>
<tr><td>J</td><td>Circular arc center coordinate 2</td></tr>
<tr><td>R</td><td>Radius of R point specified circular arc</td><td>0 to 214748.3647 (mm)
0 to 21474.83647 (inch)
0 to 359.99999 (degree)</td><td>0 to 2147483647
0 to 35999999</td></tr>
<tr><td rowspan="2">Speed</td><td rowspan="2">F</td><td rowspan="2">Interpolation feed composite speed</td><td>0.01 to 6000000.00 (mm/min)
0.001 to 600000.000 (inch/min)</td><td>1 to 600000000</td></tr>
<tr><td>0.001 to 2147483.647 (degree/min)</td><td>1 to 2147483647</td></tr>
<tr><td rowspan="11">Others</td><td>G</td><td>G instruction</td><td>00, 01, 02, 03, 04, 09, 24, 25, 26, 28, 30, 32, 43, 44, 49, 53, 54, 55, 56, 57, 58, 59, 61, 64, 90, 91, 92</td><td>——</td></tr>
<tr><td>H</td><td>Subprogram call sequence number
Tool length offset data number</td><td>1 to 9999
1 to 20</td><td>1 to 9999
1 to 20</td></tr>
<tr><td>L</td><td>Repeat count</td><td>0 to 9999</td><td>0 to 9999</td></tr>
<tr><td>M</td><td>Miscellaneous function (M code)</td><td>0 to 9999</td><td>0 to 9999</td></tr>
<tr><td>N</td><td>Sequence number</td><td>1 to 9999</td><td>——</td></tr>
<tr><td>O</td><td>Motion program number</td><td>1 to 256</td><td>——</td></tr>
<tr><td rowspan="3">P</td><td>Dwell time</td><td>1 to 65535</td><td>1 to 65535</td></tr>
<tr><td>Start program No.</td><td>1 to 256</td><td>1 to 256</td></tr>
<tr><td>Subprogram call number</td><td>1 to 256</td><td>1 to 256</td></tr>
<tr><td>PB</td><td>Parameter block No.</td><td>1 to 16</td><td>1 to 16</td></tr>
<tr><td>TL</td><td>Torque limit value</td><td>1 to 500</td><td>1 to 500</td></tr>
<tr><td rowspan="5">Operational expression</td><td>+</td><td>Addition</td><td rowspan="5">-2147483648 to 2147483647</td><td rowspan="5">-2147483648 to 2147483647</td></tr>
<tr><td>-</td><td>Subtraction</td></tr>
<tr><td>*</td><td>Multiplication</td></tr>
<tr><td>/</td><td>Division</td></tr>
<tr><td>MOD</td><td>Remainder</td></tr>
</table>

REMARK

(1) About the command unit

A decimal point can be entered in the motion program input information which define the command address, speed, etc.

[Example] 123456.7890

A decimal point may also be omitted.

When a decimal point is omitted, a command address is represented in 0.0001mm, 0.00001 inch or 0.00001 degree increments, for example.

<For command address>

○○○○○○. ○○○○

[Example] 10. ······ 10mm
10 ······· 0.001mm (unit: mm)

<For feedrate (F)>

○○○○○○○○. ○○

[Example] 10. ······ 10mm/min
10 ······· 0.1mm/min (unit: mm)

Any value may be specified up to 10 digits. (Decimal point not included)
Specifying more than 10 digits will result in an error.

The numbers of significant decimal places are listed below. Digits after the significant decimal places are ignored. Note that specifying 10 or more digits will result in an error.

Unit Command	mm	inch	degree
Command address	4	5	5
Command speed	2	3	3

6.6.5 Positioning control unit for 1 axis

For one axis, positioning control is exercised in the control unit specified in the fixed parameter.
(The control unit specified in the parameter block is ignored.)

6.6.6 Control units for interpolation control

(1) A check is made on the interpolation control unit specified in the parameter block and the control unit set in the fixed parameter.
For interpolation control, if the interpolation control unit in the parameter block differs from the control unit in the fixed parameter of each axis, the result will be as described below.

	Interpolation Control Unit in Parameter Block			Starting Method
	mm	inch	degree	
Condition for normal start	There are axes whose control unit set in fixed parameter is mm.	There are axes whose control unit set in fixed parameter is inch.	There are axes whose control unit set in fixed parameter is degree.	Control starts in the interpolation control unit of the parameter block.
Condition for unit mismatch error (error code 40)	When the control unit of any axis in the fixed parameter does not match the interpolation control unit of the parameter block.			• If the control units of the axes to be interpolation-controlled are the same, control starts in the preset control unit. • If the control units of the axes to be interpolation-controlled are different, control starts in the unit of the highest priority as indicated below. Priority degree>inch>mm

(2) In interpolation control, the combinations of axis control units are classified as indicated below.

	mm	inch	degree
mm	1)	2)	2)
Inch	2)	1)	2)
degree	2)	2)	1)

REMARKS

1): Same unit
2): Unit mismatch

(a) Same unit (1))
The position command is calculated for positioning according to the preset address/travel, positioning speed and electronic gear.

(b) Unit mismatch (2))

- On a unit mismatch, the travel and positioning speed are calculated for each axis.
  - a) The travel is converted into the PLS unit using the electronic gear of its own axis.
  - b) The positioning speed is converted into the PLS/sec unit using the electronic gear of the axis whose control unit matches the interpolation control unit.

    The travel converted into PLS, the speed converted into PLS/sec, and the electronic gear are used to calculate the position command value for positioning.
- If there are two or more axes whose control units are the same as the interpolation control unit in the linear interpolation of three or more axes, the electronic gear of the lowest axis number is used to calculate the positioning speed.

POINT

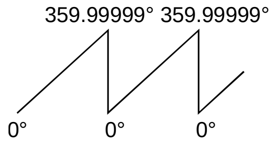
(1) For circular interpolation control

When degree is used as the control unit of one axis, degree should also be used with the other axis.

6.6.7 Control in the control unit of "degree"

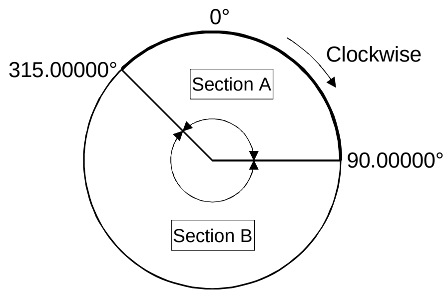
When the control unit is degree, the following items are different from those of the other control units.

(1) Present value address
The present value address in degree is the ring address of 0 to 360°.



(2) Stroke limit valid/invalid setting
The upper and lower limit values of a stroke limit in degree is between 0° and 359.99999°.

(a) Setting for making stroke limit valid
To make the stroke limit valid, set the lower limit value of the stroke limit first, then the upper limit value in the clockwise direction.



1) Set the moving range in section A as follows.
   a) Lower limit value of stroke limit ..... 315.00000°
   b) Upper limit value of stroke limit ..... 90.00000°
2) Set the moving range in section B as follows.
   a) Lower limit value of stroke limit ...... 90.00000°
   b) Upper limit value of stroke limit ...... 315.00000°

(b) Setting for making stroke limit invalid
To make the stroke limit invalid, set to make the "lower stroke limit value" equal to the "upper stroke limit value".
Control can be exercised independently of the stroke limit setting.

POINT
You cannot make circular interpolation which includes the axis whose stroke limit has been set to be invalid.

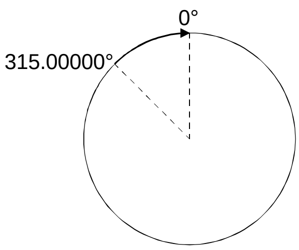
(3) Positioning control
The positioning control methods in the control unit of degree will be explained below.
(a) Absolute value command
Under the absolute value command, positioning is carried out relative to the present value in the direction nearer to the specified address.

Example

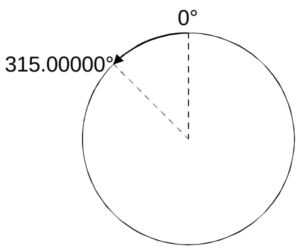
(1) When the axis is moved from the present value of 315.00000° to 0°, clockwise positioning is performed.

(2) When the axis is moved from the present value of 0° to 315.00000°, counterclockwise positioning is performed.

From 315.00000° to 0°



From 0° to 315.00000°

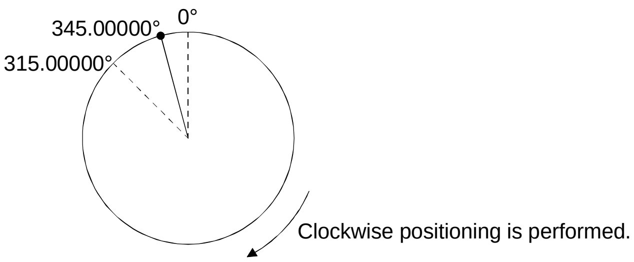


POINTS

(1) The positioning direction of the absolute value command is determined by the way of setting the stroke limit range, and positioning may not be made in the direction nearer to the specified address.

Example

When the axis is moved from the present value of 0° to 315.00000°, clockwise positioning is performed if the lower stroke limit value is 0° and the upper stroke limit value is 345.00000°.



(2) The positioning address is within the range 0° to 360°.
When carrying out positioning of more than one revolution, use the incremental value command.

(b) Incremental value command
Under the incremental value command, positioning of the specified travel is performed in the specified direction. The moving direction depends on the sign of the travel.
1) Positive moving direction .... Clockwise
2) Negative moving direction........ Counterclockwise

POINT

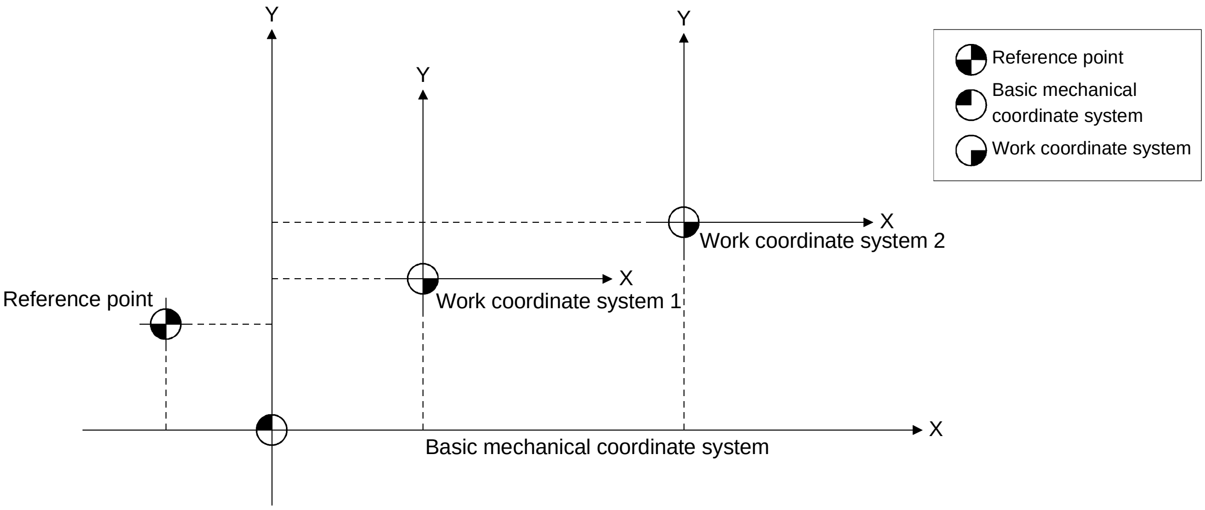
Under the incremental value command, positioning of more than 360° can be done.

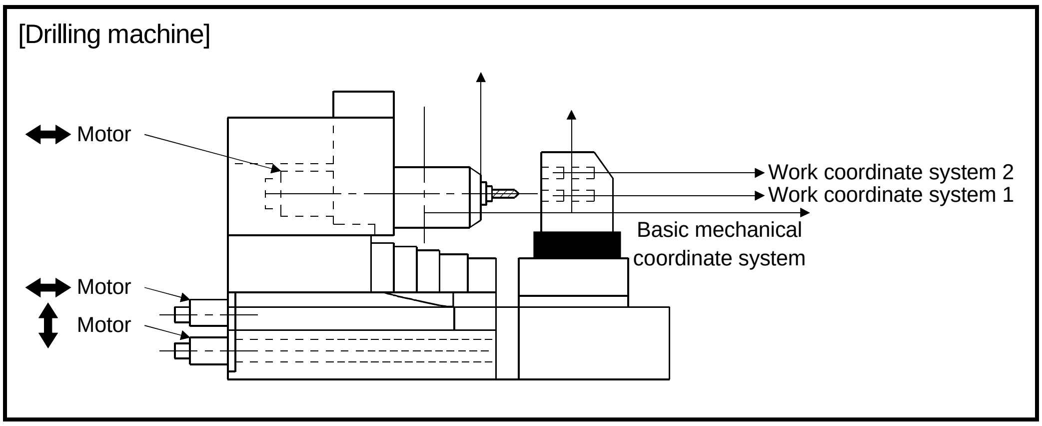
6.7 About Coordinate Systems

This section describes coordinate systems.
There are two coordinate systems: basic mechanical coordinate system and work coordinate system.

(1) Basic mechanical coordinate system
.... A coordinate system specific to a machine and indicates the position determined specifically for the machine.

(2) Work coordinate system
.... A coordinate system used by a programmer for programming to set the reference point on a work as a coordinate home position.
In the work coordinate system, a position is specified with an offset value from the basic mechanical coordinate system. The offset value is set with a distance from the mechanical coordinate system origin (0).
You can specify up to six work coordinate systems (work coordinates 1 to 6). Set them by parameter setting or work coordinate system selection (G54 to G59). (Refer to Section 4.7 or 6.8.19.)
By setting multiple work coordinates, you can easily perform multiple positioning operations with a single program.



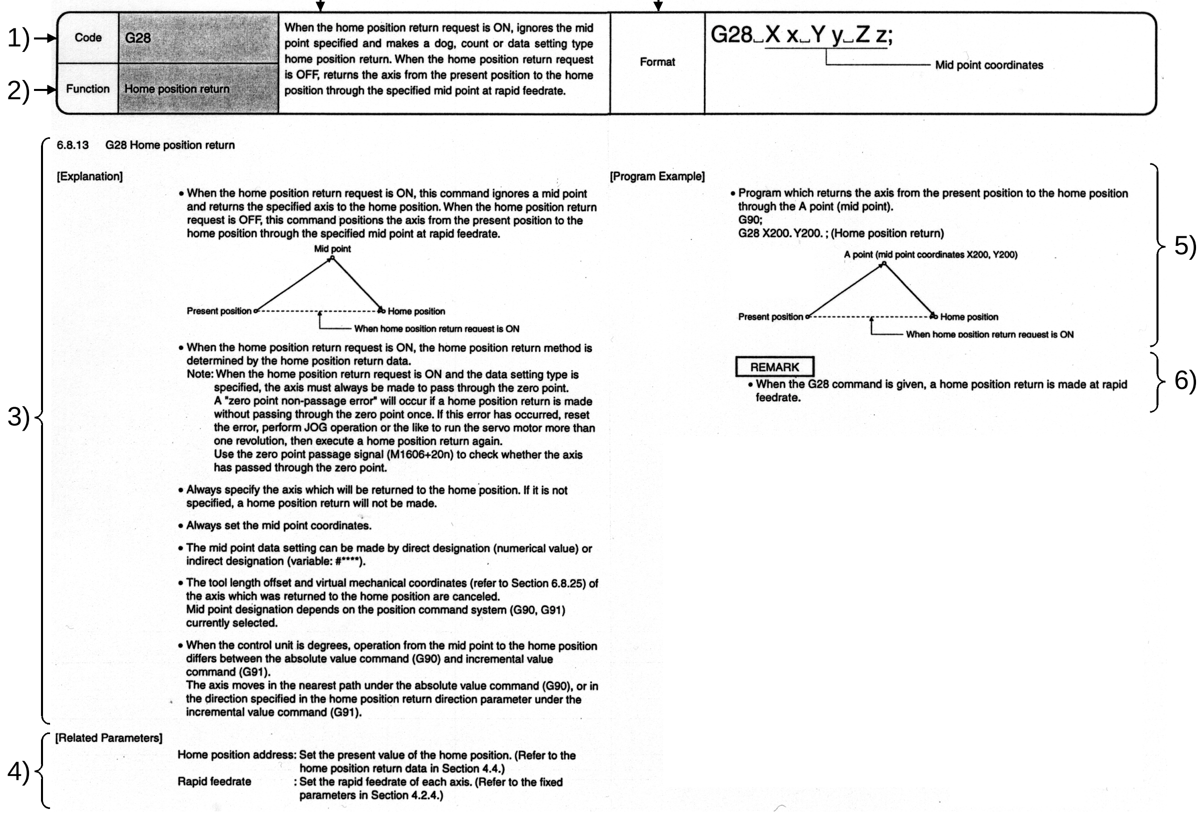


6.8 G Codes

This section explains the instruction codes used in motion programs.
Each instruction is described in the following format.

Briefly explains the function outline of the instruction.

Indicates the input or description method.
The "␣" mark indicates that a space must be placed at the time of program input.



1) Name of the instruction code.
2) Indicates the model name.
3) Gives the detailed explanation or precautions.
4) Indicates the parameters related to this instruction. (Parameters whose values must be set)
5) Shows a program example which uses this instruction.
6) Provides supplementary explanation or instructions related to this instruction.

Table 6.6 indicates the arguments of the G codes.

Table 6.6 G Code Arguments

	Axis command (*1)	Radius command (R)	Center point command (I • J)	Skip command (SKIP)	Cancel command (CAN)	Starting angle (START)	Amplitude (STRK)	M code (*2)	G code	Feed (F)	H	L	N	O	P	PB	Remarks
G00	○							○	○								Only G codes of G04, G43, G44 and G49 are available.*
G01	○							○	○	○							Only G codes of G04, G43, G44 and G49 are available.*
G02	◎		◎					○	○	○							Only G code of G04 is available. Center point command and axis command may be specified for up to 2 axes.
G02	◎	◎						○	○	○							Only G code of G04 is available. Radius command and axis command may be specified for up to 2 axes.
G03	◎		◎					○	○	○							Only G code of G04 is available. Center point command and axis command may be specified for up to 2 axes.
G03	◎	◎						○	○	○							Only G code of G04 is available. Radius command and axis command may be specified for up to 2 axes.
G04															◎		Dwell
G09									○								Only G codes of G01, G02 and G03 are available.*
G23																	
G24					◎										○	○	P: Start program number PB: Parameter block number
G25	◎					○	○			○							Specify only axis name for axis command and frequency for F.
G26	◎																Specify only axis name for axis command.
G28	○								○								Only G code of G53 is available.
G30	○								○								Only G code of G53 is available.
G32	○			◎				○	○	○					○		P must not be specified for axis command and M code simultaneously.
G43	○										◎						
G44	○										◎						
G49	○								○								Only G code of G28 is available.
G53	○								○								Only G code of G28 is available.
G54	○								○								Only G codes of G00, G01, G02, G03 and G92 are available.*
G55	○								○								Only G codes of G00, G01, G02, G03 and G92 are available.*
G56	○								○								Only G codes of G00, G01, G02, G03 and G92 are available.*
G57	○								○								Only G codes of G00, G01, G02, G03 and G92 are available.*
G58	○								○								Only G codes of G00, G01, G02, G03 and G92 are available.*
G59	○								○								Only G codes of G00, G01, G02, G03 and G92 are available.*
G61									○								Only G codes of G00, G01, G02 and G03 are available.*
G64									○								Only G codes of G00, G01, G02 and G03 are available.*
G90	○								○								Only G codes of G00, G01, G02 and G03 are available.*
G91	○								○								Only G codes of G00, G01, G02 and G03 are available.*
G92	○																
G100																	
G101																	

○ : May be specified.
◎ : Must be specified.
Blank: Must not be specified.

For G43, G44, G49, G54 to G59, G90 and G91, use the currently selected modal group 01 to set the specifiable arguments.
For *, the G code may be set in the first parameter only.

*1 The axis commands are X, Y, Z, U, V, W, A, B and C.
*2 The M codes are other than M00, M01, M02, M30, M98, M99 and M100.

Code	G00	Positions the specified axes. (PTP)
Function	PTP positioning at rapid feedrate	

6.8.1 G00 PTP positioning at rapid feedrate

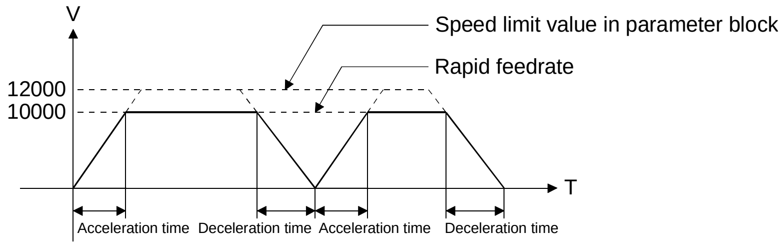
[Explanation]

- Linearly positions all the specified axes from the present value to the specified coordinate axis position at the fixed speed.

- Being a modal instruction, this command is valid until another G code in the same group is used. Hence, if the next command is the same G code, it may be enabled by specifying only the axis name. (Group (01) is made up of G00, G01, G02 and G03.)

- This command always increases or decreases speed at the starting or end point of a block and proceeds to the next block.

- The positioning speed is not more than the rapid feedrate of each axis.
  [Example]
  G00 X100. ;
  X150. ;
  (When rapid feedrate is 10000mm/min and speed limit value in parameter block is 12000mm/min)



- Acceleration-fixed acceleration/deceleration is made.
  Acceleration is calculated from the lower speed of the rapid feedrate and speed limit value and the acceleration time and deceleration time in the parameter block.

- The positioning data can be set by direct designation (numerical value) or indirect designation (variable: #****).

- Commanding the M code in G00 also causes acceleration/deceleration to be made in the acceleration time of the parameter block as in G01. (Example G00 X□ M□;)

[Related Parameters]

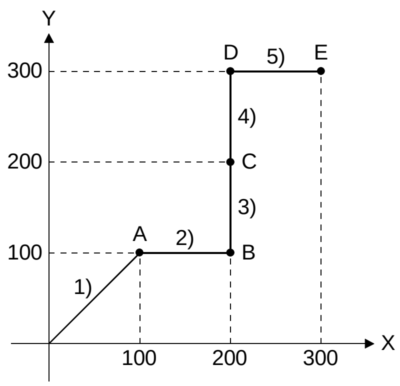
Rapid feedrate: Set the maximum feedrate of each axis.
(Refer to Section 4.2.4 for the rapid feedrate setting in the fixed parameter.)
When G00 is executed, positioning takes place in the shortest path which connects the starting point and end point.
The positioning speed is within the rapid feedrate of each axis.

Format	G00␣X x␣Y y␣Z z; Axis names Positioning addresses

[Program Example]

- Program used to position the axes at points A, B, C, D and E. (Under absolute value command)

1) G00 X100. Y100. ;	(A point positioning)	Travel under G00
2) X200. ;	(B point positioning)	
3) Y200. ;	(C point positioning)	
4) G01 Y300. F100. ;	(D point positioning)	Travel under G01
5) X300. ;	(E point positioning)	



(Unit: mm)

REMARKS

- To determine the feedrate of G00, the axis whose time to reach the target position is the longest in the travel/rapid feedrate (fixed parameter) of all axes is used as the reference axis, and interpolation is made in the reference axis speed interpolation mode phase or the like. (Refer to Section 4.2.4.)
- The rapid feedrate of each axis is clamped at the speed limit value if it is larger than the speed limit value of the parameter block. The calculation of the reference axis is also made using the clamped value.

Code	G01	Linearly interpolates the axes from the present value to the specified end point at the specified feedrate. (CP) As the feedrate, specify the linear speed (composite speed) in the advance direction.
Function	CP positioning at speed specified in F	

6.8.2 G01 CP positioning at speed specified in F

[Explanation]

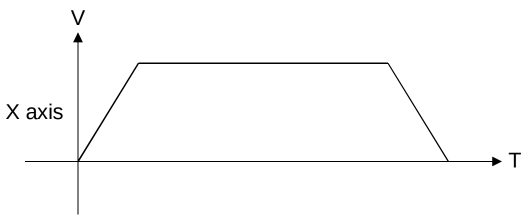
- Being a modal instruction, this command is valid until another G code in the same group is used. Hence, when the next command is G01, it may be enabled by specifying only the axis name, unless the feedrate is changed.

- As the command unit of the feedrate, specify the interpolation control unit of the parameter block.

- The maximum command value of the feedrate is the speed limit value set in the parameter block.

- If the F command is not set in the first G01 command, a program error (error code: 501) occurs.

- When this command is executed continuously, the feedrate is not increased or decreased at the starting or end point of a block since the status is not the exact stop check mode.
  [Example]
  G01 X100. F200. ;
  X150. ;



- The positioning data can be set by direct designation (numerical value) or indirect designation (variable: #****).

- Specify G61 when making acceleration/deceleration at block switching.

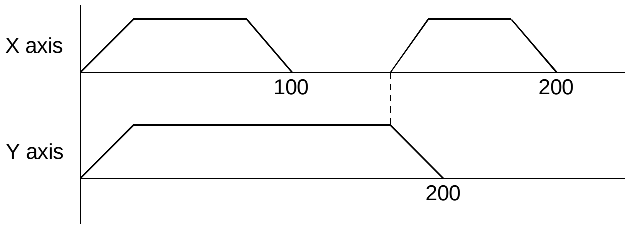
- The axes do not decelerate to a stop if the G02 or G03 command is given between the G01 commands (CP positioning).
  [Example]
  G01 X100. Y100. Z100. ;
  G02 X0. Y0. I0. J50. F500. ;
  G03 X0. Y0. I0. J50. F500. ;
  G01 X100. ;
  } Constant-speed control is exercised in this area.

Format	G01␣X x␣Y y␣Z z␣F f; f: Feedrate F: Feedrate command x, y, z: Positioning addresses X, Y, Z: Axis names

- Acceleration/deceleration processing under G01 command

G91 G01 X100. Y100. F100. ;	CP positioning of X, Y ....Block 1
Y100. ;	CP positioning of Y ....Block 2
X100. ;	CP positioning of X ....Block 3

When the above program is run, the acceleration/deceleration processing of the X and Y axes is performed as shown below.



Note: • Both the acceleration and deceleration times are the acceleration time of the parameter block.
• As under the M code command, the acceleration/deceleration time under the G0 command is the acceleration time of the parameter block.

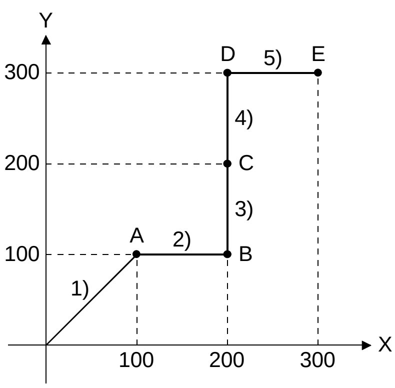
[Related Parameters]

Speed limit value: Set the maximum feedrate of each axis.
(Refer to the speed limit value of the parameter block in Section 4.6.)

[Program Example]

- Program which performs positioning to A, B, C, D and E points. (Under absolute value command)

1) G01 X100. Y100. F100. ;	(A point positioning)	Travel under G01 (Travel at feedrate of 100mm/min)
2) X200. ;	(B point positioning)	
3) Y200. ;	(C point positioning)	
4) G00 Y300. F100. ;	(D point positioning)	Travel under G00
5) X300. ;	(E point positioning)	



(Unit: mm)

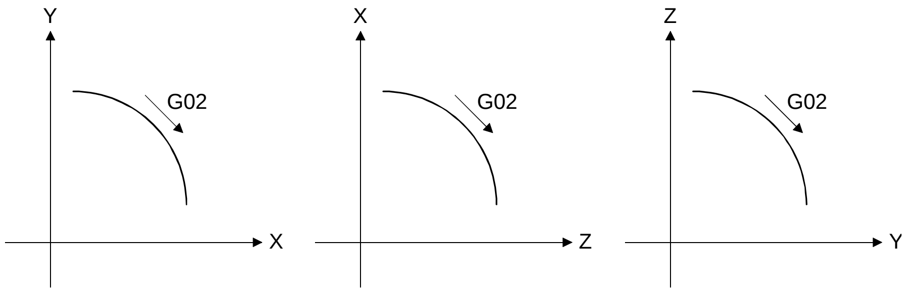
Code	G02	Moves the axes from the current position (starting point) to the specified coordinate position (end point) along a circular arc (CW). The travel speed is the specified feedrate.
Function	Circular interpolation (CW) Circular arc center coordinate designation	

6.8.3 G02 Circular interpolation CW (Circular arc center coordinate designation)

[Explanation]

- Use the incremental values (always use incremental values) from the current position (starting point) to command the circular arc center coordinates.
  For G02 (CW), give the end point coordinates of the circular arc with the address (must be specified for 2 axes) and specify the center coordinates of the circular arc with I and J.
  The center coordinates 1, 2 are I and J in order of lower axis numbers.
  When X=Axis 1, Y=Axis 2, I=1(X), J=2(Y)
  When X=Axis 2, Y=Axis 1, I=1(Y), J=2(X)

- Always specify the end point coordinates for 2 axes as they cannot be omitted.
  G02 (CW): Clockwise



- If the end point is in the same position as the starting point, the circular arc is 360 degrees (perfect circle).

- If they cannot be linked by a circular arc,
  Within the permissible circular arc error range: The starting and end points are connected by helical interpolation.
  Beyond the permissible circular arc error range: An error occurs at the circular arc starting point.

- When this command is executed continuously, the feedrate is not increased or decreased at the starting or end point of a block since the status is not the exact stop check mode.

- When the circular arc center coordinates and radius are specified for G02 (CW) at the same time, the radius-specified circular interpolation has priority.

- The positioning data can be set by direct designation (numerical value) or indirect designation (variable: #****).

	Format	G02␣X x␣Y y␣I i␣J j␣F f; F: Feedrate command f: Feedrate I i␣J j: Circular arc center coordinates 1, 2 X x␣Y y: End point X, Y coordinates

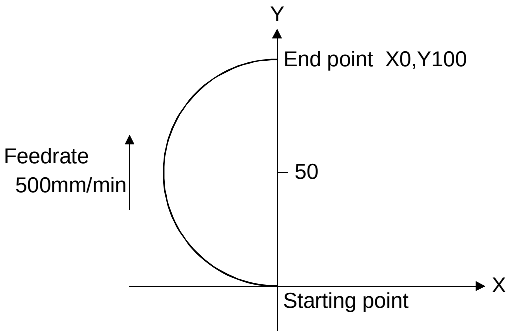
[Related Parameters]

Speed limit value: Set the maximum feedrate of each axis.
(Refer to the speed limit value of the parameter block in Section 4.6.)

Circular arc error: Set the permissible circular arc error range.
(Refer to the permissible circular arc error range of the parameter block in Section 4.6.3.)

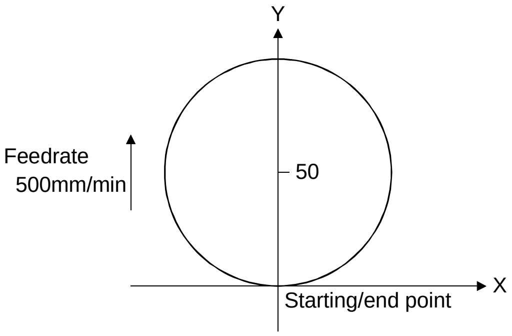
[Program Example]

- Program which performs circular interpolation from the current position to draw a half circle.
  G91 G02 X0. Y100. I0. J50. F500. ;



(Unit: mm)

- Program which performs circular interpolation from the current position to draw a perfect circle.
  G02 X0. Y0. I0. J50. F500. ; (Perfect circular command)



(Unit: mm)

REMARKS

- The end point and circular arc center coordinates cannot be omitted. Always specify them for two axes.
- Circular interpolation cannot be made if it includes the degree axis whose stroke limit is set to be invalid.
- Circular interpolation cannot be made for the unit combination of mm and degree or inch and degree.

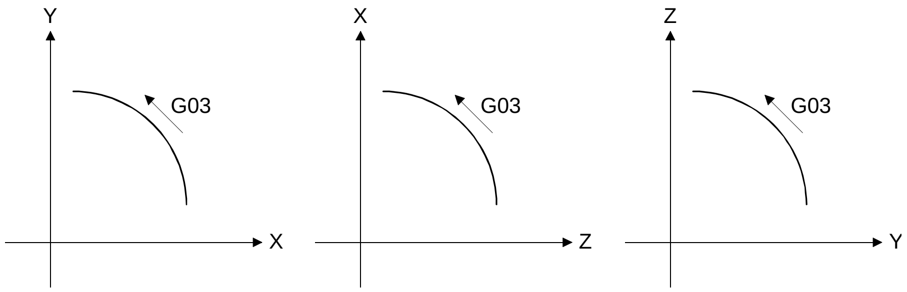
Code	G03	Moves the axes from the current position (starting point) to the specified coordinate position (end point) along a circular arc (CCW). The travel speed is the specified feedrate.
Function	Circular interpolation (CCW) Circular arc center coordinate designation	

6.8.4 G03 Circular interpolation CCW (Circular arc center coordinate designation)

[Explanation]

- Use the incremental values (always use incremental values) from the current position (starting point) to command the circular arc center coordinates.
  For G03 (CCW), give the end point coordinates of the circular arc with the address (must be specified for 2 axes) and specify the center coordinates of the circular arc with I and J.
  The center coordinates 1, 2 are I and J in order of lower axis numbers.
  When X=Axis 1, Y=Axis 2, I=1(X), J=2(Y)
  When X=Axis 2, Y=Axis 1, I=1(Y), J=2(X)

- Always specify the end point coordinates for 2 axes as they cannot be omitted.
  G03 (CCW): Counterclockwise



- If the end point is in the same position as the starting point, the circular arc is 360 degrees (perfect circle).

- If they cannot be linked by a circular arc,
  Within the permissible circular arc error range: The starting and end points are connected by helical interpolation.
  Beyond the permissible circular arc error range: An error occurs at the circular arc starting point.

- When this command is executed continuously, the feedrate is not increased or decreased at the starting or end point of a block since the status is not the exact stop check mode.

- When the circular arc center coordinates and radius are specified for G03 (CCW) at the same time, the radius-specified circular interpolation has priority.

- The positioning data can be set by direct designation (numerical value) or indirect designation (variable: #****).

	Format	G03␣X x␣Y y␣I i␣J j␣F f; X x, Y y: End point X, Y coordinates I i, J j: Circular arc center coordinates 1, 2 F: Feedrate command f: Feedrate

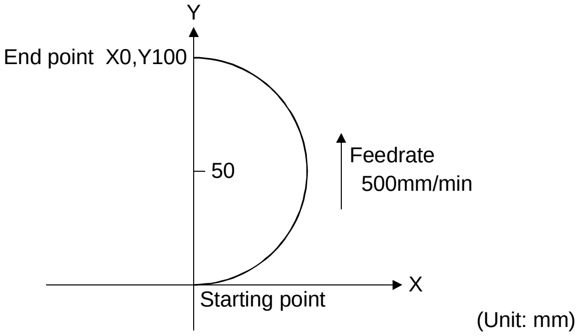
[Related Parameters]

Speed limit value: Set the maximum feedrate of each axis.
(Refer to the speed limit value of the parameter block in Section 4.6.)

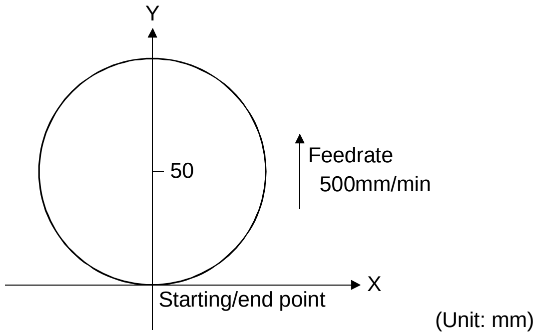
Circular arc error: Set the permissible circular arc error range.
(Refer to the permissible circular arc error range of the parameter block in Section 4.6.3.)

[Program Example]

- Program which performs circular interpolation from the current position to draw a half circle.
  G91 G03 X0. Y100. I0. J50. F500. ;



- Program which performs circular interpolation from the current position to draw a perfect circle.
  G03 X0. Y0. I0. J50. F500. ; (Perfect circular command)



REMARKS

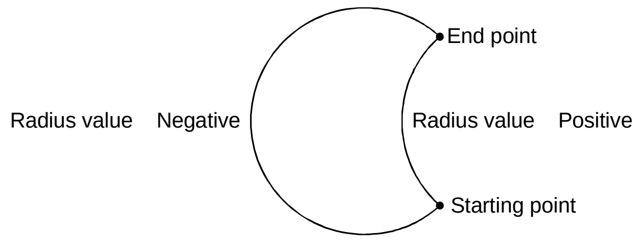
- The end point and circular arc center coordinates cannot be omitted. Always specify them for two axes.
- Circular interpolation cannot be made if it includes the degree axis whose stroke limit is set to be invalid.
- Circular interpolation cannot be made for the unit combination of mm and degree or inch and degree.

Code	G02	Moves the axes from the current position (starting point) to the specified coordinate position (end point) along a circular arc of the specified radius (CW). The travel speed is the specified feedrate.
Function	Circular interpolation (CW) Radius specified circular interpolation	

6.8.5 G02 Circular interpolation CW (Radius designation)

[Explanation]

- A less than half-circle circular arc command is given at a positive R (circular arc radius) value, or a more than half-circle circular arc command is given at a negative R value.
  Always use an incremental value to command the R value.



  An error occurs if the distance between
  starting and end points - radius × 2 > circular arc error.

- If a perfect circuit command (the starting point is the same as the end point) is specified in R-specified circular interpolation, an error (error code: 108) occurs and no operation is performed. Therefore, specify the circular arc center coordinates for the perfect circuit command.

- A circular arc of more than 180° is drawn at a negative circular arc radius (R) value, or a circular arc of less than 180° is drawn at a positive R value.

- When this command is executed continuously, the feedrate is not increased or decreased at the starting or end point of a block since the status is not the exact stop check mode.

- When the circular arc center coordinates and radius are specified for G02 (CW) at the same time, the radius-specified circular interpolation has priority.

- The positioning data can be set by direct designation (numerical value) or indirect designation (variable: #****).

[Related Parameters]

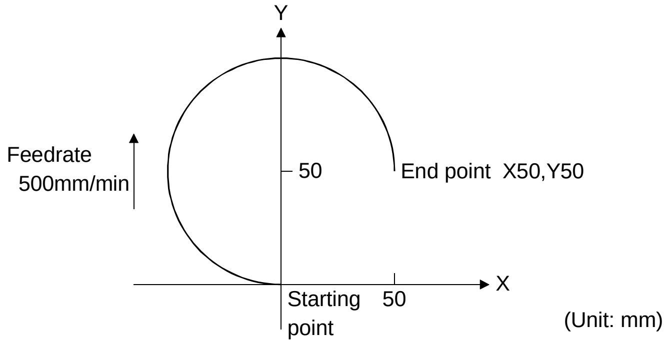
Speed limit value: Set the maximum feedrate of each axis.
(Refer to the speed limit value of the parameter block in Section 4.6.)

Circular arc error: Set the permissible circular arc error range.
(Refer to the permissible circular arc error range of the parameter block in Section 4.6.3.)

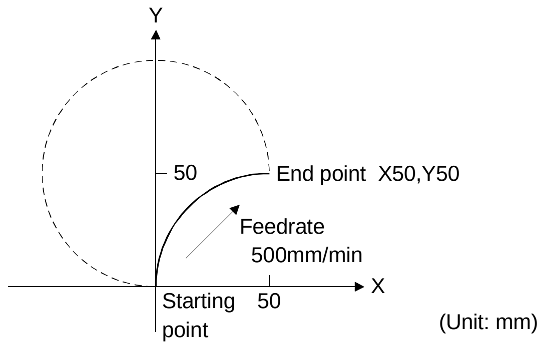
Format	G02␣X x␣Y y␣R r␣F f; f: Feedrate F: Feedrate command R r: Circular arc radius X x␣Y y: End point X, Y coordinates

[Program Example]

- Program which draws a circular arc of more than 180° at a negative circular arc radius (R) value.
  G91 G02 X50. Y50. R-50. F500. ;



- Program which draws a circular arc of less than 180° at a positive circular arc radius (R) value.
  G91 G02 X50. Y50. R50. F500. ;



REMARKS

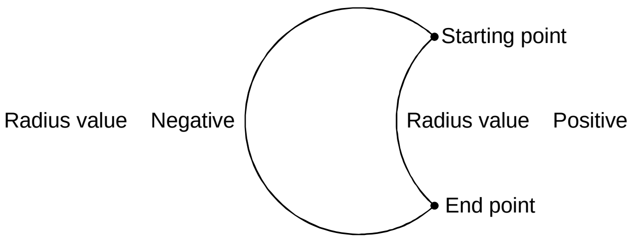
- The end point coordinates and circular arc radius cannot be omitted. Always specify the end point coordinates and circular arc radius.
- Circular interpolation cannot be made if it includes the degree axis whose stroke limit is set to be invalid.
- Circular interpolation cannot be made for the unit combination of mm and degree or inch and degree.

Code	G03	Moves the axes from the current position (starting point) to the specified coordinate position (end point) along a circular arc of the specified radius (CCW). The travel speed is the specified feedrate.
Function	Circular interpolation (CCW) Radius specified circular interpolation	

6.8.6 G03 Circular interpolation CCW (Radius designation)

[Explanation]

- A less than half-circle circular arc command is given at a positive R (circular arc radius) value, or a more than half-circle circular arc command is given at a negative R value.
  Always use an incremental value to command the R value.



  An error occurs if the distance between
  starting and end points - radius $\times$ 2 $>$ circular arc error.

- If a perfect circuit command (the starting point is the same as the end point) is specified in R-specified circular interpolation, an error (error code: 108) occurs and no operation is performed. Therefore, specify the circular arc center coordinates for the perfect circuit command.

- A circular arc of more than 180° is drawn at a negative circular arc radius (R) value, or a circular arc of less than 180° is drawn at a positive R value.

- When this command is executed continuously, the feedrate is not increased or decreased at the starting or end point of a block since the status is not the exact stop check mode.

- When the circular arc center coordinates and radius are specified for G03 (CCW) at the same time, the radius-specified circular interpolation has priority.

- The positioning data can be set by direct designation (numerical value) or indirect designation (variable: #****).

[Related Parameters]

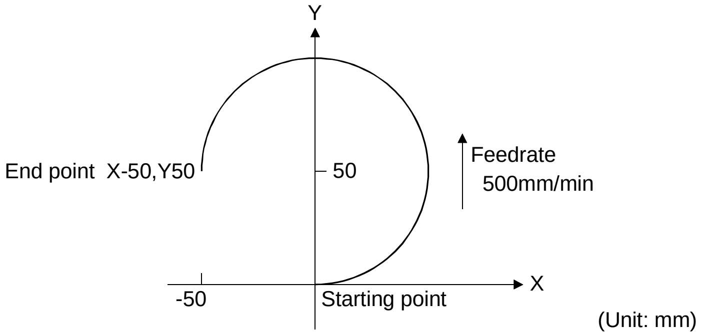
Speed limit value: Set the maximum feedrate of each axis.
(Refer to the speed limit value of the parameter block in Section 4.6.)

Circular arc error: Set the permissible circular arc error range.
(Refer to the permissible circular arc error range of the parameter block in Section 4.6.3.)

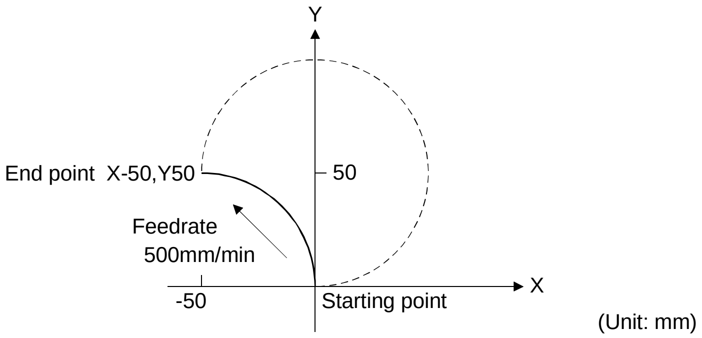
	Format	G03␣X x␣Y y␣R r␣F f; Feedrate Feedrate command Circular arc radius End point X, Y coordinates

[Program Example]

- Program which draws a circular arc of more than 180° at a negative circular arc radius (R) value.
  G91 G03 X-50. Y50. R-50. F500. ;



- Program which draws a circular arc of less than 180° at a positive circular arc radius (R) value.
  G91 G03 X-50. Y50. R50. F500. ;



REMARKS

- The end point coordinates and circular arc radius cannot be omitted. Always specify the end point coordinates and circular arc radius.
- Circular interpolation cannot be made if it includes the degree axis whose stroke limit is set to be invalid.
- Circular interpolation cannot be made for the unit combination of mm and degree or inch and degree.

Code	G04	Waits for the next block to be executed for the specified period of time.
Function	Dwell	

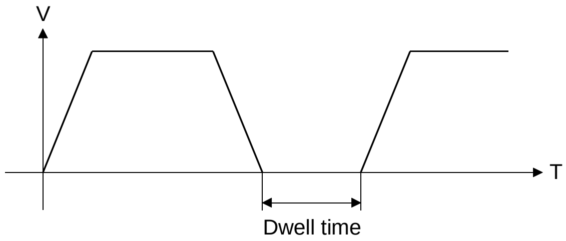
6.8.7 G04 Dwell

[Explanation]

- For the dwell command, specify the time from a stop after deceleration under the preceding move command until the next block starts.

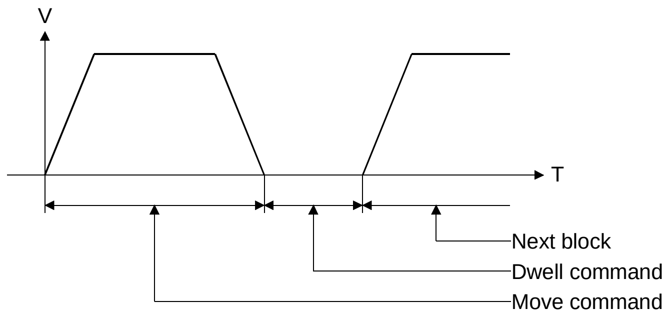
- The symbol indicating the dwell time is "P".

- The dwell time can be specified in the range 1 to 65535 in increments of 0.001 seconds.
  Therefore, setting of G04 P1000 indicates a wait time of 1 second.



- The dwell time can be set by direct designation (numerical value) or indirect designation (variable: #****).

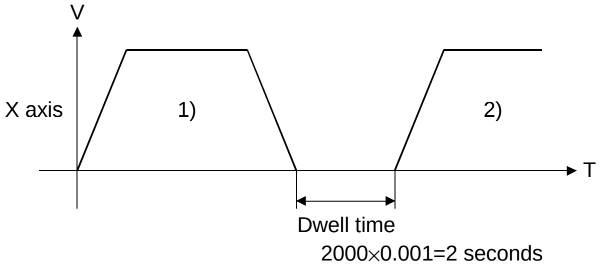
- When specifying dwell in the same block as the move block, describe dwell after the move command.
  Also, describe the dwell time (P) after G04.
  [Example]
  G00 X100 Y100 G04 P2000;
  G04 P2000 — Dwell command
  G00 X100 Y100 — Move command
  (G00, G02, G01 or G03 can be specified)



	Format	G04␣P p; Dwell time (1 to 65535)

[Program Example]

- Program in which dwell time is placed between positioning operation instructions.
  1) G01 X100. F10. ; (Positioning)
  2) G04 P2000; (Dwell time set to 2 seconds)
  3) G01 X200. ; (Positioning)



The X axis is positioned to 100., stops there for 2 seconds, and starts positioning operation to 200. again.

REMARK

- A decimal point cannot be specified for the dwell time.

Code	G09	Moves the axis in the specified block point-to-point.
Function	Exact stop check	

6.8.8 G09 Exact stop check

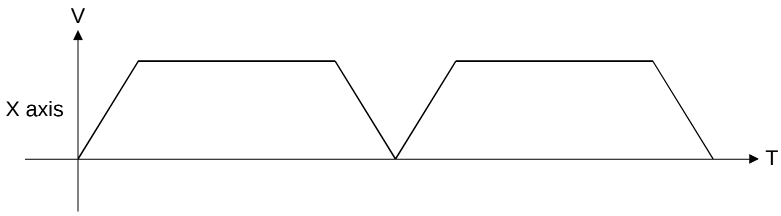
[Explanation]

- This command is used with the interpolation instruction. Executing this command moves the axis point-to-point in only the specified block.
  The interpolation instruction codes usable with this command are G01, G02 and G03 only.

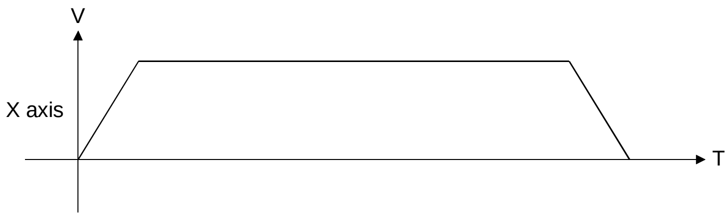
- In this system, the next block is executed after deceleration to a stop in the specified coordinate position.

- Not being a modal instruction, this command is valid for the specified block only.

<When exact stop check is used>
G09 G01 X100. F300. ;
X200. ;



<When exact stop check is not used>
G01 X100. F300. ;
X200. ;



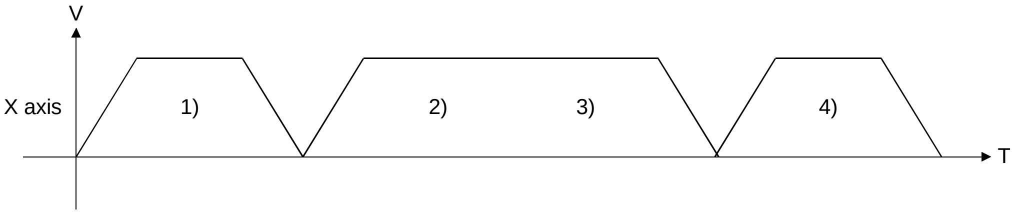
- The positioning data can be set by direct designation (numerical value) or indirect designation (variable: #****).

	Format	G09␣G01␣X x␣F f; May be used only in the G01, G02 or G03 program

[Program Example]

- Program which uses the exact stop check for positioning.

  1) G09 G01 X100. F500. ; (Positioning using exact stop check)
  2) X200. ; (Positioning)
  3) X300. ; (Positioning)
  4) G09 G01 X400. ; (Positioning using exact stop check)



Code	G23	Makes invalid G24 (cancel function, cancel start function) which has already been made valid. Valid until G24 (cancel function, cancel start function) is executed.
Function	Cancel, cancel start invalidity	

6.8.9 G23 Cancel, cancel start invalidity

[Explanation]

- This command makes invalid the cancel or cancel start function which has already been made valid.

- This function is also valid for the high-speed oscillation axis.

```
N1 G24 CAN #X100;
N2 G01 X200. F200. ;               Cancel function is valid for N2 and N3.
N3 G25 Y START90. STRK1. F10;
N4 G23; ------------------------> Cancel function invalid
                                   (Cancel function is also made invalid for
                                   the high-speed oscillation axis.)
```

	Format	G23;

[Program Example]

- Program which makes the cancel start function valid/invalid during execution of a 010 program.
  010
  G24 CAN #X100 P100 PB1; Execution of cancel start function
  G90 G01 X200. F1000. ;
  G23; Cancel start function invalid

Code	G24	Cancels the running program and automatically starts the specified start program. This function is valid until cancel or cancel start function invalidity (G23) is executed.
Function	Cancel, cancel start	

6.8.10 G24 Cancel, cancel start

[Explanation]

- Turning ON the cancel device signal during execution of this command decelerates the axis to a stop and cancels the running program (cancel function). When the start program number Pn has been set, turning ON the cancel signal decelerates the axis to a stop and automatically starts the specified program (cancel start function).

- This command cannot be used with the home position return (G28) instruction.

- In a waiting status for a restart (single block, M00, M01) during macro processing, this command is made valid after completion of the processing.

- If the cancel device turns ON during move block switching, a cancel start is made valid at the processing of the next move block when there are no operating axes (no high-speed oscillation axes).

- The devices that may be used for cancel are X, Y, M, TC, TT, CC, CT, B and F. By assigning the input signal designed for high-speed read function to the cancel device, response is made faster than the input from the PC.

- The setting range of the program number Pn for a start is 1 to 256.

- The parameter block of the start program can be set with PBn. The setting range of the parameter block number PBn is 1 to 16. If the setting of the parameter block number PBn is omitted, it is fixed to parameter block number 1.

- The program number Pn and parameter block number PBn set for a start can be set by indirect designation using a variable, D or W (2-word data).

- When G24 exists at any point between continuous CP blocks, the axis decelerates to a stop once.

N1 G24 CAN #X100; N2 G01 X200. F2000. ; N3 X300.Y200. ;	Cancel function for N1 is valid until G24 or G23 is specified.
N4 G24 CAN #X101;	Cancel function for N1 is made invalid and the axis decelerates to a stop.
N5 G01 X50.Y50 F1000. ;	Cancel function for N4 is valid until G24 or G23 is specified.

	Format	G24␣CAN␣#X x␣P n␣ PBn;
		PBn: Parameter block number (can be specified indirectly) P n: Start program number (can be specified indirectly) #X x: Cancel device (X, Y, M, TC, TT, CC, CT, B, F) CAN: Cancel designation

- When G24 is executed after high-speed oscillation (G25), the high-speed oscillation axis also stops.

  N1 G25 X START90. STRKI. F10;
  N2 G24 CAN #X100 P100;
  N3 G01 Y100. Z100. F1000. ;
  N4 G26 X;
  N5 G01 X0. Y0. Z0. F1000. ;
  N6 G23;

  Cancel function for N2 is valid between N3 and N5. Note that the high-speed oscillation axis also stops if cancel is made invalid in this area.

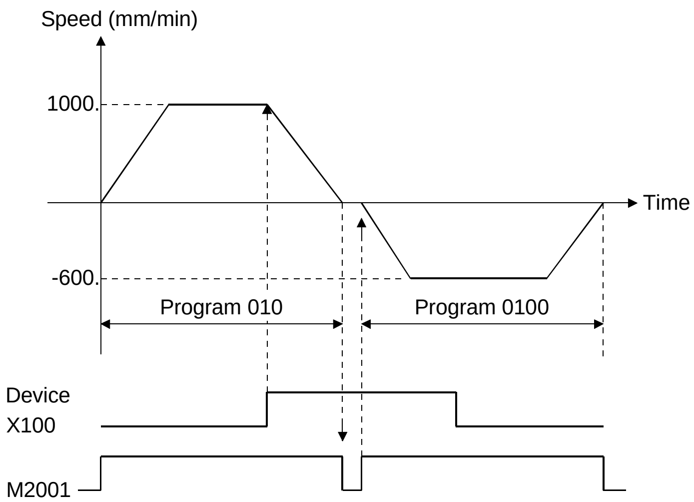
- If the start program number Pn is omitted (cancel function), the running program ends when the cancel device turns ON.

- When setting the start axes in the SVST instruction, also include the axis number to be executed in the start program. Making a start turns ON the start acceptance flag of the set axis. The start acceptance flag turns OFF once at a cancel time, but it turns ON again when the axis is started in the original program at a start program run.

[Program Example]

- Program which cancels program operation during a 010 program run and starts 0100. (Command unit is mm)

  010;
  1) G24 CAN #X100 P100 PB1; Execution of cancel start function
  2) G90 G01 X200. F1000. ; Cancel device X100 turns ON midway. After deceleration to stop, 0100 starts.
  0100;
  3) G90 G01 X50. F600. ; X axis moves to 50mm position at 600mm/min.

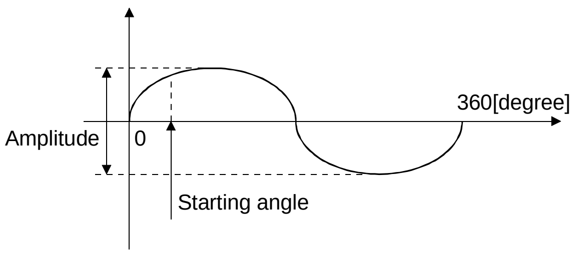


Code	G25	Oscillates the specified axis in a sine curve.
Function	High-speed oscillation	

6.8.11 G25 High-speed oscillation

[Explanation]

- The specified axis oscillates in a sine curve.



Amplitude : Specify the oscillating amplitude in the setting unit. It can be specified indirectly by a variable, D or W (2-word data). The setting range is 1 to 2147483647. If the setting is outside the range, a minor error (error code: 585) occurs, disabling a start.

Starting angle: Specify the starting position with the angular position of a sine curve. It can be specified indirectly by a variable, D or W (2-word data). Set it within the range is 0 to 359.9 [degrees] in 0.1 degree increments. If the setting is outside the range, a minor error (error code: 586) occurs, disabling a start.

Frequency : Specify the number of cycles in which the axis will be operated for 1 minute in a sine curve. It can be specified indirectly by a variable, D or W (2-word data). The setting range is 1 to 5000 [CPM]. If the setting is outside the range, a minor error (error code: 587) occurs, disabling a start.

	Format	G25␣X␣START␣s␣STRK␣a␣F f;

- f: Frequency (can be specified indirectly)
- F: Frequency designation (can be specified indirectly)
- a: Amplitude (can be specified indirectly)
- STRK: Amplitude designation
- s: Starting angle (can be specified indirectly)
- START: Starting angle designation
- X: Axis name

• This command is valid for the specified block only (group 00).

• After a start, operation continues until G26, high-speed oscillation stop, is executed or the stop command is entered.

• Acceleration/deceleration processing is not performed. When you want to avoid a sudden start, set the starting angle to 90.0 [degrees] or 270.0 [degrees].

[Program Example]

• Program in which the X axis oscillates in the sine curve of 10 [mm] amplitude, 90 [degree] starting angle and 30 [CPM] frequency.
(Command unit is mm)
G25 X START 90. STRK 10. F30;

Note: The starting angle (START) is valid to the first decimal place.
Example (1) START 90. .... Means 90.0 (degrees).
(2) START 90. .... Means 9.0 (degrees).
(3) In START #10
#10 = 900 .... Means 90.0 (degrees).
#10 = 1 .... Means 0.1 (degrees).

Code	G26	Terminates the high-speed oscillation of the axis which is performing high-speed oscillation.
Function	High-speed oscillation stop function	

6.8.12 G26 High-speed oscillation stop

[Explanation]

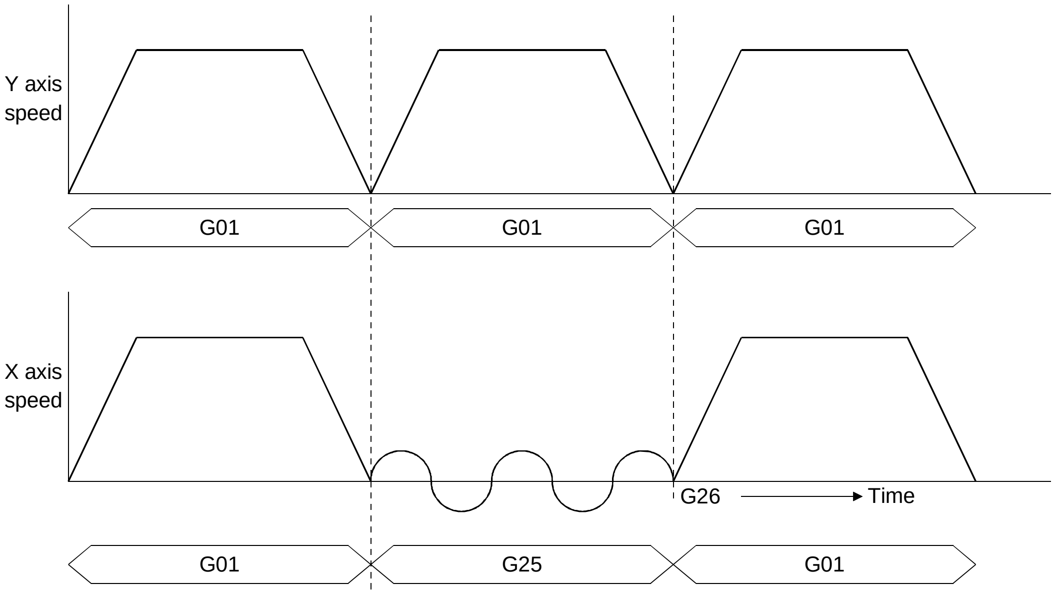
- Stops the high-speed oscillation of the axis which is performing high-speed oscillation.

- Use this command in pairs with a high-speed oscillation start.
  When the corresponding axis is not stopped up to a program END (M02, M30) after a high-speed oscillation start, high-speed oscillation is kept performed at a program END.
  Also, do not set a stop to the axis which has not made a high-speed oscillation start. In that case, a minor error (error code: 582) is displayed and execution proceeds to the next block.

	Format	G26␣X; Axis name

[Program Example]

```
N01 G91 G01 X10. Y10. F100. ;
N02 G25 X START 0. STRK 1000. F100. ;
N03 G01 Y10. ;
N04 G26 X;
N05 G01 X10. Y10. ;
M02;
```



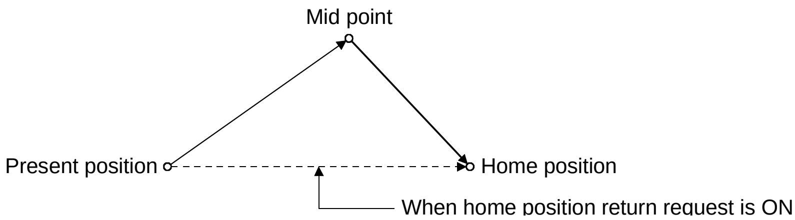
- If the start command of the X axis (high-speed oscillation start axis) is described in the N03 block, a minor error (error code: 581) is displayed when this block is executed, and this program is suspended.

Code	G28	When the home position return request is ON, ignores the mid point specified and makes a dog, count or data setting type home position return. When the home position return request is OFF, returns the axis from the present position to the home position through the specified mid point at rapid feedrate.
Function	Home position return	

6.8.13 G28 Home position return

[Explanation]

- When the home position return request is ON, this command ignores a mid point and returns the specified axis to the home position. When the home position return request is OFF, this command positions the axis from the present position to the home position through the specified mid point at rapid feedrate.



- When the home position return request is ON, the home position return method is determined by the home position return data.
  Note: When the home position return request is ON and the data setting type is specified, the axis must always be made to pass through the zero point.
  A "zero point non-passage error" will occur if a home position return is made without passing through the zero point once. If this error has occurred, reset the error, perform JOG operation or the like to run the servo motor more than one revolution, then execute a home position return again.
  Use the zero point passage signal (M1606+20n) to check whether the axis has passed through the zero point.

- Always specify the axis which will be returned to the home position. If it is not specified, a home position return will not be made.

- Always set the mid point coordinates.

- The mid point data setting can be made by direct designation (numerical value) or indirect designation (variable: #****).

- The tool length offset and virtual mechanical coordinates (refer to Section 6.8.25) of the axis which was returned to the home position are canceled.
  Mid point designation depends on the position command system (G90, G91) currently selected.

- When the control unit is degrees, operation from the mid point to the home position differs between the absolute value command (G90) and incremental value command (G91).
  The axis moves in the nearest path under the absolute value command (G90), or in the direction specified in the home position return direction parameter under the incremental value command (G91).

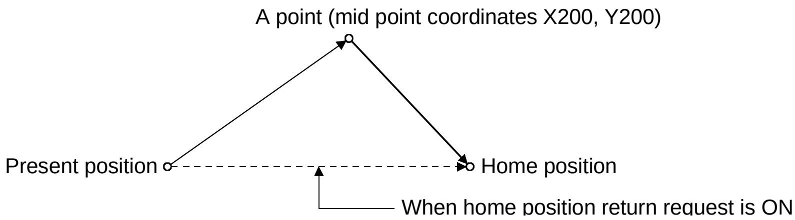
[Related Parameters]

Home position address: Set the present value of the home position. (Refer to the home position return data in Section 4.4.)
Rapid feedrate : Set the rapid feedrate of each axis. (Refer to the fixed parameters in Section 4.2.4.)

Format	G28␣X x␣Y y␣Z z; Mid point coordinates

[Program Example]

- Program which returns the axis from the present position to the home position through the A point (mid point).
  G90;
  G28 X200. Y200. ; (Home position return)



REMARK

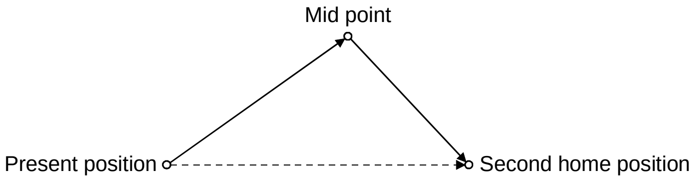
- When the G28 command is given, a home position return is made at rapid feedrate.

Code	G30	Returns the axis from the present position to the second home position through the specified mid point at rapid feedrate.
Function	Second home position return	

6.8.14 G30 Second home position return

[Explanation]

- This command positions the specified axis from the present position to the second home position through the specified mid point at rapid feedrate.



- Always specify the axis which will be returned to the second home position. If it is not specified, a second home position return will not be made.

- Always set the mid point coordinates.

- The mid point data setting can be made by direct designation (numerical value) or indirect designation (variable: #****).

- The tool length offset and virtual mechanical coordinates (refer to Section 6.8.25) of the axis which was returned to the second home position are canceled.
  Mid point designation depends on the position command system (G90, G91) currently selected.

- When the control unit is degrees, operation from the mid point to the second home position differs between the absolute value command (G90) and incremental value command (G91).
  The axis moves in the nearest path under the absolute value command (G90), or in the direction specified in the home position return direction parameter under the incremental value command (G91).

[Related Parameters]

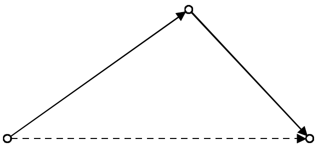
Second home position address: Set the present value of the second home position. (Refer to the home position return data in Section 4.4.)

Rapid feedrate : Set the rapid feedrate of each axis. (Refer to the fixed parameters in Section 4.2.4.)

	Format	G30␣X x␣Y y␣Z z;
		$\underbrace{\text{X x␣Y y␣Z z}}$ Mid point coordinates

[Program Example]

- Program which returns the axis from the present position to the second home position through the A point (mid point).
  G90;
  G30 X200. Y200. ; (Second home position return)



REMARK

- When the G30 command is given, a second home position return is made at rapid feedrate.

Code	G32	Moves the axis at the specified feedrate, suspends the remaining command at the input of an external signal, and executes the next block. Skips dwell similarly when there is only the dwell command.
Function	Skip	

6.8.15 G32 Skip

[Explanation]

- When the skip signal is entered during execution of G32, skip, the remaining motion of that block is suspended and the next block is executed. Dwell may also be skipped by giving the dwell command (P) in the G32 block without specifying the axis.
- A format error occurs if the axis command or M code and the dwell command are described at the same time.
- Specify the dwell time in the range 1 to 65535 in increments of 0.001 seconds.
- Specify the skip signal in the program.
- The skip function makes a skip when the skip signal turns ON.
- This command is valid for the specified block only (group 00). The interpolation type of this command is the CP mode.
- When the skip signal is not input until the end point of this command block, the block completes at the end point.
- For dwell/skip, the block completes on completion of the dwell processing.
- The next circular interpolation cannot be made.
- The F command is handled like G01.

	Format	<When axis is specified> **G32␣X x␣Y y␣F f␣SKIP␣#Xx;** Skip device (X,Y,M,TC,TT,CC,CT,B,F) Skip command Feedrate (can be specified indirectly) Feedrate command Positioning address (can be specified indirectly) Axis name <When dwell is specified> **G32␣P p␣SKIP␣#Xx;** Skip device (X,Y,M,TC,TT,CC,CT,B,F) Skip command Dwell time Dwell command

- The coasting value δ_A between skip signal detection and a stop is represented by the following expression.

$$\delta_A(\text{mm}) = \frac{F}{60}\left(t1 + \frac{tcl}{2} + Tr\right)$$

F : Command speed [mm/min]
t_1 : Signal import delay time ≒ 0.004 + detection delay time [sec]
tcl : Acceleration/deceleration time [sec]
Tr : Position loop time constant [sec]
(Reciprocal number of position control gain 1 value set in servo parameter. When position control gain 1 = 25, Tr = 1/25 = 0.04 [sec])

- Under the following conditions, G32 makes deceleration to a stop once, then proceeds to the next block.
  1) When the PTP mode (G00, G25, G28, G30 or the like) is executed after the G32 block
     N10 G32 X100. F1000. SKIP #X10;
     N20 G00 X200. ; → The axis decelerates to a stop before this block.
     N30 G32 X300. F1000. SKIP #X11;
  2) High-speed oscillation stop (G26) is executed after the G32 block
     N10 G25 Y START 90. STRK 1. F400. ;
     N20 G32 X100. F1000. SKIP #X10;
     N30 G26 Y; → The axis decelerates to a stop before this block.
     G32 X200. F1000. SKIP #X11;
  3) When the absolute value command (G90) or incremental value command (G91) is executed after the G32 block
     N10 G90;
     N20 G32 X100. F1000. SKIP #X10;
     N30 G91; → The axis decelerates to a stop before this block.
     N40 G32 X200. F1000. SKIP #X11;

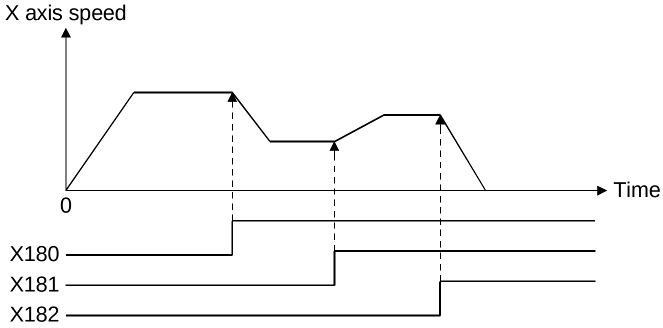
4) When the block immediately after G32 is in the CP mode but its command axes do not include the specified axis of the G32 block

N10 G32 X100. F1000. SKIP #X10;

N20 G32 X100. Z100. F1000. SKIP #X11; --------→ The axis decelerates to a stop before this block.

[Program Example]

- Program designed to make multiple skips under the control of external skip signals specified from the program midway through positioning.
  (Under incremental value command)
  - G91;
  - G32 X100. F2000 SKIP #X180;   Turns ON the X180 signal midway.
  - G32 X100. F1000 SKIP #X181;   Turns ON the X181 signal midway.
  - G32 X200. F1500 SKIP #X182;   Turns ON the X182 signal midway.



- Under dwell command
  If cancel device X100 turns ON during dwell in N01, G0 in N02 where dwell was suspended is executed.
  N01 G32 P1000 SKIP #X1000;
  N02 G90 G0 X100. ;

⚠ CAUTION

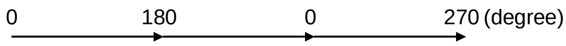
⚠ The following operation assumes that a skip (G32) is specified during constant-speed control (G01) and the degree axis without a stroke range is included.
When, under this condition, an instruction of an absolute value command exists after a skip, the last positioning point and the travel distance in the whole program are the same independently of whether a skip is executed or not. This is indicated by the following example.

(1) When the skip instruction is an incremental value command and subsequent instructions are also incremental value commands

<Program example>

```
G91;
G32 X180. SKIP#X100 F10. ;
G01 X180. ;
G01 X270. ;
```

<Motion without a skip>



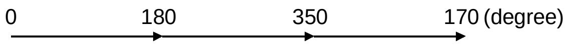
<Motion with a skip>
(When a skip is made at 100 (degree))

(2) When the skip instruction is an absolute value command and subsequent instructions are also absolute value commands

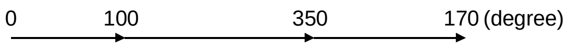
<Program example>

```
G90;
G32 X180. SKIP#X100 F10. ;
G01 X350. ;
G01 X170. ;
```

<Motion without a skip>



<Motion with a skip>
(When a skip is made at 100 (degree))



The last positioning point is the same if a skip is not provided.

(*) It should be noted that the above explanation is valid between a skip (G32) and deceleration to a stop (CP to PTP, etc.) After deceleration to a stop, operation of the ordinary degree axis is performed. The conditions of deceleration to a stop after a skip (G32) are described below. For more information, refer to "6.8.15 G32 Skip".

1) When the PTP mode (G00, G25, G28, G30 or the like) is executed after the G32 block
2) High-speed oscillation stop (G26) is executed after the G32 block
3) When the absolute value command (G90) or incremental value command (G91) is executed after the G32 block
4) When the block immediately after G32 is in the CP mode but its command axes do not include the specified axis of the G32 block

Code	G43	Moves the axis with the preset offset value added to the move command. By setting a difference between the tool length value and actual tool length as the offset value, you can create a program without being aware of the tool length.
Function	Tool length offset (+)	

6.8.16 G43 Tool length offset (+)

[Explanation]

- By executing this command, the axis moves to the position which results from adding the offset value set in the tool length offset data setting registers to the end position of the move command.

- In the following case, the tool length offset command is canceled.
  G49; ——— Tool length offset cancel command
  G43 H0;
  G44 H0; ——— Set the offset data number 0 to cancel the tool length offset.
- This command may be given to one axis only. If this command is given to two or more axes, it is valid for the last specified axis.
  G43 X1. Y1. Z1. H1; —— The Z axis is made valid.
  If no axis is specified, the last specified axis is made valid.
  G01 Z1;
  G43 H1; —— The Z axis is made valid.

- As this command is a modal instruction, the offset value is retained until the offset value is canceled (G49).

- Tool length offset may be made to only one axis simultaneously. (Both G43 and G44)
  ⋮
  G43 X100. H1;
  G43 Y100. H2; ←—— Cannot be used this way.

[Related Parameters]

Tool length offset value: Set in the tool length offset data setting registers. (Refer to Section 3.2.3.)

Format	G43␣X x␣H h; h: Offset data number x: Positioning address X: Axis name

[Program Example]

- Program designed to position the axis with the offset value added to the command position. (For absolute value command)
  (Data of the tool length offset data setting registers are as follows:
  H1 = 5mm (D560, 561 = 50000), H2 = 10mm (D562, 563 = 100000))

G90;	(Absolute value command)
G00 G43 X50. H1	(With the addition of the offset value of 5mm, the X axis is positioned to its 55mm position)
G01 X25. F500. ;	(The X axis moves to its 30mm position at 500mm/min.)
Y100. ;	(The Y axis moves to its 100mm position at 500mm/min.)
G43 X200. H2;	(With the addition of the offset value of 10mm, the X axis moves to its 210mm position (offset value change))

Code	G44	Moves the axis with the preset offset value subtracted from the move command. By setting a difference between the tool length value and actual tool length as the offset value, you can create a program without being aware of the tool length.
Function	Tool length offset (-)	

6.8.17 G44 Tool length offset (-)

[Explanation]

- By executing this command, the axis moves to the position which results from subtracting the offset value set in the tool length offset data setting registers from the end position of the move command.

- In the following case, the tool length offset command is canceled.
  G49; ——— Tool length offset cancel command
  G43 H0; ┐
  G44 H0; ┘—— Set the offset data number 0 to cancel the tool length offset.
- This command may be given to one axis only. If this command is given to two or more axes, it is valid for the last specified axis.
  G44 X1. Y1. Z1. H1; —— The Z axis is made valid.
  If no axis is specified, the last specified axis is made valid.
  G01 Z1. ;
  G44 H1; —— The Z axis is made valid.

- As this command is a modal instruction, the offset value is retained until the offset value is canceled (G49).

- Tool length offset may be made to only one axis simultaneously. (Both G43 and G44)
  ⋮
  G44 X100. H1;
  G44 Y100. H2; ←—— Cannot be used this way.

[Related Parameters]

Tool length offset value: Set in the tool length offset data setting registers. (Refer to Section 3.2.3.)

	Format	G44␣X x␣H h; H h: Offset data number x: Positioning address X: Axis name

[Program Example]

- Program designed to position the axis with the offset value subtracted from the command position. (For absolute value command)
  (Data of the tool length offset data setting registers are as follows:
  H1 = 5mm (D560, 561 = 50000), H2 = 10mm (D562, 563 = 100000))

G90;	(Absolute value command)
G00 G44 X50. H1;	(With the subtraction of the offset value of 5mm, the X axis is positioned to its 45mm position)
G01 X25. F500. ;	(The X axis moves to its 20mm position at 500mm/min.)
Y100. ;	(The Y axis moves to its 100mm position at 500mm/min.)
G44 X200. H2;	(With the subtraction of the offset value of 10mm, the X axis moves to its 190mm position (offset value change))

Code	G49	Cancels the preset tool length offset value (G43, G44).
Function	Tool length offset cancel	

6.8.18 G49 Tool length offset cancel

[Explanation]

- This command cancels the preset tool length offset value (G43, G44) and performs the specified positioning.
- Always specify the positioning address for tool length offset cancel.

[Related Parameters]

Power-on mode: At power-on, the tool length offset cancel mode is established.

	Format	G49␣X x; x: Positioning address X: Axis name

[Program Example]

- Program designed to cancel the offset value and perform the specified positioning after positioning has been executed by tool length offset. (For absolute value command)
  (Data of the tool length offset data setting registers are as follows:
  H1 = 5mm (D560, 561 = 50000), H2 = 10mm (D562, 563 = 100000))

G90;	(Absolute value command)
G00 G43 X50. H1;	(With the addition of the offset value of 5mm, the X axis is positioned to its 55mm position)
G01 X25. F500. ;	(The X axis moves to its 30mm position at 500mm/min.)
Y100. ;	(The Y axis moves to its 100mm position at 500mm/min.)
G43 X200. H2;	(With the addition of the offset value of 10mm, the X axis moves to its 210mm position (offset value change))
G49 X100. ;	(With the offset value canceled, the X axis moves to its 100mm position at 500mm/min.)

Code	G53	Moves the axes to the command position in the basic mechanical coordinate system at rapid feedrate.
Function	Mechanical coordinate system selection	

6.8.19 G53 Mechanical coordinate system selection

[Explanation]

- The basic mechanical coordinate system represents the position determined for a specific machine (e.g. tool changing position, stroke end position).
  It is automatically set relative to the predetermined reference point after a home position return is executed by the DSFLP instruction at power-on.

- Not being a modal instruction, this command is valid for the specified block only.

- When G53 and G28 are specified in the same block, the latter command is valid.
  G53 G28........; —— G28 is valid (home position return command)
  G28 G53........; —— G53 is valid (mechanical coordinate system selection command)

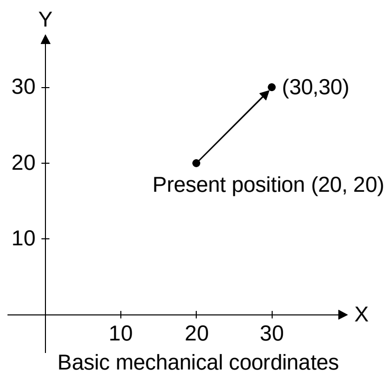
- When G53 and G30 are specified in the same block, the latter command is valid.
  G53 G30........; —— G28 is valid (second home position return command)
  G30 G53........; —— G53 is valid (mechanical coordinate system selection command)

- The offset specified in G92 is not valid.

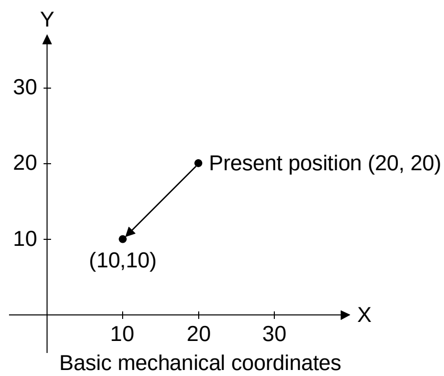
- The tool length offset specified in G43 or G44 is not valid.

- Under the incremental value command (G91), the axes move at the incremental value in the mechanical coordinate system, and under the absolute value command (G90), the axes move at the absolute value in the mechanical coordinate system.
  [Example]

  G91; (For incremental value command)
  G53 X10. Y10.;

  G90; (For absolute value command)
  G53 X10. Y10. ;

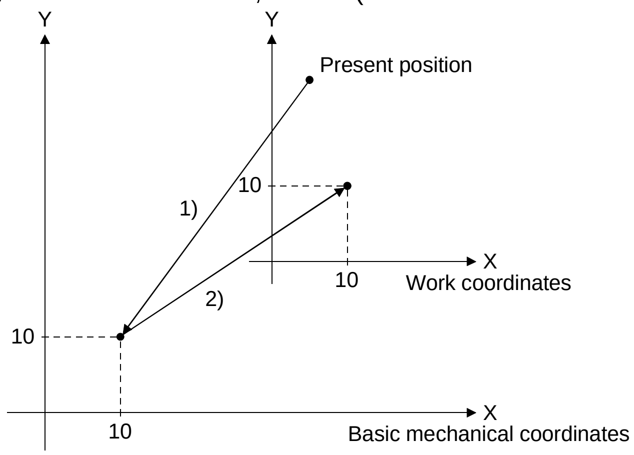
  

- Positioning data can be set by direct designation (numerical value) or indirect designation (variable: #****).

	Format	G53␣X x␣Y y␣Z z; (X x␣Y y␣Z z: Coordinates in basic mechanical coordinate system)

[Program Example]

- Program designed to position the axes to the specified position in the work coordinate system after positioning them to the specified position in the basic mechanical coordinate system in the absolute value mode.
  0) G90; (Absolute value command)
  1) G53 X10. Y10. ; (Axes move to X10. Y10. in the basic mechanical coordinates)
  2) G01 X10. Y10. F20. ; (Axes move to X10. Y10. in the work coordinates)



(Unit: mm)

REMARK

- Motion under G53 is always processed by G00. (The modal group 01 is not changed.)

Code	G54, G55, G56, G57, G58, G59	Selects the work coordinate system and moves the axes to the specified position in the work coordinate system at the speed specified in the feedrate.
Function	Work coordinate system 1 to 6 selection	

6.8.20 G54 to G59 Work coordinate system selection

[Explanation]

- Work coordinate systems 1 to 6 are coordinate systems specified in the parameters or work coordinate system setting.
  Set the offset value in the work coordinate system using the distance from the basic mechanical coordinate system origin (0).

- The coordinate system of G54 is selected at a motion program start.

- Being a modal command, any of work coordinate systems 1 to 6 is valid until the next work coordinate system 1 to 6 selection command is given.

- Giving the G92 command in any of the G54 to G59 modes allows a new work coordinate system to be set.
  Giving the G92 command causes all work coordinates systems (1 to 6) to move in parallel.

<Work coordinate system selection>
G54 Xx Yy Zz;

<Work coordinate system change>
G54 G92 Xx Yy Zz; ....Work coordinates 2 to 6 also move in parallel similarly.

- Move mode (moving method): G00 to G03 depend on the data of the modal information group 01.

- CP mode (constant-speed control): G61 and G64 depend on the the data of the modal information group 13.

- Positioning data can be set by direct designation (numerical value) and indirect designation (variable: #****).

[Related Parameters]

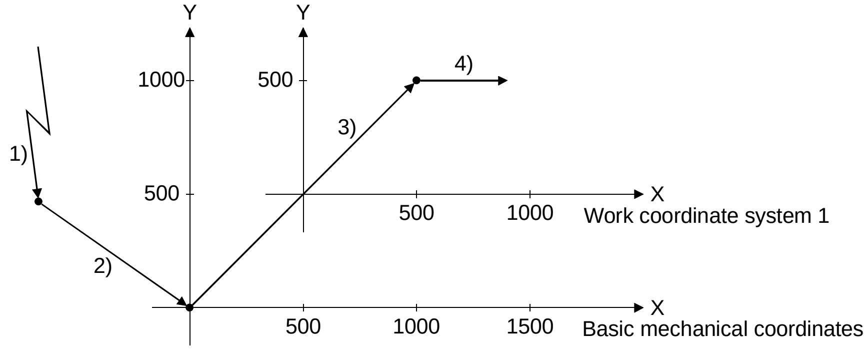
Work coordinate system offset value: Specify the offset in the work coordinate system using the distance from the basic mechanical coordinates. (Refer to the work coordinate data in Section 4.7.)
Up to six work coordinate systems may be set. (Work coordinate systems 1 to 6)

	Format	G54␣X x␣Y y␣Z z; ~ G59 X x␣Y y␣Z z: Position located in specified work coordinate system

[Program Example]

<Work coordinate system selection>

- Program designed to position the axes to the specified position in the work coordinate system 1.
  (The offset of the work coordinate system 1 is X500, Y500)
  0) G90; (Absolute value command)
  1) G28 X0. Y0.; (Home position return)
  2) G53 X0. Y0.; (Axes move to the basic mechanical coordinate origin)
  3) G54 X500. Y500.; (Axes move to the specified position in the work coordinate system 1)
  4) G91 G01 X500. F10. ; (Incremental value command positioning)

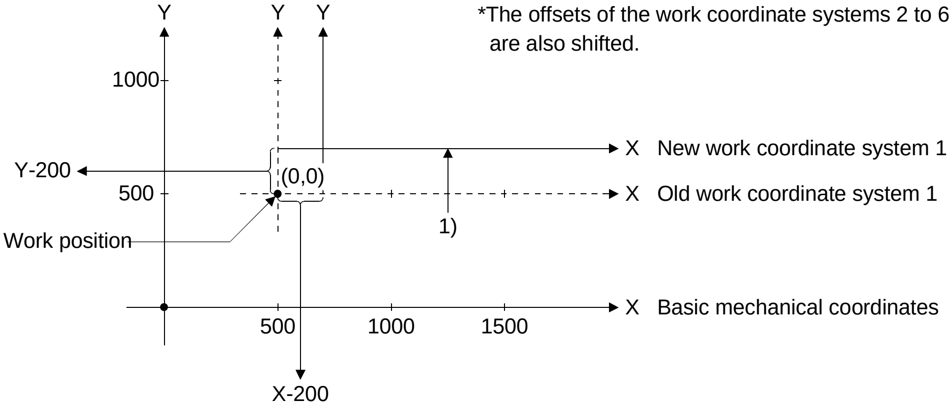


(Unit: mm)

<Work coordinate system change>

- Program designed to set the offset of the work coordinate system 1 to X500, Y500 in the parameter setting of work coordinate data, then change the work coordinate system to new work coordinate system 1.
  1) G54 G92 X-200. Y-200. ; (New work coordinate system 1 setting)
  (After execution of 1), the present value is changed to X-200, Y-200.)

*The offsets of the work coordinate systems 2 to 6 are also shifted.



Code	G61	Moves the axis point-to-point (PTP).
Function	Exact stop check mode	

6.8.21 G61 Exact stop check mode

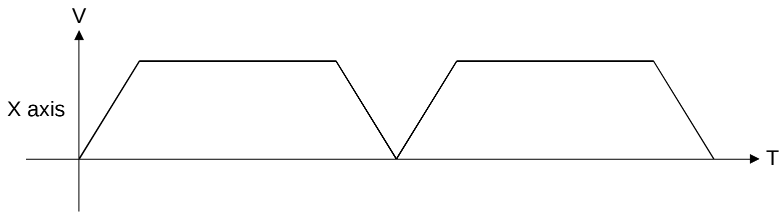
[Explanation]

- This command is used with the interpolation instruction. Executing this command moves the axis PTP.
  The instruction codes usable with this command are G01, G02 and G03 only.

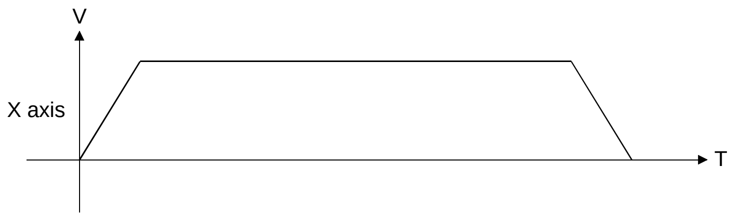
- In this system, the next block is executed after deceleration to a stop per specified coordinates.

- Being a modal instruction, this command is valid until the cutting mode (G64) is commanded.

<In exact stop check mode>
G61 G01 X100. F500.;
X200. ;



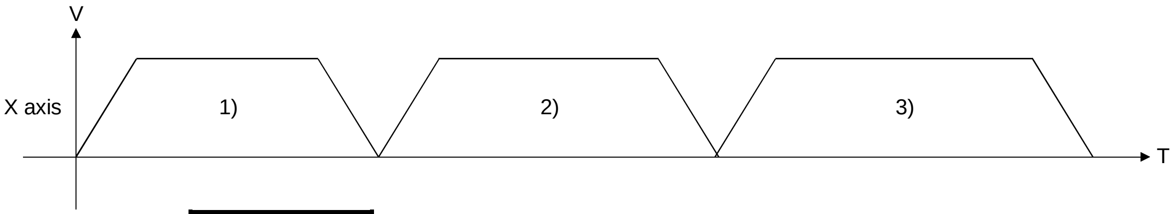
<No in exact stop check mode>
G01 X100. F500.;
X200. ;



	Format	G61;

[Program Example]

• Program designed to position the axis in the exact stop check mode.
1) G61 G01 X100. F500.; (Positioning in the exact stop check mode)
2) X200. ; (Positioning in the exact stop check mode)
3) X300. ; (Positioning in the exact stop check mode)



REMARK

• Only the rapid feedrate may be the specified speed in G00. To specify the speed every time PTP positioning is executed, you can use G61 and G01.

Code	G64	Executes the next block continuously without deceleration to a stop between cutting feed blocks.
Function	Cutting mode	

6.8.22 G64 Cutting mode

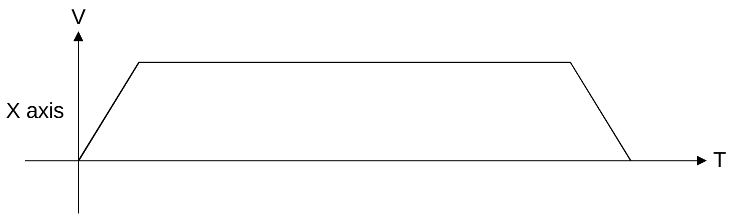
[Explanation]

- Designed to position the axis to the specified coordinate position approximately, this command performs continuous operation without deceleration to a stop per specified coordinates unlike the exact stop check mode.
  Use this command when you want to make a smooth connection with the interpolation instruction (G01, G02, G03).

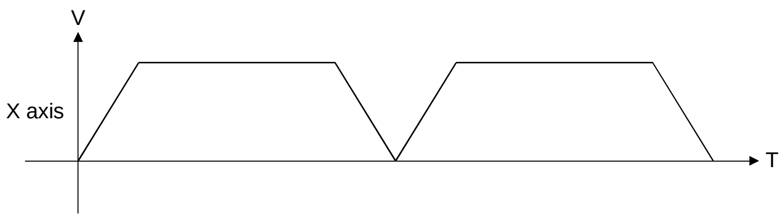
- The cutting mode is established at a motion program start.

- Being a modal instruction, this command is valid until the exact stop check mode (G61) is commanded.

<In cutting mode>
G64 G01 X100. F500. ;
X200. ;



<Not in cutting mode>
G61 G01 X100. F500. ;
X200. ;

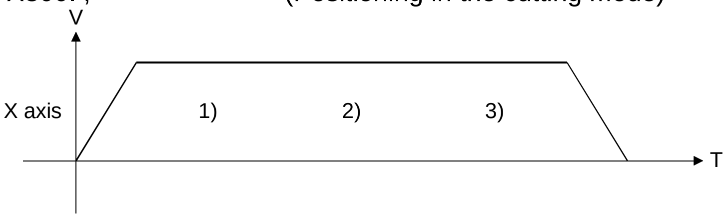


	Format	G64;

[Program Example]

- Program designed to position the axis in the cutting mode.

```
1) G64 G01 X100. F500. ;(Positioning in the cutting mode)
2) X200. ;               (Positioning in the cutting mode)
3) X300. ;               (Positioning in the cutting mode)
```



Code	G90	Sets the coordinate command as an absolute value command.
Function	Absolute value command	

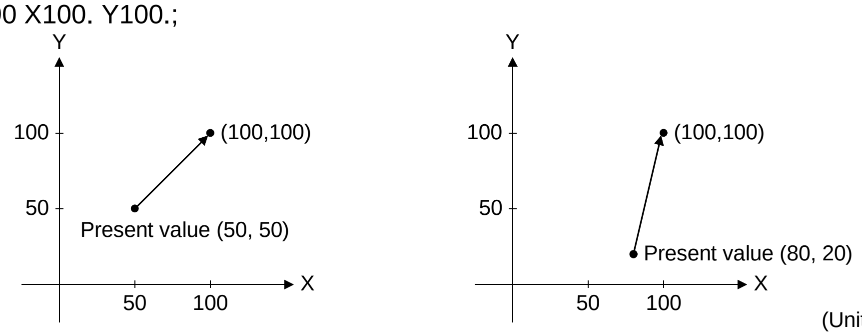
6.8.23 G90 Absolute value command

[Explanation]

- In the absolute value command mode, the axes move to the specified coordinate position independently of the present position. The positioning command set after execution of this command performs operation with the absolute value from the origin coordinates.

- Being a modal instruction, this command is valid until the incremental value command mode (G91) is commanded.

- The absolute value command mode is established at a motion program start.
  [Example]
  G90 X100. Y100.;



(Unit: mm)

At present position coordinates of X50, Y50   At present position coordinates of X80, Y20

- Positioning data can be set by direct designation (numerical value) and indirect designation (variable: #****).

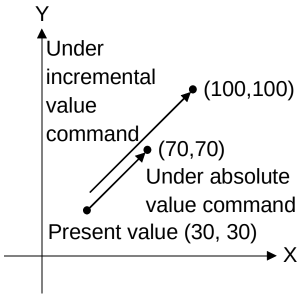
	Format	G90␣X x␣Y y␣Z z; Locating position (X x Y y Z z)

[Program Example]

- Example of comparison of positioning between the absolute value command and incremental value command

  <Incremental value command example>
  G91 X70. Y70.;

  <Absolute value command example>
  G90 X70. Y70.;



(Unit: mm)

Code	G91	Sets the coordinate command as an incremental value command.
Function	Incremental value command	

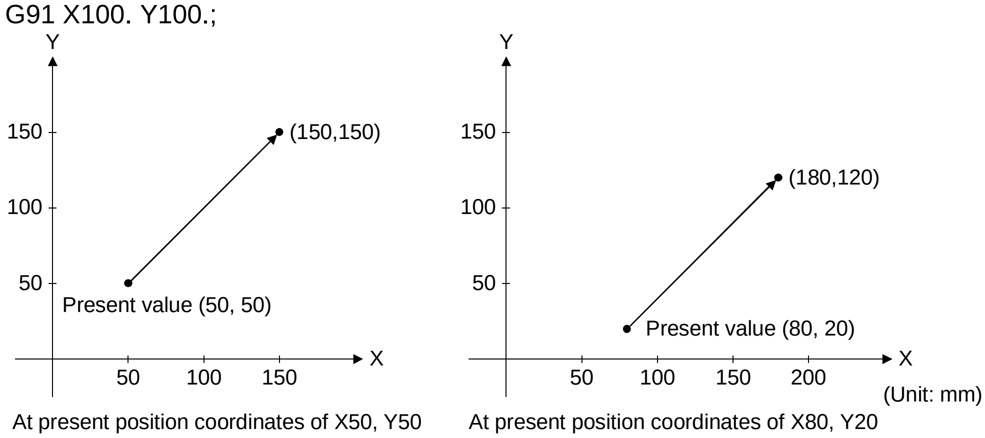
6.8.24 G91 Incremental value command

[Explanation]

- In the incremental value command mode, the axes move the distance of the specified relative value from the starting point (0) of the present position.
  The positioning command set after execution of this command performs operation with the incremental value from the present position.

- Being a modal instruction, this command is valid until the absolute value command mode (G90) is commanded.

- The absolute value command mode is established at a motion program start.
  [Example]
  G91 X100. Y100.;



- Positioning data can be set by direct designation (numerical value) and indirect designation (variable: #****).

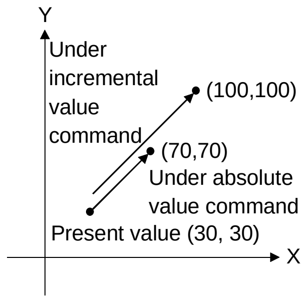
	Format	G91␣X x␣Y y␣Z z; — Locating position

[Program Example]

- Example of comparison of positioning between the incremental value command and absolute value command

  <Absolute value command example>
  G90 X70. Y70.;

  <Incremental value command example>
  G91 X70. Y70.;



(Unit: mm)

Code	G92	Sets the mechanical coordinates (virtual mechanical coordinates) simulatively. Setting the virtual mechanical coordinate system also changes the work coordinate systems 1 to 6.
Function	Coordinate system setting	

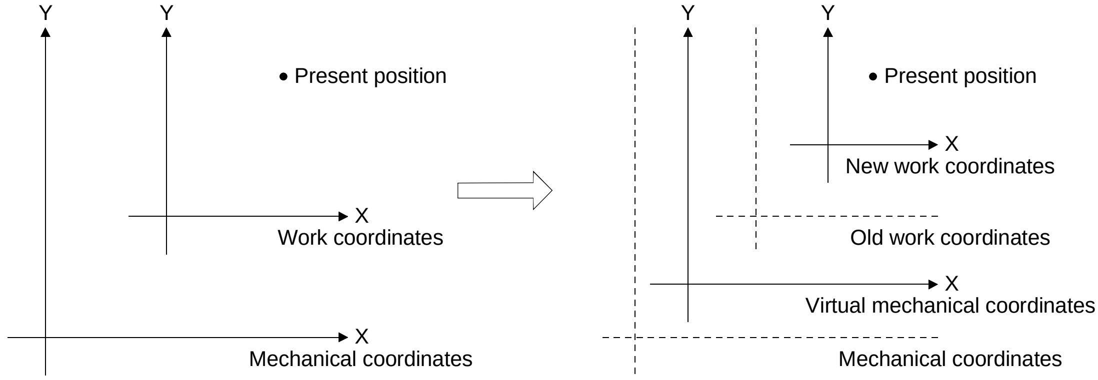
6.8.25 G92 Coordinate system setting

[Explanation]

- The present position in the work coordinate system is changed to the specified coordinate value to set new work coordinates. The work coordinate system is set in the specified position (offset from the present position).
  Making coordinate system setting sets the virtual mechanical coordinates and moves the work coordinate systems 1 to 6 in parallel.

  [Example]

  G92 X20. Y30.;



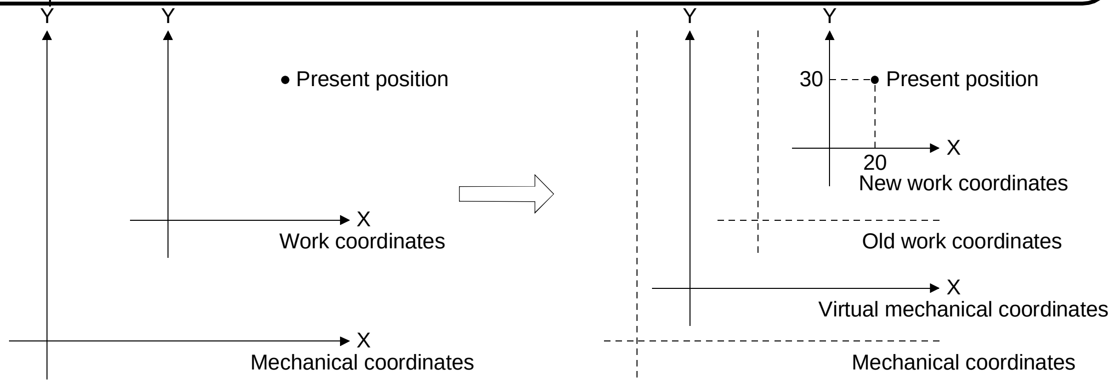
- Positioning data can be set by direct designation (numerical value) and indirect designation (variable: #****).
- When the software version of the controller operating system SV43C, SV43F, SV43U or SV43B is Ver. 00F or earlier and G92 is to be executed in the CP mode (e.g. G01), execute G92 after executing M100 (preread inhibit) to decelerate the axes to a stop once.
- When the software version of the controller operating system SV43C or SV43F is Ver. 00G or later, executing G92 in the CP mode (e.g. G01) decelerates the axes to a stop once. When G92 is executed in the single block mode with this software version or later, making a single block start twice in the same block shifts execution to the next block.

POINT
If the present value is changed in G92, the present value data restored after a power failure is based on the status prior to execution of G92.

[Program Example]

- Program designed to set the work coordinate system in the specified position.
  G92 X20. Y30.;

	Format	G92␣X x␣Y y␣Z z; Set coordinate value (Specify the offset from the present position)



(Unit :mm)

Code	G100, G101	Changes the acceleration/deceleration system to time-fixed acceleration/deceleration or acceleration-fixed acceleration/deceleration.
Function	Time-fixed acceleration/deceleration, acceleration-fixed acceleration/deceleration switching instructions	

6.8.26 G100, G101 Time-fixed acceleration/deceleration, acceleration-fixed acceleration/deceleration switching instructions

[Explanation]

- The acceleration/deceleration system of the move command G01, G02, G03, G32 or G00 (with M code) is switched to time-fixed acceleration/deceleration or acceleration-fixed acceleration/deceleration.

- Specify the G code of this command independently.

- Use G100 to choose time-fixed acceleration/deceleration.
  The G100 status is established at a start.

- Use G101 to choose acceleration-fixed acceleration/deceleration.

- Under G101, acceleration-fixed acceleration/deceleration, the M code does not wait for FIN. (The M code is output to the M code storage register but the M code outputting signal does not turn ON.)

- Acceleration/deceleration in the acceleration-fixed mode is valid until:
  1) G100, time-fixed acceleration/deceleration instruction, is executed;
  2) The program ends under M02;
  3) The program is stopped by the rapid stop command, stop command, error reset or emergency stop; or
  4) The program is stopped at error occurrence.

- When G100 is changed to G101 or G101 to G100, the axes decelerate to a stop.

	Format	G100; G101;

[Program Example]

- Program designed to make the acceleration-fixed acceleration/deceleration mode of the acceleration/deceleration system valid, then invalid midway through the program (command unit: mm)

```
010;
G91;
N1 G28 X0. Y0.;
N2 G01 X100. F1000.;   Time-fixed acceleration/deceleration (at start, operation
N3 Y100.;              is performed under G100) Deceleration to stop after execution
N4 G101;
N5 X100.;              Acceleration-fixed acceleration/deceleration
N6 Y100.;              Deceleration to stop after execution
N7 G100;
N8 X100.;              Time-fixed acceleration/deceleration
N9 Y100.;
M02;
%
```

6.9 M Codes

This section explains the M codes used in motion programs.

(1) M codes

When a motion program is run, the 4-digit code data following M is output to the data register (D) in the M command block.
The processing of the next block is not executed until the FIN signal (M1819+20n/M3219+20n) is entered.
(Refer to Section 7.11 for relationships between the M codes and FIN signal.)

<Command format>

M**** Numeral	Setting range : 0 to 9999 (except M00, M01, M02, M30, M98, M99 and M100)

The M codes usable are 9993 types since M00, M01, M02, M30, M98, M99 and M100 are fixed in functions and they are special M codes. (Refer to Section 6.10 for the special M codes.)

6.10 Special M Codes

Table 6.7 lists the arguments of the special M codes.

Table 6.7 Special M Code Argument List

	Axis Command (*1)	Radius Command (R)	Center Point Command (I,J)	M Code (*2)	G code	Feed (F)	H	L	N	O	P	Remarks
M00												
M01												
M02												
M30												
M98							○	○			○	
M99											○	
M100												
Other M codes				○	○							

○ : May be set.
Blank : Must not be set.

*1 The axis commands are X, Y, Z, U, V, W, A, B, and C.
*2 M codes indicate those other than M00, M01, M02, M30, M98, M99 and M100.

Code	M00	Stops a program run.
Function	Program stop	

6.10.1 M00 Program stop

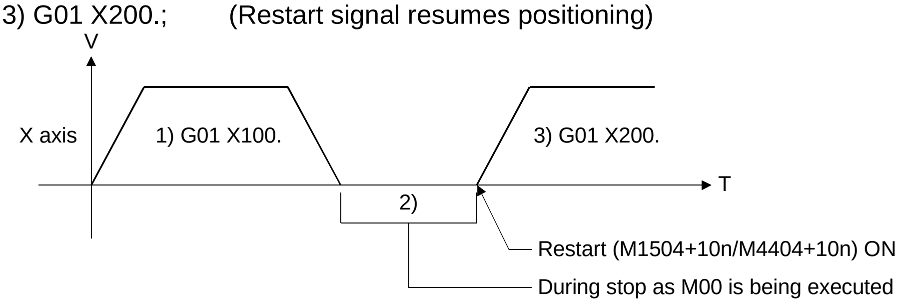
[Explanation]

- Executing this command stops the program without execution of the next block. By turning ON the restart signal (M1504+10n/M4404+10n) after a stop, execution resumes from the next block.

	Format	M00;

[Program Example]

- Program designed to make a stop during positioning operation and restart positioning.
  1) G01 X100. F10.; (Positioning)
  2) M00; (Program stop) ← Restart signal (M1504+10n/M4404+10n) ON
  3) G01 X200.; (Restart signal resumes positioning)



Code	M01	When the optional program stop is ON, executing M01 stops a program run.
Function	Optional program stop	

6.10.2 M01 Optional program stop

[Explanation]

- When the optional program stop (M1501+10n/M4401+10n) is ON, executing this command stops the program without execution of the next block.
  By turning ON the restart signal (M1504+10n/M4404+10n) after a stop, execution resumes from the next block.

- When the optional program stop (M1501+10n/M4401+10n) is OFF, the next block is executed without a program stop.

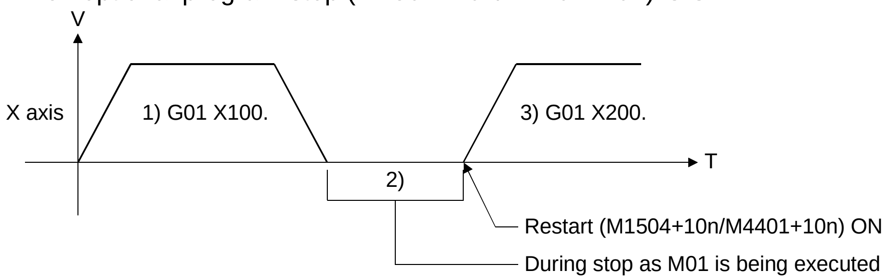
	Format	M01;

[Program Example]

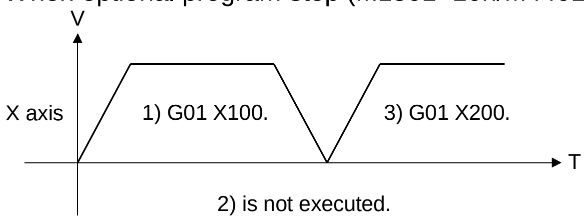
- Program which uses the optional program stop (M01).

1) G01 X100. F10.;	(Positioning)
2) M01;	(Optional program stop)
3) G01 X200.;	(Positioning)

<When optional program stop (M1501+10n/M4401+10n) is ON>



<When optional program stop (M1501+10n/M4401+10n) is OFF>



REMARK

- M01 performs the same operation as "M00" when the optional program stop (M1501+10n/M4401+10n) is ON.

Code	M02	Ends a program.
Function	Program end	

6.10.3 M02 Program end

[Explanation]

- Executing this command ends a program run.
  This command is required at the end of a program.

	Format	M02;

[Program Example]

- Program which is ended after positioning control.

```
G90;                       (Absolute value command)
G01 X100. Y200. F100.;     (Positioning)
X200. Y300.;               (Positioning)
G00 X0. Y0.;               (Positioning)
M02;                       (Program end) ..... Also be enabled by M30.
%
```

REMARK

- M02 and M30 have the same function.

Code	M30	Ends a program.
Function	Program end	

6.10.4 M30 Program end

[Explanation]

- Executing this command ends a program run.
  This command is required at the end of a program.

	Format	M30;

[Program Example]

- Program which is ended after positioning control.

G90;	(Absolute value command)
G01 X100. Y200. F100.;	(Positioning)
X200. Y300.;	(Positioning)
G00 X0. Y0.;	(Positioning)
M30;	(Program end) ..... Also be enabled by M02.
%	

REMARK

- M30 and M02 have the same function.

Code	M98, M99	Make subprogram call (M98) and subprogram end (M99).
Function	Subprogram call, subprogram end	

6.10.5 M98, M99 Subprogram call, subprogram end

[Explanation]

- A program of the same pattern can be registered as a single subprogram and called as required from the main program.

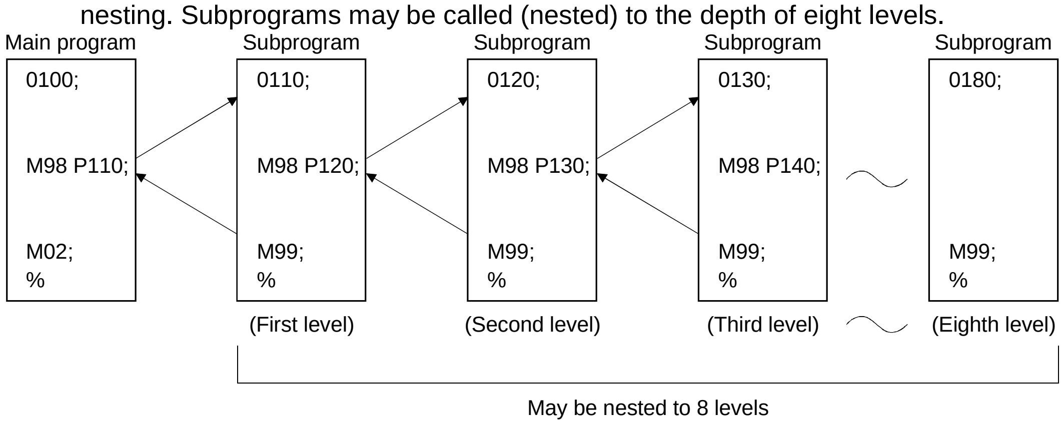
<Program call> (M98)

- Argument program number, sequence number and repeat number may be omitted. When omitted, these numbers are as follows.
  Program number : Main program
  Sequence number : First
  Repeat count : Once

[Example]

M98; Executes once from the beginning of the main program.

- A subprogram can be called from another subprogram. This is called subprogram nesting. Subprograms may be called (nested) to the depth of eight levels.



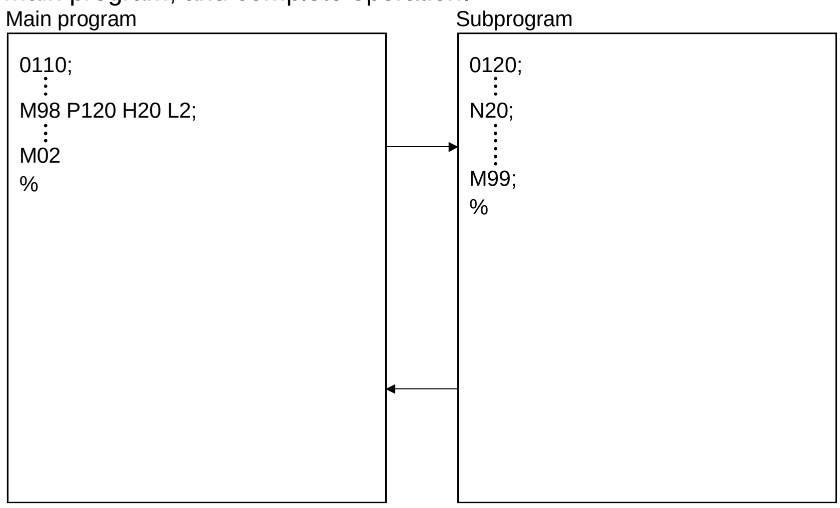
<Subprogram end> (M99)

- Returns to the block next to the call block.

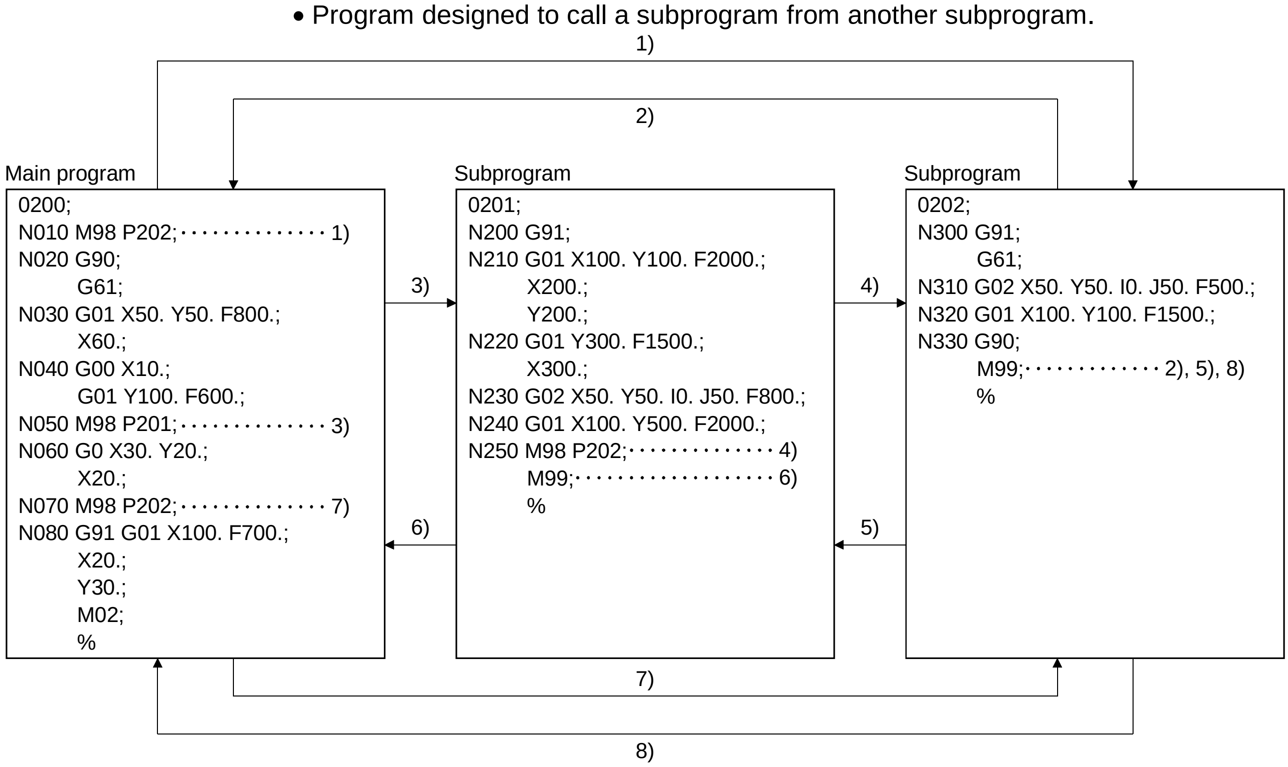
Format	M98␣P p␣H h␣L l; l: Subprogram repeat count (0 to 9999) h: Subprogram call sequence number (1 to 9999) p: Subprogram call program number (1 to 256) M99;

[Program Example]

- Program designed to run the specified subprogram twice repeatedly, return to the main program, and complete operation.



- Program designed to call a subprogram from another subprogram.



Code	M100	Does not execute preread on the G code software.
Function	Preread inhibit	

6.10.6 M100 Preread inhibit

[Explanation]

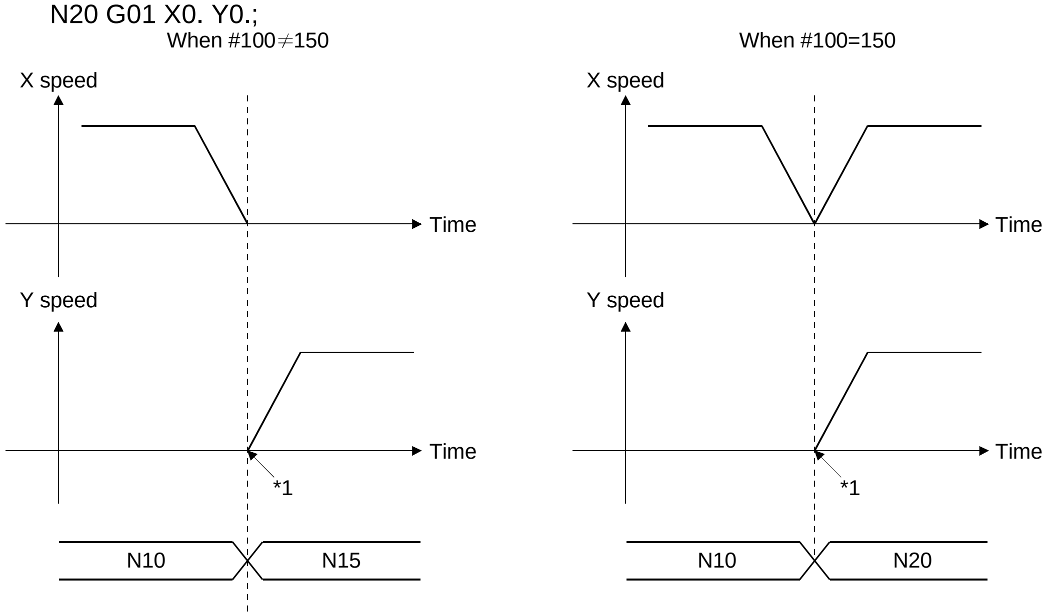
- Executing this command does not execute preread on the G code software.
  After completion of motion up to the preceding block, the next block is processed.

	Format	M100;

[Program Example]

```
N10 G01 X10. F10.;
    M100;
    IF [#100 EQ150] GOTO20;
N15 G01 Y10.;
N20 G01 X0. Y0.;
```

Since M100 exists in the next block, a change in #100 during execution of the command on this line is reflected on the IF statement below.



*1 When M100 is executed, CP does not continue from N10 to N15 or from N10 to N20 and the axis decelerates to a stop once after execution of N10.

6.11 Miscellaneous

Table 6.8 lists the arguments that may be specified in the first character.

Table 6.8 Argument List

	( )	[ ]	Operator	Logical Operator	Assignment (=)	GOTO	G	M	Remarks
#	○	○	○	○	○				——
IF	○	◎	○	○		◎			——
GOTO	○	○	○						——
/							○	○	Depends on the data after "/".
G									Refer to Section 6.8.
M	○	○	○						Refer to Section 6.10 for M00, M01, M02, M30, M98, M99 and M100.
Axis command	○	○	○						Depends on the G code in the modal 01 group.
Feed	○	○	○						Depends on the G code in the modal 01 group.
O	○	○	○						——
N	○	○	○						Regards the line number and later as the fist character.
( )	○								Handles data between "(" and ")" as a comment.
IF	○	○	○	○					
ELSE	○		○						
END	○		○						
WHILE	○	○	○	○					
DO	○		○						

○ : May be specified.
◎ : Must be specified.
Blank : Must not be specified.

Code	IF, GOTO	Controls the flow of a run program according to the condition.
Function	Program control function	

6.11.1 Program control function (IF, GOTO statement)

[Explanation]

- If the specified expression is true (1) (condition is satisfied), execution jumps to the sequence number specified in GOTO.
  If the expression is false (0), the next line is executed.

  IF [#100 EQ1] GOTO100;
    If #100 is 1, execution jumps to N100.
    If it is other than 1, the next line is executed.
  IF [#100] GOTO100;
    If #100 is 1 (true), execution jumps to N100.
    If it is 0 (false), the next line is executed.

- The following comparison instructions may be used in the expression.

Code	Meaning
EQ	Equal to (=)
NE	Not equal to (!=)
GT	Greater than (>)
LT	Less than (<)
GE	Greater than or equal to (>=)
LE	Less than or equal to (<=)

- The expression must be enclosed in "[", "]".

- The line number specified in GOTO must exist in the same program. If it does not, an error (error code: 541) occurs.

- If only GOTOn is specified, execution jumps to the specified line number unconditionally.

	Format	IF␣[expression]␣GOTOn n: Sequence number

[Program Example]

- Program designed to cause a jump to the specified line if the condition is satisfied.

```
            0201;
            N200 G91;
            N210 G01 X100. Y100. F2000.;
                 X200.;
                 Y200.;
          ┌──    IF [#100] GOTO230;          (If #100 if true, execution jumps to N230)
Jump to   │ N220 G01 Y300. F1500.;
 N230     │      X300.;
          └─→N230 G02 X50. Y50. I0. J50. F800.;
            N240 G01 X100. Y500. F2000.;
          ┌──    IF [#110 EQ 180] GOTO260;  (If #110 if 180, execution jumps to N260)
Jump to   │ N250 G00 X10.;
 N260     │      Y100.;
          └─→N260 G28 X0. Y0.;
                 M02;
                 %
```

REMARK

- Only one comparison instruction may be used in one block.

Code	IF, THEN, ELSE, END	Controls the flow of a run program according to the condition.
Function	Program control function	

6.11.2 Program control function (IF, THEN, ELSE, END statements)

[Explanation]

- If the specified expression is true (1) (condition is satisfied), the THEN statement (block group up to ELSE) is executed. If it is false (0) (condition is not satisfied), the ELSE statement (block group up to END) is executed.

  IF [#110 EQ1] THEN 1;
    If #100 is 1, the block group described here is executed.
  ELSE1;
    If #100 is not 1, the block group described here is executed.
  END1;

- When ELSE is omitted, the block group up to END is executed only if the conditional expression is true.

  IF [#100 EQ1] THEN 1;
    If #100 is 1, the block group described here is executed.
  END1;

- The multiprogramming depth is up to three levels including that of the WHILE statement.

```
IF [ ] THEN1 ;
    IF [ ] THEN2 ;
        IF [ ] THEN3 ;

        END3 ;
    END2 ;
END1 ;
```

- The GOTO statement cannot cause execution to go into or come out of the THEN and ELSE statements.

	Format	IF␣[expression]␣THENm; Block U group ELSEm; Block U group ENDm; m: IF identification number (1 to 32)

[Program Example]

```
     01;
N1   G91;
N2   G01 X100. Y100. F2000;
N3   X200.;
N4   Y200.;
N5    IF [#100 EQ0] THEN1;
N6   G01 Y300. F1500;            When #100=0, THEN1 to END1 are executed.
N7   X300.;
N8   END1;
N9   G02 X50. Y50. I0. J50. F800;
N10  G01 X100. Y500. F2000;
N11  IF [#110] THEN2;
N12  G00 X10.;                   When #110 is true, THEN2 to ELSE2 are executed.
N13  Y100.;
N14  ELSE2;
N15  G28 X0. Y0.;                When #110 is false, ELSE2 to ELSE2 are executed.
N16  END2;
N17  M02;
     %
```

Caution: Note that if the sequence number (N**) is omitted in the above program, the block number changes as indicated below.

Program	Execution Block No. (A)	Execution Block No. (B)	Execution Block No. (C)	Execution Block No. (D)
01;	0	0	0	0
G91;	1	1	1	1
G01 X100. Y100. F2000;	2	2	2	2
X200.;	3	3	3	3
Y200.;	4	4	4	4
IF [#100 EQ0] THEN1;	5	5	5	5
G01 Y300. F1500;	6	——	6	——
X300.;	7	——	7	——
END1;	8	——	8	——
G02 X50. Y50. I0. J50. F800;	9	6	9	6
G01 X100. Y500. F2000;	10	7	10	7
IF [#110] THEN2;	11	8	11	8
G00 X10.;	12	9	——	——
Y100.;	13	10	——	——
ELSE2;	14	11	——	——
G28 X0. Y0.;	——	——	12	9
END2;	——	——	13	10
M02;	15	12	14	11
%	——	——	——	——

(A) indicates that #100=0 and #110 is true. (B) indicates that #100≠0 and #110 is true.
(C) indicates that #100=0 and #110 is false. (D) indicates that #100≠0 and #110 is false.

Code	WHILE, DO	Controls the flow of a run program according to the condition.
Function	Program control function	

6.11.3 WHILE DO statement

[Explanation]

- While the [conditional expression] holds, blocks between the next block and ENDm block are executed repeatedly, and when it does not hold, execution shifts to the block next to ENDm.

- WHILE [conditional expression] DOm and ENDm are used in pairs.
  The identification number m range is 1 to 32.

- The multiprogramming depth of the WHILE statement is up to three levels.

[Example 1] The identification number m can be used any number of times as desired.

```
WHILE [   ] DO1;
  to
END1;
  to
WHILE [   ] DO5;
  to
END5;
  to
WHILE [   ] DO1;
  to
END1;
```

[Example 2] The multiprogramming depth is up to three levels.

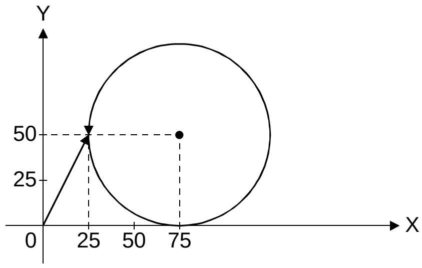
```
WHILE [   ] DO1;
  to
WHILE [   ] DO2;
  to
WHILE [   ] DO3;
  to                (Third level) (Second level) (First level)
END3;
  to
END2;
  to
END1;
```

	Format	WHILE␣[conditional expression]␣DOm m: WHILE identification number (1 to 32)

[Program Example]

- Program designed to cause a jump to the specified line if the condition is satisfied.

```
     0110;
N1   #0=0;
N2   G91 G00 X25. Y50.;
N3   WHILE [#0 LT3] DO1;<----+
N4   G03 X0. Y0. I25. J0. F100.;  |  *1
N5   #0=#0+1; ..........................*2
N6   END1;-------------------+
N7   G28 X0. Y0.;
N8   M02;
     %
```



*1: N3 to N6 are repeated while variable #0<3 holds.

*2: Every time this block is executed once, 1 is added to variable #0.
The program on the left ends after drawing a circle three times.

Caution: Note that if the sequence number (N**) is omitted in the above program, the block number changes as indicated below.

Program	Execution Block No.
0110;	0
#0=0;	1
G91 G00 X25. Y50.;	2
WHILE [#0 LT3] DO1;	3
G03 X0. Y0. I25. J0. F100.;	4
#0=#0+1;	5
END1;	——
G28 X0. Y0.;	4
M02;	5
%	——

Code	+, -, *, /, MOD, =	Perform addition (+), subtraction (-), multiplication (*), division (/), remainder (MOD) and assignment (=).
Function	Four fundamental operators, assignment operator	

6.11.4 Four fundamental operators, assignment operator (+, -, *, /, MOD, =)

[Explanation]

- Calculation of the specified operator is performed.
- The priority of operations is in order of function, multiplication type operation and addition type operation.

  #100 = #110 + #120 * SIN [#130];

  1) Function

  2) Multiplication type operation

  3) Addition type operation

- The area of operation where you want to give priority can be enclosed in [ ]. [ ] can be five levels deep including [ ] of a function. An operational expression may be described in up to 72 characters. (Up to the maximum number of characters in one block)

  #100 = SQRT [[[#110 - #120] * SIN [#130] + #140] * #150];

  First level

  Second level

  Third level

- For +, -, * and /, the operation result type is used for operation. Operation data 1, 2 are converted into the operation result type. The operation result can be the 16-, 32- or 64-bit type.

  Operation result = operation data 1 operator operation data 2;

  Operation result is stored

  Operation is performed after conversion of operation data 1, 2 into operation result type.

- For MOD, the 16- or 32-bit type is used for operation. If operation data 1, 2 are the 64-bit type, they are converted into the 32-bit type.
  The operation result can be the 16-, 32- or 64-bit type, but if the operation result is the 64-bit type, the result of operation performed with the 32-bit type is converted into the 64-bit type and the result of conversion is stored.

  Operation result = operation data 1 operator operation data 2;

  Operation result is stored
  Note that if operation result is 64-bit type, 32-bit type is converted into 64-bit type.

  Operation is performed after conversion of operation data 1, 2 into operation result type.
  Note that if operation result is 64-bit type, 32-bit type is used to perform operation.

- The following operational expressions will result in an error (560: format error).

  #10 = ##20; → Possible if #10 = # [#20];

  #10 = #20 +- #30; → Possible if #10 = #20 + [- #30];

- If there is no operation result (if operation exists in the operation result, or for conditional expression such as the IF statement), the 32-bit type is used to perform operation.

	Format	n1␣Operator␣n2 n2: Numerical value or variable Operator: Operator (+, -, *, /, MOD) n1: Numerical value or variable

[Program Example]

- Program designed to carry out positioning according to the result of the specified operation.

```
0200;
        #40L = 1000000;
        #60L = 767;
        #80L = 10000;
        #30L = [#40L + 50000] * 2;
        #50L = #60L MOD 256;
        #70L = #80L * 2;
N060 G00 X#30L Y#50L;
        X20.;
N080 G91 G01 X100. F#70L;
        X20.;
        Y30.;
M02;
%
```

Code	SIN, COS, TAN, ASIN, ACOS, ATAN	Perform operations of SIN (sine), COS (cosine), TAN (tangent), ASIN (arcsine), ACOS (arccosine) and ATAN (arctangent).
Function	Trigonometric functions	

6.11.5 Trigonometric functions (SIN, COS, TAN, ASIN, ACOS, ATAN)

[Explanation]

- The operation of the specified trigonometric function is performed.
- The operation result is a 32-bit integer (BIN value) including four decimal places.
- When the argument of the trigonometric function has no decimal point, the operation result is similarly a BIN value including four decimal places.

	Format	function␣[n]; [n]: Numerical value (can be specified indirectly) function: Trigonometric function (SIN, COS, TAN, ASIN, ACOS, ATAN)

[Program Example]

#10:L = SIN [60.];	#10:L = 8660
#16:L = SIN [600000];	#16:L = 8660
#20:L = COS [45.];	#20:L = 7071
#26:L = COS [450000];	#26:L = 7071
#30:L = TAN [30.];	#30:L = 5773
#36:L = TAN [300000];	#36:L = 5773
#40:L = ASIN [0.8660];	#40:L = 599970
#46:L = ASIN [8660];	#46:L = 599970
#50:L = ACOS [0.7071];	#50:L = 450005
#56:L = ACOS [7071];	#56:L = 450005
#60:L = ATAN [1.];	#60:L = 450000
#66:L = ATAN [10000];	#66:L = 450000

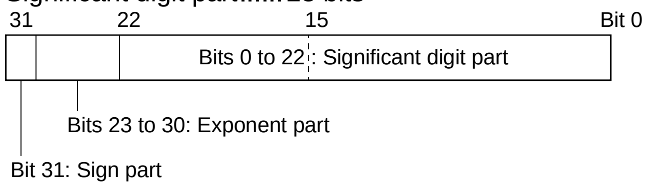
Code	INT	Converts a floating-point type real number into a 32-bit integer (BIN value) including four decimal places.
Function	Floating-point type real number processing instruction Real number to BIN value	

6.11.6 Real number to BIN value conversion (INT)

[Explanation]

- A floating-point type real number is converted into a 32-bit integer (BIN value) including four decimal places.

- A floating-point type real number is processed as single precision (32 bit) in the binary floating-point format of the IEEE Standard.
  Sign part .... 1 bit
  Exponent part .... 8 bits
  Significant digit part...... 23 bits



- The following values can be handled as floating-point type real numbers.
  $-1.0\times2^{128}<\text{value}\leq-1.0\times2^{-126}$, 0, $1.0\times2^{-126}\leq\text{value}<1.0\times2^{128}$

	Format	INT␣[n] ; [n]: Indirect designation only INT: Real number to 32-bit integer (BIN value) conversion command

[Program Example]

```
#2:L = 10000;
#4:L = FLT[#2:L];     #4:L = (461C4000)16
                      (D4,5 = (461C4000)16)
#6:L = INT[#4:L];     #6:L = 10000
```

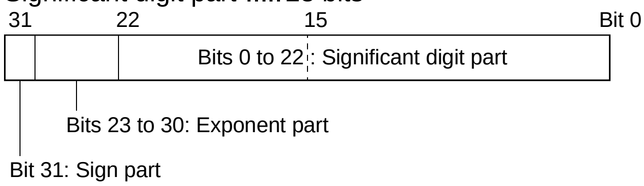
Code	FLT	Converts a 32-bit integer (BIN value) including four decimal places into a floating-point type real number.
Function	Floating-point type real number processing instruction BIN value to real number conversion	

6.11.7 BIN value to real number conversion (FLT)

[Explanation]

- A 32-bit integer (BIN value) including four decimal places is converted into a floating-point type real number.

- A floating-point type real number is processed as single precision (32 bit) in the binary floating-point format of the IEEE Standard.
  Sign part .... 1 bit
  Exponent part ....8 bits
  Significant digit part .....23 bits



- The following values can be handled as floating-point type real numbers.
  $-1.0\times2^{128}<\text{value}\leq-1.0\times2^{-126}$, 0, $1.0\times2^{-126}\leq\text{value}<1.0\times2^{128}$

	Format	FLT␣[n] ;
		FLT: 32-bit integer (BIN value) to real number conversion command [n]: Indirect designation only

[Program Example]

```
#2:L = 10000;
#4:L = FLT[#2];    #4:L = (461C4000)16
                   (D4,5 = (461C4000) 16)
#6:L = INT[#4];    #6:L = 10000
```

Code	SQRT, ABS, BIN, BCD, LN, EXP, RND, FIX,	Perform operations of SQRT (square root), ABS (absolute value), BIN (BCD to BINARY conversion), BCD (BINARY to BCD conversion), LN (natural logarithm), EXP (base e exponent), RND (round off), FIX (round down) and FUP (round up).
Function	Functions	

6.11.8 Functions (SQRT, ABS, BIN, BCD, LN, EXP, RND, FIX, FUP)

[Explanation]

- Operation of the specified function is performed.

- For the operation result, refer to Items (5), (6), (7) in Section 6.3.3.

	Format	function␣[n]; [n]: Numerical value (can be specified indirectly) function: Trigonometric Function (SQRT, ABS, BIN, BCD, LN, EXP, RND, FIX, FUP)

[Program Example]

#10L = SQRT [100]   10 enters [D11, D10].
#20L = ABS [-25]   25 enters [D21, D20].
#30L = BIN [100]   64 enters [D31, D30].
#40L = BCD [100]   256 enters [D41, D40].
#50L = LN [1000000] 13 enters [D51, D50].
#60L = EXP [20]   485165195 enters [D61, D60].
#70F = RND [14/3]   5 enters [D73, D72, D71, D70] (64-bit floating-point type).
#80F = FIX [14/3]   4 enters [D83, D82, D81, D80] (64-bit floating-point type).
#90F = FUP [14/3]   5 enters [D93, D92, D91, D90] (64-bit floating-point type).
#170F = RND [-14/3] -5 enters [D173, D172, D171, D170] (64-bit floating-point type).
#180F = FIX [-14/3]   -5 enters [D183, D182, D181, D180] (64-bit floating-point type).
#190F = FUP [-14/3]   -4 enters [D193, D192, D191, D190] (64-bit floating-point type).

Code	AND, OR, XOR, NOT, <<, >>	Perform logical product (AND), logical add (OR), exclusive logical add (XOR), logical NOT (NOT) and shift operations (<<, >>).
Function	Logical operators	

6.11.9 Logical operators (AND, OR, XOR, NOT, <<, >>)

[Explanation]

- Operation of the specified logical operator is performed.
- Only the integer types (16-bit type, 32-bit type) may be used to perform logical operation. Logical operation including the 64-bit floating-point type cannot be performed. (Error 560: Format error)
  The operation result can be 16- or 32-bit type, but it is converted into the operation result type for operation.
- The area of operation where you want to give priority can be enclosed in [ ]. [ ] can be five levels deep including [ ] of a function. An operational expression may be described in up to 72 characters. (Up to the maximum number of characters in one block)

  <For AND, OR, XOR, <<, >> >

  Operation result = operation data 1 operator operation data 2;

  Operation result is stored

  Operation is performed after conversion of operation data 1, 2 into operation result type.
  Note that operation including 64-bit floating-point type cannot be performed.

  <For NOT>

  Operation result = NOT [operation data 1];

  Each bit of operation data 1 is inverted and result of inversion is stored into operation result.
- The logical operators can be used with the conditional expressions of the IF and WHILE statements.

  IF[[ON #M1000] AND [OFF #M1100]] GOTO1;

  If M1000 is ON and M1100 is OFF, the N1 line is executed.

  IF[[# 100 AND #200] EQ #300] GOTO2;

  If the result of ANDing #100 and #200 contents is equal to #300, the N2 line is executed.

Format	<For AND, OR, XOR, <<, >> > **n1␣Operator␣n2;** n2: Numerical value or variable Operator: Operator (AND, OR, XOR, <<, >>) n1: Numerical value or variable <For NOT> **NOT␣[n1] ;** n1: Numerical value or variable

[Program Example]

Operator	Program Example	Operation
AND	#10L = 100; #20L = #10L AND 15;	#10L = 01100100 15 = 00001111 #20L = 00000100 = 4
OR	#10L = 100; #20L = #10L OR 14;	#10L = 01100100 14 = 00001110 #20L = 01100100 = 110
XOR	#10L = 100; #20L = #10L XOR 14;	#10L = 01100100 14 = 00001110 #20L = 01101010 = 106
NOT	#10L = 90; #20L = NOT [#10L];	#10L = 01011010 #20L = 10100101 = 165
<<	#10L = 20; #20L = #10L << 2;	#10L = 00010100 #20L = 01010000 = 80
>>	#10L = 80; #20L = #10L >> 2;	#10L = 01010000 #20L = 00010100 = 20

Code	WAITON, WAITOFF	Executes the next move block when the ON/OFF condition of the specified device holds.
Function	Move block wait functions	

6.11.10 Move block wait functions (WAITON, WAITOFF)

[Explanation]

- Execution waits the next move block to be executed until the ON/OFF condition of the specified device holds. Note that the operation block is executed.

- The response time of WAITON/WAITOFF is the operation cycle time (approx. 3.5msec for 8 or less axes).

- It takes about 7 to 64msec from when a program is started until the program is actually run. Therefore, WAITON/WAITOFF can be used to start a motion program fast. By setting a wait for a shift to the next block with WAITON or WAITOFF after a program start has been made by the SVST instruction in a sequence program, prereading of the next block has been completed, and therefore, the next block can be executed at high speed (approx. 3.5msec for 8 or less axes) after the device condition has held, improving the variation or delay in a program start.
  [Example]

```
WAITON #X10;  ----------> When X10 turns ON, N1 block is executed.
N1 G01 X100. Y200. F1000.;
WAITOFF #X11; ----------> When X11 turns OFF, N2 block is executed.
N2 G01 X200. Y300. F500
 :
 :
M02;
%
```

- The grammar is indicated below.
  <WAITON statement>: WAITON #<device>
  [Example] WAITON #X10;
  <WAITOFF statement>: WAITOFF #<device>
  [Example] WAITOFF #X11;

- WAITON/WAITOFF cannot be used with the home position return instruction.

	Format	WAITON␣#Xx ; Device (X, Y, M, TC, TT, CC, CT, B, F) WAITOFF␣#Xx ; Device (X, Y, M, TC, TT, CC, CT, B, F)

[Program Example]

Program which executes the next block when a condition holds.

```
00001  WAITON #X10;
00002  N1 G01 X100. Y200. F1000.;
00003  WAITOFF #X11;
00004  N2 #10 = 5
00005  G00 X0. Y-10.;
00006  WAITON #X12;
00007  GOTO 10;
  :
00015  N10 G00 X0. Y0.;
  :
00020  #0 = 5;
00021  WAITOFF #XFF;
00022  IF [#0 EQ 5] GOTO 20;
00023  N15 G01 X200. Y200. F2000.;
  :
00027  N20 G01 X100. Y100. F2000.;
00028  M02;
00029  %
```

The above program is run as described below.

1. Line 1 When device X10 turns ON, line 2 is executed.
2. Line 3 When device X11 turns OFF, line 5 is executed.
   (Line 4 is being executed.)
3. Line 6 When device X12 turns ON, N10 is executed.
4. Line 21 When device XFF turns OFF, #0=5 to line 27 are executed. Because of preread processing, N15 is not executed and execution jumps to N20 if the #0(D0) value is changed from sequence program while execution waits for XFF to turn from ON to OFF in the WAITOFF statement.

Code	PB	Uses the parameter block of the specified number.
Function	Parameter block change	

6.11.11 Parameter block change (PB)

[Explanation]

- The numerical value following PB is used as a parameter block number.
- The parameter block value may also be specified indirectly by a variable, D or W (2-word data).
- Any of 1 to 16 may be specified as the parameter block value.
  Specifying any other value than the above will result in a "format error". (Error code 560)
- Once given, the parameter block change command is valid until the parameter block change command is given again.
  However, when a torque limit value change (TL) is made, the specified torque limit value is used.
- When a parameter block change (PB) is made during a torque limit value change (TL), the torque limit value in the new parameter block is used.
- When a parameter block change is made during a CP motion, the axis decelerates to a stop once and the next CP motion is executed.

```
G01 X100. F500. ; ←──── Deceleration to a stop at X100.
PB3 ; ←──────────────── After that, parameter block 3 is used.
G01 X200. ;
```

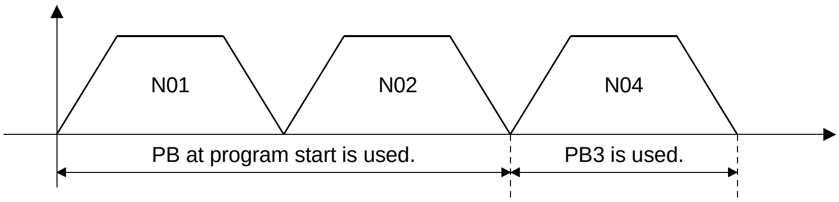
- At a home position return (G28), the parameter block at a program start is used.
- The parameter block change command cannot be described in the same block as another command.
- If a cancel start is made during a parameter block change, the start program uses the parameter block for execution of the start program.
- A parameter block change (PB) is valid for the next travel.

	Format	PB␣pb; pb: Parameter block number PB: Parameter block change command

[Program Example]

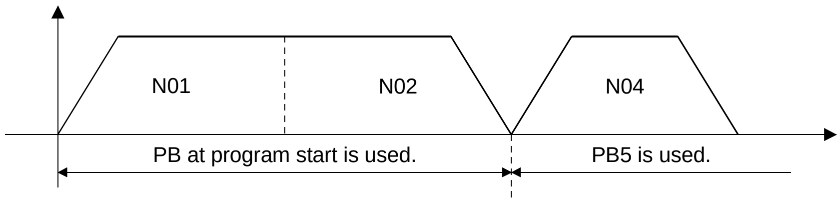
1) When a parameter block change is made during PTP

```
N01  G00 X0.;       Uses the parameter block at a program start.
N02  G00 X100.;
N03  PB3;           Changes to parameter block 3.
N04  G00 X300.;
```



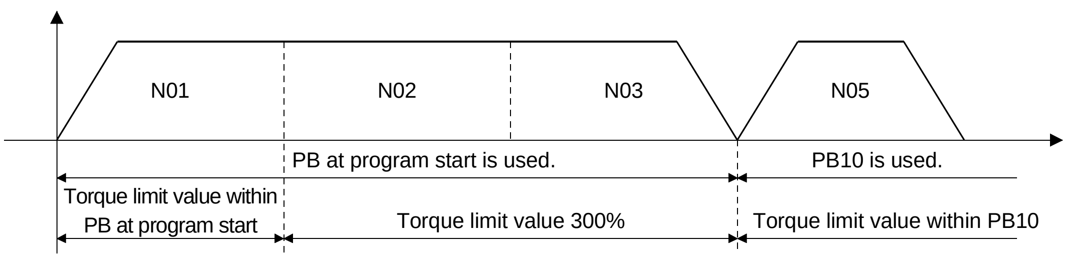
2) When a parameter block change is made during CP

```
N01 G01 X0. F200.;  Uses the parameter block at a program start.
N02 G01 X100.;
N03 PB5;            Changes to parameter block 5.
N04 G01 X200.;
```



3) When torque limit value is being changed

```
N01  G01 X0. F200.
N02  G01 X100. TL300;
N03  G01 X200.;
N04  PB10;
N05  G01 X300.;
```



Code	TL	Changes the torque limit value to the specified value.
Function	Torque limit value change	

6.11.12 Torque limit value change (TL)

[Explanation]

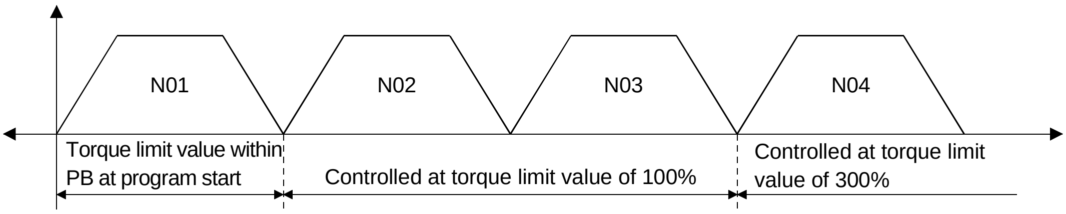
- The numerical value following TL is commanded as a torque limit value. The torque limit value may also be specified indirectly by a variable, D or W (2-word data).
  (After the TL code, the torque limit value in the parameter block is not used.)
- Any of 1 to 500(%) may be specified as the torque limit value.
  Specifying any other value than the above will result in a "format error". (Error code 560)
- Once given, the TL command is valid until the TL command is given again or the parameter block or CHGT command is given. However, at a program start, the torque limit value in the specified parameter block or the specified torque limit value is used.
- At a home position return (G28), the torque limit value in the parameter block at a program start is used.
- If a cancel start is made during a torque limit value change, the start program uses the torque limit value in the parameter block for execution of the start program.
- If a torque limit value change (TL) is specified in G32 (skip) and the skip device is already ON before execution of G32, the torque limit value change command (TL) is also skipped and the torque limit value specified previously remains unchanged.
- The torque limit value change (TL) is valid for all axes specified in SVST. However, if the torque limit value specified in the torque limit value change (TL) for the axis whose torque limit value is specified in the CHGT command is greater than the torque limit value in the CHGT command, torque is clamped at the torque limit value of the CHGT command.
- The axis operating under the high-speed oscillation (G25) is not made valid. That axis is made valid from the move command or M code after the high-speed oscillation stop (G26) is executed.
- If specified in a move block, the torque limit value (TL) is made valid from that motion. When the torque limit value is independent (no block motion specified), it is made valid for the next motion.

	Format	TL␣t; t: Torque limit value TL: Torque limit value change command

[Program Example]

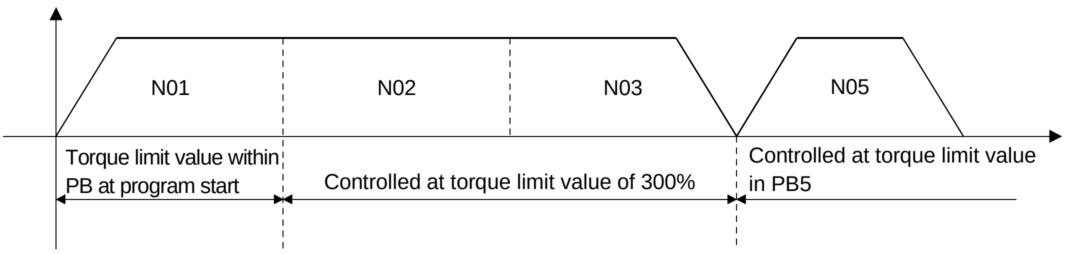
1) When torque limit value change is made

```
N01  G00 X0.;              ← Controls at the torque limit value in the parameter block at a program start.
N02  G00 X100.  TL100;
N03  G00 X200.;            ← Controls at the torque limit value of 100%.
N04  G00 X300.  TL300;     ← Controls at the torque limit value of 300%.
```



2) When parameter block change is made

```
N01  G01 X0.  F200. ;      ← Controls at the torque limit value in the parameter block at a program start.
N02  G01 X100.  TL200;
N03  G01 X200.;            ← Controls at the torque limit value of 200%.
N04  PB5;                  ← Changes to parameter block 5.
N05  G01 X300. ;           ← Controls at the torque limit value in parameter block 5.
```



Code	SET, RST	Turns the specified device ON/OFF.
Function	Bit device set, reset functions	

6.11.13 Bit device set, reset functions (SET, RST)

[Explanation]

- The specified device can be turned ON/OFF from the G code program.
- Refer to Section 6.6.2 (6) for the usable device ranges.

	Format	SET␣#Yy; #Yy: ON device (Y, M) SET: Device ON command RST␣#Yy; #Yy: OFF device (Y, M) RST: Device OFF command

[Program Example]

1) SET #M0; Turns ON device M0.
2) RST #M0; Turns OFF device M0.
3) SET#Y10; Turns ON device Y10.

Code	ON, OFF	By describing this command in the conditional expression of IF or WHILE, branches processing according to the ON/OFF status of the specified bit device.
Function	Bit device conditional branch	

6.11.14 Conditional branch using bit device (ON, OFF)

[Explanation]

- The ON/OFF status of the specified bit device is judged by the ON/OFF command to see if it is true (1) or false (0).
  By using this command in the conditional expression of IF or WHILE, a conditional branch can be made with a bit device.
  When used with a logical operator, this command enables a conditional branch with multiple bit devices.

- [ ] of the conditional expression can be five levels deep including [ ] of a function.
  An operational expression may be described in up to 72 characters in all. (Up to the maximum number of characters in one block)

  <When "ON" is specified>

  IF [ON #M100] GOTO1;

  When M100 is ON, the result is true (1) and a branch to N01 is taken.
  When M100 is OFF, the result is false (0) and the next block is executed.

  <When "OFF" is specified>

  IF [OFF #M100] GOTO1;

  When M100 is ON, the result is false (0) and the next block is executed.
  When M100 is OFF, the result is true (1) and a branch to N01 is taken.

  <When used with logical operator>

  IF [[ON #M100] AND [ON #M110]] GOTO1;

  When M100 is ON and M110 is ON, a branch to N01 is taken.
  If either of them is OFF, the next line is executed.

- The device that may be specified after the ON/OFF command is the bit device only.
  If a word device is specified, a "format error" (error code: 560) occurs.

- The bit devices usable in the ON/OFF command are X, Y, M, TC, TT, CC, CT, B and F.

- The ON/OFF command is available for the conditional expressions of the program control functions (IF GOTO, IF THEN, WHILE).

	Format	IF␣[ON␣#M100]␣GOTO1; ON/OFF device (X, Y, M, TC, TT, CC, CT, B, F) — #M100 ON/OFF command (describe OFF for OFF) — ON *Conditional expression of IF THEN or WHILE can also be described similarly.

[Program Example]

1) When M100 is ON, a branch to line N03 is taken.

```
N01  IF [ON #M100] GOTO3;  ←  Branches to line N03 if M100 is ON.
N02  G01 X100. F200.;         Executes the next line (N02) if M100 is OFF.
N03  G00 X0.;
```

2) Execution starts from the next line (THEN1 and later) if M100 is ON, or from ELSE1 if it is OFF.

```
N01  IF [ON #M200] THEN1;
N02  G01 X100. F200.;  ←  Executed when M200 is ON.
N03  ELSE1;
N04  G00 X200.;  ←  Executed when M200 is OFF.
N05  END1;
```

3) While M300 is OFF, the blocks within WHILE (N02, N03, N04) are executed repeatedly.

```
N01  WHILE [OFF #M300] DO2;  ←  Executes blocks within WHILE while M300 is OFF.
N02  G91 G01 X10. F100.;
N03  #10 = #10 + 1;
N04  END2;
N05  G90 G00 X0.;  ←  Executed when M300 turns ON.
```

7. AUXILIARY AND APPLIED FUNCTIONS

This section describes the auxiliary and applied functions available for positioning control by the servo system CPU.

7.1 Limit Switch Output Function

The limit switch output function allows the A1SY42 output module or AY42 output module to output ON/OFF signals corresponding to the positioning address set for each axis.

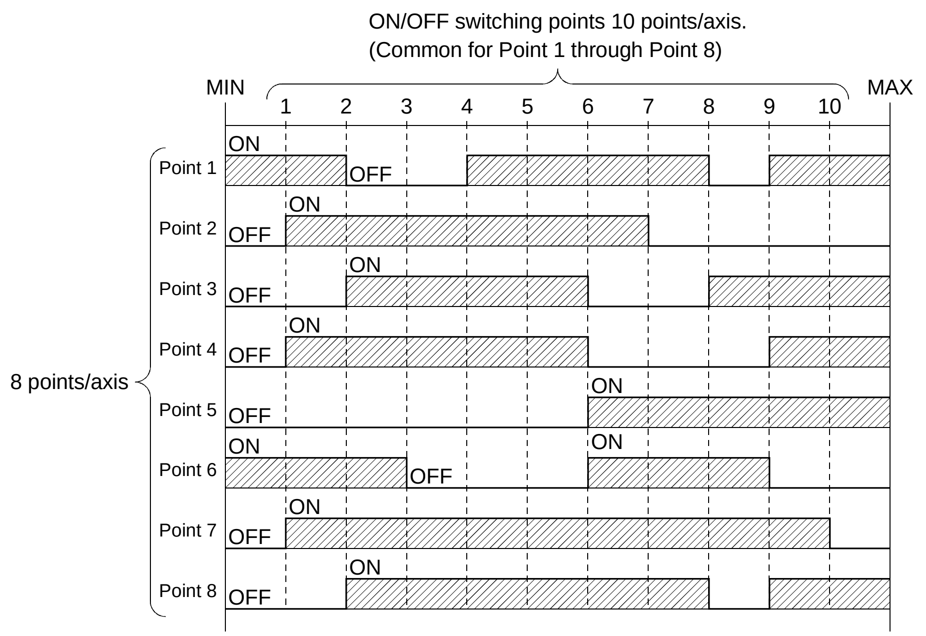
7.1.1 Limit switch output data

Item	Settings	Units	Initial Value	Remarks
ON/OFF point setting	• –2147483648 to 2147483647 (× 10^{-4}mm, × 10^{-5}inch) • 0 to 35999999 (× 10^{-5}degree)	×10^{-4}mm ×10^{-5}inch ×10^{-5}degree	0	• Up to 10 points can be set for each axis.

7.1.2 Limit switch output function

[Control Details]

(1) The limit switch function outputs the ON/OFF pattern from the A1SY42/ AY42 at the set addresses.
Before running the limit switch output function, the ON/OFF point addresses and the ON/OFF pattern must be set from a peripheral device.
(Settings cannot be made by the sequence program.)
The number of limit switch outputs per axis and the ON/OFF points are as follows:
(a) Number of limit switch output points .... 8 points/axis, total 64 points
(b) ON/OFF points.. 10 points/axis
Set an address in the stroke limit range for each point.



(2) Limit Switch Enable/Disable Setting
The following devices can be used to enable or disable the limit switch output from each axis or each point.

Table 7.1 Limit Switch Enable/Disable Settings

Set Data/Device	Setting Unit	Processing	Set Data Valid Timing
Limit switch output used/not used setting in the fixed parameters.	Axis	Used Set ON/OFF pattern can be output for the appropriate axis. Not Used All outputs OFF for the appropriate axis.	(1) Leading edge of PC ready (M2000) (2) When test mode is started
Limit switch output enable signal (M1806 + 20n/M3206 + 20n)	Axis	ON ON/OFF pattern is output for the appropriate axis based on the set ON/OFF pattern and the limit switch output disable setting registers (D1008 and D1009). OFF All outputs OFF for the appropriate axis.	Limit switch output used/not used setting in the fixed parameters is set to "used."
Limit switch output disable setting registers (D1008 and D1009/D760 to D775)	Point	Disable bit (1) Outputs corresponding to disable bits set to "1" are OFF. Enable bit (0) Outputs corresponding to enable bits set to "0" output an ON/OFF pattern based on the set ON/OFF pattern.	While M1806 + 20n/M3206+20n is ON.

REMARK

The data in Table 7.1 is also valid during the test mode set by a peripheral device.

(3) Cautions
(a) The limit switch output is based on the "feed present value" for each axis after PC ready (M2000) turns ON and the PCPU ready flag (M9074) is ON. All points turn OFF when the PCPU ready flag (M9074) turns OFF.
(b) While the PCPU ready flag (M9074) is ON and the feed present value is outside the set stroke limits, the limit switch output is based on M1806 + 20n/M3206+20n.
Consequently, the user should apply an interlock to ensure that the sequence program turns M1806 + 20n/M3206+20n ON inside the stroke limit range only.

7.2 Backlash Compensation Function

The backlash compensation function compensates for the backlash amount in the mechanical system. When the backlash compensation amount is set, extra pulses equivalent to the backlash compensation amount are output after a change in travel direction resulting from positioning control, JOG operation, or manual pulse generator operation.

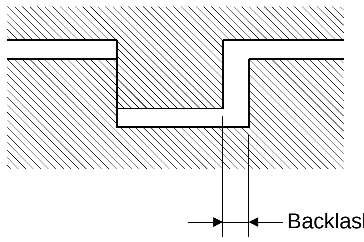


Figure 7.1 Backlash Compensation Amount

(1) Setting the backlash compensation amount
The backlash compensation amount is one of the fixed parameters, and is set for each axis using a peripheral device.
The setting range differs according to whether mm, inch, or degree, units are used, as shown below.

(a) Millimeter units

- 0 to 6.5535
- $0 \leq \frac{\text{(Backlash compensation amount)}}{\text{(Travel value per pulse)}} \leq 65535\text{(PLS)}$

(Decimal fraction rounded down.)

(b) Inch or Degree Units

- 0 to 0.65535
- $0 \leq \frac{\text{(Backlash compensation amount)}}{\text{(Travel value per pulse)}} \leq 65535\text{(PLS)}$

(Decimal fraction rounded down.)

(2) Backlash compensation processing
The details of backlash compensation processing are shown in the table 7.2.

Table 7.2 Details of Backlash Compensation Processing

Condition	Processing
First motion after power on	• No backlash compensation if travel direction = home position return direction. • Backlash compensation if travel direction ≠ home position return direction.
JOG operation start	• Minimum backlash amount on first JOG operation after travel direction change.
Positioning start	• Backlash compensation if travel direction changed.
Manual pulse generator operation	• If travel direction changed.
Home position return start	• Backlash compensation amount is valid after home position return is started.
Absolute position system	• Status stored at power off and applied to absolute position system.

POINTS

(1) The feed pulses equivalent to the backlash compensation amount are not added to the feed present value.
(2) Home position return is required after the backlash compensation amount is changed.
The original backlash compensation amount is retained until home position return is carried out.

7.3 Torque Limit Function

The torque limit function controls the torque generated by the servomotor within the set range.
The torque is controlled to the set torque limit value if the torque required during positioning control exceeds the set limit value.

(1) Torque limit value set range
Set the torque limit value between 1% and 500% of the rated torque.

7.3.1 Torque limit value changing function

At a program start or jog start, the torque limit value can be changed from the motion program or sequence program.

(1) At a program start or for jog operation, the torque limit value is changed to the value in the specified parameter block.

(2) From the motion program, the TL or PB instruction is used to change the torque limit value.
When the PB instruction is used, the torque limit value is changed to the one in the specified parameter block.

(3) From the sequence program, the CHGT instruction (refer to Section 5.6) is used to change.

[Control Details]

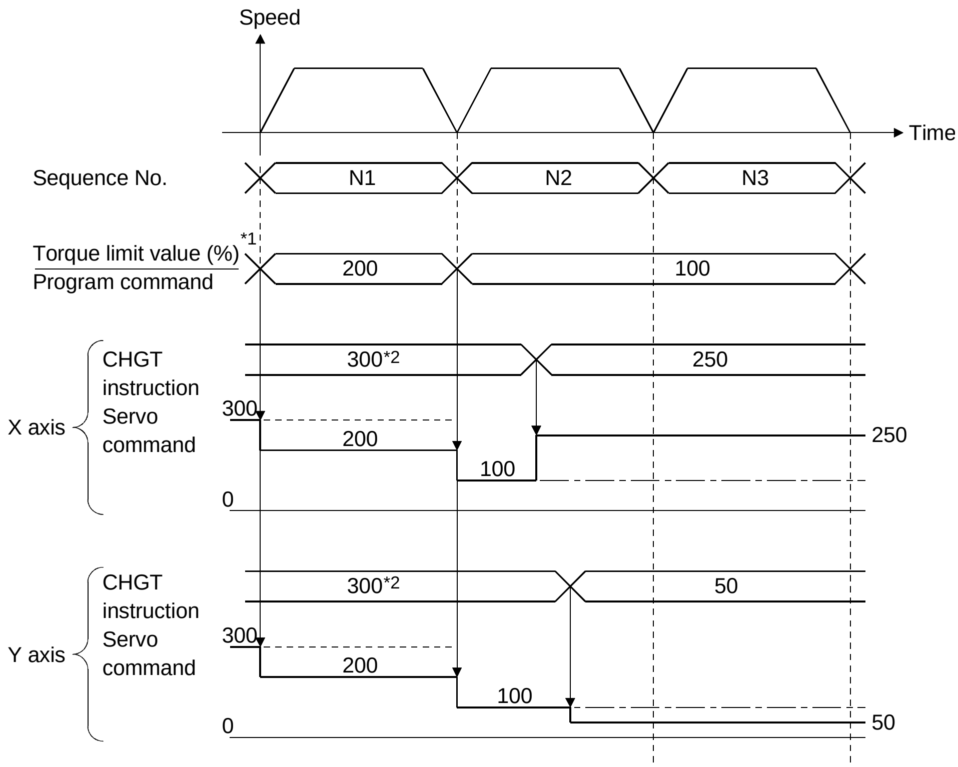
(1) The torque limit value at a motion program start or jog start is changed to the value specified in the parameter block.

(2) When the TL or PB instruction is used to change the torque limit value, the new value is valid until the next TL or PB instruction is executed. However, it is clamped at the torque limit value of the CHGT instruction.

[Program Example]

- It is supposed that before a program start, the torque limit value has been set to 300% for each axis in the CHGT instruction.
- The program is run with the torque limit value of the parameter block set to 200%.
- After execution of N1, the torque limit value is changed to 100% by the TL instruction.
- During execution of N2, the torque limit values of the X and Y axes are changed to 250% and 50%, respectively, by the CHGT instruction.

```
010;
G90;
N1 G00 X100. Y100.;
TL100;
N2 G00 X200. Y200.;
N3 G00 X300. Y300.;
M02;
%
```



*1: Indicates the torque limit value changes from the program and CHGT and the resultant command to the servo in %.

(1) The program command indicates a change of the torque limit value by the TL or PB instruction at a SVST start. The torque limit value under the program command is given to all the operating axes.

(2) Torque limit value changed by the CHGT instruction. Given to the corresponding axes.

(3) The servo command indicates the torque limit value given actually to the servo amplifier.

*2: When the CHGT instruction is not executed after power-on, the torque limit value is 300%.

Explanation

1) The torque limit value given at a program start is the lower value of the torque limit value of the parameter block specified in the SVST instruction and the value in the preceding CHGT instruction. In this case, the value is 200% in each axis.

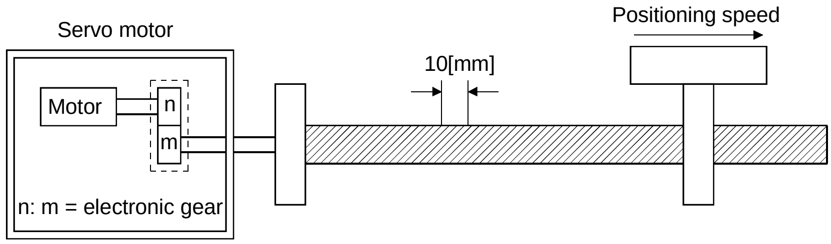
2) The torque limit value of the TL instruction at N2 execution is 100% in each axis.

3) During N1 execution, the torque limit value is changed by the CHGT instruction to 250% in the X axis and to 50% in the Y axis.

7.4 Electronic Gear Function

The electronic gear function changes the travel value per pulse.
The electronic gear is set by setting the travel value per pulse (see Section 4.2.1).
Using the electronic gear function allows positioning control without the need to select the encoder to match the mechanical system.

[Example]



Pulses per motor revolution.................10000 [PLS]
Travel value per motor revolution ....10 [mm]

(1) Electronic gear 1:1 (electronic gear setting = 1)

$$\text{Travel value per pulse} = \frac{\text{Travel value per motor revolution}}{\text{Pulses per motor revolution}} = \frac{10\ [\text{mm}]}{10000\ [\text{PLS}]}$$

$$= 0.001\ [\text{mm/PLS}]$$

Positioning control is executed at the commanded speed.

(2) Electronic gear 2:1 (electronic gear setting = 0.5)

$$\text{Travel value per pulse} = \frac{\text{Travel value per motor revolution}}{\text{Pulses per motor revolution}} = \frac{5\ [\text{mm}]}{10000\ [\text{PLS}]}$$

$$= 0.0005\ [\text{mm/PLS}]$$

Positioning control is executed faster than the commanded speed.

(3) Electronic gear 1:2 (electronic gear setting = 2)

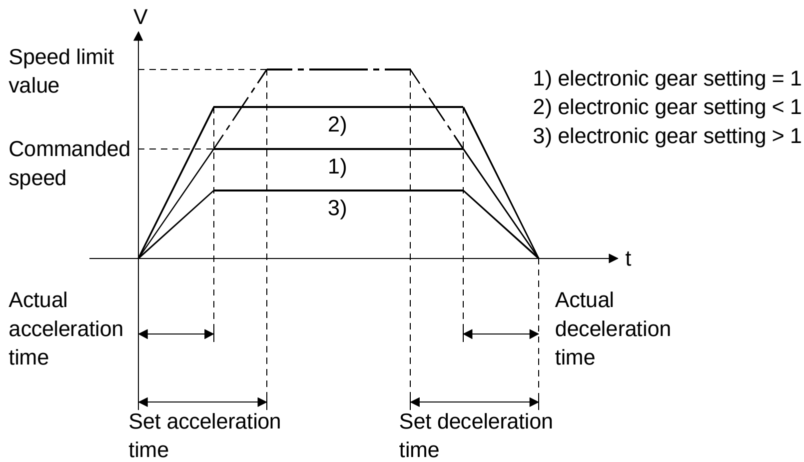
$$\text{Travel value per pulse} = \frac{\text{Travel value per motor revolution}}{\text{Pulses per motor revolution}} = \frac{20\ [\text{mm}]}{10000\ [\text{PLS}]}$$

$$= 0.002\ [\text{mm/PLS}]$$

Positioning control is executed slower than the commanded speed.

The relationship between the commanded speed (positioning speed set in the servo program) and actual speed (actual positioning speed) is shown below for different electronic gear settings.

- if electronic gear setting = 1, commanded speed = actual speed
- if electronic gear setting < 1, commanded speed < actual speed
- if electronic gear setting > 1, commanded speed > actual speed



The speed limit value, acceleration time and deceleration time are data from the designated parameter block.

Figure 7.2 Relationship Between Commanded Speed and Actual Speed

7.5 Absolute Positioning System

The absolute positioning system can be used for positioning control when using an absolute-position-compatible servomotor and MR-[ ]-B.
Home position return is not necessary using the absolute positioning system because after the machine position is initially established at system startup, the absolute position is sensed each time the power is turned on.
The machine position is established using a home position return initiated from the sequence program or a peripheral device.

(1) Absolute position system startup procedure
The system startup procedure is shown below.

Absolute position system startup

↓

For MR-□-B
- Connect CPU to absolute-position-compateible MR-□-B
- Connect FLS, RLS, STOP etc. wiring to A172SENC/A171SENC.

↓

Adjust Machine Position
Adjust machine to home position.

↓

Turn On Power
Turn on the servo amplifier and servo system CPU power.

↓

Set Positioning Parameters
Set the following positioning parameters:
- System setting*
- Fixed parameters*
- Servo parameters*
- Home position return data*
- Parameter block
- Work coordinate data*

… • Parametersf marked* must be set. (Do not change them after they are set.)
• Default values are used if the parameter block is not set.

↓

Sense Absolute Position
Make a sense absolute position request from the sequence program or a peripheral device.

… • Turn ON M2000 in a sequence program.
• Select test mode with a peripheral device.

↓

Adjust the Machine Position
Make any required adjustments to the machine position using the manual pulse generator or JOG operation.

… • Enable manual pulse generator operation from the sequence program or a peripheral device.
• JOG operation is possible from the sequence program or a peripheral device.

↓

Establish Absolute Position
Establish the absolute position using home position return by data set method.

… • The methods of home position return are two types indicated below.
• DSFLP/CHGA instructions of a sequence program.
• Test mode in a peripheral device.

↓

End

(2) In the absolute positioning system, the absolute position may be lost under the following conditions:
Re-establish the absolute position using home position return or by aligning the machine position and using present value change.
(a) After removing or replacing the battery unit.
(b) On occurrence of a servo battery error (detected at servo amplifier power on).
(c) After the mechanical system is disturbed by a shock.

(3) Power OFF Allowed Traveling Points can be monitored in the system setting mode of a peripheral device, and the present value history can be monitored in the monitor mode.
(For details on monitoring Power OFF Allowed Traveling Points and the present value history, refer to the operating manual for the peripheral device being used.)
(a) Present value history monitor
1) Month/day/hour/minute
The time when a home position return is completed or the servo amplifier power is turned ON or OFF is indicated.
In order to display the time correctly, it is necessary to first set the clock data at the programmable controller side, then switch ON M9028 (clock data read request) from the sequence program.
2) Encoder present value
When using MR-H-B (version BCD-B13W000-B2 or later) or MR-J2-B (version BCD-B20W200-A1 or later), the multiple revolution data and within-one-revolution data read from the encoder is displayed.
Note: For the encoder present value in the home position data area, the encoder present value when the motor is within the in-position range after completion of a home position return is displayed (not the encoder value at the home position).
3) Servo command value
The command value issued to the servo amplifier is displayed.
4) Monitor present value
The present value controlled within the servo system CPU is displayed.
Note: A value close to the feed present value is displayed, but, since the monitor present value and feed present value are different data, the display of different values does not indicate an error.
5) Alarms
When an error involving resetting of the present value occurs while the servo amplifier power is ON, an error code is displayed. For details of the error, refer to the error contents area (related error list) at the bottom of the screen.

⚠ CAUTION
⚠ After removing or replacing the battery unit, correctly install the new unit and establish the absolute position. ⚠ After a servo battery error occurs, eliminate the cause of the error and ensure operation is safe before establishing the absolute position. ⚠ After the mechanical system is disturbed by a shock, make the necessary checks and repairs, and ensure operation is safe before establishing the absolute position.

POINTS

(1) The address setting range in the absolute position system is -2147483648 to 2147483647. It is not possible to restore position commands that exceed this limit, or present values after a power failure.
When performing an infinite feed operation, solve this problem by setting the units to degrees.

(2) If the present value address is changed by the coordinate system setting instruction (G92), the restored data of the present value after a power failure is the value based on the status prior to execution of the coordinate system setting instruction.

(3) When home position return has not been completed, restoration of the present value after a power failure is not done properly.

7.6 Home Position Return

(1) Make a home position return when the machine origin must be checked, e.g. at power-on.

(2) The following three methods are available for a home position return.
- Near-zero point dog type } Used in other than an absolute position system.
- Count type }
- Data setting type .... Recommended for use in an absolute position system.

(3) Before starting a home position return, the home position return data (refer to Section 4.4) must be set to each axis.

7.6.1 Near-zero point dog type home position return

[Control Details]

(1) Near-zero point dog type
The near-zero point dog type is a method in which the home position is a zero point after the near-zero point dog has turned from ON to OFF.

(2) Near-zero point dog type home position return
The operation of the near-zero point dog type home position return is shown in Fig. 7.3.

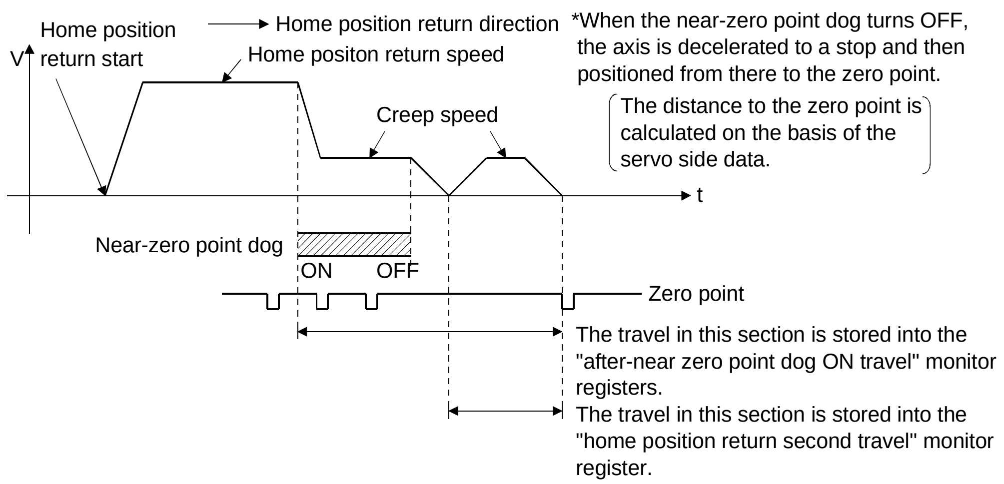


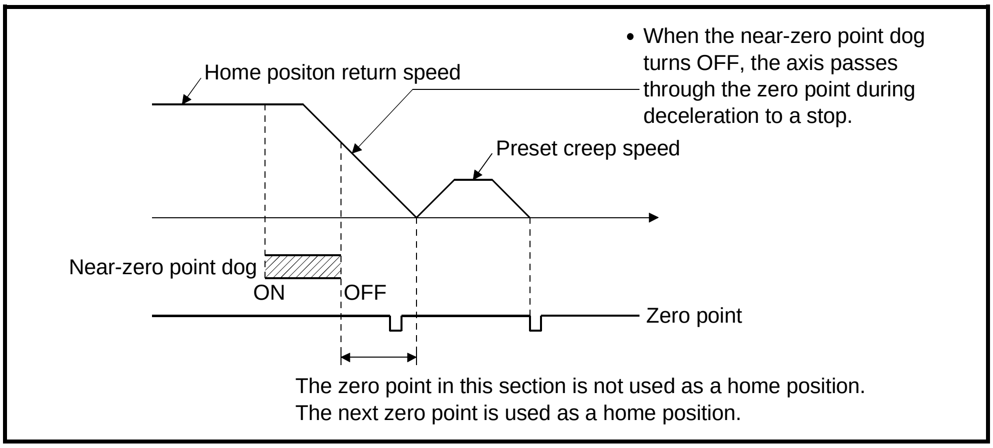
Fig. 7.3 Near-Zero Point Dog Type Home Position Return Operation

(3) Execution of home position return
Execute a home position return using the DSFLP/CHGA instruction in Section 7.6.4.
When the home position return request is ON, a near-zero point dog/count/data setting type home position return is also made under G28 of a motion program.

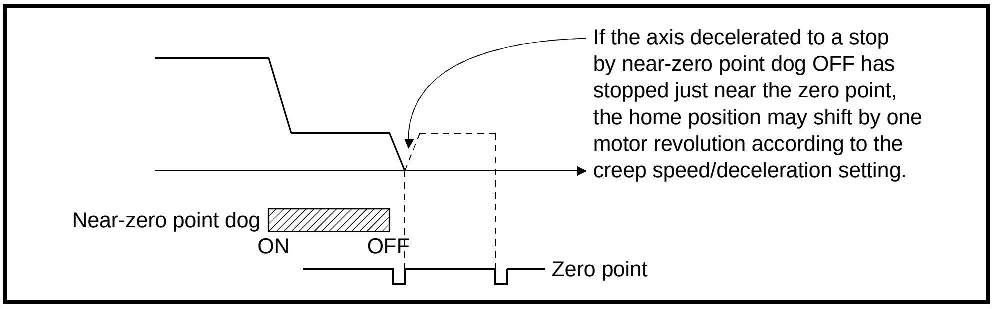
[Cautions]

The following instructions are given for a near-zero point dog type home position return.

(1) Keep the near-zero point dog ON until the axis decelerates from the home position speed to the creep speed.
If the near-zero point dog turns OFF before the axis decelerates to the creep speed, the axis decelerates to a stop and the next zero point is defined as a home position.



(2) Adjust the position where the near-zero point dog turns OFF so that the "home position return second travel" becomes half of the travel corresponding to one motor revolution.
If the "home position return second travel" is not half of the travel corresponding to one motor revolution, the home position may shift by one motor revolution as shown below.



IMPORTANT

(1) In either of the following cases, make a home position return after performing JOG operation or the like to return the axis to the position before the near-zero point dog turned ON.
A home position return cannot be made without returning the axis to the position before the near-zero point dog.
  (a) Home position return in the position after the near-zero point dog has turned from ON to OFF
  (b) Home position return when power is switched from OFF to ON after completion of a home position return

7.6.2 Count type home position return

[Control Details]

(1) Count type
The count type is a method in which the home position is a zero point in the specified distance (travel after near-zero point dog ON) after the near-zero point dog has turned ON.
Set the travel after near-zero point dog ON to the home position return data (refer to Section 4.4).

(2) Count type home position return
The operation of the count type home position return is shown in Fig. 7.4.

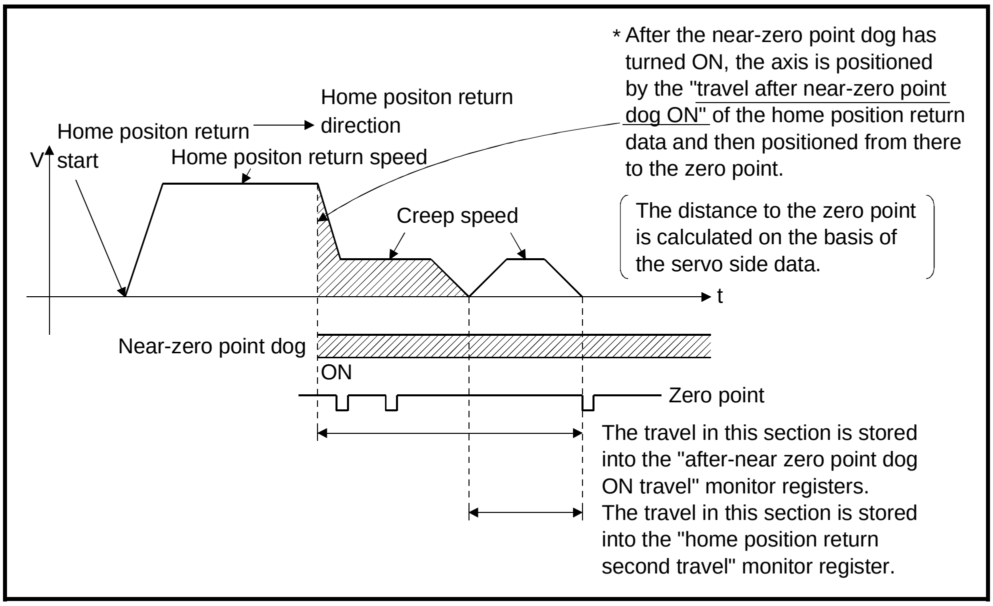


Fig. 7.4 Count Type Home Position Return Operation

(3) Execution of home position return
Execute a home position return using the DSFLP/CHGA instruction in Section 7.6.4.

[Cautions]

(1) The near-zero point dog should be turned OFF a sufficient distance away from the home position.

(2) In the count type, you can execute a home position return on the near-zero point dog or consecutive starts of a home position return.
When a home position return on the near-zero point dog or consecutive starts of a home position return have been executed, the axis is returned to the OFF position of the near-zero point dog once and makes a home position return.

7.6.3 Data setting type home position return

[Control Details]

(1) Data setting type
The data setting type is a method which does not use a near-zero point dog and can be used in an absolute position system.

(2) Data setting type home position return
The home position address is the present value during execution of a home position return made by the DSFRP/CHGA instruction.

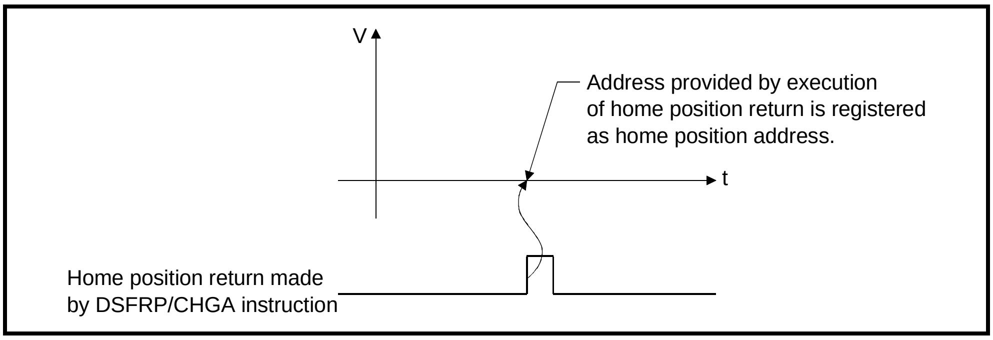


Fig. 7.5 Data Setting Type Home Position Return Operation

(3) Execution of home position return
Execute a home position return using the DSFLP/CHGA instruction in Section 7.6.4.

[Cautions]

(1) The axis must have passed through the zero point from power-on till the execution of a home position return.
A "zero point non-passage error" occurs if a home position return is executed without the axis passing through the zero point once. If the "zero point non-passage error" has occurred, reset the error, perform JOG operation or the like to run the servo motor one revolution or more, then make a home position return again.
Whether the axis has passed through the zero point or not can be checked by the zero pass signal (M1606+20n/M2406+20n).

(2) In a system other than an absolute position system, a data setting type home position return start has the same function as a present value change.

(3) The home position return data used for the data setting type are the home position return method and home position address.

7.6.4 Execution of home position return

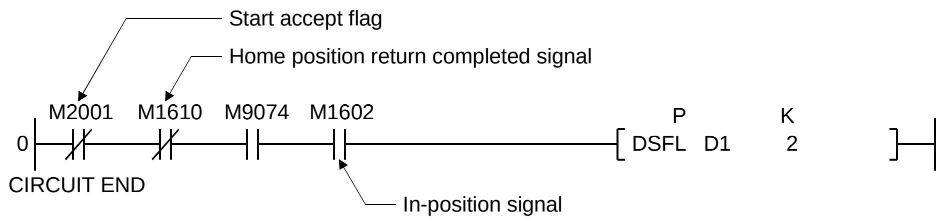
Use the DSFLP/CHGA instruction to execute a home position return.

[Control Details]

(1) A home position return is made in the home position return method specified in the home position return data (refer to Section 4.4). For details of the home position return method, refer to the following sections.
- Near-zero point dog type .... Section 7.6.1
- Count type .... Section 7.6.2
- Data setting type.......................... Section 7.6.3

[Cautions]

(1) After the PC ready flag (M2000) has turned ON, making a near-zero point dog type home position return in the following ladder before the PCPU ready flag (M9074) turns ON causes a home position return request to be given again after a home position return.
When making a home position return, use M9074 and M1602+20n or M2402+20n (in-position signal) as interlock conditions.
(Refer to the program example.)

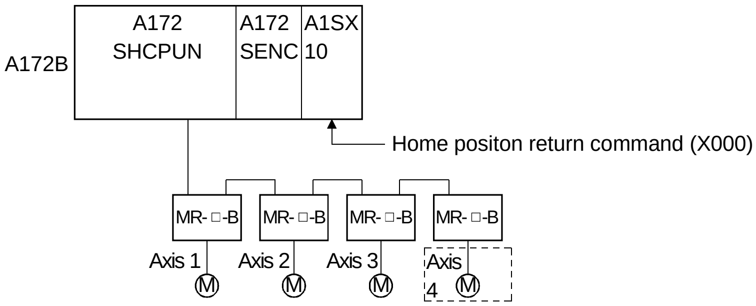


[Program Example]

A program using the DSFLP/CHGA instruction to make a home position return is explained under the following conditions.

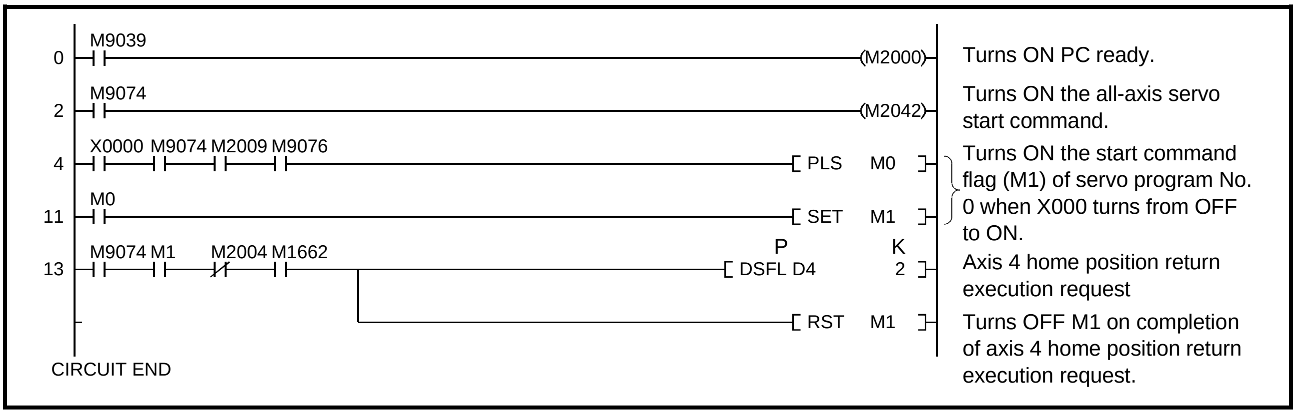
(1) System configuration

Axis 4 is returned to the home position.



(2) Sequence program example

A sequence program used to execute a home position return is shown below.



7.7 Speed Change

Used to change speed during positioning control or JOG operation.
A speed change is made with the DSFLP or CHGV instruction in a sequence program.

[Control Details]

(1) The speed of an operating axis is forcibly changed to the speed specified in the speed changing registers.

(2) A speed change is made using the DSFLP or CHGV instruction. Refer to Section 5.4 for details of the DSFLP or CHGV instruction.

(3) A speed change should be made in the range - speed limit value to + speed limit value. Error "305" will occur if it is made outside the range.

(4) Make the override invalid when making a speed change during positioning control for program operation. When the override is valid, a speed change is not made.

(5) During a temporary stop, a speed change is not made.

(6) A speed change during CP control (when the axis moves through mid points consecutively during execution of G01, G02, G03 or G32) should be made within the range -F command to +F command. If a speed change is made outside the range, the speed is controlled by the F command.

(7) The F command after a speed change during CP control is made valid within the range of not higher than the new speed.

(8) If a speed change is made during positioning control for program operation, the new speed is used for operation up to the instruction in the next move block. Depending on the type of the mode of the move block to be executed next, whether the speed change value is maintained or the command speed in the program will be used changes as indicated in Table 7.3.

(9) A speed change is invalid for the high-speed oscillation axis.

Table 7.3 Command Speed after Execution of Speed Change

	Move Mode at Speed Change *1	Move Mode after Speed Change *1	Command Speed at Execution of Move Instruction after Speed Change
1	PTP *2	PTP/OSC *2	Program command speed*6 is used.
2		CP *3	
3	CP *3	PTP/OSC *2	Program command speed*6 is used.
4		CP *3 with F command	Program command speed*7 is used.
5		CP *3 Without F command and without special M code*4	New speed is maintained.
6		CP *3 Without F command and with special M code*5	Program command speed*6 is used.

*1: A speed change is valid only for execution of move in the PTP or CP move mode.

*2: The PTP mode is a move mode executed under G00, G28, G30 or G53. The OSC mode is a move mode executed under G25.

*3: The CP mode is a move mode executed under G01, G2, G3 or G32. The independent M code is also handled as the CP mode.

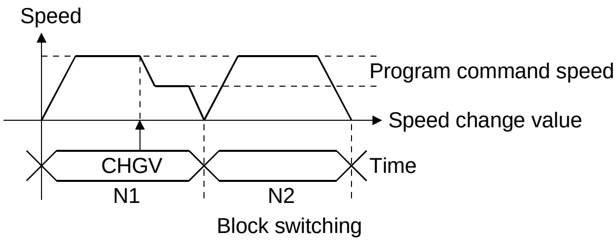
*4: CP without special M code indicates that the special M code (M00, M01, M02, M30, M98, M99, M100) is not executed during the CP mode after a speed change.

*5: CP with special M code indicates that the special M code (M00, M01, M02, M30, M98, M99, M100) is executed during the CP mode after a speed change.
The axis decelerates to a stop as soon as the special M code is executed.

*6: The program command speed indicates the rapid feedrate in the PTP mode, the F (frequency) command in the OSC mode, or the F (speed) command in the CP mode.

Example (CHGV executed during N1)

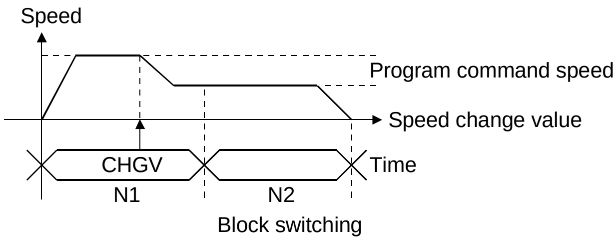
```
010;
N1 G00 X100. ;
N2 G00 X200. ;
M02;
%
```



*7: The F (speed) command is used. Note that it is clamped at the speed change value.

Example (CHGV executed during N1)

```
011;
N1 G01 X100. F1000. ;
N2 G01 X200. F1000. ;
M02;
%
```



[Data setting]

(1) The speed changing registers of each axis are indicated below. (A172SHCPUN/A171SHCPUN only)

<A172SHCPUN>

Axis No.	Speed Change Registers	
	Upper	Lower
1	D963	D962
2	D969	D968
3	D975	D974
4	D981	D980
5	D987	D986
6	D993	D992
7	D999	D998
8	D1005	D1004

<A171SHCPUN>

Axis No.	Speed Change Registers	
	Upper	Lower
1	D963	D962
2	D969	D968
3	D975	D974
4	D981	D980

(2) The setting ranges to the speed change registers are indicated below.

Unit / Item	mm		inch		degree	
	Setting range	Unit	Setting range	Unit	Setting range	Unit
Speed change value	0 to 600000000	$\times10^{-2}$ mm/min	0 to 600000000	$\times10^{-3}$ inch/min	0 to 2147483.647	$\times10^{-3}$ degree/min

POINT

When setting the speed in a sequence program, store into the speed change registers a value which is 100 times (unit: mm)/1000 times (unit: inch, degree) the actual speed.

Example

To change the speed to 10000.00mm/min, store "1000000" into the speed change registers.

[Cautions]

A speed change will not be made if any of the following errors occurs.
(A check is made at execution of the DSFLP/CHGV instruction.)

Error Definition		Error Processing	Error Code
Data setting error	Axis No. setting is other than 1 to 8/1 to 4.	• Error step is stored into D9010 or D9011. • M9010 or M9011 turns ON.	–
	Axis No. setting is indirectly specified by index qualification.		
	Preset speed is outside the range 0 to speed limit value.	• Error detection flag (M1607+20n) turns ON. • Error code given on the right is stored into the minor error code storage register of the corresponding axis.	305
Speed change error	Specified axis was making home position return.	• Error detection flag (M1607+20n) turns ON. • Error code given on the right is stored into the minor error code storage register of the corresponding axis.	301
	Deceleration was being made due to OFF of the JOG operation signal.		304

(1) If a speed change is made, the preset speed is ignored in any of the following cases. (An error will not occur.)
   (a) During motion program execution
   (b) During deceleration under the stop command
   (c) During a stop
   (d) During manual pulse generator operation

[Operation Timing]

The operation timing for making a speed change is shown in Fig. 7.6.

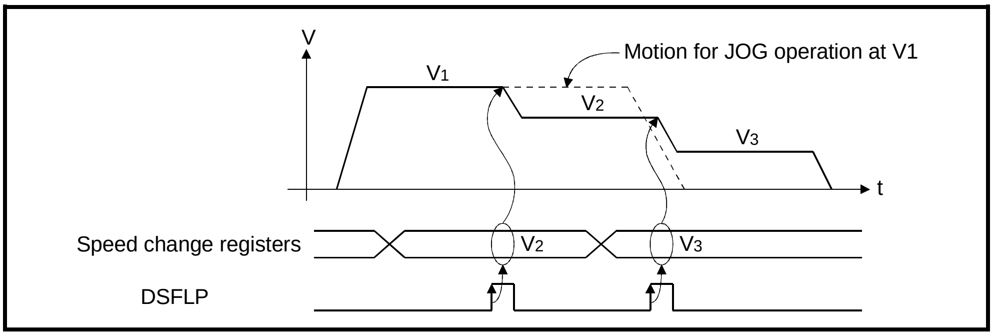


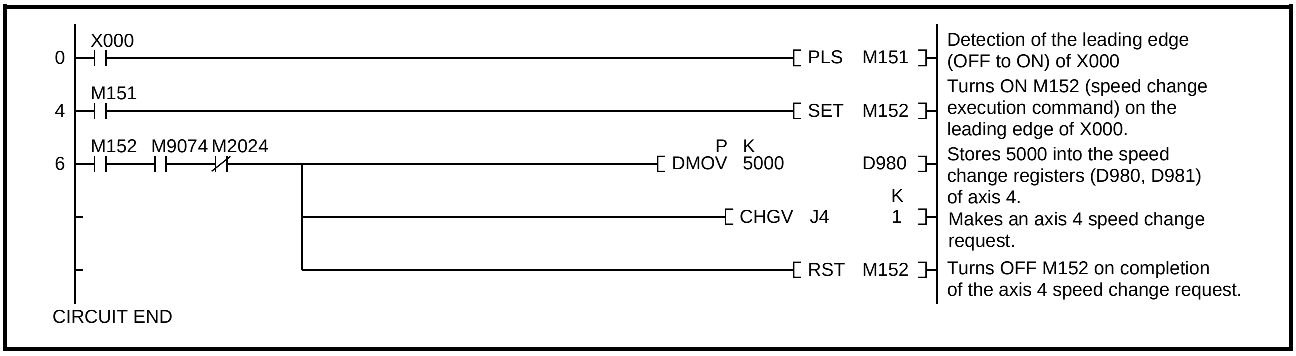
Fig. 7.6 Operation Timing for Speed Change

[Program Example]

A program example for making a speed change is described under the following conditions.

(1) Speed changing conditions
- (a) Axis No. whose speed is changed................ Axis 4
- (b) New speed.. 5000
- (c) Speed change command.............................. X000

(2) Sequence program



7.8 JOG Operation

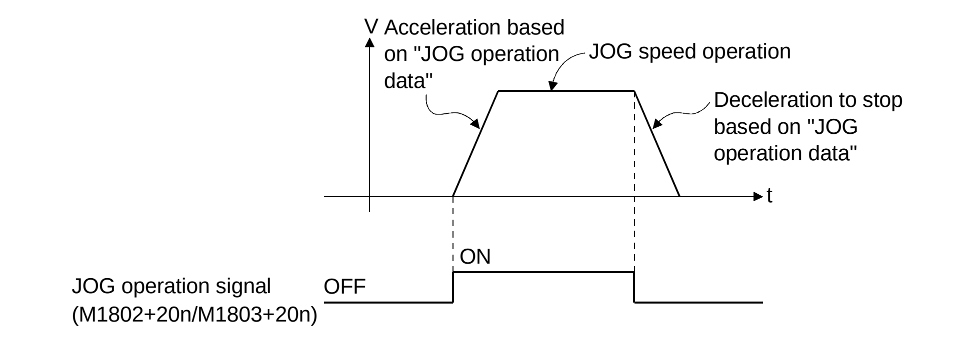
Preset JOG operation is performed.
Individual start or simultaneous start can be made for JOG operation.
JOG operation can be performed from a sequence program or in the test mode of the peripheral device.
(For the JOG operation method in the test mode of the peripheral device, refer to the operating manual of the peripheral device used.)
To perform JOG operation, the JOG operation data (refer to Section 4.5) must be set to each axis.

7.8.1 Individual start

JOG operation of the specified axis is started.
The following JOG operation signals are used for JOG operation.
- Forward rotation JOG operation........M1802+20n
- Reverse rotation JOG operation ....M1803+20n

[Control Details]

(1) While the JOG operation signal is ON, JOG operation is performed using the JOG operation speed setting register value. When the JOG operation signal turns OFF, the axis decelerates to a stop.
Acceleration/deceleration is controlled in accordance with the data set to the JOG operation data.



JOG operation of the axis whose JOG operation speed is ON is performed.

(2) The following table lists the JOG operation signal, JOG operation setting registers and setting range of each axis.

<A172SHCPUN/A171SHCPUN>

No.	A172SHCPUN				A171SHCPUN				Setting range					
	JOG operation		JOG operation speed setting registers		JOG operation		JOG operation speed setting registers		mm		inch		degree	
	Forward rotation JOG	Reverse rotation JOG	Upper	Lower	Forward rotation JOG	Reverse rotation JOG	Upper	Lower	Setting range	Unit	Setting range	Unit	Setting range	Unit
1	M1802	M1803	D965	D964	M1802	M1803	D965	D964	1 to 600000000	10^{-2} mm/min	1 to 600000000	10^{-3} inch/min	1 to 2147483647	10^{-3} degree /min
2	M1822	M1823	D971	D970	M1822	M1823	D971	D970						
3	M1842	M1843	D977	D976	M1842	M1843	D977	D976						
4	M1862	M1863	D983	D982	M1862	M1863	D983	D982						
5	M1882	M1883	D987	D986	–	–	–	–						
6	M1902	M1903	D993	D992	–	–	–	–						
7	M1922	M1923	D999	D998	–	–	–	–						
8	M1942	M1943	D1005	D1004	–	–	–	–						

<A273UHCPU (32-axis feature)/A173UHCPU(S1)>

No.	JOG operation		JOG operation speed setting registers		Setting range							
					mm		inch		degree		PULSE	
	Forward rotation JOG	Reverse rotation JOG	Upper	Lower	Setting range	Unit	Setting range	Unit	Setting range	Unit	Setting range	Unit
1	M3202	M3203	D641	D640	1 to 600000000	10^{-2} mm/min	1 to 600000000	10^{-2} inch/min	1 to 2147483647	10^{-2} degree/min	1 to 10000000	PLS/sec
2	M3222	M3223	D643	D642								
3	M3242	M3243	D645	D644								
4	M3262	M3263	D647	D646								
5	M3282	M3283	D649	D648								
6	M3302	M3303	D651	D650								
7	M3322	M3323	D653	D652								
8	M3342	M3343	D655	D654								
9	M3362	M3363	D657	D656								
10	M3382	M3383	D659	D658								
11	M3402	M3403	D661	D660								
12	M3422	M3423	D663	D662								
13	M3442	M3443	D665	D664								
14	M3462	M3463	D667	D666								
15	M3482	M3483	D669	D668								
16	M3502	M3503	D671	D670								
17	M3522	M3523	D673	D672								
18	M3542	M3543	D675	D674								
19	M3562	M3563	D677	D676								
20	M3582	M3583	D679	D678								
21	M3602	M3603	D681	D680								
22	M3622	M3623	D683	D682								
23	M3642	M3643	D685	D684								
24	M3662	M3663	D687	D686								
25	M3682	M3683	D689	D688								
26	M3702	M3703	D691	D690								
27	M3722	M3723	D693	D692								
28	M3742	M3743	D695	D694								
29	M3762	M3763	D697	D696								
30	M3782	M3783	D699	D698								
31	M3802	M3803	D701	D700								
32	M3822	M3823	D703	D702								

POINT

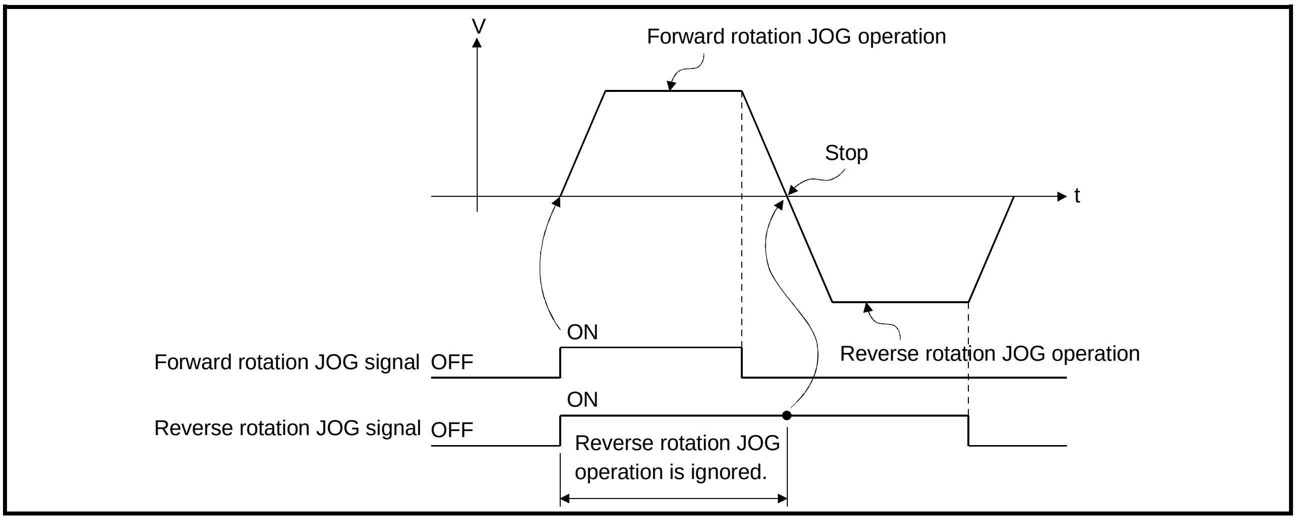
When setting the JOG operation speed in a sequence program, store into the JOG operation speed setting registers a value which is 100 times (unit: mm)/1000 times (unit: inch, degree) the actual speed.

Example

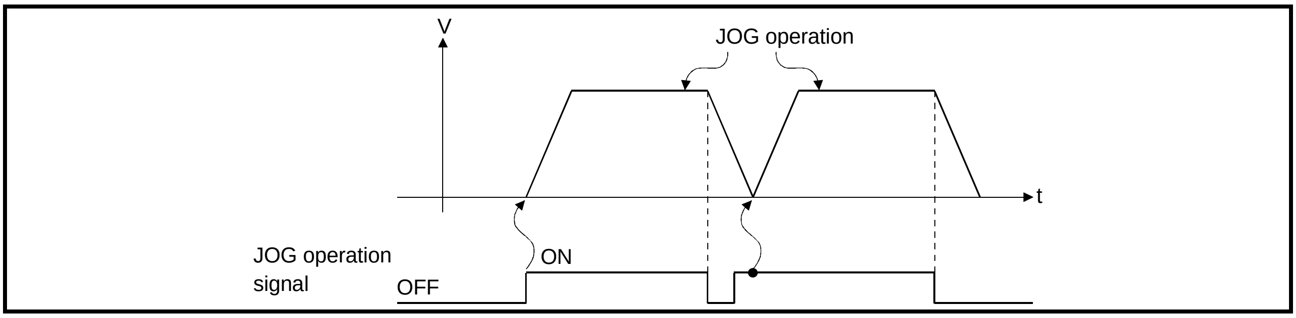
To set the JOG operation speed to 6000.00mm/min, store "600000" into the JOG operation speed setting registers.

[Cautions]

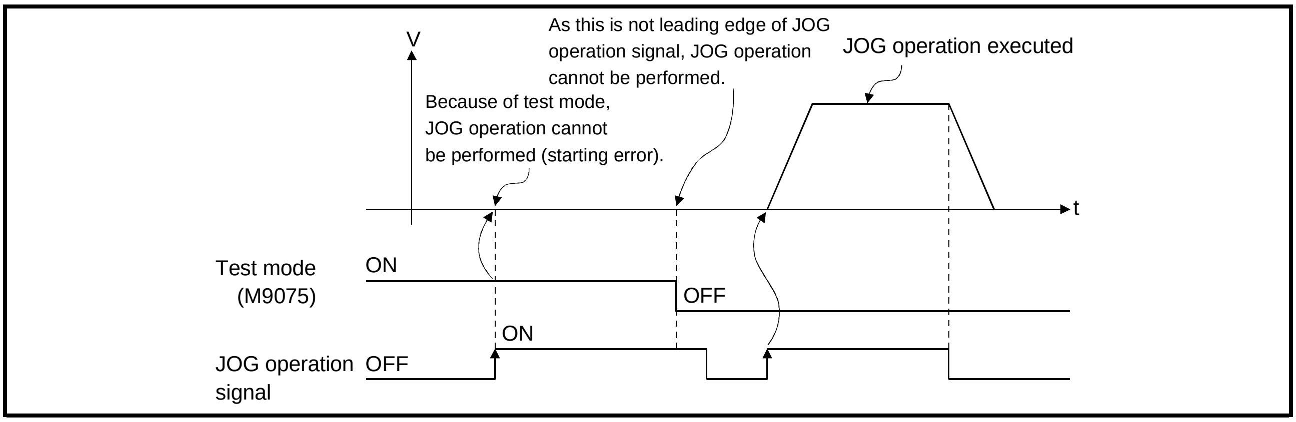
(1) Forward rotation JOG operation will be performed if the forward rotation JOG signal (M1802+20n/M3202+20n) and reverse rotation JOG signal (M1803+20n/ M3203+20n) of one axis have turned ON at the same time.
When the axis is decelerated to a stop after the forward rotation JOG signal has turned OFF, reverse rotation JOG operation is performed if the reverse rotation JOG signal is ON.



(2) If the JOG operation signal turns ON during deceleration due to OFF of the JOG operation signal, the axis decelerates to a stop down to speed 0 and then resumes JOG operation.



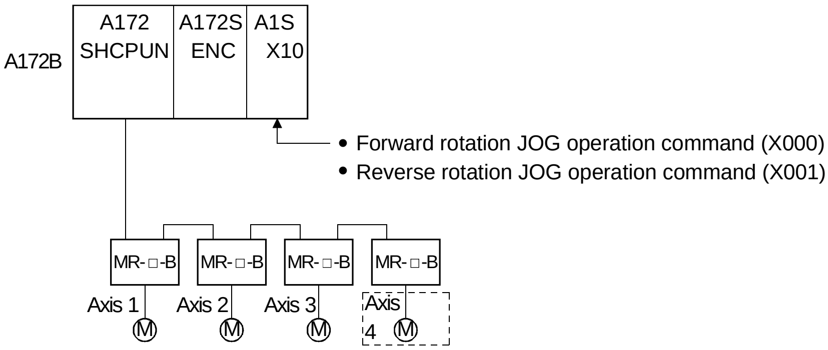
(3) In the test mode using the peripheral device, JOG operation under control of the JOG operation signal (M1802+20n/M1803+20n/M3202+20n/M3203+20n) is not performed.
After the test mode is canceled, JOG operation is started on the leading edge (OFF to ON) of the JOG operation signal.



[Program Example]

A program for JOG operation is described under the following conditions.

(1) System configuration
JOG operation of axis 4 is performed.

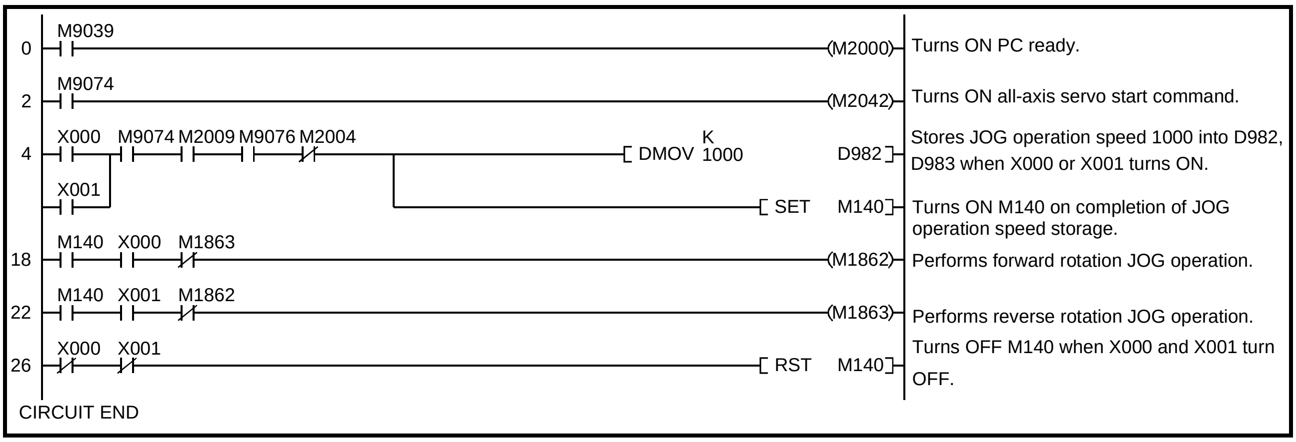


(2) JOG operation conditions
(a) Axis No .... Axis 4

(b) JOG operation speed 1000

(c) JOG operation commands
1) Forward rotation JOG operation..... During ON of X000
2) Reverse rotation JOG operation .... During ON of X001

(3) Sequence program



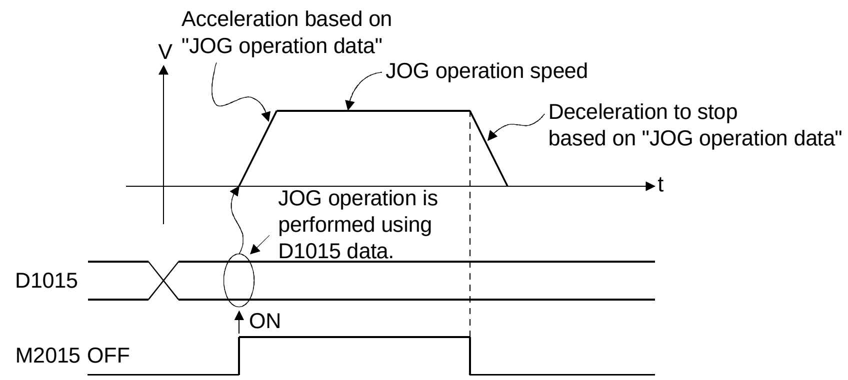
7.8.2 Simultaneous start

JOG operations of the specified multiple axes are started simultaneously.

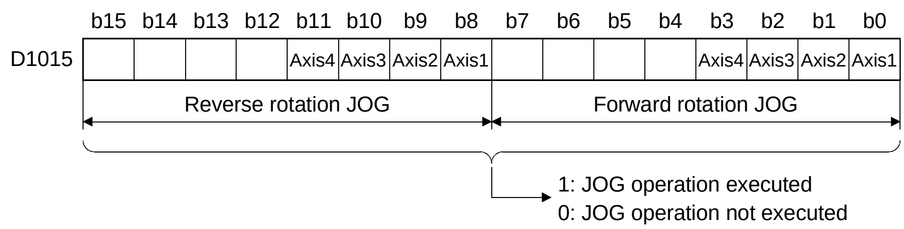
[Control Details]

• A172SHCPUN/A171SHCPUN

(1) While the JOG simultaneous start command flag (M2015) is ON, JOG operation is performed using the JOG operation speed setting register value of each axis. When M2015 turns OFF, the axes decelerate to a stop. Acceleration/deceleration is controlled in accordance with the data set to the JOG operation data.



(2) Set the axes for JOG operation to the JOG operation simultaneous start axis setting area (D1015).



Example

Make setting as follows when using the MOV instruction to perform forward rotation JOG operation of axes 1 and 2 and reverse rotation JOG operation of axis 4.

(1) For setting in hexadecimal (H) •••••• MOV H0803 D1015

(2) For setting in decimal (K) •••••••••• MOV K2051 D1015

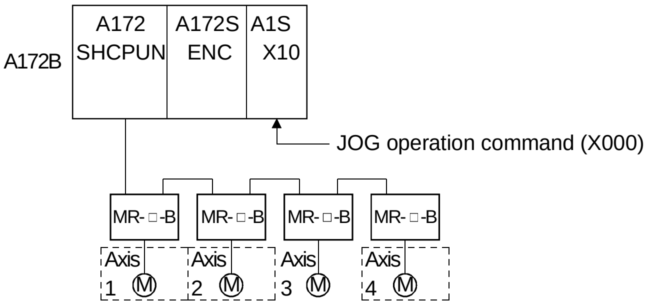
(3) The following table lists the JOG operation speed setting registers.

No.	A172SHCPUN				A171SHCPUN				Setting range					
	JOG operation		JOG operation speed setting registers		JOG operation		JOG operation speed setting registers		mm		inch		degree	
	Forward rotation JOG	Reverse rotation JOG	Upper	Lower	Forward rotation JOG	Reverse rotation JOG	Upper	Lower	Setting range	Unit	Setting range	Unit	Setting range	Unit
1	M1802	M1803	D965	D964	M1802	M1803	D965	D964	1 to 600000000	10^{-2} mm/min	1 to 600000000	10^{-2} inch/min	1 to 2147483647	10^{-2} degree /min
2	M1822	M1823	D971	D970	M1822	M1823	D971	D970						
3	M1842	M1843	D977	D976	M1842	M1843	D977	D976						
4	M1862	M1863	D983	D982	M1862	M1863	D983	D982						
5	M1882	M1883	D987	D986	–	–	–	–						
6	M1902	M1903	D993	D992	–	–	–	–						
7	M1922	M1923	D999	D998	–	–	–	–						
8	M1942	M1943	D1005	D1004	–	–	–	–						

[Program Example]

A program for simultaneous start of JOG operations is described under the following conditions.

(1) System configuration
JOG operations of axes 1, 2 and 4 are performed.

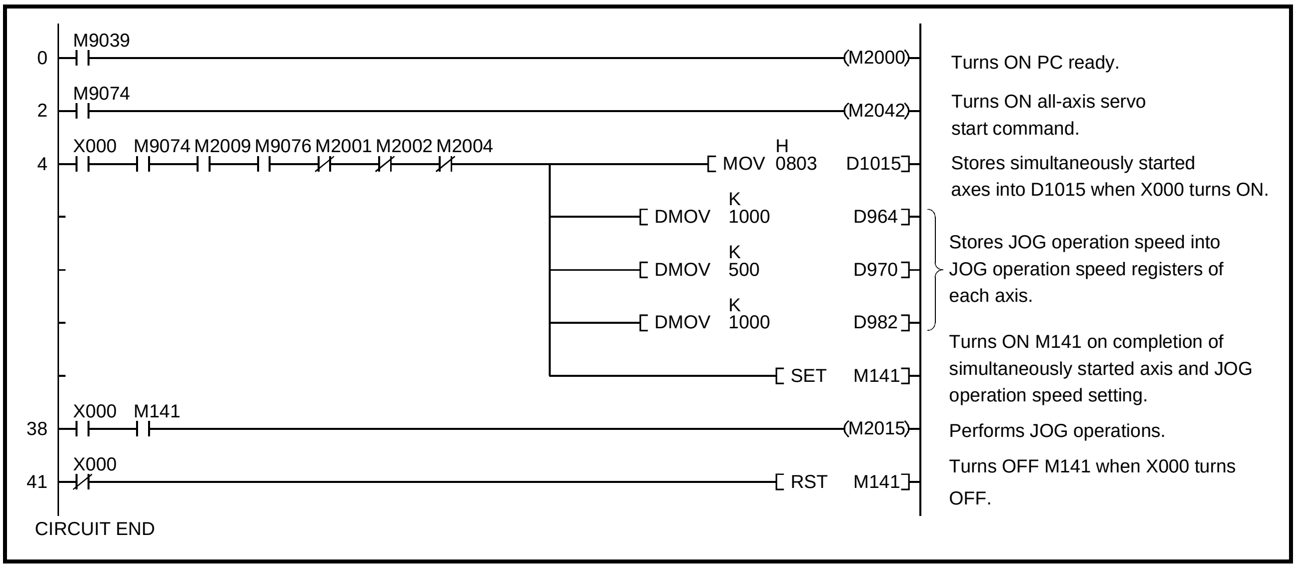


(2) JOG operation conditions
(a) JOG operation conditions are listed below.

Item	JOG Operation Conditions		
Control axis	Axis 1	Axis 2	Axis 4
JOG operation speed	1000	500	1000
JOG operation direction	Forward	Forward	Reverse

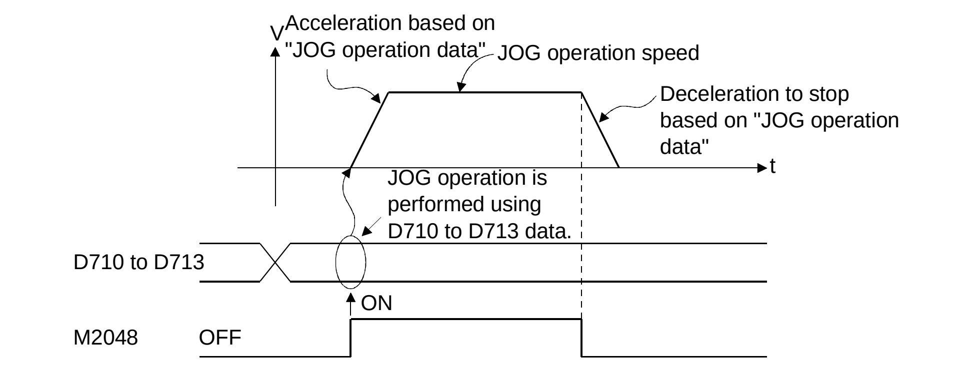
(b) JOG operation command During ON of X000

(3) Sequence program



• A273UHCPU (32-axis feature)/A173UHCPU (S1)

(1) While the JOG simultaneous start command flag (M2048) is ON, JOG operation is performed using the JOG operation speed setting register value of each axis. When M2048 turns OFF, the axes decelerate to a stop. Acceleration/deceleration is controlled in accordance with the data set to the JOG operation data.

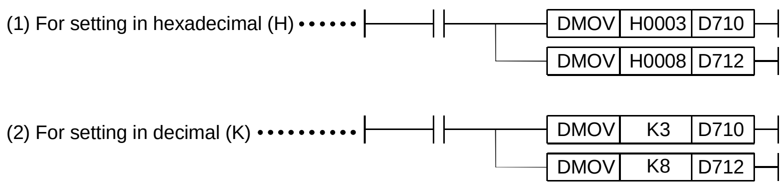


(2) Set the axes for JOG operation to the JOG operation simultaneous start axis setting areas (D710 to D713).

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	
D710	Axis16	Axis15	Axis14	Axis13	Axis12	Axis11	Axis10	Axis9	Axis8	Axis7	Axis6	Axis5	Axis4	Axis3	Axis2	Axis1	Forward rotation JOG
D711	Axis32	Axis31	Axis30	Axis29	Axis28	Axis27	Axis26	Axis25	Axis24	Axis23	Axis22	Axis21	Axis20	Axis19	Axis18	Axis17	
D712	Axis16	Axis15	Axis14	Axis13	Axis12	Axis11	Axis10	Axis9	Axis8	Axis7	Axis6	Axis5	Axis4	Axis3	Axis2	Axis1	Reverse rotation JOG
D713	Axis32	Axis31	Axis30	Axis29	Axis28	Axis27	Axis26	Axis25	Axis24	Axis23	Axis22	Axis21	Axis20	Axis19	Axis18	Axis17	

*Set 1/0 to specify JOG operation simultaneous start axes.
1: Simultaneous start executed
0: Simultaneous start not executed

Make setting as follows when using the MOV instruction to perform forward rotation JOG operation of axes 1 and 2 and reverse rotation JOG operation of axis 4.



(3) The following table lists the JOG operation speed setting registers.

No.	JOG operation		JOG operation speed setting registers		Setting range							
					mm		inch		degree		PLUSE	
	Forward rotation JOG	Reverse rotation JOG	Upper	Lower	Setting range	Unit	Setting range	Unit	Setting range	Unit	Setting range	Unit
1	M3202	M3203	D641	D640	1 to 600000000	10^{-2} mm/min	1 to 600000000	10^{-3} inch/min	1 to 2147483647	10^{-3} degree/min	1 to 10000000	PLS/sec
2	M3222	M3223	D643	D642								
3	M3242	M3243	D645	D644								
4	M3262	M3263	D647	D646								
5	M3282	M3283	D649	D648								
6	M3302	M3303	D651	D650								
7	M3322	M3323	D653	D652								
8	M3342	M3343	D655	D654								
9	M3362	M3363	D657	D656								
10	M3382	M3383	D659	D658								
11	M3402	M3403	D661	D660								
12	M3422	M3423	D663	D662								
13	M3442	M3443	D665	D664								
14	M3462	M3463	D667	D666								
15	M3482	M3483	D669	D668								
16	M3502	M3503	D671	D670								
17	M3522	M3523	D673	D672								
18	M3542	M3543	D675	D674								
19	M3562	M3563	D677	D676								
20	M3582	M3583	D679	D678								
21	M3602	M3603	D681	D680								
22	M3622	M3623	D683	D682								
23	M3642	M3643	D685	D684								
24	M3662	M3663	D687	D686								
25	M3682	M3683	D689	D688								
26	M3702	M3703	D691	D690								
27	M3722	M3723	D693	D692								
28	M3742	M3743	D695	D694								
29	M3762	M3763	D697	D696								
30	M3782	M3783	D699	D698								
31	M3802	M3803	D701	D700								
32	M3822	M3823	D703	D702								

7.9 Manual Pulse Generator Operation

Positioning control is exercised according to the number of pulses entered from the manual pulse generator.
One manual pulse generator enables simultaneous operation of 1 to 3 axes and the number of manual pulse generators connected is as follows.

Number of Connectable Manual Pulse Generators	
A172SHCPUN/A171SHCPUN	A273UHCPU (32-axis feature)/A173UHCPU(S1)
1	3

[Control Details]

• A172SHCPUN/A171SHCPUN

(1) The axes set to the manual pulse generator axis setting register are positioned according to the pulse input from the manual pulse generator.
Manual pulse generator operation is made valid only when the manual pulse generator enable flag is ON.

Manual Pulse Generator Axis Setting Register	Manual Pulse Generator Enable Flag
D1012	M2012

(2) The travel and output speed of positioning control according to the input from the manual pulse generator are as follows.

(a) Travel
The travel according to the pulses input from the manual pulse generator is calculated by the following expression.

[Travel] = [travel per pulse] × [number of input pulses] × [manual pulse generator 1-pulse input magnification setting]

The travels per pulse in manual pulse generator operation are as indicated below.

Unit	Travel
mm	0.0001mm
inch	0.00001inch
degree	0.00001degree

When the unit is mm, the input of one pulse commands the travel of (0.0001mm) × (1 pulse) × (manual pulse generator 1-pulse input magnification setting).

(b) Output speed
In manual pulse generator operation, the axis is positioned at the speed which meets the number of input pulses per unit time.

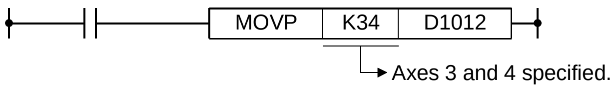
[Output speed] = [number of input pulses per 1ms] × [manual pulse generator 1-pulse input magnification setting]

(3) Setting of control axes operated by manual pulse generator

(a) Set the axes to be controlled by the manual pulse generator to the manual pulse generator axis setting register (D1012).
Set the axis to be controlled (1 to 8/1 to 4) in each digit of up to 3 decimal digits.
(The set number of digits indicates the number of axes to be operated simultaneously.)

Example

Make the following setting to control axes 3 and 4 by the manual pulse generator.



(4) Manual pulse generator 1-pulse input magnification setting

(a) Set to each axis the magnification at input of one pulse from the manual pulse generator.
<A172SHCPUN>

1-Pulse Input Magnification Setting Register	Corresponding Axis No.	Setting Range
D1016	Axis 1	1 to 10000
D1017	Axis 2	
D1018	Axis 3	
D1019	Axis 4	
D1020	Axis 5	
D1021	Axis 6	
D1022	Axis 7	
D1023	Axis 8	

<A171SHCPUN>

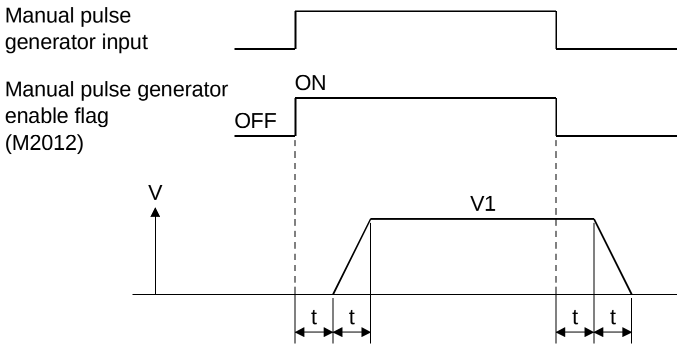
1-Pulse Input Magnification Setting Register	Corresponding Axis No.	Setting Range
D1016	Axis 1	1 to 10000
D1017	Axis 2	
D1018	Axis 3	
D1019	Axis 4	

(5) For the manual pulse generator 1-pulse input magnification which has been set, the "manual pulse generator 1-pulse input magnification setting register" of the corresponding axis is checked on the leading edge of the manual pulse generator enable flag.
If the value is outside the setting range, the manual pulse generator axis setting error storage register (D9187) and manual pulse generator axis setting error flag (M9077) are set and the magnification is controlled as "1".

(6) Manual pulse generator smoothing magnification setting
Set the magnification for smoothing the leading and trailing edges of manual pulse generator operation.

Manual Pulse Generator Smoothing Magnification Setting Register	Setting Range
D9192	0 to 59

(a) Operation



Output speed (V1) = (number of input pulses/ms)×(manual pulse generator 1-pulse input magnification setting)
Travel (L) = (travel per pulse) × number of input pulses×(manual pulse generator 1-pulse input magnification setting)

REMARKS

1) The travel per pulse of the manual pulse generator is as indicated below.

• Setting unit
- mm :0.0001mm
- inch :0.00001inch
- degree :0.00001degree

2) The smoothing time constant is 56.8ms to 3408ms.

(7) The definitions of errors at manual pulse generator operation data setting are indicated below.

Error Definition	Error Processing
Axis setting specified in any digit is other than 1 to 8/1 to 4.	• Only the digit in error is ignored. • The axes of the digits where any of 1 to 8/1 to 4 is set are made valid and perform manual pulse generator operation.
Axis set to manual pulse generator operation is specified.	• Axis of overlapped designation is ignored. • Manual pulse generator operation specified first is performed.
Setting is made in 4 or more digits.	• All axes set are ignored.

• A273UHCPU (32-axis feature)/A173UHCPU (S1)

POINTS

• When the A273UHCPU is used and two or more A273EX modules are loaded, connect the manual pulse generator to the first A273EX (starting from slot 0 of the main base).
(The manual pulse generator is valid for the first module only).
• When the A173UHCPU is used, one A172SENC is required for one manual pulse generator. Connect manual pulse generators to the first to third A172SENCs.

(1) The axes set to the manual pulse generator axis setting register are positioned according to the pulse input from the manual pulse generator.
Manual pulse generator operation is made valid only when the manual pulse generator enable flag is ON.

Manual Pulse Generator Connecting Position	Manual Pulse Generator Axis Setting Registers	Manual Pulse Generator Enable Flag
P1	D714, D715	M2051
P2	D716, D717	M2052
P3	D718, D719	M2053

(2) The travel and output speed of positioning control according to the input from the manual pulse generator are as follows.

(a) Travel
The travel according to the pulses input from the manual pulse generator is calculated by the following expression.

[Travel] = [travel per pulse] × [number of input pulses] × [manual pulse generator 1-pulse input magnification setting]

The travels per pulse in manual pulse generator operation are as indicated below.

Unit	Travel
mm	0.1μm
inch	0.00001inch
degree	0.00001degree
PULSE	1PULSE

When the unit is mm, the input of one pulse commands the travel of (0.1μm) × (1 pulse) × (manual pulse generator 1-pulse input magnification setting).

(b) Output speed
In manual pulse generator operation, the axis is positioned at the speed which meets the number of input pulses per unit time.

[Output speed] = [number of input pulses per 1ms] × [manual pulse generator 1-pulse input magnification setting]

(3) Setting of control axes operated by manual pulse generator

(a) Set the axes to be controlled by the manual pulse generator to the manual pulse generator axis setting registers (D714 to D719).
Set the bits corresponding to the controlled axes (1 to 32).

Example

Make the following setting to control axes 1, 22 and 30 by the manual pulse generator 1.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
D714	Axis16	Axis15	Axis14	Axis13	Axis12	Axis11	Axis10	Axis9	Axis8	Axis7	Axis6	Axis5	Axis4	Axis3	Axis2	Axis1
D715	Axis32	Axis31	Axis30	Axis29	Axis28	Axis27	Axis26	Axis25	Axis24	Axis23	Axis22	Axis21	Axis20	Axis19	Axis18	Axis17

(1) For setting in hexadecimal (H) •••••• DMOV H20200001 D714

(2) For setting in decimal (K) •••••••••• DMOV K538968065 D714

(4) Manual pulse generator 1-pulse input magnification setting

(a) Set to each axis the magnification at input of one pulse from the manual pulse generator.

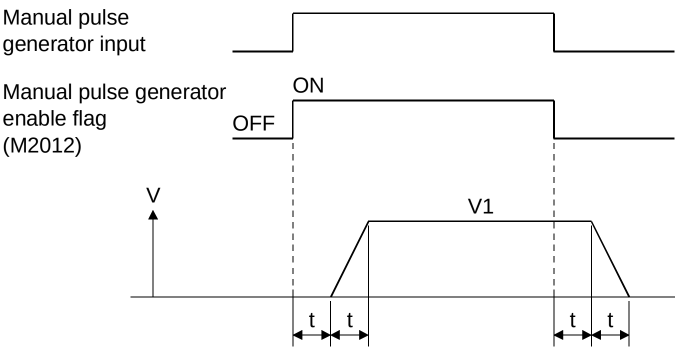
1-Pulse Input Magnification Setting Register	Corresponding Axis No.	Setting range
D720	Axis 1	1 to 100
D721	Axis 2	
D722	Axis 3	
D723	Axis 4	
D724	Axis 5	
D725	Axis 6	
D726	Axis 7	
D727	Axis 8	
D728	Axis 9	
D729	Axis 10	
D730	Axis 11	
D731	Axis 12	
D732	Axis 13	
D733	Axis 14	
D734	Axis 15	
D735	Axis 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	

(5) For the manual pulse generator 1-pulse input magnification which has been set, the "manual pulse generator 1-pulse input magnification setting register" of the corresponding axis is checked on the leading edge of the manual pulse generator enable flag.
If the value is outside the setting range, the manual pulse generator axis setting error storage registers (D9185 to D9187) and manual pulse generator axis setting error flag (M9077) are set and the magnification is controlled as "1".

(6) Manual pulse generator smoothing magnification setting
Set the magnification for smoothing the leading and trailing edges of manual pulse generator operation.

Manual Pulse Generator Smoothing Magnification Setting Register	Setting range
Manual pulse generator 1 (P1): D752	0 to 59
Manual pulse generator 2 (P2): D753	
Manual pulse generator 3 (P3): D754	

(a) Operation



Output speed (V1) = (number of input pulses/ms) × (manual pulse generator 1-pulse input magnification setting)
Travel (L) = (travel per pulse) × number of input pulses × (manual pulse generator 1-pulse input magnification setting)

REMARKS

1) The travel per pulse of the manual pulse generator is as indicated below.

• Setting unit
- mm :0.1μm
- inch :0.00001inch
- degree :0.00001degree
- PULSE :1pulse

2) The smoothing time constant is 56.8ms to 3408ms.

(7) The definitions of errors at manual pulse generator operation data setting are indicated below.

Error Definition	Error Processing
Axis setting specified in any digit is other than 1 to 32.	• Only the digit in error is ignored. • The axes of the digits where any of 1 to 32 is set are made valid and perform manual pulse generator operation.
Axis set to manual pulse generator operation is specified.	• Axis of overlapped designation is ignored. • Manual pulse generator operation specified first is performed.
The axes set are 4 or more axes.	• Only three axes starting from the lower number of the manual pulse generator axis setting registers are made valid and operated.

[Cautions]

(1) The start acceptance flag turns ON for the axis set to manual pulse generator operation.
Therefore, positioning control, home position return or the like cannot be started by the servo system CPU or peripheral device.
Turn OFF the manual pulse generator enable flag after manual pulse generator operation is finished.

(2) The torque limit value is fixed at 300% during manual pulse generator operation.

(3) When the manual pulse generator enable flag is turned ON for the axis which is being operated by positioning control, JOG operation or the like, error 214 is set to the corresponding axis and manual pulse generator input is not enabled.
After the axis has stopped, the rise of the manual pulse generator enable flag is made valid to enable the manual pulse generator input, and the start acceptance flag turns ON to import the input from the manual pulse generator.

(4) If the manual pulse generator enable flag of another manual pulse generator is turned ON for the axis which is performing manual pulse generator operation, error 214 is set to the corresponding axis and input is not enabled for that manual pulse generator. After the manual pulse generator operation enabled for input first has stopped, turn ON the manual pulse generator enable flag again.

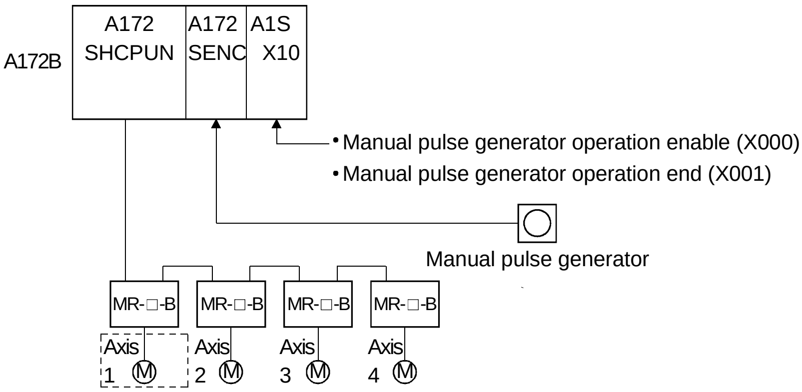
(5) If, after the manual pulse generator enable flag has been turned OFF, the manual pulse generator enable flag is turned ON again for the axis which is making smoothing deceleration, error 214 is set and manual pulse generator input is not enabled. After the axis has stopped after smoothing deceleration (the start acceptance flag has turned OFF), turn ON the manual pulse generator enable flag.

(6) If, after the manual pulse generator enable flag has been turned OFF, you set another axis and turn ON the same manual pulse generator enable flag during smoothing deceleration, manual pulse generator input is not enabled. At this time, the manual pulse generator axis setting error bit of the manual pulse generator axis setting error storage register (D9187/D9185 to D9187) turns ON and the manual pulse generator axis setting error flag (M9077) turns ON. As the condition to turn ON the manual pulse generator enable flag, provide OFF of the start acceptance flag of the specified axis as an interlock.

[Program Example]

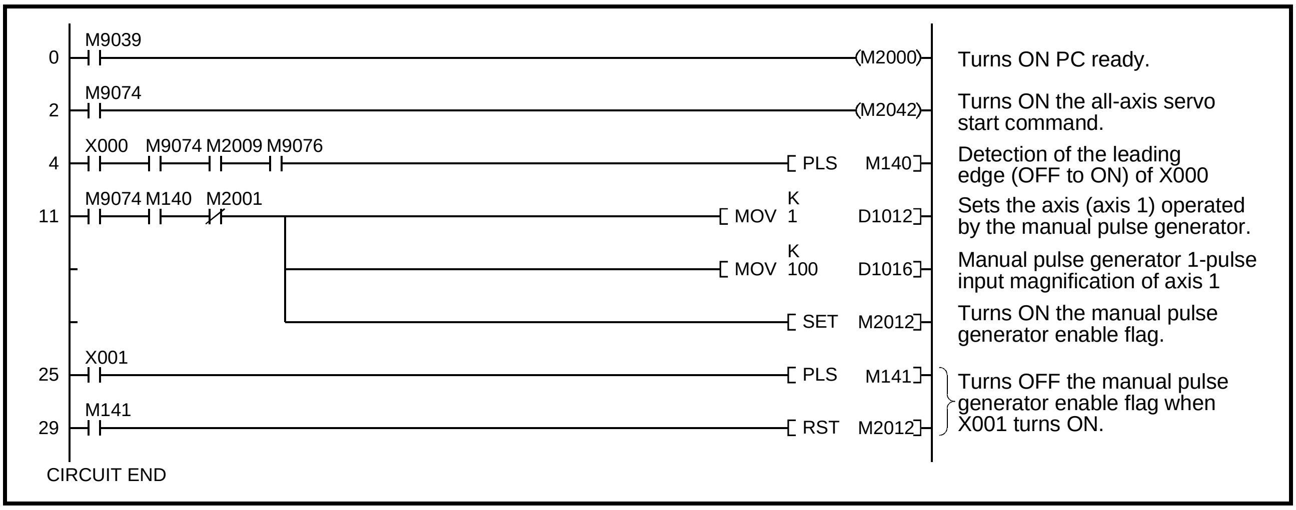
A program for manual pulse generator operation is described under the following conditions.

(1) System configuration
Manual pulse generator operation of axis 1 is performed.



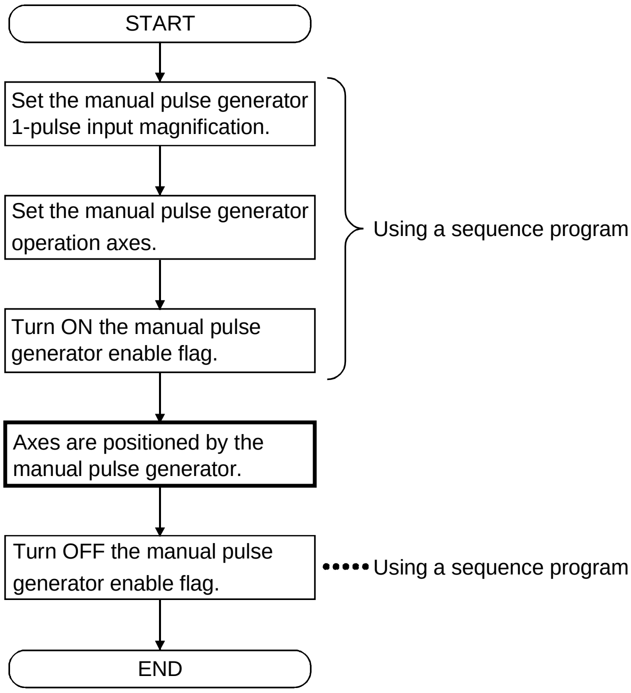
(2) Manual pulse generator operation conditions
(a) Manual pulse generator operation axis...Axis 1
(b) Manual pulse generator 1-pulse input magnification 100
(c) Manual pulse generator enable .... Leading edge (OFF to ON) of X000
(d) Manual pulse generator end.................... Leading edge (OFF to ON) of X001

(3) Sequence program example
A sequence program used to perform manual pulse generator operation is shown below.



[Manual Pulse Generator Operation Procedure]

The manual pulse generator operation procedure is indicated below.



7.10 Override Ratio Setting Function

With the override ratio setting function, you can set the ratio of override to the command speed in a motion program to change the speed.

[Control Details]

(1) To the command speed in a motion program, set the override ratio in the range 0 to 100% in 1% increments. The value obtained by multiplying the command speed by the override value is the actual feedrate.

(2) Set the override ratio to each axis.
The default value is 100% in all axes.

[Data Setting]

(1) Use the override ratio setting register to change the speed with the override ratio setting function.
The following table lists the override ratio setting register of each axis.
<A172SHCPUN/A171SHCPUN>

Axis No.	Override Ratio Setting Register
1	D500
2	D506
3	D512
4	D518
5	D524
6	D530
7	D536
8	D542

<A273UHCPU (32-axis feature)/A173UHCPU(S1)>

Axis No.	Override Ratio Setting Register	Axis No.	Override Ratio Setting Register	Axis No.	Override Ratio Setting Register	Axis No.	Override Ratio Setting Register
1	D1440	9	D1488	17	D1536	25	D1584
2	D1446	10	D1494	18	D1542	26	D1590
3	D1452	11	D1500	19	D1548	27	D1596
4	D1458	12	D1506	20	D1554	28	D1602
5	D1464	13	D1512	21	D1560	29	D1608
6	D1470	14	D1518	22	D1566	30	D1614
7	D1476	15	D1524	23	D1572	31	D1620
8	D1482	16	D1530	24	D1578	32	D1626

(2) Set the ratio to the override ratio setting register in the range 0 to 100%.

(3) When the override ratio enable/disable (M1505+10n) is ON, the content of the override ratio setting register is valid. When M1505+10n is OFF, the speed is controlled at the override ratio of 100%.

[Cautions]

(1) When the DSFRP/SVST instruction is executed, the override ratio setting register data of the operating axis having the lowest number is made valid.
[Example]
Axis 2, 3, 4 start instruction

[SVST J2J3J4 K100]

• When the above DSFRP/SVST instruction is executed, the data of axis 2 is made valid. (The data of axes 3, 4 are made invalid.)

(2) When the speed is changed by the override ratio setting function, acceleration/deceleration processing is performed according to the "acceleration time" and "deceleration time" in the parameter block.

(3) The override ratio setting is valid only for motion program operation. (Invalid for JOG operation and so on.)

(4) The definitions of errors at override ratio data setting are indicated below.

Error Definition	Error Processing	Error Code
At a start, the value set in the override ratio setting register is other than 0 to 100%.	• Operation is performed at 100%. (Operation is performed at command speed in motion program.)	190
During operation, the value set in the override ratio setting register is other than 0 to 100%.		290

[Operation Timing]

The speed change timing by the override ratio setting function is shown in Fig. 7.7.

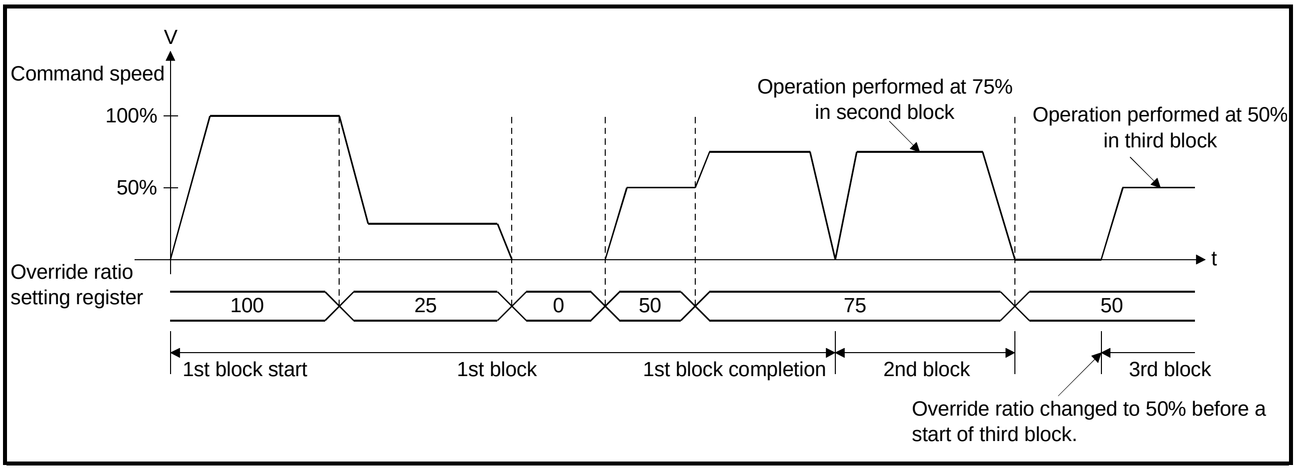


Fig. 7.7 Operation Timing at Override Ratio Setting

[Program Example]

A program example using the override ratio setting function is described under the following conditions.

(1) Override ratio setting conditions

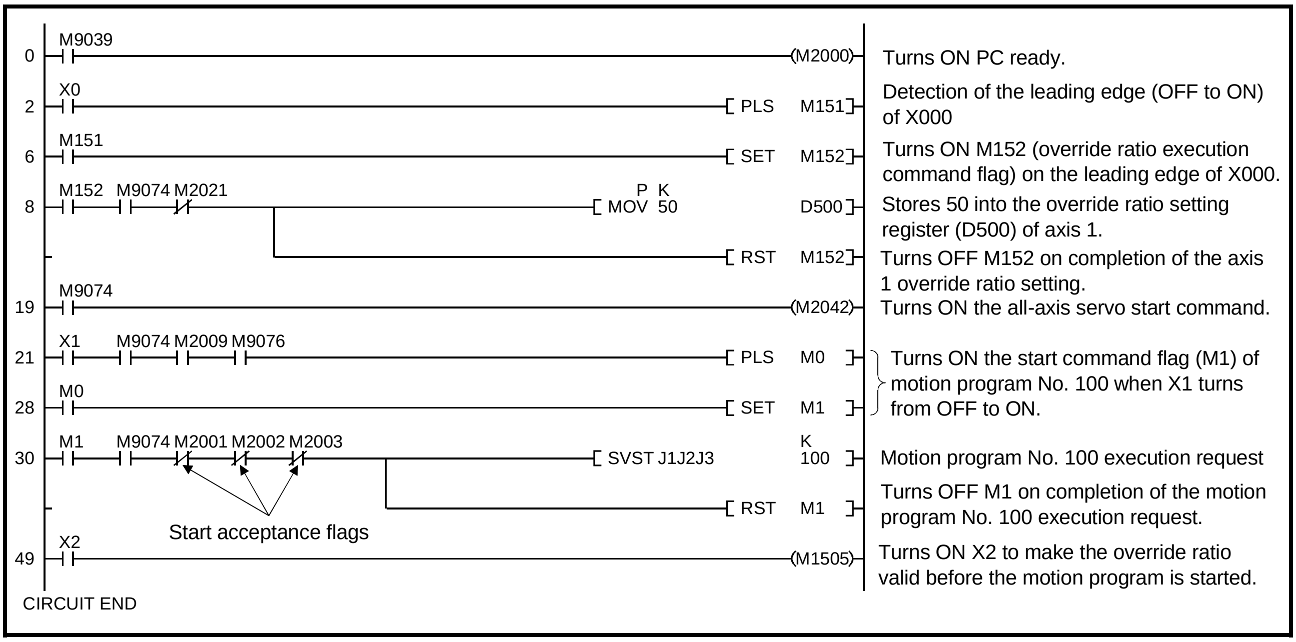
(a) Axis No. .... Axis 1

(b) Override ratio 50%

(c) Override ratio setting command.................... X180

(d) Motion program start command X181

(2) Sequence program



7.11 FIN Signal Waiting Function

The FIN signal waiting function is designed to synchronize the processing completion of each mid point with the FIN signal.
By setting the M code to each mid point for positioning, the execution of each point can be controlled by the FIN signal.

[Data Setting]

(1) The FIN signal and M code outputting signal correspond to the following devices of each axis.
<A172SHCPUN/A171SHCPUN>

Axis No.		1	2	3	4	5	6	7	8
FIN signal	A172SHCPUN	M1819	M1839	M1859	M1879	M1899	M1919	M1939	M1959
	A171SHCPUN	M1819	M1839	M1859	M1879	–	–	–	–
M code outputting signal	A172SHCPUN	M1619	M1639	M1659	M1679	M1699	M1719	M1739	M1759
	A171SHCPUN	M1619	M1639	M1659	M1679	–	–	–	–

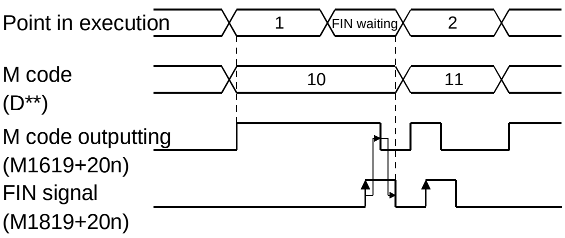
<A273UHCPU (32-axis feature)/A173UHCPU(S1)>

Axis No.	1	2	3	4	5	6	7	8
FIN signal	M3219	M3239	M3259	M3279	M3299	M3319	M3339	M3359
M code outputting signal	M2419	M2439	M2459	M2479	M2499	M2519	M2539	M2559
Axis No.	9	10	11	12	13	14	15	16
FIN signal	M3379	M3399	M3419	M3439	M3459	M3479	M3499	M3519
M code outputting signal	M2579	M2599	M2619	M2639	M2659	M2679	M2699	M2719
Axis No.	17	18	19	20	21	22	23	24
FIN signal	M3539	M3559	M3579	M3599	M3619	M3639	M3659	M3679
M code outputting signal	M2739	M2759	M2779	M2799	M2819	M2839	M2859	M2879
Axis No.	25	26	27	28	29	30	31	32
FIN signal	M3699	M3719	M3739	M3459	M3779	M3799	M3819	M3839
M code outputting signal	M2899	M2919	M2939	M2959	M2979	M2999	M3019	M3039

(2) The acceleration/deceleration system is the fixed acceleration/deceleration time mode.
The acceleration/deceleration time used is the acceleration time in the selected parameter block.

[Program Example]

```
01;
G01 X20. Y20. F100. M10; (Point 1)
X30. Y25. M11;           (Point 2)
X35. Y30. M12;           (Point 3)
X40. Y40;                (Point 4)
M02;
%
```



Operation explanation chart

1. When positioning of the axis to point 1 starts, the M code is output and the M code outputting signal turns ON.
2. In response to this, the PLC performs necessary processing and then turns ON the FIN signal. Until the FIN signal turns ON, the axis does not move to the next point.
3. When the PLC turns ON the FIN signal, the M code outputting signal turns OFF.
4. After the M code outputting signal has turned OFF, the PLC turns OFF the FIN signal. After this, positioning to next point 2 starts.

[Cautions]

(1) The M code outputting signal turns OFF when the stop command (external, M1800+20n, M1801+20n), cancel signal or skip signal is entered.

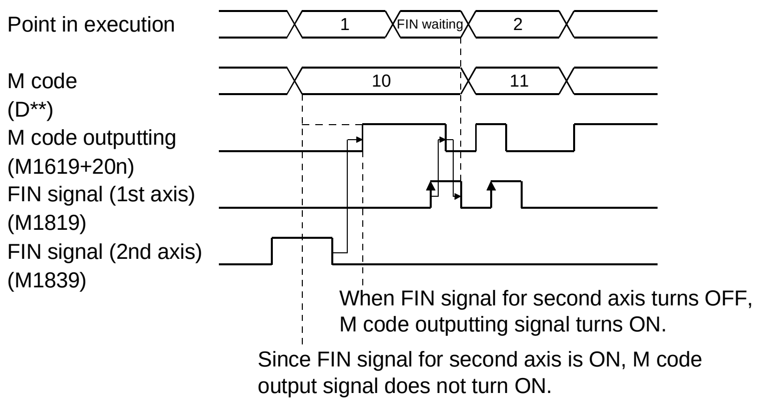
(2) When the M code is set to the last point, positioning is completed after the FIN signal is turned from OFF to ON to OFF.

(3) When the FIN waiting function is used , a shift to a point is made under the command before acceleration or deceleration. (Refer to the chart in (6) 2).)

(4) During interpolation, the M code outputting signal is output to all interpolation axes.
When inputting the FIN signal to interpolation axes, turn ON the signal of any of the interpolation axes.
Note that the FIN signal for the high-speed oscillation execution axis is ignored.

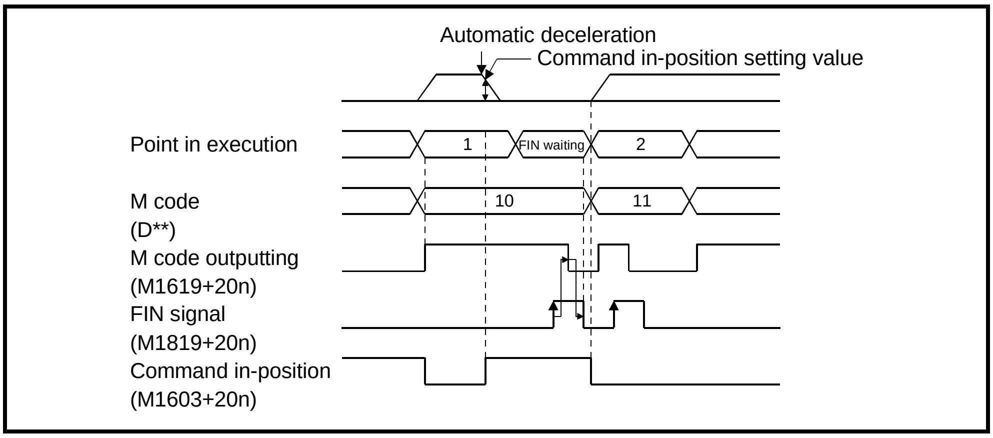
(5) When the FIN signal for any one of the interpolation axes is ON, the M code outputting signal is not output if the FIN waiting function is executed.

Example: When the FIN waiting function for point 1 is executed with the signal for the second axis kept ON

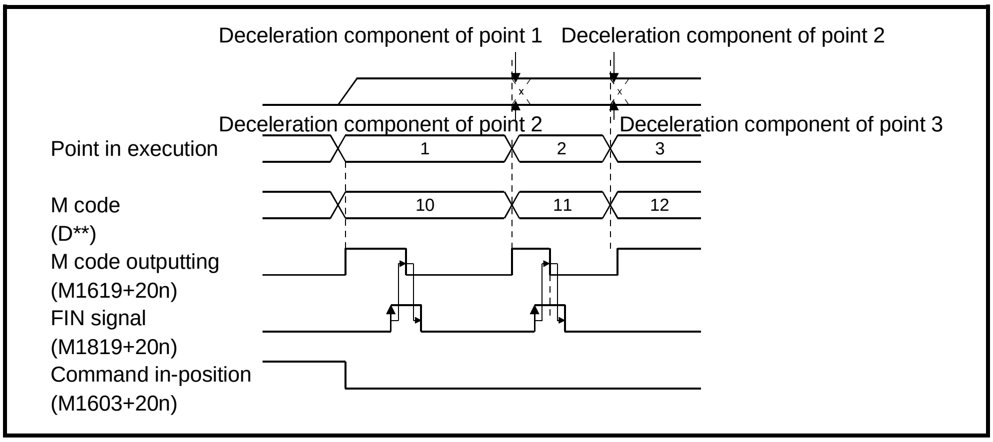


(6) When the FIN waiting function is used, the command in-position signal is output as described below.

1) When automatic deceleration is started by positioning to the executed point (including the last point) during FIN waiting
If the difference between the positioning address (command position) of the executed point and the feed present value falls within the command in-position range during FIN waiting, the command in-position signal (M1603+20n/M2403+20n) turns ON.
When the axis moves to the next point, the command in-position signal turns OFF.

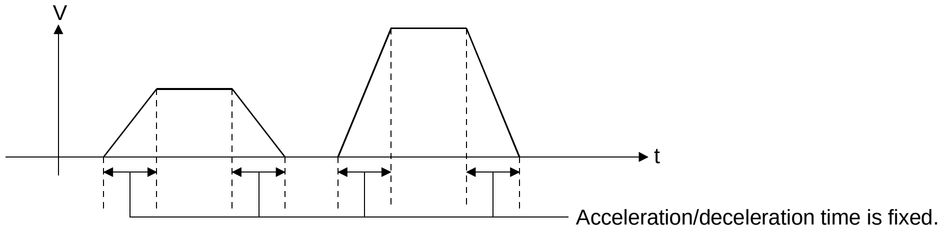


2) When the axis moves to the next point without automatic deceleration being made by positioning to the executed point during FIN waiting
If the axis moves to the next point without automatic deceleration, the command in-position signal does not turn ON.



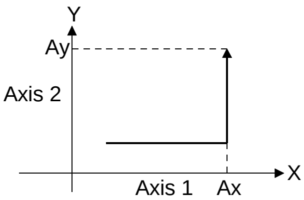
POINT

In the fixed acceleration/deceleration mode, the time required for acceleration/deceleration is fixed at different speeds.

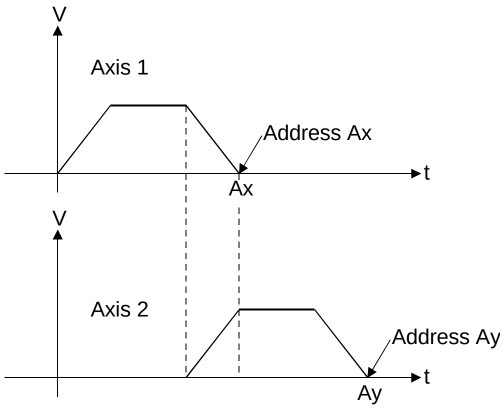


(1) In the fixed acceleration/deceleration mode, the following processing and parameters are invalid.
- Deceleration time and rapid stop deceleration time in parameter block
- S-pattern acceleration/deceleration

(2) When positioning operation (constant-speed control) as shown below is to be performed, speed processing of each axis is as shown below.



Positioning operation

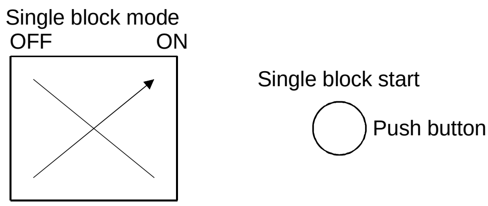


Constant-Speed Control Processing of Each Axis

7.12 Single Block

The single block function is designed to execute program operation block-by-block to check of run of a motion program.
The single block function is available in either of the following two modes. One is the mode in which the single block function is specified before a program start and the other is the mode in which the single block function is executed midway through a program run.
This section explains the latter mode where the single block function is executed midway through a program run.

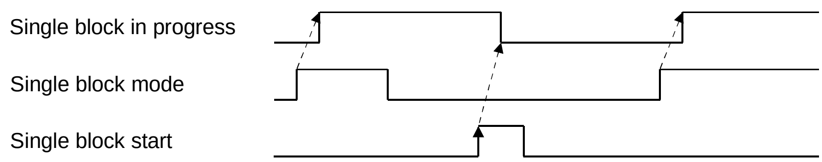
[Control Details]



During continuous operation, turn ON the single block mode signal and turn the single block start signal from OFF to ON to start single block operation at any point during operation.

(1) Single block signal devices
The following signals are related to the single block function.

Signal Name	Device No.		Signal Direction
	A172SHCPUN/A171SHCPUN	A273UHCPU (32-axis feature)/A173UHCPU	
Single block in progress	M1409	M4009	SCPU ← PCPU
Single block mode	M1508	M4408	SCPU → PCPU
Single block start	M1509	M4409	



These signals are valid for all program operations executed concurrently.

1) Single block in progress (M1409/M4009)
The single block in progress signal indicates that the single block function can be executed. When the single block in progress signal is ON, the single block function is executed. When the single block in progress signal is OFF, turn the SVST start or single block start signal from OFF to ON to start continuous operation.
When the single block mode signal is turned ON, the single block in progress signal turns ON.
When the single block mode signal is turned OFF and the single block start signal is then turned from OFF to ON, the single block in progress signal turns OFF.

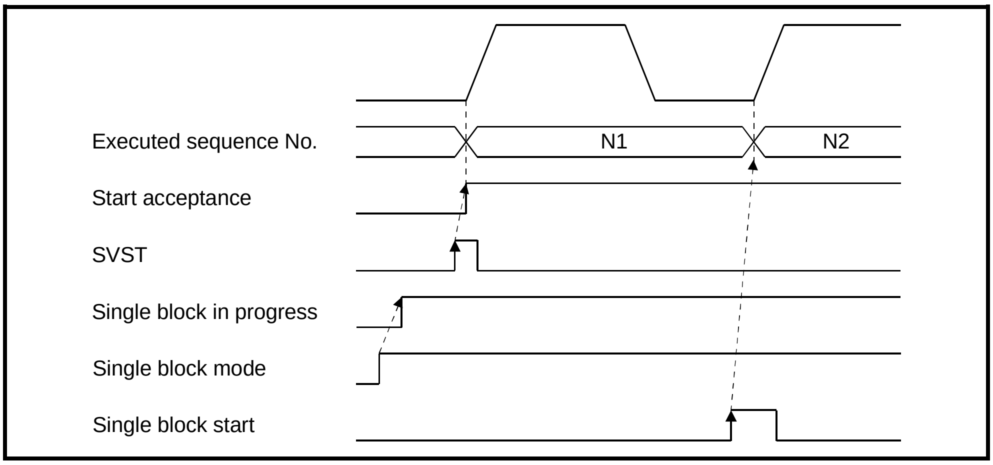
2) Single block mode (M1508/M4408)
The single block mode signal is designed to make the single block function valid.

3) Single block start (M1509/M4409)
The single block start signal is designed to start a program in a single block waiting status.

(2) How to execute single block from a start

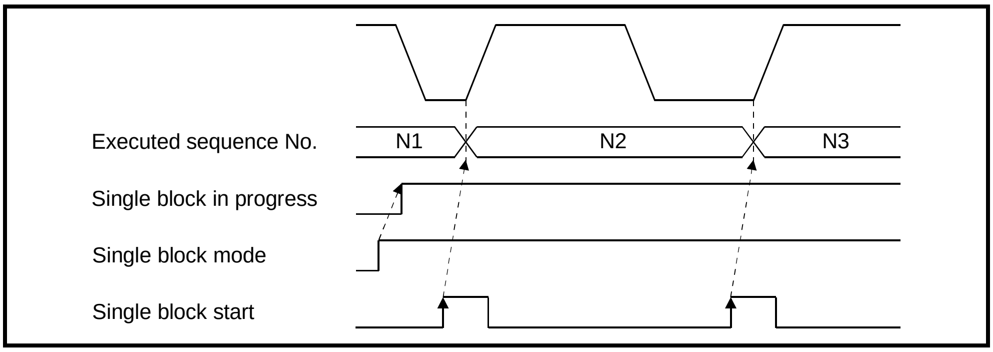
Turning ON the single block mode signal turns ON the single block in progress signal. In this status, turn ON the SVST start signal.

After the first block is executed, execution waits for the single block start signal to turn from OFF to ON.



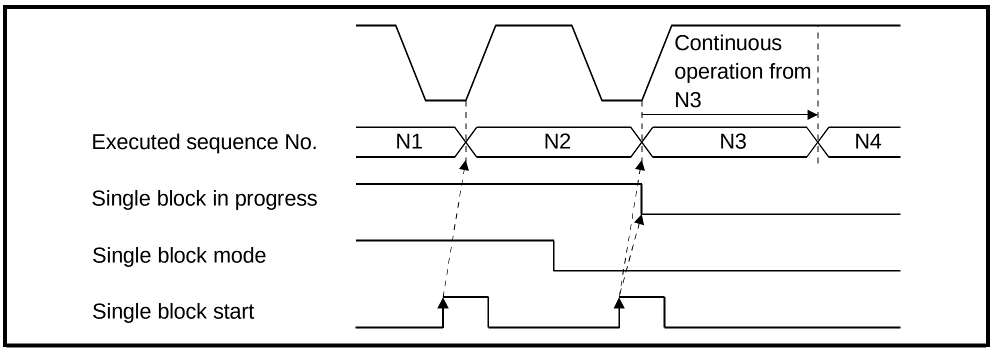
(3) How to continue single block

With the single block in progress signal ON, turn the single block start signal from OFF to ON. After one block program is run, execution waits for the single block start signal to turn ON.

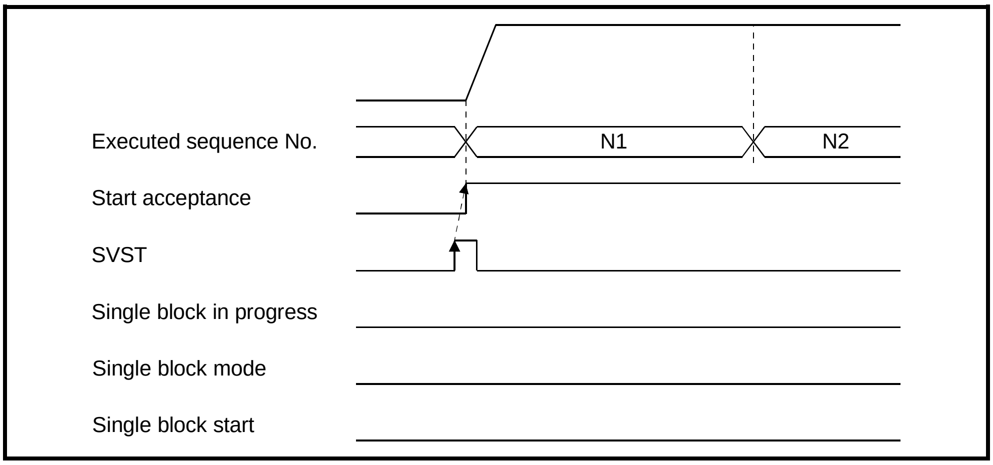


(4) How to start operation continuously during execution of single block

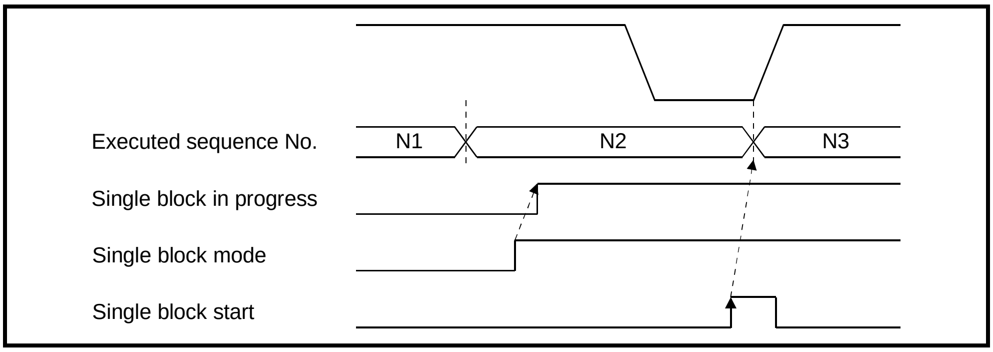
Turn ON the single block mode signal. In this state, turn the single block start signal from OFF to ON. This turns OFF the single block in progress signal and starts the program running continuously.



(5) How to perform continuous operation from a start (Ordinary operation)
With the single block in progress signal OFF, start a program with SVST to run the program continuously.



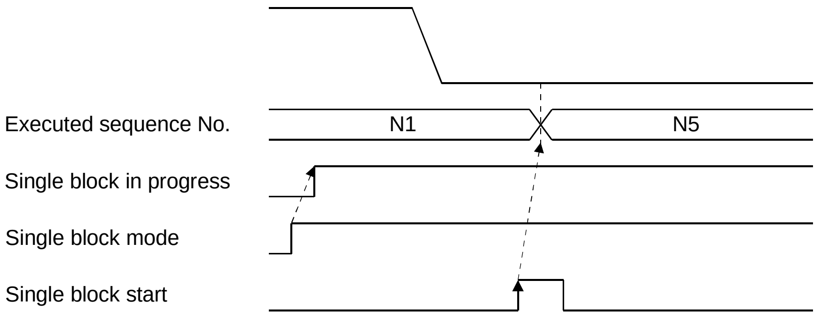
(6) How to execute single block during continuous operation
Turn ON the single block mode signal during program operation.
During move block execution, the program is stopped after termination of that block and execution waits for the single block start signal to turn from OFF to ON.



A macro instruction block, e.g. arithmetic operation, is preread during execution of the move instruction for PTP (e.g. G00) or CP (e.g. G01). Therefore, if the single block function is executed while the macro instructions are preread during motion, the executed block number and executed sequence number displayed are those in the preread area.

```
010;
N1 G01 X100. F100.; (Single block in progress is ON)
N2 #D0 = 0;
N3 #D2 = 1;
N4 #D3 = 2;
N5 #D4 = 3; (Preread completion block)
M02;
%
```

During N1 execution, the single block in progress signal is turned ON. If the macro instructions in up to N5 have been preread at this time, making a single block start for one block changes the executed sequence No. from N1 to N5.



[Cautions]

(1) Single block mode (M1508/M4408) and single block command (M1503+10n/M4403+10n)
If the single block mode signal (M1508/M4408) and single block command (M1503+10n/M4403+10n) are used to execute the single block function simultaneously, the operation performed by the single block command (M1503+10n/M4403+10n) is made invalid.

(2) Emergency stop, stop command, rapid stop command and error when single block in progress is ON
When the single block in progress signal is ON, it does not turn OFF if an emergency stop is made, the stop command or rapid stop command is given, or an error occurs.
The single block in progress signal turns OFF by turning OFF the single block mode signal and then turning the single block start signal from OFF to ON.

(3) Status at termination of one block execution when single block in progress is ON
If one block execution ends when the single block in progress signal is ON, the automatically operating signal (M1402+10n/M4002+10n) does not turn OFF. At this time, the command in-position signal (M1603+20n/M2403+20n) turns ON.

(4) Single block start during move instruction execution
During axis motion (except high-speed oscillation), the single block start signal is not accepted. Make a block start after the axis has been stopped by the single block function.

7.13 Enhanced Present Value Control

The following functions have been added to provide enhanced present value control when the ABS encode is used.

(1) Enhanced functions

(a) Function for checking the validity of an encoder during operation

- Checks whether encoder's variance in a 3.5ms time interval is within 180 degrees at the motor axis. (An error is indicated when the variance is not within 180 degrees.)
- Checks whether encoder data matches feed-back positions managed by the servo amplifier. (An error is indicated when the data does not match the feed-back positions.)

(b) Present value log monitor for checking the following values with peripheral devices

- Encoder present value, servo commanded value, and monitor present value (mechanical value) at power-on sequence
- Encoder present value, servo commanded value, and monitor present value (mechanical value) at power-off sequence
- Encoder present value, servo commanded value, and monitor present value (mechanical value) at home position return

(c) If an allowable travel value is set at power-off sequence, whether encoder data has changed exceeding the setting range at power-off sequence can be checked at servo amplifier power-on sequence. (An error is indicated when the encoder data has exceeded the setting range.)

(2) Restrictions on the servo amplifier

The following restrictions are imposed according to the servo amplifier combinations:

Servo amplifier	Restrictions
MR-H-B : BCD-B13W000-B2 and after MR-J2-B : BCDB20W200-A1 and after	No restrictions
MR-H-B : BCD-B13W000-B1 and after MR-J2-B : BCD-B20W200-A0 and before MR-J-B : All types ADU : All types	All enhanced functions cannot be used.

7.14 High–Speed Reading of Designated Data

This function stores the designated positioning data in the designated device (D, W) with the signal from an input module mounted on the motion base as the trigger.
It can be set in the system setting of a peripheral device software package.

(1) Positioning data that can be set
1. Positioning command
2. Actual present value
3. Position droop
4. M codes
5. Torque limit value
6. Motor current
7. Motor rpm
8. Servo command value

(2) Modules and signals used
<A172SHCPUN/A171SHCPUN>

Input Module	Signal	Reading Timing	Number of Points Settable
A172SENC/A171SENC	TREN	0.8ms	1
PC input module	X device		8

Note: Only one PC input module can be used.

<A273UHCPU (32 axis feature)/A173UHCPU (S1)>

Input Module	Signal	Reading Timing	Number of Points Settable
A273EX	TREN	0.8ms	3
A172SENC			1
PC input module	X device		8

Note: Only one PC input module can be used.

APPENDICES

APPENDIX 1 SCPU ERROR CODE LIST

If an error occurs when the PC is switched to the RUN status or is in the RUN status, the error indication and error code (including the step number) are stored in a special register by the self-diagnosis function. When an error occurs, refer to Table 1.1 for its cause and the corrective action to take.
Eliminate the cause of the error by taking the appropriate corrective action.
Error codes can be read at a peripheral device; for details on the relevant operation, see the Operating Manual for the peripheral device.

⚠ CAUTION
⚠ When an error occurs, check the points stated in this manual and reset the error.

Appendix 1.1 SCPU Error Code List

The list presented below gives the error numbers, and the error contents, causes, and corrective actions for each error message.

Table 1.1 Error Code List

Error Message (When an A273UHCPU is Used)	Contents of Special Register D9008 (BIN Value)	CPU Status	Error Contents and Cause	Corrective Action
"INSTRCT.CODE ERR" (When an instruction is executed.)	10	Stopped	An instruction code that cannot be decoded has been included in the program. (1) A ROM which includes undecodable instruction codes has been installed. (2) The memory contents have changed for some reason and now include an undecodable instruction code.	(1) Read the error step with a peripheral device, and correct the program at that step. (2) If the ROM is the problem, either rewrite its contents or replace it with a ROM into which the correct contents have been written.
"PARAMETER ERROR" (On switching on the power or resetting. On switching from {STOP/PAUSE} to {RUN/STEP RUN})	11	Stopped	The parameter data in the CPU's memory has been changed due to noise or incorrect installation of the memory.	(1) Check the installation of the memory and install it correctly. (2) Read the parameter data of the CPU memory at a peripheral device, check the data, correct it, and write the corrected data back into the memory.
"MISSING END INS." (When M9056 or M9057 is ON. On switching from {STOP/PAUSE} to {RUN/STEP RUN})	12	Stopped	(1) There is no END (FEND) instruction in the program. (2) When a subprogram is set in the parameters, there is no END instruction in the subprogram.	(1) Write an END instruction at the end of the program.
"CAN'T EXECUTE (P)" (When a CJ/SCJ/JMP/CALL(P)/ FOR-NEXT instruction is executed. On switching from {STOP/PAUSE} to {RUN/STEP RUN})	13	Stopped	(1) The jump destination designated with a CJ/SCJ/CALL/CALLP/JMP instruction does not exist, or more than one exists. (2) There is a CHG instruction but no subprogram is set. (3) Although there is no CALL instruction, there is a RET instruction in the program and is has been executed. (4) A CJ/SCJ/CALL/CALLP/JMP instruction whose jump destination is at or beyond the END instruction has been executed. (5) The number of FOR instructions does not match the number of NEXT instructions. (6) A JMP instruction has been included between a FOR and NEXT command, exiting the FOR - NEXT sequence. (7) The subroutine has been exited by execution of a JMP instruction before execution of a RET instruction. (8) Execution of a JMP instruction has caused a jump into a step in a FOR - NEXT range, or into a subroutine.	(1) Read the error step with a peripheral device, and correct the program at that step.(Correct, for example, by inserting a jump destination, or making sure there is only one jump destination.)

Table 1.1 Error Code List (Continued)

Error Message (When an A273UHCPU is Used)	Contents of Special Register D9008 (BIN Value)	CPU Status	Error Contents and Cause	Corrective Action
"CHK FORMAT ERR." (On switching from {STOP PAUSE} to {RUN STEP RUN})	14	Stopped	(1) An instruction other than an LDX, LDIX, ANDX, or ANIX instruction (including NOP) has been included in the same ladder block as a CHK instruction. (2) More than one CHK instruction exists. (3) The number of contacts in a CHK instruction ladder block exceeds 150. (4) The device number of an X device in a CHK instruction ladder block exceeds X7FE when using an A373CPU or X1FFE when using an A373U/A273U. (5) The following ladder block CJ□ has not been inserted before the CHK instruction ladder block. (6) The D1 device (number) of a CHK D1 D2 instruction is not the same as the device (number) of the contact before the CJ[ ] instruction. (7) The pointer P254 is not appended at the head of a CHK instruction ladder block. P254 CHK D1 D2	(1) Check if any of items (1) to (6) in the column to the left apply to the program with the CHK instruction ladder block, correct any problem in the program with a peripheral device, then restart program operation. (2) This error code is only valid when the I/O control method used is the direct method.
"CAN'T EXECUTE (I)" (When an interruption occurs. On switching from {STOP PAUSE} to {RUN STEP RUN})	15	Stopped	(1) An interrupt module is used but there is no number for the corresponding interrupt pointer I in the program. Or, more than one exists. (2) There is no IRET instruction in the interrupt program. (3) There is an IRET instruction other than in the interrupt program.	(1) Check the whether or not an interrupt program corresponding to the interrupt module exists and either create an interrupt program or eliminate the duplicated I number. (2) Check if there is an IRET instruction in the interrupt program: if there is not, insert one. (3) Check if there is an IRET instruction other than in the interrupt program: if there is, delete it.
"CASSETTE ERROR" (On switching on the power or resetting.)	16	Stopped	No memory cassette is installed.	Install a memory cassette and reset.
"RAM ERROR" (On switching on the power or resetting. When M9084 is turned ON in the STOP status.)	20	Stopped	(1) On checking if data can be read from and written to the CPU data memory area normally, it is determined that one or both are not possible.	There is a hardware fault. Contact your nearest Mitsubishi service center, agent, or office, and explain the problem.
"OPE.CIRCUIT ERR." (On switching on the power or resetting.)	21	Stopped	(1) The operation circuit that executes sequence processing in the CPU does not operate normally.	
"WDT ERROR" (At any time)	22	Stopped	The scan time has exceeded the watchdog error monitor time. (1) The user program scan time has been exceeded due to the conditions. (2) A momentary power interruption has occurred during scanning, extending the scan time.	(1) Calculate and check the scan time for the user program and shorten the scan time, e.g. by using a CJ instruction. (2) Monitor the contents of special register D9005 with a peripheral device. If the contents are other than "0" the power supply voltage is unstable: in this case check the power supply and reduce voltage fluctuation.
"END NOT EXECUTE" (When END processing is executed.)”	24	Stopped	(1) When the END instruction is executed it is read as another instruction code, e.g. due to noise. (2) The END instruction has been changed to another instruction code somehow.	(1) Reset and establish the RUN status again.If the same error is displayed again, the cause is a CPU hardware error. Contact your nearest Mitsubishi service center, agent, or office, and explain the problem.
"WDT ERROR" (At any time)	25	Stopped	A loop has been established for execution of the sequence program, due for example to a CJ instruction, and the END instruction cannot be executed.	Check if any program will be run in an endless loop: if there is such a program, modify the program.

Table 1.1 CPU Error Code List (Continued)

Error Message (When an A273UHCPU is Used)	Contents of Special Register D9008 (BIN Value)	CPU Status	Error Contents and Cause	Corrective Action
"UNIT VERIFY ERR." (When an END instruction is executed. However, no check is performed when M9084 or M9094 is ON.)	31	Stopped (RUN)	The I/O information does not match a loaded module when the power is switched ON. (1) An I/O module (this includes special function modules) is loose, or has become detached, during operation. Or, a completely different module has been loaded.	(1) The bit in special registers D9116 to D9123 that corresponds to the module for which the verification error occurred will be set to "1": check for the module whose bit is set to "1" by monitoring these registers with a peripheral device and replace that module. (2) If the current arrangement of loaded modules is acceptable, reset with the reset switch.
"FUSE BREAK OFF" (When an END instruction is executed. However, no check is performed when M9084 or M9094 is ON.)	32	RUN (Stopped)	There is an output module with a blown fuse.	(1) Check the blown fuse indicator LEDs of the output modules and replace the fuse of the module whose indicator LED is lit. (2) Modules with blown fuses can also be detected by using a peripheral device. The bit in special registers D9100 to D9107 that corresponds a module whose fuse has blown will be set to "1": monitor these registers to check.
"CONTROL-BUS ERR." (When FROM, TO instruction are executed. On switching on the power or resetting. On switching from {STOP, PAUSE} to {RUN, STEP RUN})	40	Stopped	FROM, TO instructions cannot be executed. (1) Fault in the control bus to the special function module.	(1) There is a hardware fault of the special function module, CPU module, or base unit: replace each module/unit to find the defective one. Contact your nearest Mitsubishi service center, agent, or office, and explain the problem with the defective module/unit.
"SP.UNIT DOWN" (When FROM, TO instruction are executed. On switching on the power or resetting. On switching from {STOP, PAUSE} to {RUN, STEP RUN})	41	Stopped	On execution of a FROM, TO instruction, a special function module was accessed but no response was received. (1) The accessed special function module is faulty.	There is a hardware fault in the accessed special function module: contact your nearest Mitsubishi service center, agent, or office, and explain the problem.
"LINK UNIT ERROR" (On switching on the power or resetting. On switching from {STOP, PAUSE} to {RUN, STEP RUN})	42	Stopped	(1) A data link module for use with MELSECNET has been loaded at the master station.	(1) Remove the data link module for MELSECNET from the master station. After making this correction, reset and start operation from the initial status.
"I/O INT.ERROR" (When an interruption occurs.)	43	Stopped	An interruption has occurred although there is no interrupt module.	(1) There is a hardware fault in one of the modules: replace each module in turn to determine which one is defective. Contact your nearest Mitsubishi service center, agent, or office, and explain the problem with the defective module.
"SP.UNIT LAY.ERR." (On switching on the power or resetting. On switching from {STOP, PAUSE} to {RUN, STEP RUN})	44	Stopped	(1) Three or more computer link modules have been installed for one CPU module. (2) Two or more data link modules for MELSECNET have been installed. (3) Two or more interrupt modules have been installed. (4) In the parameter settings made at a peripheral device, an allocation for a special function module has been made where there is in fact an I/O module, or vice versa.	(1) Do not install more than two computer link modules. (2) Do not install more than one data link module for MELSECNET. (3) Install only one interrupt module. (4) Re-set the I/O allocations in the parameter settings made at the peripheral device so that they agree with the loaded modules.

Table 1.1 CPU Error Code List (Continued)

Error Message (When an A273UHCPU is Used)	Contents of Special Register D9008 (BIN Value)	CPU Status	Error Contents and Cause	Corrective Action
"SP.UNIT ERROR" (When a FROM, TO instruction is executed)	46	Stopped (RUN)	(1) A location where there is no special function module has been accessed (when the FROM, TO instruction was executed).	(1) Read the error step using a peripheral device, check the contents of the FROM, TO instruction at that step, and correct it using the peripheral device.
"LINK PARA.ERROR" (On switching on the power or resetting. On switching from {STOP PAUSE} to {RUN STEP RUN})	47	RUN	(1) The data written to the link parameter area when link range settings are made by parameter setting at a peripheral device differ for some reason from the parameter data read by the CPU. (2) The setting for the total number of slave stations is "0".	(1) Write the parameters again and check. (2) If the error is displayed again, there is a hardware fault. Contact your nearest Mitsubishi service center, agent, or office, and explain the problem.
"OPERATION ERROR" (When a command is executed)	50	RUN (Stopped)	(1) The result of BCD conversion is outside the stipulated range (max. 9999 or 99999999). (2) A setting exceeding the stipulated device range has been made and operation is therefore impossible. (3) A file register has been used in the program without having made a file register capacity setting.	(1) Read the error step with a peripheral device, and correct the program at that step. (Check the device setting range, BCD conversion value, etc.)
"BATTERY ERROR" (At any time (However, no check is performed when M9084 is ON.))	70	RUN	(1) The battery voltage has fallen below the stipulated value. (2) The battery's lead connector has not been installed.	(1) Replace the battery. (2) If the battery is used to back up the RAM memory or to retain memory contents during momentary power interruptions, install a lead connector.

APPENDIX 2 ERROR CODES STORED BY THE PCPU

The errors that are detected at the PCPU are servo program setting errors and positioning errors.

(1) Motion program setting errors
Motion program setting errors are errors as the results of checking a parameter block No. or an axis No. when executing SVST instructions.
When an error occurs, the following happens:
- The motion program setting error flag (M9079) comes ON.
- The program number of the program in which the error occurred is stored in the error program No. register (D9189).
- The error code is stored in the error point block No. register (D9195).
- The error code is stored in the error item information register (D9190).

(2) Positioning error
(a) Positioning errors are errors that occur when positioning starts or during positioning: they are classified into minor errors, major errors, and servo errors.
1) Minor errors............. These are errors generated by sequence programs or servo programs; they are assigned error codes 1 to 999.
The cause of minor errors can be eliminated by checking the error code and correcting the sequence program or servo program.
2) Major error.............. These are errors generated by external input signals or control commands from the SCPU; they are assigned error codes 1000 to 1999.
When a major error occurs, check the error code and eliminate the error cause in the external input signal status or sequence program.
3) Servo error .... These are errors detected by the servo amplifier; they are assigned error codes 2000 to 2999.
When a servo error occurs, check the error code and eliminate the error cause at the servo side.

(b) When an error occurs, the error detection signal for the relevant axis comes ON, and the error code is stored in the minor error code, major error code, or servo error code register.

<A172SHCPUN> Table 2.1 Error Code Registers, Error Flags

Device / Error Class	Error Code Register								Error Detection Signal
	Axis 1	Axis 2	Axis 3	Axis 4	Axis 1	Axis 2	Axis 3	Axis 4	
Minor error	D806	D826	D846	D866	D886	D906	D926	D946	M1607+20n
Major error	D807	D827	D847	D867	D887	D907	D927	D947	
Servo error	D808	D828	D848	D868	D888	D908	D928	D948	M1608+20n

<A171SHCPUN> Table 2.2 Error Code Registers, Error Detection Flags

Device / Error Class	Error Code Register				Error Detection Signal
	Axis 1	Axis 2	Axis 3	Axis 4	
Minor error	D806	D826	D846	D866	M1607+20n
Major error	D807	D827	D847	D867	
Servo error	D808	D828	D848	D868	M1608+20n

<A273UHCPU (32 axis feature)/A173UHCPU (S1)>

Table 2.3 Error Code Registers, Error Flags

Error Class \ Device	Error Code Register								Error Detection Signal
	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5	Axis 6	Axis 7	Axis 8	
Minor error	D6	D26	D46	D66	D86	D101	D126	D146	M2407+20n
Major error	D7	D27	D47	D67	D87	D107	D127	D147	
Servo error	D8	D28	D48	D68	D88	D108	D128	D148	M2408+20n

Error Class \ Device	Error Code Register								Error Detection Signal
	Axis 9	Axis 10	Axis 11	Axis 12	Axis 13	Axis 14	Axis 15	Axis 16	
Minor error	D166	D186	D206	D226	D246	D266	D286	D306	M2407+20n
Major error	D167	D187	D207	D227	D247	D267	D287	D307	
Servo error	D168	D188	D208	D228	D248	D268	D288	D308	M2408+20n

Error Class \ Device	Error Code Register								Error Detection Signal
	Axis 17	Axis 18	Axis 19	Axis 20	Axis 21	Axis 22	Axis 23	Axis 24	
Minor error	D326	D346	D366	D386	D406	D426	D446	D466	M2407+20n
Major error	D327	D347	D367	D387	D407	D427	D447	D467	
Servo error	D328	D348	D368	D388	D408	D428	D448	D468	M2408+20n

Error Class \ Device	Error Code Register								Error Detection Signal
	Axis 25	Axis 26	Axis 27	Axis 28	Axis 29	Axis 30	Axis 31	Axis 32	
Minor error	D486	D506	D526	D546	D566	D586	D606	D626	M2407+20n
Major error	D487	D507	D527	D547	D567	D587	D607	D627	
Servo error	D488	D508	D528	D548	D568	D588	D608	D628	M2408+20n

(c) If another error occurs after an error code has been stored, the existing error code is overwritten, deleting it.
However, it is possible to check the history of error occurrence by using a peripheral device started up with the GSV43P software.

(d) Error detection flags and error codes are latched until the error code reset signal (M1807+20n/M3207+20n) or servo error reset signal (M1808+20n/M3208+20n) comes ON.

POINTS

(1) When some servo errors occur, the same error code will be stored again even if the servo error reset signal (M1808+20n/M3208+20n: ON) is issued.

(2) When a servo error occurs, reset the servo error after first eliminating the error cause at the servo side.

Appendix 2.1 Motion Program Setting Errors

The error codes, error definitions and corrective actions for motion program setting errors are indicated in Table 2.4.

Table 2.4 Motion Program Setting Errors

Error Code Stored in D9190	Error Name	Definition	Error Processing	Corrective Action
1	Parameter block number setting error	The specified parameter block number is outside the range 1 to 16.	The motion program is executed with the parameter block number set to the default value of "1".	Specify the parameter block number in the range 1 to 16.
906	Axis number setting error	The axis not used in the system settings has been specified for the motion program set in the DSFRP/SVST instruction.	Positioning control does not start.	Set the axis number that was specified in the system settings.
3300	Start program excess error	An attempt was made to start and run 9 or more programs simultaneously with the DSFRP/SVST instruction.	Positioning control does not start.	Set up to 8 programs as the simultaneously run programs.

Appendix 2.2 Minor Errors

Minor errors are those that occur in the sequence program or servo program.
The error codes for these errors are from 1 to 999.
Minor errors include set data errors, positioning control start-up errors, positioning control errors, and control change errors.

(1) Set data errors (1 to 99)
These errors occur when the data set in the parameters for positioning control is not correct.
The error codes, causes, processing, and corrective actions are shown in Table 2.5 below.

Table 2.5 Set Data Error List (1 to 99)

Error Code	Data Where Error Occurred	Check Timing	Error Cause	Error Processing	Corrective Action
21	Home position return data	When count type, near-zero-point dog type, or data set type or home position return is started.	The home position address of a degree axis is outside the range 0 to 35999999 ($\times 10^{-5}$degrees).	Home position return is not started.	Set the home position address within the permissible range with a peripheral device.
22		When a count type, or near-zero-point dog type home position return is started.	The home position return speed is set outside the range of 1 to the speed limit value.		Set the home position return speed at or below the speed limit value by using a peripheral device.
23			The creep speed is set outside the range of 1 to the home position return speed.		Set the creep speed at or below the home position return speed by using a peripheral device.
24		When a count type home position return is started.	The travel value after the near-zero-point dog comes ON is outside the range of 0 to $2^{31}-1$($\times$ unit).		Set the travel value after the near-zero-point dog to within the permissible range with a peripheral device.
25		When a count type, near-zero-point dog type or home position return is started.	The parameter block No. is outside the range of 1 to the maximum No.		Set the parameter block No. within the permissible range with a peripheral device.
40	Parameter block	When interpolation control is started	The unit for interpolation control designated in the parameter block is different from the control unit designated in the fixed parameters.	Control is executed using the control unit designated in the fixed parameters.	Designate the same control unit in the fixed parameters and servo parameters.

POINT

Sometimes, if the interpolation control unit designated in the parameter block and the control unit designated in the fixed parameters are different, no error code is stored; this depends on the combination of units designated.
For details, see Section 6.6.6.

(2) Positioning control start-up errors (100 to 199)

The errors shown in this section are those detected when positioning control is started.

Error codes, causes, processing, and corrective actions are shown in Table 2.6 below.

*: When interpolation control is being executed, the error codes are stored in the error code storage areas of all the axes involved in the interpolation.

Table 2.6 Positioning Control Start-Up Error List (100 to 199)

Error Code	Control Mode				Error Cause	Error Processing	Corrective Action
	Positioning	JOG	Manual Pulse Generator	Home Position Return			
100	○	○	○	○	• The PC ready flag (M2000) or PCPU ready flag (M9074) is OFF.	Positioning control does not start.	• Set the servo system CPU to RUN. • Turn the PC ready flag (M2000) ON.
101	○	○	○	○	• The start accept flag (M2001 to M2008/M2001 to M2004) of the relevant axis has been turned ON.		• Provide an interlock in the program to prevent the axis from being started while in motion (use the turning OFF of the start accept signal for the axis as the interlock condition).
103	○	○	○	○	• The stop command (M1800+20n) of the relevant axis has been turned ON.		• Turn the stop command (M1800+20n) OFF and start positioning.
104	○	○	○	○	• The rapid stop command (M1801+20n) of the relevant axis has been turned ON.		• Turn the rapid stop command (M1801+20n) OFF and start positioning.
105	○				• On starting, the feed present value is outside the stroke limit range.		• Move back inside the stroke range using JOG operation. • Enter inside the stroke range by executing a home position return or present value change.
106*	○				• Positioning outside the stroke limit has been designated.		• Positioning end point must be within the specified stroke limit.
107	○				• An address that does not generate an arc was designated in circular interpolation for which an auxiliary point is designated. (Error in relationship between the start point, auxiliary point, and end point)		• Designate correct addresses in the servo program.
108*	○				• An address that does not make an arc was designated in circular interpolation for which a radius is designated. (Error in relationship between the start point, auxiliary point, and end point)		
109	○				• An address that does not generate an arc was designated in circular interpolation for which a center point is designated. (Error in relationship between the start point, auxiliary point, and end point)		
110*	○				• In circular interpolation, the difference between the end point address and the ideal end point exceeded the allowable error range for circular interpolation.		
115				○	• The home position return completed signal (M1610+20n) has been turned ON during a near-zero point dog type home position return operation.		• Resumptive starts are not possible for home position return operations. Use JOG operation or positioning operation to return the axis to a point before the near-zero point dog signal was output, then retry the home position return operation.

Table 2.6 Positioning Control Start-Up Error List (100 to 199) (Continued)

Error Code	Control Mode: Positioning	JOG	Manual Pulse Generator	Home Position Return	Error Cause	Error Processing	Corrective Action
116		○			• The set JOG speed is 0.	Positioning control does not start.	• Set a correct speed (within the specified range).
					• The set JOG speed exceeds the JOG speed limit value.	Control is executed at the JOG speed limit value.	
117		○			• Both forward and reverse motion were designated when simultaneously starting JOG operation programs.	Only the axis set to move in the forward direction starts.	• Set correct data.
120				○	ZCT not set During second travel in dog type or count type home position return, or when data set type home position return is started, the zero pass signal (M1606+20n) is OFF.	Home position return is not completed correctly.	• Carry out the home position return after the home position has been passed.
140	○				• In linear interpolation for which a reference axis is designated the travel value of the reference axis is set at "0".	Positioning control does not start.	• Do not set an axis whose travel value is 0 as the reference axis.
142				○	• An external input signal has come ON although external input signal setting has not been performed for that signal in the system settings.		• Perform external input signal setting in system setting.
160	○				• The operating axis is specified in the SVST instruction.		• Start after the operating signal has turned OFF. Provide an SVST instruction operating interlock.
161	○				• An attempt was made to start the program whose number is outside the range 1 to 256.		• Reconsider the SVST instruction.
163	○				• The sequence number specified in SVST is outside the range 0 to 9999.	Positioning control starts from the beginning of the program.	• Set the sequence number within the range 0 to 9999.
190	○				• At a start, the override ratio is outside the range 0 to 100%.	Operation is performed at 100%.	• Set the override ratio within the range 0 to 100%.

(3) Positioning control errors (200 to 299)
The errors shown in this section are those detected during positioning control.
Error codes, causes and corrective actions are shown in Table 2.7.

Table 2.7 Positioning Control Start-Up Error List (200 to 299)

Error Code	Control Mode: Positioning	Control Mode: JOG	Control Mode: Manual Pulse Generator	Control Mode: Home Position Return	Error Cause	Error Processing	Corrective Action
200	○	○	○		• The PC ready flag (M2000) was turned OFF while positioning was being started in response to a start request issued by a sequence program.	Axis motion decelerates to a stop.	• Turn the PC ready flag (M2000) ON after all axes have stopped.
201				○	• The PC ready flag (M2000) was turned OFF during a home position return operation.		• After turning the PC ready flag (M2000) ON or turning the stop command (M1800+20n) or rapid stop command (M1801+20n) OFF, re-attempt home position return. (In the case of a near-zero-point dog type home position return, use JOG operation or positioning operation to return the axis to the point before the near-zero-point dog signal was output, and re-attempt home position return.)
202				○	• The stop command (M1800+20n) has been turned ON during a home position return operation.		
203				○	• The rapid stop command (M1801+20n) has been turned ON during a home position return operation.	Axis motion stops immediately.,	
204	○	○	○	○	• The PC ready flag (M2000) was turned back ON during deceleration initiated by turning OFF the PC ready flag (M2000).	No processing	• Turn the PC ready flag (M2000) ON after all axes have stopped. (Turning ON the PC ready flag (M2000) during deceleration is ignored.)
206				○	• While a home position return operation was in progress, an emergency stop was executed in the test mode at a peripheral device by pressing the [Back Space] key.	Axis motion stops immediately.	• In the case of a near-zero point dog type home position return, use JOG operation or positioning operation to return the axis to the point before the near-zero point dog signal was output, and re-attempt home position return. • If the near-zero point dog signal is turned OFF when executing a count type home position return, use JOG operation or positioning operation to return the axis to the point before the near-zero point dog signal was output, and re-attempt home position return. (In the near-zero-point dog signal is turned ON when executing count type home position return, re-attempt the home position return.)
207	○	○			• The feed present value exceeded the stroke limit during positioning. In the case of circular interpolation, an error code is stored only for axes whose feed present value exceeded the stroke limit. In the case of linear interpolation, error codes are stored for all axes involved in the interpolation.	Axis motion decelerates to a stop.	• Correct the stroke limit or travel value setting so that positioning is executed within the stroke limit.
208	○		○		• During circular interpolation or during simultaneous operation of multiple manual pulse generators, the feed present value of another axis exceeded the stroke limit value. (For detection of other axis errors).		
209				○	• An overrun has occurred because the set travel value exceeds the deceleration distance when a speed/position change (CHANGE) signal is input during speed/position switching control, or when the near-zero-point dog signal is input during count type home position return.		• Correct the speed setting so that overrun does not occur. • Set a travel value which will not cause an overrun.

Table 2.7 Positioning Control Error List (200 to 299) (Continued)

Error Code	Control Mode: Positioning	Control Mode: JOG	Control Mode: Manual Pulse Generator	Control Mode: Home Position Return	Error Cause	Error Processing	Corrective Action
211	○				• During positioning, an overrun occurs because the deceleration distance for the output speed is not attained at the point where the final positioning address is detected.	Axis motion decelerates to a stop.	• Set a speed at which overrun does not occur. • Set a travel value which will not cause an overrun.
214			○		• An attempt was made to control an axis already being moved by the manual pulse generator by setting the manual pulse generator operation enable flag for that axis.	The manual pulse generator input is ignored until the axis stops.	• Perform the manual pulse generator operation after the axis has stopped.
290	○				• At a start, the override ratio is outside the range 0 to 100%.	Operation is performed at 100%.	• Set the override ratio within the range 0 to 100%.

(4) Errors occurring at speed changes and torque limit value changes (300 to 399)
The errors shown in this section are those that occur on execution of speed changes and torque limit value changes.
Error codes, causes, processing, and corrective actions are shown in table 2.8.

Table 2.8 List of Errors that Occur at Speed Changes and Torque Limit Value Changes

Error Code	Control Mode: Positioning	Control Mode: JOG	Control Mode: Manual Pulse Generator	Control Mode: Home Position Return	Error Cause	Error Processing	Corrective Action
301				○	• An attempt was made to change the speed of an axis executing a home position return.	The speed is not changed.	• The speed of an axis executing a home position return cannot be changed.
303	○				• An attempt was made to change the speed of an axis after automatic deceleration had started in positioning.		• The speed of an axis cannot be changed after automatic deceleration has started.
304		○			• An attempt was made to change the speed of an axis during deceleration initiated by turning OFF the JOG operation start signal (M1802+20n, M1803+20n).		• Do not attempt a speed change during deceleration initiated by turning OFF the JOG operation start signal (M1802+20n, M1803+20n).
305		○			• The speed to be changed to in a speed change was set outside the range of 0 to the speed limit value.	The speed is kept at the speed limit value.	• Set the speed within the range from 0 to the speed limit value.
	○				• The absolute value of speed to be changed to in a speed change was set outside the range of 0 to the speed limit value.		• Set the absolute value of speed within the range from 0 to the speed limit value.
310					• A speed change was attempted during high-speed oscillation. • A speed change to "0" request was issued during high-speed oscillation.	The speed is not changed.	• Do not perform speed changes during high-speed oscillation.
311					• A value outside the range 1 to 500% was set in the torque limit value change request (CHGT).	The torque limit value is not changed.	• Make a change request within the range 1 to 500% .
312					• A torque limit change request (CHGT) was made for an axis not started yet.		• Make a change request for a started axis.

(5) Motion program running errors (500 to 599)
These errors are detected during motion program execution.
Check the executed motion program number, executed sequence number and executed block number, and correct the motion program.
Table 2.9 lists the processings and corrective actions for motion program running errors.

Table 2.9 Motion Program Running Error (500 to 599) List

Error Code	Control Mode: Positioning	Control Mode: JOG	Control Mode: Manual Pulse Generator	Control Mode: Home Position Return	Error Cause	Error Processing	Corrective Action
500	○				• 0 is specified as the N number.		• Set the N number of a sequence program within the range 1 to 9999.
501	○				• There is no F command. Speed is "0".	Deceleration to stop	• Specify F before and during execution of G01, G02, G03. Specify the speed of "1" or higher.
502	○				• The command value is greater than the range.		• Set the address, speed, dwell time, etc. within the ranges.
503	○				• The speed command specified is greater than the speed limit value of the parameter block.	Speed is clamped at speed limit value for operation.	• Set the correct speed (within the range).
504	○				• 5 or more axes were specified in 1 block.		• 5 or more axes cannot be interpolated. Set the number of interpolation axes up to 4 axes.
510	○				• Unauthorized G code was specified.		• Specify the correct G code.
513	○				• The interpolation length is greater than the range.		• Specify the axis address within the range.
525	○				• Subprogram level excess. Subprogram calling depth exceeded 8 levels.		• Set the calling depth within 8 levels.
530	○				• Arithmetic expression is not correct.		• Use a correct arithmetic expression.
531	○				• Integer value overflow. The integer value exceeded the range during arithmetic operation.		• Reconsider the variable value and arithmetic expression.
532	○				• The numbers of "[" and "]" specified in one block differ.		• Set the numbers of "[" and "]" in pairs.
533	○				• The denominator of division is 0.		• Set the denominator to other than 0.
535	○				• The IF [condition] GOTO statement is in error.		• Reconsider the IF statement.
536	○				• The variable number exceeds the range.		• Set the variable within the range.
537	○				• The variable definition statement does not have "=".	Deceleration to stop	• Add "=".
541	○				• The sequence number specified for subprogram call, return from subprogram or GOTO is not set.		• Set the sequence number.
542	○				• In the specified motion program, the WHILE[]DOm-ENDm statement is in error.		• Reconsider the motion program.
543	○				• In the specified motion program, the nesting of the DOm-ENDm statement is greater than the limit.		
544	○				• In the specified motion program, DOm-ENDm are not in pairs.		
545	○				• In the specified motion program, the IF[]THENm-ENDm statement is in error.		
546	○				• In the specified motion program, the nesting of the IF[]THENm-ENDm statement is greater than the limit.		
547	○				• In the specified motion program, IF[]THENm, ELSEm and ENDm are not in pairs.		
555	○				• At a subprogram call, the specified subprogram is not registered.		• Create the specified subprogram. Change the call number.

Table 2.9 Motion Program Running Error (500 to 599) List (Continued)

Error Code	Control Mode: Positioning	JOG	Manual Pulse Generator	Home Position Return	Error Cause	Error Processing	Corrective Action
560	○				• The command format in the motion program is not correct.		• Reconsider the motion program. Reconsider the argument following G**.
562	○				• There is no M02/M30 at the end of the motion program. There is no M99 at the end of the subprogram.		• Put M02, M30 or M99 before %.
570	○				• For the tool length offset (G43, G44) command, the offset data number is not specified. The offset data number is not correct.	Deceleration to stop	• Reconsider the offset data number.
571	○				• For the tool length offset (G43, G44) or tool offset cancel (G49) command, the axis corresponding to compensation is not specified.		• Specify the axis corresponding to compensation.
580	○				• The command beyond the preset stroke range was executed.		• Give the command within the preset stroke range.
581	○				• The move command was given to the high-speed oscillation operation axis.		• Do not give the move command to the high-speed oscillation operation axis.
582	○				• High-speed oscillation cancel was given to the axis which was not operating in high-speed oscillation.	No processing	• High-speed oscillation cancel is invalid.
584	○				• Cancel start (G24) program number error		• Reconsider the motion program number.
585	○				• High-speed oscillation (G25) amplitude range error		• Reconsider the high-speed oscillation (G25) amplitude range.
586	○				• High-speed oscillation (G25) starting angle range error		• Reconsider the high-speed oscillation (G25) starting angle range.
587	○				• High-speed oscillation (G25) frequency range error		• Reconsider the high-speed oscillation (G25) frequency range.
591	○				• A fault occurred in the system.	Deceleration to stop	• Consult your sales representative.
592	○				• The axis name is incorrect.		• Use X, Y, Z, U, V, W, A, B, C. Match the axis name with the one in the system settings.
593	○				• 0 number designated in the specified motion program is incorrect.		• Reconsider the 0***; part.
594	○				• The axis not specified in SVST is specified in the motion program.		• Reconsider the SVST instruction. Reconsider the motion program.

(6) System errors (900 to 999)

Table 2.10 System Error List (900 to 999)

Error Code	Control Mode: Positioning	JOG	Manual Pulse Generator	Home Position Return	Error Cause	Error Processing	Corrective Action
900					• When the servo amplifier power is switched ON, the motor type set in the "system settings" differs from the motor type actually installed. (Checked only when using MR-J2-B)	Further operation is impossible.	• Correct the motor type setting in the system settings.
901					• When the servo amplifier power is switched ON, the motor travel value while the power was OFF is found to have exceeded the "Power OFF Allowed Traveling Points" setting made in the system settings.		• Check the position. Check the encoder battery.

Appendix 2.3 Major Errors

Major errors are caused by external input signals or by control commands from the SCPU. The error codes for major errors are 1000 to 1999.
Major errors consist of control start-up errors, positioning errors, absolute system errors, and system errors.

(1) Positioning control start-up errors (1000 to 1099)

The errors shown in this section are those detected when positioning control is started.

Error codes, error causes, error processing and corrective actions are shown in Table 2.11.

Table 2.11 Positioning Control Start-Up Error List (1000 to 1099)

Error Code	Control Mode: Positioning	Control Mode: JOG	Control Mode: Manual Pulse Generator	Control Mode: Home Position Return	Error Cause	Error Processing	Corrective Action
1000	○	○	○	○	• The external stop signal of the corresponding axis was turned ON.	Positioning control does not start.	• Turn OFF the STOP signal.
1001		○	○	○	• When positioning was started in the forward direction (addresses increasing), the external FLS (upper limit LS) signal was turned OFF.		• Move the axis in the reverse direction in the JOG mode until it enters the external limit range.
1002		○	○	○	• When positioning was started in the reverse direction (addresses decreasing), the external RLS (lower limit LS) signal was turned OFF.		• Move the axis in the forward direction in the JOG mode until it enters the external limit range.
1003				○	• When near-zero point type home position return was started, the external DOG (near-zero point dog) signal was turned ON.		• Move the axis to a point before the near-zero point dog in the JOG mode and then execute a home position return.
1004	○	○	○	○	• The servo state of the corresponding axis is not servo READY. (M1615+20n: OFF). (1) The power supply to the servo amplifier is OFF. (2) Initial processing is in progress after turning on the servo amplifier. (3) The servo amplifier has not been installed. (4) A servo error has occurred. (5) Cable fault.		• Wait until the servo status is READY (M1615+20n: OFF).
1005	○	○	○	○	• The servo error detection signal of the corresponding axis (M1608+20n) was turned ON.		• Eliminate the error at the servo side, reset the servo error detection signal (M1608+20n) by using the servo error reset command (M1808+20n), then start operation.

(2) Positioning control errors (1100 to 1199)
The errors shown in this section are those detected during positioning.
Error codes, error causes, error processing, and corrective actions are shown in Table 2.12.

Table 2.12 Positioning Control Error List (1100 to 1199)

<table>
<tr><th rowspan="2">Error Code</th><th colspan="4">Control Mode</th><th rowspan="2">Error Cause</th><th rowspan="2">Error Processing</th><th rowspan="2">Corrective Action</th></tr>
<tr><th>Positioning</th><th>JOG</th><th>Manual Pulse Generator</th><th>Home Position Return</th></tr>
<tr><td>1101</td><td>○</td><td>○</td><td>○</td><td>○</td><td>• When positioning was started in the forward direction (addresses increasing), the external FLS (upper limit LS) signal was turned OFF.</td><td rowspan="3">Axis motion decelerates to a stop in accordance with the "deceleration processing on STOP input" setting in the parameter block.</td><td>• Move axis in the reverse direction in the JOG mode until it enters the external limit range.</td></tr>
<tr><td>1102</td><td>○</td><td>○</td><td>○</td><td>○</td><td>• When positioning was started in the reverse direction (addresses decreasing), the external RLS (lower limit LS) signal was turned OFF.</td><td>• Move the axis in the forward direction in the JOG mode until it enters the external limit range.</td></tr>
<tr><td>1103</td><td></td><td></td><td></td><td>○</td><td>• The external STOP signal (stop signal) was turned ON while the axis was moving.</td><td>• When executing a near-zero point dog type home position return, move the axis to a point before the near-zero point dog in the JOG mode and then execute a home position return.</td></tr>
<tr><td>1104</td><td>○</td><td>○</td><td>○</td><td>○</td><td>• The servo error detection signal (M1608+20n) was turned ON while an axis was in motion.</td><td>The axis stops immediately without decelerating.</td><td>• After taking the appropriate corrective action for the servo error, the axis can be restarted.</td></tr>
<tr><td>1105</td><td>○</td><td>○</td><td>○</td><td>○</td><td>• The power supply to the servo amplifier was turned OFF while an axis was in motion. (Servo not installed status detected, cable fault, etc.)</td><td>M1615+20n turned OFF.</td><td>• Turn ON the power supply to the servo amplifier.
• Check the cable to servo amplifier connecting cable.</td></tr>
</table>

(3) Absolute System Errors (1200 to 1299)
The errors shown in this section are those detected in an absolute system.
Error codes, error causes, error processing, and corrective actions are shown in Table 2.13.

Table 2.13 Absolute System Error List (1200 to 1299)

Error Code	Control Mode: Positioning	Fixed-Pitch Feed	Speed	Speed/Position Switching	Speed Switching	Constant Speed	JOG	Manual Pulse Generator	Home Position Return	Position Follow-Up Control	OSC	Error Cause	Error Processing	Corrective Action
1201												• When the servo amplifier power was switched ON, a sum check error occurred with the backup data in the controller. • Home position return has not been performed. • CPU module battery error. • Home position return has been performed, but not completed.	Home position return request ON	• Check the battery of the CPU module and execute a home position return.
1202*												• When the servo amplifier power is turned ON, a communication error in communication between the servo amplifier and encoder occurs.	Home position return request ON, servo error 2016 set.	• Check the motor and encoder cables and perform home position return again.
1203*												• During operation, the amount of change in the encoder present value complies with the following expression: "Amount of change in encoder present value/3.5 ms > 180° of motor revolution" After the servo amplifier power has been turned ON, a continual check is performed (in both servo ON and OFF states).	No Processing	• Check the motor and encoder cables.
1204*												• During operation, the following expression holds: "Encoder present value (PLS) ≠ feedback present value (PLS) (encoder effective bit number)". After the servo amplifier power has been turned ON , a continual check is performed (in both servo ON and OFF states).		

*: These errors occur only when using MR-H-B and MR-J2-B servo amplifiers.

(4) System errors (1300 to 1399/1500 to 1599)
These are errors which are detected at power-on.
Table 2.14 lists the error codes, error causes, error processings and corrective actions.

Table 2.14 Main Base Side (1300 to 1399/1500 to 1599) List

Error Code	Control Mode: Positioning	JOG	Manual Pulse Generator	Home Position Return	Error Cause	Error Processing	Corrective Action
1310					• Initial communication with the servo system CPU is not completed normally. • Servo system CPU fault	Positioning control does not start.	• Change the servo system CPU.

Appendix 2.4 Servo Errors

The servo errors include the servo amplifier errors and servo power supply module errors.
You can set to each line the processings to be performed on detection of servo errors. (Only the servo errors detected by the ADU (when A273UHCPU is used))
Specify the processings and lines in the system settings of the peripheral device.

	Setting	Control
1	Line-by-line servo OFF (default)	• When a servo error has occurred in any of the ADU axes, all axes in that line are put in servo OFF status. (Control exercised is the same as at all-axis servo OFF.)
2	Only own axis servo OFF	• Only the ADU axis where a servo error has occurred is placed in servo OFF status and no influence is given to the other axes. • However, note that: 1) For the 2 axes/1 module type, both axes are put in servo OFF status if a servo error has occurred in one axis. 2) The line-by-line servo OFF status is established if any of the following servo errors occurs. Overcurrent (2032) Undervoltage (2810) Excessive regeneration (2830) Overvoltage (2833) Amplifier power supply overheat (2847)

(1) Servo amplifier errors (2000 to 2799)
The servo amplifier errors are detected by the servo amplifier and assigned error codes 2000 to 2799.
The servo errors include errors in the ADU and errors in the MR-□-B.
For the servo amplifier types, the ADU is abbreviated to Ⓐ and the MR-□-B to Ⓜ.
When any of the servo amplifier errors occurs, the servo error detection signal (M2408+20n) turns ON. Eliminate the error cause and turn ON the servo error reset (M3208+20n) to reset the servo error, and make a restart. (However, the servo error detection signal will not turn ON for any of the error codes 2100 to 2499 as they are warning.)
Note: 1. For regenerative alarm protection (error code 2030) and overload protection 1, 2 (error code 2050, 2051), the status when the protective circuit was activated is still retained in the servo amplifier after activation. The data stored is cleared when the external power is switched OFF, but is not cleared by the RESET signal.
2. If the external power is switched OFF repeatedly to reset any of the error codes 2030, 2050 and 2051, overheat may lead to damage to the devices. Therefore, resume operation after removing the cause without fail.
The servo error definitions are given in Table 2.15.

⚠ CAUTION

⚠ If a controller or servo amplifier self-diagnostic error has occurred, make check in accordance with this manual and restore to normal.

Table 2.15 Servo Amplifier Error (2000 to 2799) List

Error code	Amplifier Type	Error Cause		Error Check Timing	Processing	Corrective Action
		Name	Definition			
2010	A	P-N non-wiring	• P-N of the servo power supply module are not wired to P-N of the ADU.	Any time	Imme-diate stop	• Reconsider wiring.
	M	Undervoltage	• The power supply voltage is less than 160VAC.			• Measure the input voltage (R, S, T) with a voltmeter.
			• Instantaneous power failure occurred for longer than 15msec.			• On an oscilloscope, check for an instantaneous power failure.
			• Due to power supply capacity shortage, the power supply voltage dropped at a start or the like.			• Reconsider the power supply capacity.
2012	A	Internal memory alarm	• ADU's SRAM fault.	• At power-on of servo amplifier		• Change the ADU.
	M	Memory alarm 1	• Servo amplifier's SRAM is faulty. • Servo amplifier's EPROM checksum does not match	• At power-on of servo amplifier • On PC ready (M2000) leading edge • At servo error reset • At power-on of servo system CPU		• Change the servo amplifier.
2013	M	Clock alarm	• Servo amplifier's clock is faulty.	Any time		• Change the servo amplifier.
2014	A	Watchdog	• Servo control system fault.			• Reset and recheck the servo system CPU.
			• ADU fault.			• Change the ADU.
	M		• Servo amplifier hardware is faulty. • Servo system CPU hardware is faulty.			• Change the servo amplifier. • Change the servo system CPU.
2015	A	2-port memory alarm	• ADU's 2-port memory fault.	• At power-on of servo amplifier • At servo error reset		• Reset and recheck the servo system CPU. • Change the ADU.
	M	Memory alarm 2	• Servo amplifier's EEPROM is faulty.	• At power-on of servo amplifier • On PC ready (M2000) leading edge • At servo error reset • At power-on of servo system CPU		• Change the servo amplifier.
2016	A	Detector alarm 1	• At initialization, communication with encoder is not normal. • The encoder type (ABS/INC) set in system settings differs from the actual encoder type.	• At power-on of servo amplifier • At servo error reset		• Reset and recheck the servo system CPU. • Change the servo motor (encoder). • Reconsider the system settings.
	M		• Communication with encoder is in error.	• At power-on of servo amplifier • On PC ready (M2000) leading edge • At servo error reset • At power-on of servo system CPU		• Check the detector cable connector for disconnection. • Change the servo motor. • Change the detector cable. • Check the combination of detector cable type (2-wire/4-wire type) and servo parameter.

Table 2.15 Servo Amplifier Error (2000 to 2799) List (Continued)

Error Code	Amplifier Type	Error Cause		Error Check Timing	Processing	Corrective Action
		Name	Definition			
2017	A	Board alarm	• ADU's analog-to-digital converter is faulty.	• At power-on of servo amplifier • At servo error reset	Immediate stop	• Reset and recheck the servo system CPU. • Change the ADU.
	M		• Device on the servo amplifier board is faulty.	• At power-on of servo amplifier • On PC ready (M2000) leading edge • At servo error reset • At power-on of servo system CPU		• Change the servo amplifier.
2019	M	Memory alarm 3	• Servo amplifier's flash ROM checksum does not match.	• At power-on of servo amplifier • On PC ready (M2000) leading edge • At servo error reset • At power-on of servo system CPU		• Change the servo amplifier.
2020	A	Detector alarm 2	• During operation, communication with the encoder is not normal.	Any time		• Check wiring between the encoder and ADU. • Change the servo motor (encoder).
	M	Detector alarm 2	• Communication with the encoder is in error.			• Check the detector cable connector for disconnection. • Change the servo motor. • Change the detector cable.
2024	M	Output side ground fault	• U, V or W of the servo amplifier is in ground fault.			• Use a multimeter to check across U, V, W terminals and earth. • Use a multimeter and megger to check across U, V, W terminals and core.
2025	A	Absolute position erase	• In the absolute value encoder, the voltage of the super capacitor in the encoder is less than 2.5±0.2V. • In the absolute value encoder, speed was 500rpm or higher during a power failure.	• At power-on of servo amplifier • At servo error reset		• Change the battery (MR-JBAT-□). • Check the wiring between encoder and ADU.
	M	Battery alarm	• Reduction in the voltage of the super capacitor in the absolute value encoder	• At power-on of servo amplifier • On PC ready (M2000) leading edge • At servo error reset • At power-on of servo system CPU		• Switch power on for a few minutes, charge the super capacitor, then switch power from OFF to ON, and make home position setting.
			• Battery voltage reduction.			• After powering off the servo amplifier, measure the battery voltage.
			• Battery cable or battery fault. (After deactivating the error, home position return must be made again.)			• Change the servo amplifier battery.
2026	A	Module mismatch	• The servo parameter (system settings) does not match the actual servo amplifier.	• At power-on of servo amplifier • At servo error reset		• Reconsider the system settings.

Table 2.15 Servo Amplifier Error (2000 to 2799) List (Continued)

Error Code	Amplifier Type	Error Cause: Name	Error Cause: Definition	Error Check Timing	Processing	Corrective Action
2030	M	Excessive regeneration	• The ON/OFF frequencies of the regenerative power transistor are too high. (Be careful as the regenerative brake resistor may overheat.)	Any time	Immediate stop	• Check the regenerative level (%) of the servo monitor and reduce the acceleration/deceleration frequencies or feedrate. • Decrease the load. • Increase the servo motor capacity.
			• Servo parameter (system settings) setting mistake.			• Check the servo parameters (regenerative brake resistor and motor type set in system settings).
			• Regenerative brake resistor wiring mistake.			• Connect the regenerative brake resistor properly.
			• Regenerative brake resistor fault.			• Change the regenerative brake.
			• The regenerative power transistor was damaged in short circuit status.			• Change the servo amplifier.
2031	A	Overspeed	• The command speed is too high.			• Reconsider the command speed.
			• Overshoot occurred during acceleration.			• Reconsider the servo parameter.
			• Encoder fault.			• Change the encoder.
			• Encoder cable fault or wiring mistake.			• Check the wiring between encoder and ADU.
	M		• The motor speed is higher than 115% of the rated speed.			• Check the motor speed in the servo parameter. • Check whether the number of pulses per revolution and the travel per revolution in the fixed parameters match the machine specifications.
			• The acceleration/deceleration time constant is too small, resulting in overshoot.			• If overshoot occurs during acceleration/deceleration, check the acceleration and deceleration times in the fixed parameters.
			• The servo system is instable to cause overshoot.			• If overshoot occurs, adjust the position loop gain/position control gain 1, 2, speed loop gain/speed control gain 1, 2 in the servo parameters or increase the speed integral compensation.
			• Detector fault.			• Check the detector cable for wire breakage. • Change the servo motor.
2032	A	Overcurrent	• The servo motor connected is not as set.	• At power-on of servo amplifier • At servo error reset		• Reconsider the system settings.
			• The U, V, and W phases of the ADU output resulted in a short circuit or ground fault.			• Check the servo motor cable.
			• Wiring mistake of the U, V, and W phases of the ADU output.			• Correct the servo motor wiring.
			• Damage to the ADU's transistor module. • ADU fault.			• Change the ADU.
			• Coupling fault of servo motor and encoder.			• Change the servo motor.
			• The servo motor oscillated.			• Reconsider the servo parameters.

Table 2.15 Servo Amplifier Error (2000 to 2799) List (Continued)

Error Code	Amplifier Type	Error Cause: Name	Error Cause: Definition	Error Check Timing	Processing	Corrective Action
2032	M	Overcurrent	• U, V, and W of the servo amplifier output resulted in a short circuit.	Any time	Immediate stop	• Check U, V, and W of the servo amplifier output for a short circuit.
			• U, V, and W of the servo amplifier output resulted in a ground fault.			• Check U, V, and W of the servo amplifier output and the earth for a ground fault. Check U, V, and W of the servo amplifier output and the core for a ground fault. If a ground fault is found, change the servo amplifier and motor.
			• Wiring mistake of the U, V, and W phases of the servo amplifier output.			• Correct the wiring.
			• Damage to the servo amplifier transistor.			• Change the servo amplifier.
			• Coupling fault of servo motor and encoder.			• Change the servo motor.
			• Encoder cable fault.			• Change the encoder cable.
			• The servo motor connected differs from the setting.			• Check the connected motor in the system settings.
			• The servo motor oscillated.			• Check and adjust the gain settings in the servo parameters.
			• Noise entered the overcurrent detection circuit.			• Check for the actuated relay or valve in the peripheral.
2033	M	Overvoltage	• The converter bus voltage exceeded 400V. • The acceleration frequency was too high and exceeded the regenerative capability. • Regenerative brake resistor connection mistake.			• Increase the acceleration and deceleration times in the fixed parameters. • Check connection across C-P of the regenerative terminal block.
			• The regenerative brake resistor in the servo amplifier is dead.			• Measure the voltage across C-P of the regenerative terminal block with a multimeter. If the voltage is abnormal, change the servo amplifier. (Make measurement about 3 minutes after the charge lamp has gone off.)
			• The regenerative power transistor has been damaged.			• Change the servo amplifier.
			• The power supply voltage is high.			• Measure the input voltage (R, S, T) with a voltmeter.
2034	M	Communication alarm	• Receive data from the servo system CPU is in error.			• Check the motion bus cable. • Check the motion bus cable for wire breakage. • Check whether the motion bus cable is clamped properly.
2035	A	Data alarm	• The command speed is too high.			• Reconsider the command speed.
	A		• Servo system CPU fault.			• Change the servo system CPU.
	M		• The position command variation from the servo system CPU is too large or the command speed is too high.			• Check the command speed and the number of pulses per revolution and travel per revolution in the fixed parameters.
	M		• Noise entered the command from the servo system CPU.			• Check connection of the motion bus cable connector. • Check the motion bus cable for wire breakage. • Check whether the motion bus cable is clamped properly. • Check for the actuated relay or valve in the peripheral.

Table 2.15 Servo Amplifier Error (2000 to 2799) List (Continued)

Error Code	Amplifier Type	Error Cause Name	Error Cause Definition	Error Check Timing	Processing	Corrective Action
2036	A	Transfer alarm	• Servo system CPU fault.	Any time	Imme-diate stop	• Change the servo system CPU.
2036	M	Transfer alarm	• Communication with the servo system CPU is in error.	Any time	Imme-diate stop	• Check connection of the motion bus cable connector. • Check the motion bus cable for wire breakage. • Check whether the motion bus cable is clamped properly.
2042	M	Feedback alarm	• Encoder signal is in error.	Any time	Imme-diate stop	• Change the servo motor.
2045	A	Amplifier fin overheat	• The ADU fan is at a stop.	Any time	Imme-diate stop	• Change the ADU fan.
2045	A	Amplifier fin overheat	• The continuous output current of the ADU is exceeded.	Any time	Imme-diate stop	• Reduce the load.
2045	A	Amplifier fin overheat	• ADU's thermal sensor fault.	Any time	Imme-diate stop	• Change the ADU.
2045	M	Fin overheat	• The heat sin in the servo amplifier is overheated. • Amplifier fault (rated output excess). • Power ON and OFF are repeated in an overload status. • Cooling fault.	Any time	Imme-diate stop	• If the effective torque of the servo motor is large, reduce the load. • Reduce the acceleration/deceleration frequencies. • Check whether the amplifier fan is at a stop. (MR-H150B or more) • Check for ventilation obstruction. • Check whether the temperature in the panel is proper (0 to +55°C). • Check whether the electromagnetic brake is operated externally during operation. • Change the servo amplifier.
2046	A	Servo motor overheat	• The thermal protector built in the servo motor malfunctioned.	Any time	Imme-diate stop	• Change the servo motor.
2046	A	Servo motor overheat	• The continuous output of the servo motor is exceeded.	Any time	Imme-diate stop	• Reduce the load.
2046	M	Servo motor overheat	• The servo motor is overloaded.	Any time	Imme-diate stop	• If the effective torque of the servo motor is large, reduce the load.
2046	M	Servo motor overheat	• The servo motor and regenerative brake option are overheated.	Any time	Imme-diate stop	• Check the ambient temperature (0 to +40°C) of the servo motor.
2046	M	Servo motor overheat	• The thermal protector built in the encoder is faulty.	Any time	Imme-diate stop	• Change the servo motor.
2050	A	Overload	• The rated current of the servo motor is exceeded. • Load inertia or friction is too large.	Any time	Imme-diate stop	• Reduce the load.
2050	A	Overload	• Hunting due to parameter setting mistake.	Any time	Imme-diate stop	• Reconsider the servo parameters.
2050	M	Overload 1	• Overload current of about 200% flew continuously in the servo amplifier and servo motor.	Any time	Imme-diate stop	• Check for machine collision. • If the load inertia is extremely large, increase the acceleration/deceleration time constant or reduce the load. • If hunting has occurred, adjust the position loop gain in the servo parameter. • Check the U, V, W connections of the servo amplifier and servo motor. • Check the detector cable for wire breakage. • Change the servo motor.

Table 2.15 Servo Amplifier Error (2000 to 2799) List (Continued)

Error Code	Amplifier Type	Error Cause: Name	Error Cause: Definition	Error Check Timing	Processing	Corrective Action
2051	M	Overload 2	• The servo amplifier and servo motor are overloaded near the maximum torque (more than 95% of the current limit value).	Any time	Immediate stop	• Check for machine collision. • If the load inertia is extremely large, increase the acceleration/deceleration time constant or reduce the load. • If hunting has occurred, adjust the position loop gain/position control gain 1, 2, speed loop gain/speed control gain 1, 2 in the servo parameters. • Check the U, V, W connections of the servo amplifier and servo motor. • Check the detector cable for wire breakage. • Change the servo motor. • If the bus voltage in the servo amplifier is low (the charge lamp is off), change the servo amplifier.
2052	A	Error excessive	• The deviation counter value exceeded the specified value. • Inertia is too large to make enough acceleration.			• Reconsider the servo parameters.
			• Encoder or cable fault.			• Change the encoder or cable.
	M		• A difference between servo amplifier command pulses and feedback pulses exceeded 80000 pulses.			• Check for machine collision. • Increase the acceleration/deceleration time constant. • Increase the position loop gain/position control gain 1, 2 in the servo parameters. • Check the detector cable for wire breakage. • Change the servo motor. • If the bus voltage in the servo amplifier is low (the charge lamp is off), change the servo amplifier.
2057	A	Hardware alarm	• ADU hardware fault.			• Change the ADU.
2086	M	RS232 communication alarm	• Parameter unit communication error		Continued	• Check the parameter unit cable for wire breakage. • Change the parameter unit.
2102	A	Battery warning	• The absolute value encoder battery voltage dropped.			• Change the battery (MR-JBAT-□).
	M		• The voltage of the battery loaded in the servo amplifier dropped.			• Change the battery.
2103	M	Open battery cable warning	• The power supply voltage supplied to the absolute position detector dropped.			• Change the battery. • Check the detector cable for wire breakage. • Change the servo motor. • Change the servo amplifier.
2140	M	Excessive regeneration warning	• An excessive regeneration error (2030) may occur. (The 85% level of the max. load capacity was detected in the regenerative brake resistor)			• Refer to details of the excessive regeneration error (2030).
2141	A	Overload warning	• The 80% level of the overload error (2050) level was detected.			• Refer to details of the overload error (2050).
	M		• An overload error (2050, 2051) may occur. (85% level was detected)			• Refer to details of the overload error (2050, 2051).
2143	A	Absolute value counter warning	• Encoder fault.			• Change the encoder.
2146	M	Servo emergency stop	• 1A-1B (emergency stop input) of the servo amplifier connector CN6 were disconnected.			• Short 1A-1B of the servo amplifier connector CN6.

Table 2.15 Servo Amplifier Error (2000 to 2799) List (Continued)

Error Code	Amplifier Type	Error Cause		Error Check Timing	Processing	Corrective Action
		Name	Definition			
2147	A	Emergency stop	• Brought to an emergency stop.	Any time	Immediate stop	• Reset the emergency stop.
	M		• The emergency stop (EMG) signal is input from the servo system CPU.			
2149	M	Main circuit OFF warning	• The servo ON (SON) signal was turned ON when the contactor is OFF. • At not more than 50RPM, the main circuit bus voltage dropped to or below 215V.		Continued	• Turn ON the main circuit contactor or main circuit power.
2196	M	Home position setting error warning	• After the home position setting command is given, the droop pulse value did not fall within the in-position range.			• Make a home position return again.
2201 to 2224	A	Parameter warning	• The parameter that was set is unauthorized. 2201 Amplifier setting 2202 Motor type 2203 Motor capacity 2204 Number of feedback pulses 2205 In-position range 2206 Position control gain 2 (actual position gain) 2207 Speed control gain 2 (actual speed gain) 2208 Speed integral compensation 2209 Forward rotation torque limit value 2210 Reverse rotation torque limit value 2211 Emergency stop time delay 2212 Position control gain 1 (model position gain) 2213 Speed control gain 1 (model speed gain) 2214 Load inertia ratio 2215 Error excessive alarm level 2216 Special compensation processing 2217 Special servo processing 2218 Td dead zone compensation 2219 Feed forward gain 2220 Unbalance torque compensation 2221 Dither command 2222 Gain operation time 2223 Servo response level setting 2224 –			• Reconsider the system settings and servo parameters.

Table 2.15 Servo Amplifier Error (2000 to 2799) List (Continued)

Error Code	Amplifier Type	Error Cause: Name	Error Cause: Definition	Error Check Timing	Process-ing	Corrective Action
2301 to 2336	M	Parameter alarm	• The servo parameter value is outside the setting range. (Any unauthorized parameter is ignored and the value before setting is retained.) 2301 Amplifier setting 2302 Regenerative brake resistor 2303 Motor type 2304 Motor capacity 2305 Motor speed 2306 Number of feedback pulses 2307 Rotation direction setting 2308 Auto tuning setting 2309 Servo response level setting 2310 Forward rotation torque limit value 2311 Reverse rotation torque limit value 2312 Load inertia ratio 2313 Position control gain 1 2314 Speed control gain 1 2315 Position control gain 2 2316 Speed control gain 2 2317 Speed integral compensation 2318 Notch filter selection 2319 Feed forward gain 2320 In-position range 2321 Electromagnetic brake sequence output 2322 Monitor output mode selection 2323 Optional function 1 2324 Optional function 2 2325 Optional function 3 2326 Optional function 4 2327 Monitor output 1 offset 2328 Monitor output 2 offset 2329 Prealarm data selection 2330 Zero speed 2331 Error excessive alarm level 2332 Optional function 5 2333 Optional function 6 2334 PI-PID switching position droop 2335 Torque limit compensation factor 2336 Speed differential compensation (actual speed differential compensation)	Any time	Con-tinued	• Reconsider the setting ranges of the servo parameters.

Table 2.15 Servo Amplifier Error (2000 to 2799) List (Continued)

<table>
<tr><th rowspan="2">Error Code</th><th rowspan="2">Amplifier Type</th><th colspan="2">Error Cause</th><th rowspan="2">Error Check Timing</th><th rowspan="2">Process-ing</th><th rowspan="2">Corrective Action</th></tr>
<tr><th>Name</th><th>Definition</th></tr>
<tr><td>2301
to
2324</td><td>A</td><td>Parameter alarm</td><td>• The servo parameter value is outside the setting range. (Any unauthorized parameter is ignored and the value before setting is retained.)
<table>
<tr><td>2301</td><td>Amplifier setting</td></tr>
<tr><td>2302</td><td>Motor type</td></tr>
<tr><td>2303</td><td>Motor capacity</td></tr>
<tr><td>2304</td><td>Number of feedback pulses</td></tr>
<tr><td>2305</td><td>In-position range</td></tr>
<tr><td>2306</td><td>Position control gain 2 (actual position gain)</td></tr>
<tr><td>2307</td><td>Speed control gain 2 (actual speed gain)</td></tr>
<tr><td>2308</td><td>Speed integral compensation</td></tr>
<tr><td>2309</td><td>Forward rotation torque limit value</td></tr>
<tr><td>2310</td><td>Reverse rotation torque limit value</td></tr>
<tr><td>2311</td><td>Emergency stop time delay</td></tr>
<tr><td>2312</td><td>Position control gain 1 (model position gain)</td></tr>
<tr><td>2313</td><td>Speed control gain 1 (model speed gain)</td></tr>
<tr><td>2314</td><td>Load inertia ratio</td></tr>
<tr><td>2315</td><td>Error excessive alarm level</td></tr>
<tr><td>2316</td><td>Special compensation processing</td></tr>
<tr><td>2317</td><td>Special servo processing</td></tr>
<tr><td>2318</td><td>Td dead zone compensation</td></tr>
<tr><td>2319</td><td>Feed forward gain</td></tr>
<tr><td>2320</td><td>Unbalance torque compensation</td></tr>
<tr><td>2321</td><td>Dither command</td></tr>
<tr><td>2322</td><td>Gain operation time</td></tr>
<tr><td>2323</td><td>Servo response level setting</td></tr>
<tr><td>2324</td><td>–</td></tr>
</table></td><td>Any time</td><td rowspan="2">Con-tinued</td><td>• Reconsider the setting ranges of the servo parameters.</td></tr>
<tr><td>2500</td><td>A</td><td>Parameter alarm</td><td>• Among the servo parameters, any of the following items is unauthorized.
• Amplifier
• External regenerative brake resistor setting
• Motor type
• Motor capacity</td><td>• At power-on of servo amplifier
• At servo error reset</td><td>• Reconsider the system settings and servo parameters.</td></tr>
</table>

Table 2.15 Servo Amplifier Error (2000 to 2799) List (Continued)

<table>
<tr><th rowspan="2">Error Code</th><th rowspan="2">Amplifier Type</th><th colspan="2">Error Cause</th><th rowspan="2">Error Check Timing</th><th rowspan="2">Process-ing</th><th rowspan="2">Corrective Action</th></tr>
<tr><th>Name</th><th>Definition</th></tr>
<tr><td>2501
to
2524</td><td>A</td><td>Parameter alarm</td><td>• The parameter that was set is unauthorized.
<table>
<tr><td>2501</td><td>Amplifier setting</td></tr>
<tr><td>2502</td><td>Motor type</td></tr>
<tr><td>2503</td><td>Motor capacity</td></tr>
<tr><td>2504</td><td>Number of feedback pulses</td></tr>
<tr><td>2505</td><td>In-position range</td></tr>
<tr><td>2506</td><td>Position control gain 2
(actual position gain)</td></tr>
<tr><td>2507</td><td>Speed control gain 2
(actual speed gain)</td></tr>
<tr><td>2508</td><td>Speed integral compensation</td></tr>
<tr><td>2509</td><td>Forward rotation torque limit value</td></tr>
<tr><td>2510</td><td>Reverse rotation torque limit value</td></tr>
<tr><td>2511</td><td>Emergency stop time delay</td></tr>
<tr><td>2512</td><td>Position control gain 1
(model position gain)</td></tr>
<tr><td>2513</td><td>Speed control gain 1
(model speed gain)</td></tr>
<tr><td>2514</td><td>Load inertia ratio</td></tr>
<tr><td>2515</td><td>Error excessive alarm level</td></tr>
<tr><td>2516</td><td>Special compensation processing</td></tr>
<tr><td>2517</td><td>Special servo processing</td></tr>
<tr><td>2518</td><td>Td dead zone compensation</td></tr>
<tr><td>2519</td><td>Feed forward gain</td></tr>
<tr><td>2520</td><td>Unbalance torque compensation</td></tr>
<tr><td>2521</td><td>Dither command</td></tr>
<tr><td>2522</td><td>Gain operation time</td></tr>
<tr><td>2523</td><td>Servo response level setting</td></tr>
<tr><td>2524</td><td>–</td></tr>
</table></td><td>• At power-on of servo amplifier
• On PC ready (M2000) leading edge
• At servo error reset</td><td>Con-tinued</td><td>• Reconsider the system settings and servo parameters.</td></tr>
</table>

Table 2.15 Servo Amplifier Error (2000 to 2799) List (Continued)

<table>
<tr><th rowspan="2">Error Code</th><th rowspan="2">Amplifier Type</th><th colspan="2">Error Cause</th><th rowspan="2">Error Check Timing</th><th rowspan="2">Process-ing</th><th rowspan="2">Corrective Action</th></tr>
<tr><th>Name</th><th>Definition</th></tr>
<tr><td>2601
to
2636</td><td>M</td><td>Initial parameter alarm</td><td>• The parameter setting is wrong.
• The parameter data was corrupted.
<table>
<tr><td>2601</td><td>Amplifier setting</td></tr>
<tr><td>2602</td><td>Regenerative brake resistor</td></tr>
<tr><td>2603</td><td>Motor type</td></tr>
<tr><td>2604</td><td>Motor capacity</td></tr>
<tr><td>2605</td><td>Motor speed</td></tr>
<tr><td>2606</td><td>Number of feedback pulses</td></tr>
<tr><td>2607</td><td>Rotation direction setting</td></tr>
<tr><td>2608</td><td>Auto tuning setting</td></tr>
<tr><td>2609</td><td>Servo response level setting</td></tr>
<tr><td>2610</td><td>Forward rotation torque limit value</td></tr>
<tr><td>2611</td><td>Reverse rotation torque limit value</td></tr>
<tr><td>2612</td><td>Load inertia ratio</td></tr>
<tr><td>2613</td><td>Position control gain 1</td></tr>
<tr><td>2614</td><td>Speed control gain 1</td></tr>
<tr><td>2615</td><td>Position control gain 2</td></tr>
<tr><td>2616</td><td>Speed control gain 2</td></tr>
<tr><td>2617</td><td>Speed integral compensation</td></tr>
<tr><td>2618</td><td>Notch filter selection</td></tr>
<tr><td>2619</td><td>Feed forward gain</td></tr>
<tr><td>2620</td><td>In-position range</td></tr>
<tr><td>2621</td><td>Electromagnetic brake sequence output</td></tr>
<tr><td>2622</td><td>Monitor output mode</td></tr>
<tr><td>2623</td><td>Optional function 1</td></tr>
<tr><td>2624</td><td>Optional function 2</td></tr>
<tr><td>2625</td><td>Optional function 3</td></tr>
<tr><td>2626</td><td>Optional function 4</td></tr>
<tr><td>2627</td><td>Monitor output 1 offset</td></tr>
<tr><td>2628</td><td>Monitor output 2 offset</td></tr>
<tr><td>2629</td><td>Prealarm data selection</td></tr>
<tr><td>2630</td><td>Zero speed</td></tr>
<tr><td>2631</td><td>Error excessive alarm level</td></tr>
<tr><td>2632</td><td>Optional function 5</td></tr>
<tr><td>2633</td><td>Optional function 6</td></tr>
<tr><td>2634</td><td>PI-PID switching position droop</td></tr>
<tr><td>2635</td><td>Torque limit compensation factor</td></tr>
<tr><td>2636</td><td>Speed differential compensation (actual speed differential compensation)</td></tr>
</table></td><td>• At power-on of servo amplifier
• On PC ready (M2000) leading edge
• At servo error reset
• At power-on of servo system CPU</td><td>Imme-diate stop</td><td>• After checking and correcting the parameter setting, turn the servo system CPU power from OFF to ON or turn PC ready (M2000) from OFF to ON.</td></tr>
</table>

Table 2.15 Servo Amplifier Error (2000 to 2799) List (Continued)

Error Code	Amplifier Type	Error Cause: Name	Error Cause: Definition	Error Check Timing	Process-ing	Corrective Action
2601 to 2624	A	Initial parameter alarm	• The parameter setting is wrong. • The parameter data was corrupted. 2601 Amplifier setting 2602 Motor type 2603 Motor capacity 2604 Number of feedback pulses 2605 In-position range 2606 Position control gain 2 (actual position gain) 2607 Speed control gain 2 (actual speed gain) 2608 Speed integral compensation 2609 Forward rotation torque limit value 2610 Reverse rotation torque limit value 2611 Emergency stop time delay 2612 Position control gain 1 (model position gain) 2613 Speed control gain 1 (model speed gain) 2614 Load inertia ratio 2615 Error excessive alarm level 2616 Special compensation processing 2617 Special servo processing 2618 Td dead zone compensation 2619 Feed forward gain 2620 Unbalance torque compensation 2621 Dither command 2622 Gain operation time 2623 Servo response level setting 2624 –	• At power-on of servo amplifier • On PC ready (M2000) leading edge • At servo error reset • At power-on of servo system CPU	Imme-diate stop	• After checking and correcting the parameter setting, turn the servo system CPU power from OFF to ON or turn PC ready (M2000) from OFF to ON.

(2) Servo power supply module errors (2800 to 2999)

The servo power supply module errors are detected by the servo amplifier and assigned error codes 2800 to 2999.

When any of the servo errors occurs, the servo error detection signal (M2408+20n) turns ON. Eliminate the error cause and turn ON the servo error reset (M3208+20n) to reset the servo error, and make a restart. (However, the servo error detection signal will not turn ON for any of the error codes 2900 to 2999 as they are warning.)

Note: 1. For regenerative alarm protection (error code 2830), the status when the protective circuit was activated is still retained in the servo amplifier after activation. The data stored is cleared when the external power is switched OFF, but is not cleared by the RESET signal.

2. If the external power is switched OFF repeatedly to reset the error code 2830, overheat may lead to damage to the devices. Therefore, resume operation after removing the cause without fail.

The servo power supply module error definitions are given in Table 2.16.

Table 2.16 Servo Power Supply Module Error (2800 to 2999) List

Error code	Error Cause		Error Check Timing	Processing	Corrective Action
	Name	Definition			
2810	Undervoltage	• The power supply voltage of the servo power supply module fell below 170VAC. • Instantaneous power failure occurred.	Any time	Immediate stop	• Reconsider the power supply equipment.
		• Load is too large.			• Reconsider the power supply equipment.
2830	Excessive regeneration	• High-duty operation or continuous regenerative operation caused the max. load capacity of the regenerative brake resistor to be exceeded.			• Reconsider the operation pattern, e.g. decrease the acceleration/deceleration frequencies or reduce the speed.
		• Regenerative power transistor was damaged.			• Change the servo power supply module.
		• Regenerative brake resistor setting mistake in system settings			• Reconsider the system settings.
		• Regenerative brake resistor wiring mistake.			• Correct the wiring.
2833	Overvoltage	• Regenerative brake resistor connection mistake.			• Correct the wiring.
		• Regenerative power transistor was damaged.			• Change the servo power supply module.
		• Regenerative brake resistor is dead.			• Change the regenerative brake resistor.
		• Power supply voltage is high.			• Reconsider the power supply equipment.
2847	Amplifier power supply overheat	• The servo power supply module fan is at a stop.			• Change the fan.
		• The continuous output current of the servo power supply module is exceeded.			• Reduce the load.
		• Thermal sensor fault.			• Change the servo power supply module.
2940	Excessive regeneration warning	• 80% level of the excessive regeneration error (2830) was detected.		Continued	• Refer to details of the excessive regeneration error (2830).

Appendix 2.5 PC Link Communication Errors

Table 2.17 PC Link Communication Error Codes

Error Codes Stored in D9196	Error Description	Action to Take
01	A receiving packet for PC link communication does not arrive. The arrival timing of the receiving packet is too late.	• Check whether the PC has been switched ON. • Check whether the communication cable has been connected firmly. • Check whether the communication cable has been broken. • Check whether the A30BD-PCF or A30CD-PCF has been mounted normally.
02	A receiving packet CRC code is invalid.	• Check whether there is a noise source near the PC. • Check whether the communication cable has been connected firmly. • Check whether the communication cable has been broken.
03	A receiving packet data ID is invalid.	• Check whether the A30BD-PCF or A30CD-PCF has been mounted normally. • Replace the A30BD-PCF or A30CD-PCF.
04	The number of received frames is invalid.	• Check whether the communication cable has been connected firmly. • Check whether the communication cable has been broken. • Check whether there is a noise source near the PC.
05	A PC communication task is not active yet.	• Start the PC communication task.

Appendix 2.6 LED Indications when Errors Occur at the PCPU

<A172SHCPUN/A171SHCPUN>
When the errors listed below occur, they are indicated by the "ERROR" LED on the front panel of the A172SHCPUN, and the LED on the front panel of the A171SHCPUN. The error message can be read on the error list monitor screen of the peripheral device.
For details on the operating procedure, refer to the operating manual for the peripheral device.

Table 2.18 LED Indications When Errors Occur at PCPU

"ERREOR"LED ●:Lit ○:Not lit	Error Cause	Error Check Timing	Operation when Error Occurs	Error Set Device	Corrective Action
●	• The slot set in the "system settings" has nothing mounted in it, or has a different module mounted in it.	When power switched ON On resetting with the RESET key switch	• Start is disabled.	• System setting error flag (M2041) ON	• Set the "system settings" correctly in accordance with the modules actually mounted, then reset with the RESET key switch.
●	• Axis number settings are duplicated in the "system settings".				
●	• Not even one axis No. has been set in the "system settings".				
●	• No system setting data has been written. • The system setting data has been written without performing a relative check. Or it has been written although an error occurred in the relative check. • There is no battery in the memory cassette.				
●	• An axis No. that exceeds the "number of controlled axes" setting in the "system settings" has been set.				
●	• The total number of I/O points of the PC I/O modules set in motion slots in the "system settings" exceeds 256.				
●	• The amplifier type set in the "system settings" (MR-H-B/MR-J-B/MR-J2-B) disagrees with the amplifier type actually installed (MR-H-B/MR-J-B/MR-J2-B).	When the servo amplifier power is turned ON	• Servo operation does not start for the relevant axis only. Starting of this axis is disabled.		
For servo error ● For warning ○	• Occurrence of a servo error or servo warming • When using the LED does not light for a warning.	At all times	• In the case of MR-H-B, MR-J-B and MR-J2-B axes, only the relevant axis enters the servo OFF status. • In the case of ADU axes, according to the setting of "corrective action for ADU servo errors".	• Servo error detection flag (M1608+20n) ON • Servo error code device (D808+20n) set	• Ellminate the error cause and perform a servo error reset. After servo error reset. If the servo status is normal at all axes, the LED display is cleared.
●	• Detection of motion slot module abnormality (module comes out, or is loose, during operation)		–	• Motion slot module error detection flag (M2047) ON	• Switch off the power and mount the module correctly.
●	• Occurrence of a PCPU WDT error		• immediate stop of all axes	• PCPU WDT error flag (M9073) ON • PCPU WDT error cause (D9184) set	• See Section 3.5.2.

REMARK

Numerical values corresponding to axis numbers are entered for "n" in Table 2.18 (error set device).

<A172SHCPUN>

Axis No.	n
1	0
2	1
3	2
4	3
5	4
6	5
7	6
8	7

<A171SHCPUN>

Axis No.	n
1	0
2	1
3	2
4	3

<A273UHCPU (32-axis feature)/A173UHCPU(S1)>
When any of the errors listed below occurs, it is indicated on the LED on the front panel of the A273UHCPU. The error message can be read on the "error list monitor" screen of the peripheral device.
For the operating procedure, refer to the operating manual of the peripheral device.

Table 2.19 LED Indications at Error Occurrence on PCPU

A173UHCPU(S1) "ERREOR"LED ●:Lit ○:Not lit	A273UHCPU Front LED Indication	Error Cause	Error Check Timing	Operation when Error Occurs	Error Set Device	Corrective Action
●	LAY ERROR (SL■■) (*1) Base No.+Slot No.	• The slot set in "system settings" contains no or different module.	At power-on At reset with reset key	• Start is disabled.	• System setting error flag (M2041) ON	• Match "system settings" with the actual module and reset with the reset key.
●	AXIS NO. MULTIDEF	• There are overlapping axis number settings in "system settings".				
●	AMP NO SETTING	• Not one axis number is set in "system settings".				
–	PW NO SETTING	• When the ADU axis is set in "system settings", the servo power supply module (A230P) is not set.				
●	SYS. SET DATA ERR	• "System settings data" is not written. • "System settings data" was written without relative check, or was written with an error found in relative check. • Memory cassette battery is dead.				
●	AXIS NO. ERROR	• The axis number set in "system settings" is greater than the number of control axes.				
●	I/O POINTS OVER	• The total I/O points of the PC I/O modules set to the motion slots in "system settings" are greater than 256 points.				
●	AMP TYPE ERROR■■ Axis No. (01 to 32)	• The amplifier type (MR-H-B/MR-J-B/MR-J2-B) set in "system settings" differs from the actual amplifier type (MR-H-B/MR-J-B/MR-J2-B).	At power-on of servo amplifier	• Only the corresponding axis is not put in servo ON status and cannot be started.		
–	ADU ERROR (SL■■) (*1) Base No.+Slot No.	• ADU hardware fault.	At power-on (At reset with reset key)	• The corresponding ADU axis cannot be placed in servo ON status.	• Servo error detection flag (M2408+20n) ON • Servo error code device (D08+20n) set	• Change the ADU.

Table 2.19 LED Indications at Error Occurrence on PCPU (Continued)

A173UHCPU(S1) "ERREOR"LED ●:Lit ○:Not lit	A273UHCPU Front LED Indication	Error Cause	Error Check Timing	Operation when Error Occurs	Error Set Device	Corrective Action
At servo error ● At warning ○	SV . ERROR■■■■(■■) Servo error code Axis No. (01 to 32) • (**) indicates that the code is common to all axes.	• Servo error or warning occurrence	Any time	• For the MR-H-B/MR-J-B/MR-J2-B axis, only that axis is put in servo OFF status. • For the ADU axis, processing is performed in accordance with the setting of "ADU servo error processing".	• Servo error detection flag (M2408+20n) ON • Servo error code device (D08+20n) set	• Remove the error cause and reset the servo error. If the servos of all axes return to normal after servo error reset, the LED indication goes off.
	SV . ERROR■■■■(P■) Servo error code Indicates the "n"th servo power supply module.	• Servo power supply module (A230P)-detected servo error or warning occurrence		• In that line, all axes are put in servo OFF status.		
	SYS . ERR . ■■■■(P■) System error code (major error) detected by servo power supply module Indicates the "n"th servo power supply module. * indicates the system error which is independent of the servo power supply module line.	• Servo power supply module (A230P)-detected system error (major error) occurrence		• In that line, all axes are put in servo OFF status.	• Major error detection flag (M2407+20n) ON • Major error code device (D07+20n) set	• Remove the error cause and give all-axis servo ON command. If all axes are put in servo ON status properly, the LED goes off.
●	SL■■ UNIT ERROR (*1) Base No.+Slot No.	• Motion slot module fault detection (During operation, the module has come off or is coming off)		–	• Motion slot module fault detection flag (M2047) ON	• Switch power off and load the module properly.
●	PCPU WDT ERR. ■■■ PCPU WDT error code	• PCPU WDT error occurrence		• All axes stop immediately.	• PCPU WDT error flag (M9073) ON • PCPU WDT error cause (D9184) set	• Refer to Sections 3.3, 3.4.

(*1) Indicates the base number, slot number and slot information in error.

(SL□ □)

Slot Number in error
0: I/O slot 0
⟆ ⟆
7: I/O slot 7

Base number in error
0: Main base
1: Motion extension base 1
2: Motion extension base 2
3: Motion extension base 3
4: Motion extension base 4

REMARKS

n in Table 2.19 (Error Set Device) is the value corresponding to the axis number.

Axis No.	n	Axis No.	n	Axis No.	n	Axis No.	n
1	0	9	8	17	16	25	24
2	1	10	9	18	17	26	25
3	2	11	10	19	18	27	26
4	3	12	11	20	19	28	27
5	4	13	12	21	20	29	28
6	5	14	13	22	21	30	29
7	6	15	14	23	22	31	30
8	7	16	15	24	23	32	31

*Calculate the device number corresponding to each axis as described below.

M2408+20n (servo error detection flag) = M2408 + 20 × 31 = M3028

D07+20n (major error code device) = D07 + 20 × 31 = D627

APPENDIX 3 SPECIAL RELAYS AND SPECIAL REGISTERS

Appendix 3.1 Special Relays (SP, M)

The special relays are internal relays with fixed applications in the programmable controller. Accordingly, they must not be turned ON and OFF in sequence programs (those marked *1 and *2 in the table are exceptions).

Table 3.1 Special Relay List

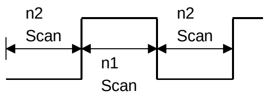
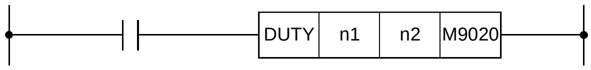
Number	Name	Stored Data	Explanation
M9000*1	Fuse blown	OFF Normal ON There is a module with a blown fuse.	● Comes ON even if there is only one output module with a blown fuse, and remains ON even after return to normal.
M9002*1	I/O unit verify error	OFF Normal ON Error	● Comes ON if there is a discrepancy between the actual I/O modules and the registered information when the power is turned on.
M9005*1	AC DOWN detection	OFF AC DOWN detected ON AC DOWN not detected	● Comes ON when there is a momentary power interruption not exceeding 20 ms; reset by turning the power OFF then ON again.
M9006	Battery low	OFF Normal ON Low battery voltage	● Comes ON when the battery voltage falls below the stipulated value; goes OFF when normal battery voltage is re-established.
M9007*1	Battery low latch	OFF Normal ON Low battery voltage	● Comes ON when the battery voltage falls below the stipulated value; remains ON even after normal battery voltage is re-established.
M9008*1	Self-diagnostic error	OFF No error ON Error	● Comes ON when an error occurs as a result of self-diagnosis.
M9009	Annunciator detection	OFF No F number detected ON F number detected	● Comes ON when OUT F, SET F instructions are executed. Goes OFF when 0 is stored in D9124.
M9010	Operation error flag	OFF No error ON Error	● Comes on when an operation error occurs during execution of an application instruction; goes OFF when the error is cleared.
M9011*1	Operation error flag	OFF No error ON Error	● Comes on when an operation error occurs during execution of an application instruction; remains ON even after the error is cleared.
M9012	Carry flag	OFF Carry OFF ON Carry ON	● Carry flag used in an application instruction.
M9016	Data memory clear flag	OFF No processing ON Output cleared	● When M9016 is ON, all data memory contents, including those in the latch range but with the exception of special relays/registers, are cleared on reception of remote RUN from a computer or other device.
M9017	Data memory clear flag	OFF No processing ON Output cleared	● When M9017 is ON, all data memory contents that are not latched, with the exception of special relays/registers, are cleared on reception of remote RUN from a computer or other device.
M9020	User timing clock No.0		● Relay repeats ON/OFF switching at fixed scan intervals. ● Starts from the OFF status when the power is turned ON or on resetting. ● The ON/OFF intervals are set with the DUTY instruction. 
M9021	User timing clock No.1		
M9022	User timing clock No.2		
M9023	User timing clock No.3		
M9024	User timing clock No.4		

Table 3.1 Special Relay List (Continued)

Number	Name	Stored Data	Explanation
M9025*1	Clock data set request	OFF No processing ON Data set request	• Writes the clock data stored in D9025 to D9028 to the clock devices after execution of the END instruction in the scan in which M9025 is switched ON.
M9026	Clock data error	OFF No error ON Error	• Comes ON when there is an error in he clock data (D9025 to D9028) values. OFF when there is no error.
M9028*2	Clock data read request	OFF No processing ON Read request	• When M9029 is ON, the clock data is read to D9025 to D9028 as BCD data.
M9030	0.1 second clock	0.05 SEC. 0.05 SEC.	• These relays generate the 0.1 second, 0.2 second, 1 second, 2 second, and 1 minute clocks. • These relays do not go ON/OFF with each scan but when their respective fixed intervals have elapsed, even during a scan. • These relays start from the OFF status when the power is turned on or resetting.
M9031	0.2 second clock	0.1 SEC. 0.1 SEC.	
M9032	1 second clock	0.5 SEC. 0.5 SEC.	
M9033	2 second clock	1 SEC. 1 SEC.	
M9034	1 minute clock	30 SEC. 30 SEC.	
M9036	Always ON	ON OFF	• Relay used for initialization during a sequence program or as a dummy contact for an application instruction. • M9036 and M9037 retain their ON or OFF status regardless of the settings of the key switch on the front of the CPU, but M9038 and M9039 change in accordance with the key switch status. They go OFF when the key switch is set to the STOP position. When the key switch is at a position other than STOP, M9038 comes ON for one scan only, and M9039 goes OFF for one scan only.
M9037	Always OFF	ON OFF	
M9038	ON for 1 scan only after RUN	ON 1 scan OFF	
M9039	RUN flag (OFF for 1 scan only after RUN)	ON 1 scan OFF	
M9040	PAUSE enable coil	OFF PAUSE disable ON PAUSE enabled	• When the RUN/STOP key switch is set to PAUSE or the remote PAUSE contact is turned on, provided M9040 is ON, the PAUSE status is established and M9041 comes ON.
M9041	PAUSE status contact	OFF PAUSE not in effect ON PAUSE in effect	
M9042	STOP status contact	OFF STOP not in effect ON STOP in effect	• ON when the RUN/STOP key switch is set to STOP.
M9043	Sampling trace completed	OFF Sampling trace in progress ON Sampling trace completed	• Comes ON on completion of the number of sampling traces set in the parameters are completed after execution of the STRA instruction. After that, it is reset by execution of the STRAR instruction.
M9046	Sampling trace	OFF Trace not in progress ON Trace in progress	• ON during execution of a sampling trace
M9047	Sampling trace preparation	OFF Sampling trace stop ON Sampling trace start	• A sampling trace cannot be executed unless M9047 has been turned ON. When M9047 is turned OFF, the sampling trace is stopped.
M9049	Number of output characters selection	OFF Output until NUL code ON 16 characters output	• When M9049 is OFF, output continues until the NUL (00H) code. When M9049 is ON, ASCII code for 16 characters is output.
M9052*2	SEG instruction switch	OFF 7-segment display ON I/O part refresh	• When M9052 is ON it is executed as the I/O partial refresh instruction. When M9052 is ON, it is executed as the 7-segment display instruction.
M9053*2	EI/DI instruction switch	OFF Sequence interrupt control ON Link interrupt control	• Turn ON when a link refresh enable/disable (EI, DI) instruction is executed.

Table 3.1 Special Relay List (Continued)

Number	Name	Stored Data	Explanation
M9054	STEP RUN flag	OFF STEP RUN not in effect ON STEP RUN in effect	● ON when the RUN/STOP key switch is set to the RUN position.
M9055	Status latch completion flag	OFF Not completed ON Completed	● Comes ON when status latch is completed. Goes OFF on execution of a reset instruction.
M9084 *2	Error check	OFF Error check executed ON No error check	● Set whether or not the error check shown below is executed on END instruction processing. (Used to shorten END instruction processing time.) (1) Blown fuse check (2) I/O module verification check (3) Battery check

POINTS

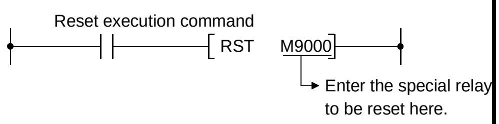
(1) All special relays, M, are turned OFF by turning the power, OFF, performing latch clear, or resetting with the RESET key switch.
When the RUN key switch is set to "STOP", the special relay settings are retained.

(2) The special relays marked "*1" in the table above remain "ON" even after a return to normal. They must therefore be turned OFF by using one of the following methods.

(a) Method using the user program
Insert the ladder block at right into the program and turn the reset execution command contact ON to clear the special relay.

(b) Method using a peripheral device
Perform a forced reset using the test function of the peripheral device.
For details on this operation, refer to the manual for the peripheral device.

(c) Turn the special relay OFF by setting the RESET key switch on the front panel of the CPU module to "RESET".



(3) The ON/OFF status of special relays marked "*2" in the table above is controlled by the sequence program.

(4) The special relays marked *3 are reset only when power is switched from OFF to ON.

Appendix 3.2 Special Registers (SP.D)

The special registers are data registers used for specific purposes in the programmable controller. Therefore, do not write data to the special registers in the program (with the exception of those whose numbers are marked *2 in the table).
Of the special relays, those from D9180 to D9199 are used for positioning control.

Table 3.2 Special Register List

Number	Name	Stored Data	Explanation
D9000	Fuse blown	Number of module with blown fuse	● When modules with a blown fuse are detected, the lowest I/O number of the detected modules is stored in hexadecimal in this special relay. (Example: Blown fuses at the output modules Y50 to 6F... "50" is stored in hexadecimal.) For monitoring at a peripheral device, use hexadecimal display monitor operations. (Cleared when the contents of D9100 are all "0".)
D9002	I/O unit verify error	I/O module verification error module number	● If I/O modules that do not match the registered data are detected when the power is turned on, the first I/O number of the lowest module number among the detected modules is stored in hexadecimal (the storage method is the same as for D9000). When monitoring with a peripheral device, use a hexadecimal display monitoring operation. (Cleared when all contents of D9116 to D9123 are reset to zero.)
D9005*1	AC DOWN counter	AC DOWN occurrence count	● 1 is added to the stored value each time the input voltage becomes 80% or less of the rating while the CPU module is performing an operation, and the value is stored in BIN code.
D9008*1	Self-diagnostic error	Self-diagnostic error number	● 1 is added to the stored value when an error is found as a result of self-diagnosis, the error number, and the value is stored in BIN code.
D9009	Annunciator detection	F number at which external failure has occurred	● When one of F0 to 255 is turned on by $\boxed{\text{OUT F}}$ or $\boxed{\text{SET F}}$, the F number detected earliest among the F numbers which have been turned on is stored in BIN code. ● D9009 can be cleared by executing a $\boxed{\text{RST F}}$ or $\boxed{\text{LEDR}}$ instruction. If another F number has been detected, the clearing of D9009 causes the next number to be stored in D9009.
D9010	Error step	Step number at which operation error has occurred	● When an operation error occurs during execution of an application instruction, the step No. where the error occurred is stored in BIN cod, and thereafter, every time an operation error occurs the contents of D9010 are updated.
D9011	Error step	Step number at which operation error has occurred	● When an operation error occurs during execution of an application instruction, the step number at which the error occurs is stored in this register in BIN code. Since storage is executed when M9011 changes from OFF to ON, the contents of D9011 cannot be updated unless it is cleared by the user program.
D9014	I/O control mode	I/O control mode number	● The set control mode is represented as follows: 0: I/O in direct mode 3: I/O in refresh mode

Table 3.2 Special Register List (Continued)

Number	Name	Stored Data	Explanation
D015	CPU operating states	Operating states of CPU	● The CPU operation states indicated in the figure below are stored in D9015. B15 • • • • B12 B11 • • • • B8 B7 • • • • B4 B3 • • • • B0 B3–B0: CPU key switch (Remains unchanged in remote run/stop mode) 0 RUN 1 STOP B7–B4: Remote RUN/STOP by parameter setting 0 RUN 1 STOP 2 PAUSE* B11–B8: Status in program 0 Other than below 1 STOP instruction execution B15–B12: Remote RUN/STOP by computer 0 RUN 1 STOP 2 PAUSE* *: When the CPU is in the RUN status and M9040 is OFF, the CPU remains in RUN mode even if set to PAUSE.
D9016	ROM/RAM setting	0: ROM 1: RAM 2: E^2PROM	● Indicates the setting for the memory selection chip; one of the values 0 to 2 is set in BIN code.
D9017	Scan time	Minimum scan time (10 ms units)	● At each END instruction, if the scan time is shorter than the contents of D9017, the new value is stored in this register. In other words, the minimum value for scan time is stored in D9017, in BIN code.
D9018	Scan time	Scan time (10 ms units)	● The scan time is stored in BIN code at each END instruction and is always rewritten.
D9019	Scan time	Maximum scan time (10 ms units)	● At each END instruction, if the scan time is longer than the contents of D9019, the new value is stored in this register. In other words, the maximum value for scan time is stored in D9019, in BIN code.
D9020[*2]	Constant scan	Constant scan time (user-specified in 10 ms units)	● When user programs are executed at fixed intervals, used to set the execution intervals, in 10 ms units. 0 : Constant scan function not used 1 to 200 : Constant scan function used program executed at intervals of (set value)×10 ms.

Table 3.2 Special Register List (Continued)

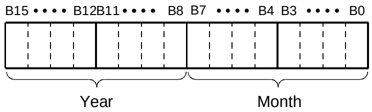
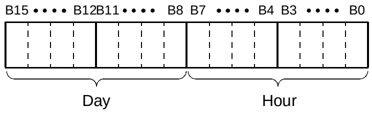
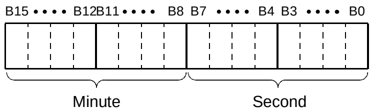
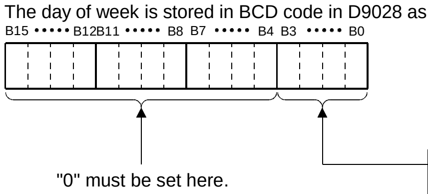
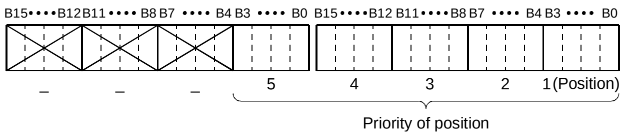
<table>
<tr><th>Number</th><th>Name</th><th>Stored Data</th><th>Explanation</th></tr>
<tr><td>D9025*2</td><td>Clock data</td><td>Clock data
(year, month)</td><td>● The year (last two digits) and month are stored in BCD code in D9025 as shown below.

Example
: July, 1993
H9307</td></tr>
<tr><td>D9026*2</td><td>Clock data</td><td>Clock data
(day, hour)</td><td>● The day and hour are stored in BCD code in D9026 as shown below.

Example
: 31st, 10th hour
H3110</td></tr>
<tr><td>D9027*2</td><td>Clock data</td><td>Clock data
(minute, second)</td><td>● The minute and second are stored in BCD code in D9027 as shown below.

Example
: 35ms, 48s
H3548</td></tr>
<tr><td>D9028*2</td><td>Clock data</td><td>Clock data
(0, day of week)</td><td>● The day of week is stored in BCD code in D9028 as shown below.

Example
: Friday
H0005
<table><tr><th colspan="2">Day of week</th></tr><tr><td>0</td><td>Sunday</td></tr><tr><td>1</td><td>Monday</td></tr><tr><td>2</td><td>Tuesday</td></tr><tr><td>3</td><td>Wednesday</td></tr><tr><td>4</td><td>Thursday</td></tr><tr><td>5</td><td>Friday</td></tr><tr><td>6</td><td>Saturday</td></tr></table></td></tr>
<tr><td>M9038*2
M9039*2</td><td>LED display priority</td><td>Priorities 1 to 4
Priorities 5 to 7</td><td>● The element numbers for priorities 1 to 4 (D9038) and 5 to 7 (D9039) for the lighting (or flashing) of the ERROR LED when an error occurs, are set and changed.

Even if "0" is set, errors which cause CPU operation to stop (including parameter settings) are unconditionally displayed on the LED display.
Default values: D9038=H4321
D9039=H0006
<table><tr><th>Element No,</th><th>Content</th></tr><tr><td>0.</td><td>Not displayed</td></tr><tr><td>1.</td><td>I/O verify, fuse blown</td></tr><tr><td>2.</td><td>Special function module, link parameters, SFC parameters, SFC operation</td></tr><tr><td>3.</td><td>CHK instruction error</td></tr><tr><td>4.</td><td>Annunciator (F)</td></tr><tr><td>5.</td><td>LED instruction related</td></tr><tr><td>6.</td><td>Parity error</td></tr></table></td></tr>
</table>

Table 3.2 Special Register List (Continued)

<table>
<tr><th>Number</th><th>Name</th><th>Stored Data</th><th>Explanation</th></tr>
<tr>
<td>D9100
to
D9101</td>
<td>Fuse blown module</td>
<td>Bit pattern of fuse blown modules in units of 16 points
(D9100 to D9101 are used for A172SHCPUN/A171SHCPUN)</td>
<td>
● Indicates the output module numbers with blown fuses (in units of 16 points) in a bit pattern. (Parameter assignment is valid)
● Also indicates the blown fuse states of the output modules in remote stations.
<table>
<tr><th></th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td>D9100</td><td>0</td><td>0</td><td>0</td><td>1 (YC0)</td><td>0</td><td>0</td><td>0</td><td>1 (Y80)</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>D9101</td><td>1 (Y1F0)</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1 (Y1A)</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>D9107</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1 (Y7B0)</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1 (Y730)</td><td>0</td><td>0</td><td>0</td></tr>
</table>
Indicates a blown fuse.
● Turn M9197 and M9198 ON/OFF to change the I/O module number range displayed.
● Clear the blown fuse module data by turning OFF M9000 (blown fuse).
</td>
</tr>
<tr>
<td>D9116
to
D9123</td>
<td>Input/Output module verification error</td>
<td>Bit pattern of verify error modules in units of 16 points
(D9116 to D9117 are used for A172SHCPUN/A171SHCPUN)</td>
<td>
● Indicates the I/O module numbers (in units of 16 points) when the I/O modules different from the registered I/O module information are detected at power-on. (Parameter assignment is valid)
● Also indicates the I/O module information in remote stations.
<table>
<tr><th></th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td>D9116</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1 (X/Y 0)</td></tr>
<tr><td>D9117</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1 (X/Y 190)</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>D9123</td><td>0</td><td>1 (X/Y 7F0)</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
</table>
Indicates an input/output module verification error.
● Turn M9197 and M9198 ON/OFF to change the I/O module number range displayed.
● Clear the verify error data by turning OFF M9002 (verify error).
</td>
</tr>
<tr>
<td>D9124</td>
<td>Annunciator detection quantity</td>
<td>Number of detected annunciators</td>
<td>
● When one of F0 to 255 is turned on by an [OUT F] or [SET F], 1 is added to the contents of D9124.
When the [RST F] or [LEDR] instruction is executed, 1 is subtracted from the contents of D9124.
The number of annunciators that has been turned on by [OUT F] or [SET F] is stored in D9124: the maximum stored value is 8.
</td>
</tr>
<tr>
<td>D9125
to
D9132</td>
<td>Annunciator detection number</td>
<td>Annunciator detection number</td>
<td>
● When F numbers in the range F0 to 255 are turned on by [OUT F] or [SET F], they
are entered in D9125 to D9132 in ascending order of register numbers.
An F number which is turned off by [RST F] is erased from D9125 to D9132, and the contents of the data registers following the one where the erased F number was stored are each shifted to the preceding data register. When the [LEDR] instruction is executed, the contents of D9125 to D9132 are shifted upward by one. When there are 8 annunciator detections, a 9th one is not stored in D9125 to D9132 even if detected.
<table>
<tr><th></th><th></th><th>SET F50</th><th>SET F25</th><th>SET F99</th><th>SET F25</th><th>SET F15</th><th>SET F70</th><th>SET F65</th><th>SET F38</th><th>SET F110</th><th>SET F151</th><th>SET F210</th><th>LEDR</th></tr>
<tr><td>D9009</td><td>0</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>99</td></tr>
<tr><td>D9124</td><td>0</td><td>1</td><td>2</td><td>3</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>8</td><td>8</td></tr>
<tr><td>D9125</td><td>0</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>99</td></tr>
<tr><td>D9126</td><td>0</td><td>0</td><td>25</td><td>25</td><td>99</td><td>99</td><td>99</td><td>99</td><td>99</td><td>99</td><td>99</td><td>99</td><td>15</td></tr>
<tr><td>D9127</td><td>0</td><td>0</td><td>0</td><td>99</td><td>0</td><td>15</td><td>15</td><td>15</td><td>15</td><td>15</td><td>15</td><td>15</td><td>70</td></tr>
<tr><td>D9128</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>70</td><td>70</td><td>70</td><td>70</td><td>70</td><td>70</td><td>65</td></tr>
<tr><td>D9129</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>65</td><td>65</td><td>65</td><td>65</td><td>65</td><td>38</td></tr>
<tr><td>D9130</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>38</td><td>38</td><td>38</td><td>38</td><td>110</td></tr>
<tr><td>D9131</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>110</td><td>110</td><td>110</td><td>151</td></tr>
<tr><td>D9132</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>151</td><td>151</td><td>210</td></tr>
</table>
</td>
</tr>
</table>

POINTS

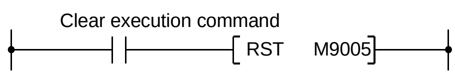
(1) All special register data is cleared by the power-off, latch clear, and reset operations. The data is retained when the RUN/STOP key switch is set to STOP.

(2) The contents of the special relays marked *1 in the table above are not cleared even after the normal status is restored. To clear the contents, use one of the following methods:

(a) Using a user program
Insert the ladder block shown at right into the program and turn on the clear execution command contact to clear the contents of the register.

(b) Using a peripheral device
Using the test function of a peripheral device, set the register to "0" by using present value change or forced reset.
For details on the operation involved, refer to the manual for the relevant peripheral device.

(c) Set the special register to "0" by setting the RESET key switch on the front of the CPU to the RESET position.



(3) For special registers marked "*2", data is written in the sequence program.

(4) The special registers marked *3 are cleared only when power is switched from OFF to ON.

Table 3.2 Special Register List (Continued)

<table>
<tr><th>Number</th><th>Name</th><th>Stored Data</th><th>Explanation</th></tr>
<tr>
<td>D9180
to
D9183</td>
<td>Limit switch output storage area</td>
<td>Limit switch output storage area
1: ON
0: OFF
(A172SHCPUN/A171SHCPUN)</td>
<td>
● The status of output (ON/OFF) to limit switch output AY42 set with a peripheral device is stored as "1" or "0".
1: ON
0: OFF
● These registers can be used to output limit switch output data to an external device using the sequence program.
(1) A172SHCPUN
<table>
<tr><td></td><td>b15</td><td>b14</td><td>b13</td><td>b12</td><td>b11</td><td>b10</td><td>b9</td><td>b8</td><td>b7</td><td>b6</td><td>b5</td><td>b4</td><td>b3</td><td>b2</td><td>b1</td><td>b0</td></tr>
<tr><td>D9180</td><td>LY0F</td><td>LY0E</td><td>LY0D</td><td>LY0C</td><td>LY0B</td><td>LY0A</td><td>LY09</td><td>LY08</td><td>LY07</td><td>LY06</td><td>LY05</td><td>LY04</td><td>LY03</td><td>LY02</td><td>LY01</td><td>LY00</td></tr>
<tr><td></td><td colspan="8">For axis 2</td><td colspan="8">For axis 1</td></tr>
<tr><td>D9181</td><td>LY1F</td><td>LY1E</td><td>LY1D</td><td>LY1C</td><td>LY1B</td><td>LY1A</td><td>LY19</td><td>LY18</td><td>LY17</td><td>LY16</td><td>LY15</td><td>LY14</td><td>LY13</td><td>LY12</td><td>LY11</td><td>LY10</td></tr>
<tr><td></td><td colspan="8">For axis 4</td><td colspan="8">For axis 3</td></tr>
<tr><td>D9182</td><td>LY2F</td><td>LY2E</td><td>LY2D</td><td>LY2C</td><td>LY2B</td><td>LY2A</td><td>LY29</td><td>LY28</td><td>LY27</td><td>LY26</td><td>LY25</td><td>LY24</td><td>LY23</td><td>LY22</td><td>LY21</td><td>LY20</td></tr>
<tr><td></td><td colspan="8">For axis 6</td><td colspan="8">For axis 5</td></tr>
<tr><td>D9183</td><td>LY3F</td><td>LY3E</td><td>LY3D</td><td>LY3C</td><td>LY3B</td><td>LY3A</td><td>LY39</td><td>LY38</td><td>LY37</td><td>LY36</td><td>LY35</td><td>LY34</td><td>LY33</td><td>LY32</td><td>LY31</td><td>LY30</td></tr>
<tr><td></td><td colspan="8">For axis 8</td><td colspan="8">For axis 7</td></tr>
</table>
* "1" or "0" is stored for each of the bits in D9180 through D9183.
1) 1..........ON
2) 0..........OFF
(2) A171SHCPUN
<table>
<tr><td></td><td>b15</td><td>b14</td><td>b13</td><td>b12</td><td>b11</td><td>b10</td><td>b9</td><td>b8</td><td>b7</td><td>b6</td><td>b5</td><td>b4</td><td>b3</td><td>b2</td><td>b1</td><td>b0</td></tr>
<tr><td>D9180</td><td>LY0F</td><td>LY0E</td><td>LY0D</td><td>LY0C</td><td>LY0B</td><td>LY0A</td><td>LY09</td><td>LY08</td><td>LY07</td><td>LY06</td><td>LY05</td><td>LY04</td><td>LY03</td><td>LY02</td><td>LY01</td><td>LY00</td></tr>
<tr><td></td><td colspan="8">For axis 2</td><td colspan="8">For axis 1</td></tr>
<tr><td>D9181</td><td>LY1F</td><td>LY1E</td><td>LY1D</td><td>LY1C</td><td>LY1B</td><td>LY1A</td><td>LY19</td><td>LY18</td><td>LY17</td><td>LY16</td><td>LY15</td><td>LY14</td><td>LY13</td><td>LY12</td><td>LY11</td><td>LY10</td></tr>
<tr><td></td><td colspan="8">For axis 4</td><td colspan="8">For axis 3</td></tr>
</table>
* "1" or "0" is stored for each of the bits in D9180 through D9181.
1) 1..........ON
2) 0..........OFF
</td>
</tr>
</table>

Table 3.2 Special Register List (Continued)

<table>
<tr><th>Number</th><th>Name</th><th>Stored Data</th><th>Explanation</th></tr>
<tr><td>D9184</td><td>Cause of PCPU error</td><td>PCPU WDT error number</td><td>
● The PCPU WDT errors tabled below are stored in D9184.
<table>
<tr><th>Error Code</th><th>Error Cause</th></tr>
<tr><td>1</td><td>PCPU software fault 1</td></tr>
<tr><td>2</td><td>PCPU excessive operation frequency</td></tr>
<tr><td>3</td><td>PCPU software fault 2</td></tr>
<tr><td>30</td><td>Hardware fault between PCPU and SCPU</td></tr>
<tr><td>100 to 107
110 to 117
120 to 127
130 to 137
140 to 147</td><td>AC motor drive unit CPU fault
100
Indicates the slot No.(0 to 7) where the AC motor drive module with the fault is loaded.
Indicates the stage No. of the base on which the AC motor drive module with the fault is loaded.
0: Main base
1: Extension base 1st stage
2: Extension base 2nd stage
3: Extension base 3rd stage
4: Extension base 4th stage</td></tr>
<tr><td>200 to 207
210 to 217
220 to 227
230 to 237
240 to 247</td><td>Motion main base/extension base-loaded module hardware fault
200
Indicates the slot No.(0 to 7) where the module with the fault is loaded.
Indicates the stage No. of the base on which the module with the fault is loaded.
0: Main base
1: Extension base 1st stage
2: Extension base 2nd stage
3: Extension base 3rd stage
4: Extension base 4th stage</td></tr>
<tr><td>250 to 253</td><td>Separated servo amplifier (MR-□-B) interface hardware fault
250
Faulty SSCNET No.
0: SSCNET 1
1: SSCNET 2
2: SSCNET 3
3: SSCNET 4</td></tr>
<tr><td>300</td><td>PCPU software fault 3</td></tr>
<tr><td>301</td><td>21 or more programs were started simultaneously by the CPSTART instruction of 8 or more points.
Up to 20 programs may be started simultaneously by the CPSTART instruction of 8 or more points.</td></tr>
</table>
</td></tr>
</table>

Table 3.2 Special Register List (Continued)

<table>
<tr><th>Number</th><th>Name</th><th>Stored Data</th><th>Explanation</th></tr>
<tr><td>D9185</td><td rowspan="2">Servo amplifier type</td><td rowspan="2">Servo amplifier type
(A172SHCPUN/A171SHCPUN)</td><td rowspan="2">● On switching the power ON or resetting, the servo amplifier type set in the system settings is set in these devices.
(1) When an A172SHCPUN is used
<table><tr><th></th><th>b15 to b12</th><th>b11 to b8</th><th>b7 to b4</th><th>b3 to b0</th></tr><tr><td>D9185</td><td>Axis 4</td><td>Axis 3</td><td>Axis 2</td><td>Axis 1</td></tr><tr><td>D9186</td><td>Axis 8</td><td>Axis 7</td><td>Axis 6</td><td>Axis 5</td></tr></table>Servo amplifier type
• 0 ••• Unused axis
• 2 ••• MR-□-B
(2) When an A171SHCPUN is used
<table><tr><th></th><th>b15 to b12</th><th>b11 to b8</th><th>b7 to b4</th><th>b3 to b0</th></tr><tr><td>D9185</td><td>Axis 4</td><td>Axis 3</td><td>Axis 2</td><td>Axis 1</td></tr><tr><td>D9186</td><td colspan="4">0</td></tr></table>Servo amplifier type
• 0 ••• Unused axis
• 2 ••• MR-□-B</td></tr>
<tr><td>D9186</td></tr>
<tr><td>D9187</td><td>Manual pulse generator axis setting error</td><td>Manual pulse generator axis setting error
(A172SHCPUN/A171SHCPUN)</td><td>● Stores the contents of the manual pulse generator axis setting error when the manual pulse generator axis setting flag (M9077) comes ON.
(1) When an A172SHCPUN is used
<table><tr><th></th><th>b15</th><th></th><th></th><th></th><th></th><th></th><th></th><th>b8</th><th></th><th>b3</th><th></th><th>b0</th></tr><tr><td>D9187</td><td>Axis8</td><td>Axis7</td><td>Axis6</td><td>Axis5</td><td>Axis4</td><td>Axis3</td><td>Axis2</td><td>Axis1</td><td>0</td><td>P1</td><td>0</td><td>P1</td></tr></table>• 1 pulse input magnification setting error
0: Normal
1: Setting error
(Outside the range 1 to 100)
• Manual pulse generator axis setting error
0: Normal
1: Setting error
(When the axis setting for each digit is outside the range 1 to 8)
• Manual pulse generator smoothing magnification setting error
0: Normal
1: Setting error
(Outside the range 0 to 59)
(2) When an A171SHCPUN is used
<table><tr><th></th><th>b15</th><th>b11</th><th></th><th></th><th>b8</th><th></th><th>b3</th><th></th><th>b0</th></tr><tr><td>D9187</td><td>0</td><td>Axis4</td><td>Axis3</td><td>Axis2</td><td>Axis1</td><td>0</td><td>P1</td><td>0</td><td>P1</td></tr></table>• 1 pulse input magnification setting error
0: Normal
1: Setting error
(Outside the range 1 to 100)
• Manual pulse generator axis setting error
0: Normal
1: Setting error
(When the axis setting for each digit is outside the range 1 to 4)
• Manual pulse generator smoothing magnification setting error
0: Normal
1: Setting error
(Outside the range 0 to 59)</td></tr>
<tr><td>D9188</td><td>Test mode request error</td><td>Test mode request error
(A172SHCPUN/A171SHCPUN)</td><td>● Stores the data of axes being operated when the test mode request error flag (M9078) comes ON.
(1) When an A172SHCPUN is used
<table><tr><th></th><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr><tr><td>D9188</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>Axis8</td><td>Axis7</td><td>Axis6</td><td>Axis5</td><td>Axis4</td><td>Axis3</td><td>Axis2</td><td>Axis1</td></tr></table>All set to "0"
Stores the operating/stopped status of each axis
• 0: Stopped
• 1: Operating
(2) When an A171SHCPUN is used
<table><tr><th></th><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr><tr><td>D9188</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>Axis4</td><td>Axis3</td><td>Axis2</td><td>Axis1</td></tr></table>All set to "0"
Stores the operating/stopped status of each axis
• 0: Stopped
• 1: Operating</td></tr>
</table>

Table 3.2 Special Register List (Continued)

<table>
<tr><th>Number</th><th>Name</th><th>Stored Data</th><th>Explanation</th></tr>
<tr><td>D9189</td><td>Error program No.</td><td>Error program number (A172SHCPUN/A171SHCPUN)</td><td>● Stores the motion program number (range: 1 to 256) affected by the error when the motion program setting error flag (M9079) comes ON.
● If, once an error program number has been stored, an error occurs in another motion program, the program number of the program with the new error is stored.</td></tr>
<tr><td>D9190</td><td>Error item information</td><td>Motion program setting error number (A172SHCPUN/A171SHCPUN)</td><td>● Stores the error code corresponding to the setting item in error when the motion program setting error flag (M9079) turns ON.
<table>
<tr><th>Error Code</th><th>Error Definition</th></tr>
<tr><td>1</td><td>The parameter block number specified is outside the range 1 to 16.</td></tr>
<tr><td>906</td><td>The motion program set in the DSFRP/SVST instruction has the unused axis in system settings.</td></tr>
<tr><td>3300</td><td>An attempt was made to start and run 9 or more programs simultaneously with the DSFRP/SVST instruction.</td></tr>
</table>
For the error processings and corrective actions, refer to Appendix 2.1.</td></tr>
<tr><td>D9191</td><td>Servo amplifier installation information</td><td>Servo amplifier installation information (A172SHCPUN/A171SHCPUN)</td><td>● When the power is turned ON, or on resetting, the servo amplifier and option slot installation statuses are checked and the results stored in this device.
(1) When an A172SHCPUN is used
<table>
<tr><th></th><th>b15 ～ b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr>
<tr><td>D9191</td><td>0</td><td>Axis8</td><td>Axis7</td><td>Axis6</td><td>Axis5</td><td>Axis4</td><td>Axis3</td><td>Axis2</td><td>Axis1</td></tr>
</table>
Stores the operating/stopped status of each axis
・Installed ・・・・・・ 1
・Not installed ・・・ 0
(2) When an A171SHCPUN is used
<table>
<tr><th></th><th>b15 ～ b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr>
<tr><td>D9191</td><td>0</td><td>Axis4</td><td>Axis3</td><td>Axis2</td><td>Axis1</td></tr>
</table>
Stores the operating/stopped status of each axis
・Installed ・・・・・・ 1
・Not installed ・・・ 0</td></tr>
<tr><td>D9192</td><td>Area for setting the smoothing magnification for manual pulse generator 1 (P1)</td><td>Areas for setting manual pulse generator smoothing magnifications (A172SHCPUN/A171SHCPUN)</td><td>● Stores the manual pulse generator smoothing time constant.
● The smoothing time constant is calculated using the following formula:
$$\text{Smoothing time constant (t)} = \left(\text{Smoothing magnification}+1\right) \times 56.8[\text{ms}]$$
The setting range for smoothing magnification is 0 to 59.</td></tr>
</table>

Table 3.2 Special Register List (Continued)

<table>
<tr><th>Number</th><th>Name</th><th>Stored Data</th><th>Explanation</th></tr>
<tr><td>D752</td><td>Manual pulse generator 1 (P1) smoothing magnification setting area</td><td rowspan="3">Manual pulse generator smoothing magnification setting area
(For A273UHCPU (32-axis feature)/A173UHCPU(S1))</td><td rowspan="3">● Stores the smoothing time constant of the manual pulse generator.
● The smoothing time constant is calculated by the following expression.
Smoothing time constant (t) = (smoothing magnification + 1) × 56.8 [ms]
Note that the setting range of the smoothing magnification is 0 to 59.</td></tr>
<tr><td>D753</td><td>Manual pulse generator 2 (P2) smoothing magnification setting area</td></tr>
<tr><td>D754</td><td>Manual pulse generator 3 (P3) smoothing magnification setting area</td></tr>
<tr><td>D776
to
D791</td><td>Axis 1 to 32 limit switch output status storing area</td><td>Limit switch output status storing area
1: ON
0: OFF
(For A273UHCPU (32-axis feature)/A173UHCPU(S1))</td><td>● Stores 1 or 0 to indicate the output status (ON/OFF) to the limit switch output AY42 set in the peripheral device.
1: ON
0: OFF
● May be used to export the limit switch output data in a sequence program.
<table>
<tr><th></th><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr>
<tr><td>D776</td><td>LY0F</td><td>LY0E</td><td>LY0D</td><td>LY0C</td><td>LY0B</td><td>LY0A</td><td>LY09</td><td>LY08</td><td>LY07</td><td>LY06</td><td>LY05</td><td>LY04</td><td>LY03</td><td>LY02</td><td>LY01</td><td>LY00</td></tr>
<tr><td></td><td colspan="8">For axis 2</td><td colspan="8">For axis 1</td></tr>
<tr><td>D777</td><td>LY1F</td><td>LY1E</td><td>LY1D</td><td>LY1C</td><td>LY1B</td><td>LY1A</td><td>LY19</td><td>LY18</td><td>LY17</td><td>LY16</td><td>LY15</td><td>LY14</td><td>LY13</td><td>LY12</td><td>LY11</td><td>LY10</td></tr>
<tr><td></td><td colspan="8">For axis 4</td><td colspan="8">For axis 3</td></tr>
<tr><th></th><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr>
<tr><td>D790</td><td>LYEF</td><td>LYEE</td><td>LYED</td><td>LYEC</td><td>LYEB</td><td>LYEA</td><td>LYE9</td><td>LYE8</td><td>LYE7</td><td>LYE6</td><td>LYE5</td><td>LYE4</td><td>LYE3</td><td>LYE2</td><td>LYE1</td><td>LYE0</td></tr>
<tr><td></td><td colspan="8">For axis 30</td><td colspan="8">For axis 29</td></tr>
<tr><td>D791</td><td>LYFF</td><td>LYFE</td><td>LYFD</td><td>LYFC</td><td>LYFB</td><td>LYFA</td><td>LYF9</td><td>LYF8</td><td>LYF7</td><td>LYF6</td><td>LYF5</td><td>LYF4</td><td>LYF3</td><td>LYF2</td><td>LYF1</td><td>LYF0</td></tr>
<tr><td></td><td colspan="8">For axis 32</td><td colspan="8">For axis 31</td></tr>
</table>
* "1" or "0" is stored for each bit of D776 to D791.
1) 1: ON
2) 0: OFF</td></tr>
<tr><td>D792
to
D799</td><td>Servo amplifier type</td><td>Servo amplifier type
(For A273UHCPU (32-axis feature)/A173UHCPU(S1))</td><td>● Stores the servo amplifier type specified in the system settings at power-on or reset.
<table>
<tr><th></th><th>b15 to b12</th><th>b11 to b8</th><th>b7 to b4</th><th>b3 to b0</th></tr>
<tr><td>D792</td><td>Axis 4</td><td>Axis 3</td><td>Axis 2</td><td>Axis 1</td></tr>
<tr><td>D793</td><td>Axis 8</td><td>Axis 7</td><td>Axis 6</td><td>Axis 5</td></tr>
<tr><td>D794</td><td>Axis 12</td><td>Axis 11</td><td>Axis 10</td><td>Axis 9</td></tr>
<tr><td>D795</td><td>Axis 16</td><td>Axis 15</td><td>Axis 14</td><td>Axis 13</td></tr>
<tr><td>D796</td><td>Axis 20</td><td>Axis 19</td><td>Axis 18</td><td>Axis 17</td></tr>
<tr><td>D797</td><td>Axis 24</td><td>Axis 23</td><td>Axis 22</td><td>Axis 21</td></tr>
<tr><td>D798</td><td>Axis 28</td><td>Axis 27</td><td>Axis 26</td><td>Axis 25</td></tr>
<tr><td>D799</td><td>Axis 32</td><td>Axis 31</td><td>Axis 30</td><td>Axis 29</td></tr>
</table>
Servo amplifier type
・0・・・Unused axis
・1・・・ADU (Main base)
・2・・・MR-□-B
・3・・・ADU (Motion extension base)</td></tr>
<tr><td>D9182
to
D9183</td><td>Test mode request error</td><td>Test mode request error
(For A273UHCPU (32-axis feature)/A173UHCPU(S1))</td><td>● Stores the operating axis data when the test mode request error flag (M9078) turns ON.
<table>
<tr><th></th><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr>
<tr><td>D9182</td><td>Axis16</td><td>Axis15</td><td>Axis14</td><td>Axis13</td><td>Axis12</td><td>Axis11</td><td>Axis10</td><td>Axis9</td><td>Axis8</td><td>Axis7</td><td>Axis6</td><td>Axis5</td><td>Axis4</td><td>Axis3</td><td>Axis2</td><td>Axis1</td></tr>
<tr><td>D9183</td><td>Axis32</td><td>Axis31</td><td>Axis30</td><td>Axis29</td><td>Axis28</td><td>Axis27</td><td>Axis26</td><td>Axis25</td><td>Axis24</td><td>Axis23</td><td>Axis22</td><td>Axis21</td><td>Axis20</td><td>Axis19</td><td>Axis18</td><td>Axis17</td></tr>
</table>
Stores the operating/stopped status of each axis
•0: Stopped
•1: Operating</td></tr>
</table>

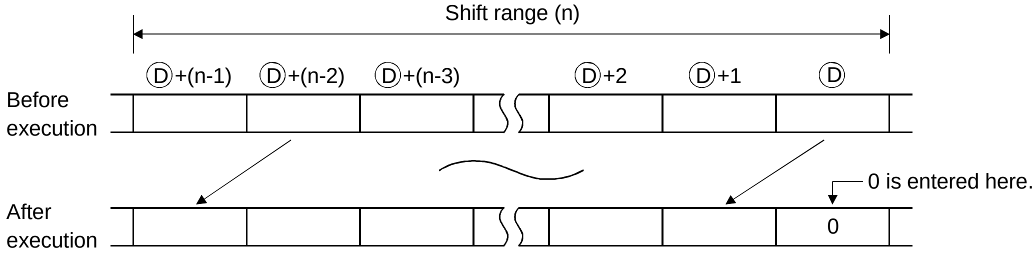
Table 3.2 Special Register List (Continued)

<table>
<tr><th>Number</th><th>Name</th><th>Stored Data</th><th>Explanation</th></tr>
<tr><td>D9185
to
D9187</td><td>Manual pulse generator axis setting error</td><td>Manual pulse generator axis setting error
(For A273UHCPU (32-axis feature)/A173UHCPU(S1))</td><td>● Stores the definitions of manual pulse generator axis setting errors when the manual pulse generator axis setting error flag (M9077) turns ON.
<table>
<tr><th></th><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr>
<tr><td>D9185</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>P3</td><td>P2</td><td>P1</td><td>P3</td><td>P2</td><td>P1</td></tr>
</table>
b15 to b6: All turn to 0.
b2 to b0: Stores the axis setting errors of the manual pulse generators connected to P1 to P3 of A273EX.
• 0: Normal
• 1: Setting error
(Axis setting in any digit is other than 1 to 8)
b5 to b3: Stores the smoothing magnification setting errors of the manual pulse generators connected to P1 to P3 of A273EX.
• 0: Normal
• 1: Setting error
(Axis setting in any digit is other than 1 to 59)
<table>
<tr><td>D9186</td><td>Axis16</td><td>Axis15</td><td>Axis14</td><td>Axis13</td><td>Axis12</td><td>Axis11</td><td>Axis10</td><td>Axis9</td><td>Axis8</td><td>Axis7</td><td>Axis6</td><td>Axis5</td><td>Axis4</td><td>Axis3</td><td>Axis2</td><td>Axis1</td></tr>
<tr><td>D9187</td><td>Axis32</td><td>Axis31</td><td>Axis30</td><td>Axis29</td><td>Axis28</td><td>Axis27</td><td>Axis26</td><td>Axis25</td><td>Axis24</td><td>Axis23</td><td>Axis22</td><td>Axis21</td><td>Axis20</td><td>Axis19</td><td>Axis18</td><td>Axis17</td></tr>
</table>
Stores 1-pulse input magnification setting error of each axis.
• 0: Normal
• 1: Setting error
(Input magnification of any axis is other than 1 to 100)</td></tr>
<tr><td>D9189</td><td>Error program number</td><td>Error program number
(For A273UHCPU (32-axis feature)/A173UHCPU(S1))</td><td>● Stores the motion program number (1 to 256) in error when the motion program setting error flag (M9079) turns ON.
● If an error occurs in another motion program when the error program number is stored, the new error program number is stored.</td></tr>
<tr><td>D9190</td><td>Error item information</td><td>Servo program setting error number
(For A273UHCPU (32-axis feature)/A173UHCPU(S1))</td><td>● Stores the error code corresponding to the setting item in error when the motion program setting error flag (M9079) turns ON.
<table>
<tr><th>Error Code</th><th>Error Definition</th></tr>
<tr><td>1</td><td>The parameter block number specified is outside the range 1 to 16.</td></tr>
<tr><td>906</td><td>The motion program set in the SVST instruction has the unused axis in system settings.</td></tr>
<tr><td>3300</td><td>An attempt was made to start and run 9 or more programs simultaneously with the SVST instruction.</td></tr>
</table>
For the error processings and corrective actions, refer to Appendix 2.1.</td></tr>
<tr><td>D9191
to
D9192</td><td>Servo amplifier loading information</td><td>Servo amplifier loading information
(For A273UHCPU (32-axis feature)/A173UHCPU(S1))</td><td>● Stores the result of servo amplifier and optional slot loading status check made at power-on or reset.
<table>
<tr><th></th><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr>
<tr><td>D9191</td><td>Axis16</td><td>Axis15</td><td>Axis14</td><td>Axis13</td><td>Axis12</td><td>Axis11</td><td>Axis10</td><td>Axis9</td><td>Axis8</td><td>Axis7</td><td>Axis6</td><td>Axis5</td><td>Axis4</td><td>Axis3</td><td>Axis2</td><td>Axis1</td></tr>
<tr><td>D9192</td><td>Axis32</td><td>Axis31</td><td>Axis30</td><td>Axis29</td><td>Axis28</td><td>Axis27</td><td>Axis26</td><td>Axis25</td><td>Axis24</td><td>Axis23</td><td>Axis22</td><td>Axis21</td><td>Axis20</td><td>Axis19</td><td>Axis18</td><td>Axis17</td></tr>
</table>
<table>
<tr><th colspan="2">Servo amplifier loading status</th></tr>
<tr><td>0</td><td>No loading or ADU fault, MR-□-B power off or connection cable fault *1</td></tr>
<tr><td>1</td><td>Servo amplifier loading status</td></tr>
</table>
*1: For the ADU, no loading causes a major error to be displayed if the axis number is set in system settings.</td></tr>
</table>

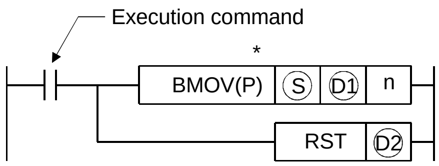
APPENDIX 4 EXAMPLE PROGRAMS

Appendix 4.1 Word Data 1 Word Shift to Left

(1) A program for shifting to the left a range of devices that comprises n points and starts with a designated word device is shown here.



(2) Word data can be shifted one word to the left by using the BMOV (P) instruction and RST instruction.
The format for a program for shifting data one word to the left by using the BMOV (P) instruction and RST instruction is shown in Figure 4.1.



(1) S, D1, D2, and n are explained below.
S, D2 : Head number of the devices to be shifted to the left
D1 : Number of (word device designated by S+1) device
n : Number of shifted devices

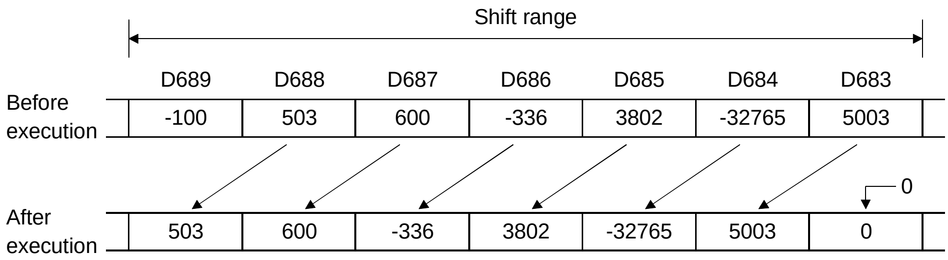
(2) *: BMOV(P) indicates the BMOV instruction and the BMOVP instruction.

Fig.4.1 Format for Left shift Using BMOV(P) Instruction and RST Instruction

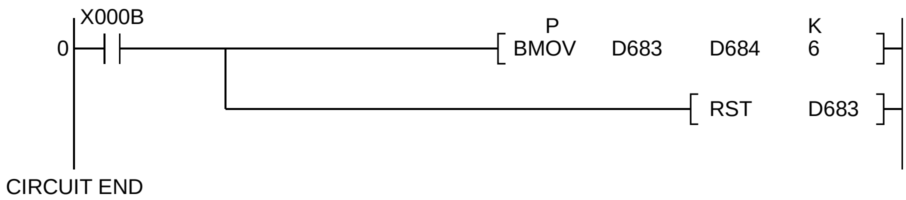
Example

(1) A program that shifts the contents of D683 to D689 one word to the left at the leading edge (OFF to ON) of XB is shown here.

[Operation]

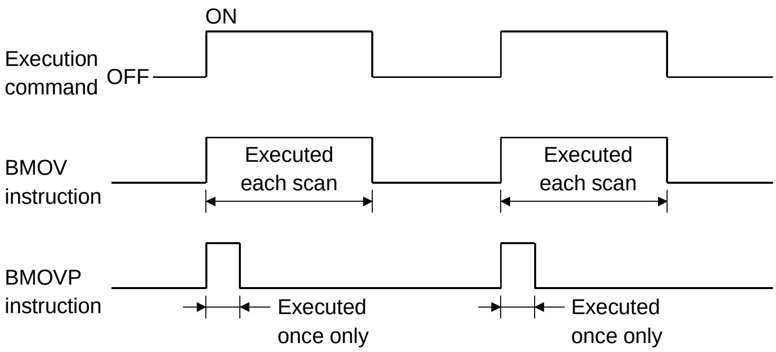


[Program Example]



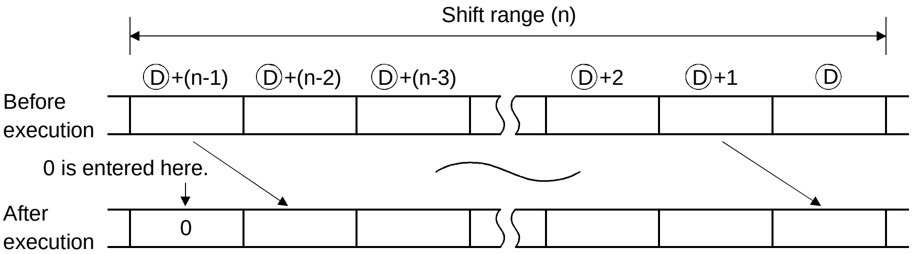
(3) Execution condition

The execution condition when the BMOV instruction and BMOVP instruction are used is as follows.

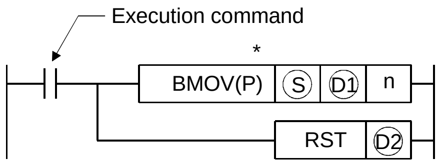


Appendix 4.2 Word Data 1 Word Shift to Right

(1) A program for shifting to the right a range of devices that comprises n points and starts with a designated word device is shown here.



(2) Word data can be shifted one word to the right by using the BMOV (P) instruction and RST instruction.
The format for a program for shifting data one word to the right by using the BMOV (P) instruction and RST instruction is shown in Figure 4.2.



(1) S, D1, D2, and n are explained below.
S : Head number of the devices to be shifted to the right
D1 : Number of (word device designated by S+1) device
D2 : Final device number of word devices to be shifted to be shifted to right
n : Number of shifted devices

(2) *: BMOV(P) indicates the BMOV instruction and the BMOVP instruction.

Fig.4.2. Format for Right Shift Using BMOV(P) Instruction and RST Instruction

Example

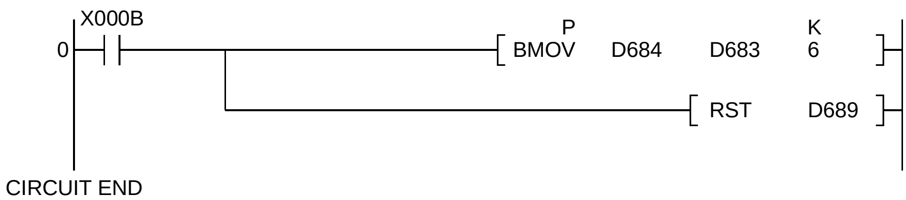
(1) A program that shifts the contents of D683 to D689 one word to the right at the leading edge (OFF to ON) of XB is shown here.

[Operation]

Shift range

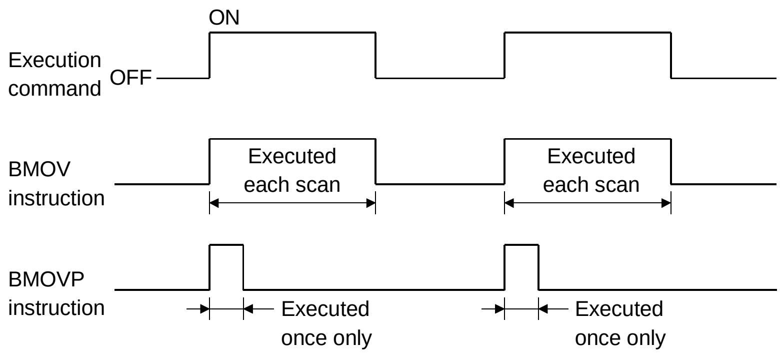
	D689	D688	D687	D686	D685	D684	D683
Before execution	-100	503	600	-336	3802	-32765	5003
After execution	0	-100	503	600	-336	3802	-32765

[Program Example]



(3) Execution condition

The execution condition when the BMOV instruction and BMOVP instruction are used is as follows.



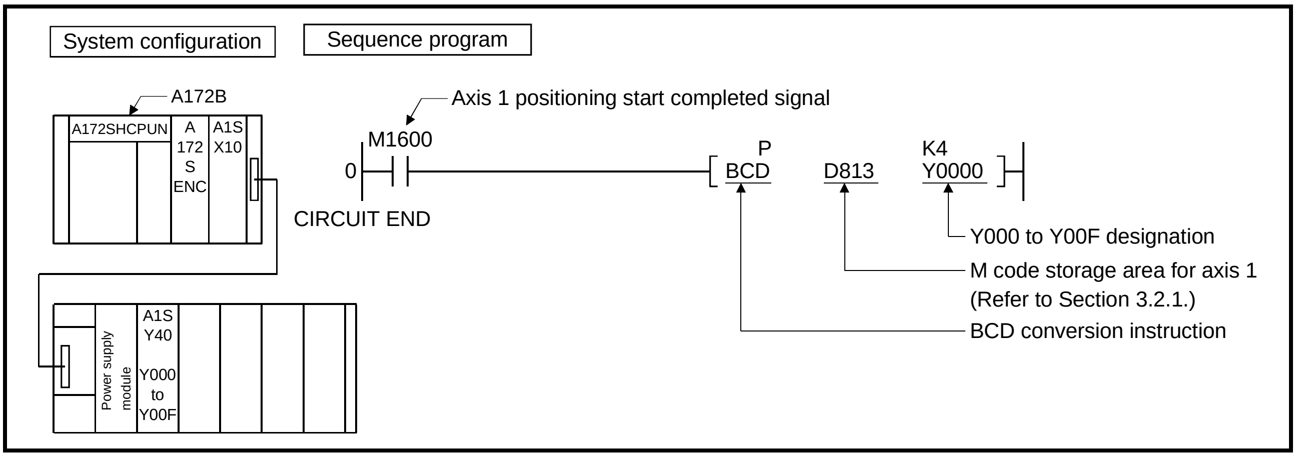
Appendix 4.3 Reading M Codes

An example of a program for reading an M code on completion of positioning start or on completion of positioning is shown here.
The distinction between positioning start completion and positioning completion is made with the following signals.

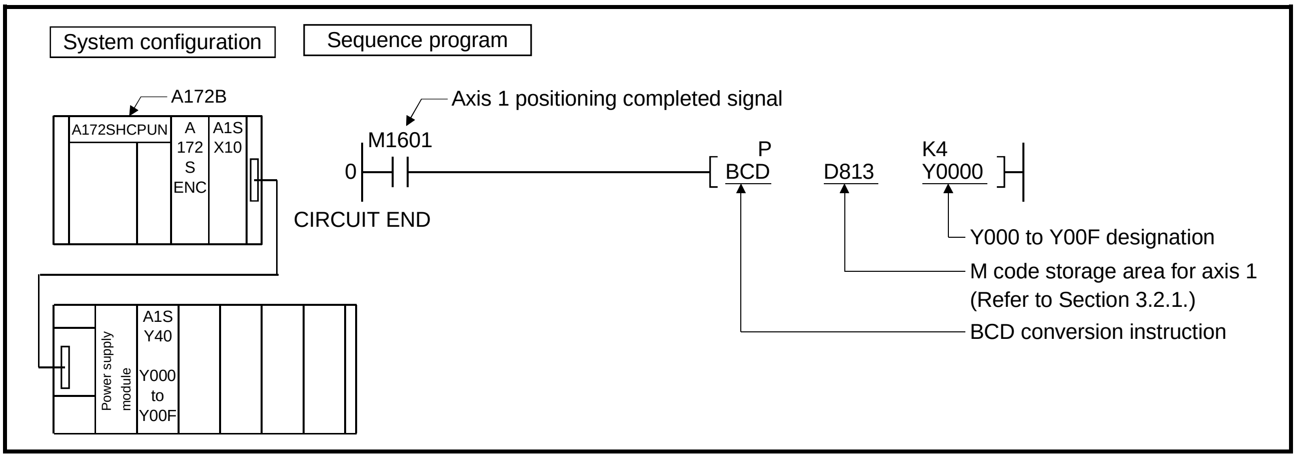
- Positioning start completed ....M1600+20n/M2400+20n (positioning start completed signal)
- Positioning completed..................M1601+20n/M2401+20n (positioning completed signal)

[Program Example]

(1) A program that outputs the M code for axis 1 from Y000 to Y00F to an external destination on completion of positioning start and after conversion to BCD code, is shown here.



(2) A program that outputs the M code for axis 1 from Y000 to Y00F to an external destination on completion of positioning and after conversion to BCD code, is shown here.



Appendix 4.4 Error Code Reading

A program that reads the error code when an error occurs is shown here.
The following signals are used to determine whether or not an error has occurred:

- Minor errors, major errors............Error detection signal (M1607+20n/M2407+20n)
- Servo errors.................................Servo error detection signal (M1608+20n/M2408+20n)

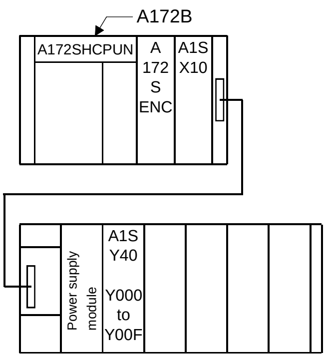
POINT

(1) The following delay occurs between the leading edge (OFF to ON) of M1607+20n/M1608+20n/M2407+20n/M2408+20n and storage of the error code.
  (a) If the sequence program scan time is less than 80 ms, there will be a delay of up to 80 ms.
  (b) If the sequence program scan time is longer than 80 ms, there will be a delay of up to one scan time.
  Program so that error code reading is executed after sufficient time has elapsed for error codes to be written in the various error code storage areas after M1607+20n/M1608+20n/M2407+20n/M2408+20n comes ON.

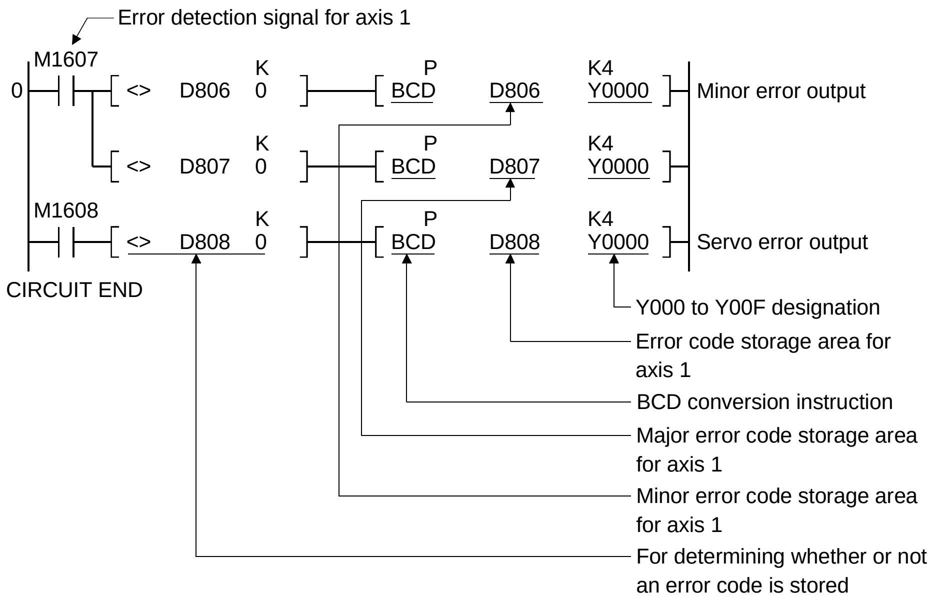
[Program Example]

(1) A program that converts the error code to BCD and outputs it to Y000 to Y00F when an axis 1 error occurs (minor error, major error) is shown here.

System configuration



Sequence program

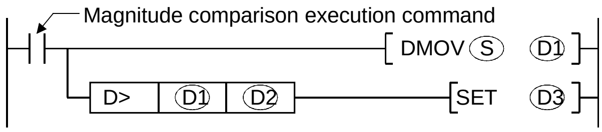


Appendix 4.5 Magnitude Comparison and Four Fundamental Operations of 32-Bit Monitor Data

When a machine value, actual present value or deviation counter value is used to perform magnitude comparison or four fundamental operations, the value must be transferred to another device memory once and the device memory of the transfer destination be used to perform processing as described below.

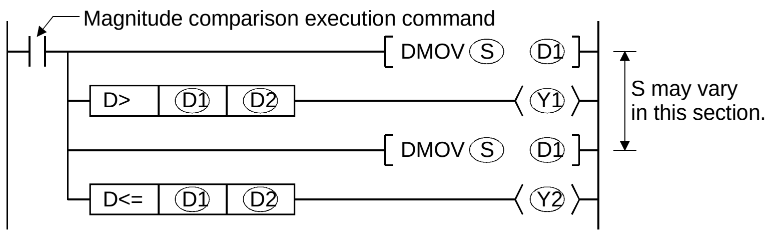
(1) Magnitude comparison example

(a) To set the device when the machine value has become greater than the set value

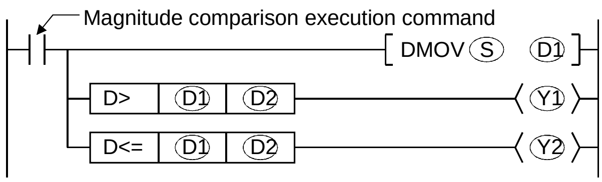


1) S, D1, D2 and D3 indicate the following.
S: Machine value
D1: Device memory for temporary storage
D2: Set value for magnitude comparison
D3: Device for setting magnitude comparison result

(b) When one piece of monitor data is referred to many times to perform comparison processing, intended operation may not be performed if the monitor data is transferred every processing as shown in program example 1.
In program example 1, neither Y1 nor Y2 may not turn ON. (This also applies to the case of 16-bit monitor data.)
This is because the S value varies asynchronously with the PC scan.
To perform such processing, transfer the monitor data to another device memory once, and after that, use that value to perform comparison processing as shown in program example 2.
[Program example 1]

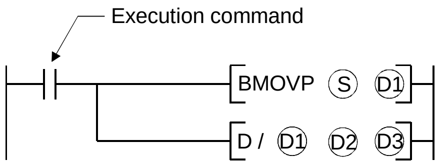


[Program example 2]



1) S, D1, D2, Y1 and Y2 indicate the following.
S: Machine value
D1: Device memory for temporary storage
D2: Set value for magnitude comparison
Y1: Magnitude comparison result output device (Result: Greater than)
Y2: Magnitude comparison result output device (Result: Equal to or less than)

(2) Four fundamental operations example
To divide the actual present value by the set value



1) S, D1, D2 and D3 indicate the following.
S: Actual present value
D1: Device memory for temporary storage
D2: Division
D3: Operation result storage device

APPENDIX 5 SERVO MOTOR TYPE-BASED RATED SPEED AND FEEDBACK PULSE COUNT LIST

Table 5.1 lists the rated speeds and feedback pulse counts on a servo motor type basis.

Table 5.1 Servo Motor Type-Based Rated Speed and Feedback Pulse Count List

<table>
<tr><th>Motor Model</th><th>Rated Speed [rpm]</th><th>Number of Feedback Pulses [PLS]</th><th>Motor Model</th><th>Rated Speed [rpm]</th><th>Number of Feedback Pulses [PLS]</th></tr>
<tr><td>HA-MH053</td><td rowspan="11">3000</td><td rowspan="11">8192</td><td>HA-LH52</td><td rowspan="17">2000</td><td rowspan="17">16384</td></tr>
<tr><td>HA-MH13</td><td>HA-LH102</td></tr>
<tr><td>HA-MH23</td><td>HA-LH152</td></tr>
<tr><td>HA-MH43</td><td>HA-LH202</td></tr>
<tr><td>HA-MH73</td><td>HA-LH302</td></tr>
<tr><td>HA-FH053</td><td>HA-LH502</td></tr>
<tr><td>HA-FH13</td><td>HA-LH702</td></tr>
<tr><td>HA-FH23</td><td>HA-LH11K2</td></tr>
<tr><td>HA-FH33</td><td>HA-LH15K2</td></tr>
<tr><td>HA-FH43</td><td>HA-LH22K2</td></tr>
<tr><td>HA-FH63</td><td>HA-UH32</td></tr>
<tr><td>HA-SH81</td><td rowspan="4">1000</td><td rowspan="19">16384</td><td>HA-UH52</td></tr>
<tr><td>HA-SH121</td><td>HA-UH102</td></tr>
<tr><td>HA-SH201</td><td>HA-UH152</td></tr>
<tr><td>HA-SH301</td><td>HA-UH222</td></tr>
<tr><td>HA-SH52</td><td rowspan="7">2000</td><td>HA-UH352</td></tr>
<tr><td>HA-SH102</td><td>HA-UH452</td></tr>
<tr><td>HA-SH152</td><td>HA-FF053</td><td rowspan="11">3000</td><td rowspan="11">8192</td></tr>
<tr><td>HA-SH202</td><td>HA-FF13</td></tr>
<tr><td>HA-SH352</td><td>HA-FF23</td></tr>
<tr><td>HA-SH502</td><td>HA-FF33</td></tr>
<tr><td>HA-SH702</td><td>HA-FF43</td></tr>
<tr><td>HA-SH53</td><td rowspan="8">3000</td><td>HA-FF63</td></tr>
<tr><td>HA-SH103</td><td>HC-MF053</td></tr>
<tr><td>HA-SH153</td><td>HC-MF13</td></tr>
<tr><td>HA-SH203</td><td>HC-MF23</td></tr>
<tr><td>HA-SH353</td><td>HC-MF43</td></tr>
<tr><td>HA-RH103</td><td>HC-MF73</td></tr>
<tr><td>HA-RH153</td><td>HC-SF52</td><td rowspan="2">2000</td><td rowspan="2">16384</td></tr>
<tr><td>HA-RH223</td><td>HC-SF102</td></tr>
</table>

APPENDIX 6 PROCESSING TIMES

The following tables list the processing time of each instruction for positioning control in the servo system CPU.

(1) Motion operation cycle (ms)

CPU	A172SHCPUN	A171SHCPUN
Number of set axes	1 to 8	1 to 4
Operation cycle	3.5ms	3.5ms

CPU	A273UHCPU (32 axis feature)			A173UHCPU(S1)		
Number of set axes (SV43)	1 to 8	9 to 18	19 to 32	1 to 12	13 to 24	25 to 32
Operation cycle	3.5ms	7.1ms	14.2ms	3.5ms	7.1ms	14.2ms

(2) SCPU instruction processing time (μs)

CPU		A172SHCPUN	A171SHCPUN	A273UHCPU (32 axis feature)	A173UHCPU (S1)
Number of set axes		1 to 8	1 to 4	1 to 32	
SVST	1 axis started	48		35	
	2 or 3 axes started	105		70	
	Error	50		150	
DSFRP	1 axis started	48			
	2 to 4 axes started	65			
	Error	60			
CHGV		27		20	
DSFLP (speed change)	Normal	28			
	Error	50			
CHGA		32		25	
DSFLP (present value change)	Normal	28			
	Error	50			
CHGT		24		20	
END		1400		Max.5000	

(3) CPU processing time (ms)

CPU	A172SHCPUN	A171SHCPUN
Number of set axes	1 to 8	1 to 4
Servo program start processing time	4 to 11	4 to 11
Speed change response	0 to 4	0 to 4
Torque limit value change response	0 to 4	0 to 4
Simultaneous start processing time (*1)	7 to 17	7 to 17
Time from PC ready flag (M2000) ON to PCPU ready flag (M9074) ON	50 to 600	50 to 350

CPU	A273UHCPU (32 axis feature)			A173UHCPU(S1)		
Number of set axes (SV43)	1 to 8	9 to 18	19 to 32	1 to 12	13 to 24	25 to 32
Servo program start processing time	4 to 11	10 to 18	14 to 21	4 to 11	10 to 18	14 to 21
Speed change response	0 to 4	0 to 8	0 to 14	0 to 4	0 to 8	0 to 14
Torque limit value change response	0 to 4	0 to 4	0 to 4	0 to 4	0 to 4	0 to 4
Simultaneous start processing time (*1)	7 to 17	10 to 24	14 to 28	7 to 17	10 to 24	14 to 28
Time from PC ready flag (M2000) ON to PCPU ready flag (M9074) ON	8 to 100	90 to 400	100 to 800	8 to 100	90 to 400	100 to 800

(*1) This processing time varies depending on the commands to be started simultaneously. Use this time merely for reference.

For other sequence program instruction processing times, refer to the ACPU Programming Manual.

(4) Axis status

• Axis status for SV43

Axis No.	A172SHCPUN Device Number	A171SHCPUN Device Number
1	M1400 to M1409	M1400 to M1409
2	M1410 to M1419	M1410 to M1419
3	M1420 to M1429	M1420 to M1429
4	M1430 to M1439	M1430 to M1439
5	M1440 to M1449	
6	M1450 to M1459	
7	M1460 to M1469	
8	M1470 to M1479	

Signal Name

	Signal Name	Fetch Cycle	Refresh Cycle	Signal Direction
0	Unusable		–	SCPU ← PCPU
1	Unusable			
2	Automatically operating		10ms	
3	Temporarily stopping			
4	Unusable		–	
5	Unusable			
6	Unusable			
7	Unusable			
8	Unusable			
9	Single block mode in progress (*1)		3.5ms	

(*1) The single block in progress is not an axis status. It is used with the first axis (M1409) only. The user cannot use it for other than the first axis.

• Axis status

Axis No.	A172SHCPUN Device Number	A171SHCPUN Device Number
1	M1600 to M1619	M1600 to M1619
2	M1620 to M1639	M1620 to M1639
3	M1640 to M1659	M1640 to M1659
4	M1660 to M1679	M1660 to M1679
5	M1680 to M1699	
6	M1700 to M1719	
7	M1720 to M1739	
8	M1740 to M1759	

Signal Name

	Signal Name	Fetch Cycle	Refresh Cycle	Signal Direction
0	Positioning start completed		3.5ms	SCPU ← PCPU
1	Positioning completed			
2	In-position			
3	Command in-position			
4	Unusable			
5	Unusable			
6	Zero pass			
7	Error detection		Immediately	
8	Servo error detection		3.5ms	
9	Home position return request		10ms	
10	Home position return completed		3.5ms	
11	External signal FLS		10ms	
12	External signal RLS			
13	External signal STOP			
14	External signal DOG/CHANGE			
15	Servo ON/OFF		3.5ms	
16	Torque control in progress			
17	(External signal DOG/CHANGE)		10ms	
18	Unusable		——	
19	M code output in progress		3.5ms	

(4) Axis status

• Axis status

Axis No.	A273UHCPU (32 axis feature) A173UHCPU (S1) Device No.
1	M2400 to M2419
2	M2420 to M2439
3	M2440 to M2459
4	M2460 to M2479
5	M2480 to M2499
6	M2500 to M2519
7	M2520 to M2539
8	M2540 to M2559
9	M2560 to M2579
10	M2580 to M2599
11	M2600 to M2619
12	M2620 to M2639
13	M2640 to M2659
14	M2660 to M2679
15	M2680 to M2699
16	M2700 to M2719
17	M2720 to M2739
18	M2740 to M2759
19	M2760 to M2779
20	M2780 to M2799
21	M2800 to M2819
22	M2820 to M2839
23	M2840 to M2859
24	M2860 to M2879
25	M2880 to M2899
26	M2900 to M2919
27	M2920 to M2939
28	M2940 to M2959
29	M2960 to M2979
30	M2980 to M2999
31	M3000 to M3019
32	M3020 to M3039

Single name

	Signal name	Refresh cycle			Fetch cycle			Signal direction
		Set number of axis			Set number of axis			
	SV43 A173UHCPU	1 to 12	13 to 24	25 to 32	1 to 12	13 to 24	25 to 32	
	SV43 A273UHCPU	1 to 8	9 to 18	19 to 32	1 to 8	9 to 18	19 to 32	
0	Positioning start completed	3.5ms	7.1ms	14.2ms				SCPU ← PCPU
1	Positioning completed							
2	In-position							
3	Command in-position							
4	Unusable							
5	Unusable							
6	Zero pass							
7	Error detection	Immediately						
8	Servo error detection	3.5ms	7.1ms	14.2ms				
9	Home position return request	10ms	20ms					
10	Home position return completed	3.5ms	7.1ms	14.2ms				
11	External signal FLS	10ms	20ms					
12	External signal RLS							
13	External signal STOP							
14	External signal DOG							
15	Servo ON/OFF	3.5ms	7.1ms	14.2ms				
16	Torque control in progress							
17	(External signal CHANGE)	10ms	20ms					
18	Unusable							
19	M code output in progress	3.5ms	7.1ms	14.2ms				

(4) Axis status

• Axis status for SV43

Axis No.	A273UHCPU (32 axis feature) A173UHCPU (S1) Device No.	Single name
1	M4000 to M4009	
2	M4010 to M4019	
3	M4020 to M4029	
4	M4030 to M4039	
5	M4040 to M4049	
6	M4050 to M4059	
7	M4060 to M4069	
8	M4070 to M4079	
9	M4080 to M4089	
10	M4090 to M4099	
11	M4100 to M4109	
12	M4110 to M4119	
13	M4120 to M4129	
14	M4130 to M4139	
15	M4140 to M4149	
16	M4150 to M4159	
17	M4160 to M4169	
18	M4170 to M4179	
19	M4180 to M4189	
20	M4190 to M4199	
21	M4200 to M4209	
22	M4210 to M4219	
23	M4220 to M4229	
24	M4230 to M4239	
25	M4240 to M4249	
26	M4250 to M4259	
27	M4260 to M4269	
28	M4270 to M4279	
29	M4280 to M4289	
30	M4290 to M4299	
31	M4300 to M4309	
32	M4310 to M4319	

	Signal name		Refresh cycle			Fetch cycle			Signal direction
			Set number of axis			Set number of axis			
	SV43	A173 UHCPU	1 to 12	13 to 24	25 to 32	1 to 12	13 to 24	25 to 32	
		A273 UHCPU	1 to 8	9 to 18	19 to 32	1 to 8	9 to 18	19 to 32	
0	Unusable		–						SCPU ← PCPU
1	Unusable								
2	Automatically operating		10ms	20ms					
3	Temporarily stopping								
4	Unusable		–						
5	Unusable								
6	Unusable								
7	Unusable								
8	Unusable								
9	Single block mode in progress (*1)		3.5ms	7.1ms	14.2ms				

(*1) The single block in progress is not an axis status. It is used with the first axis (M4009) only. The user cannot use it for other than the first axis.

(5) Axis command signals

• Axis command signals for SV43

Axis No.	A172SHCPUN Device Number	A171SHCPUN Device Number
1	M1500 to M1509	M1500 to M1509
2	M1510 to M1519	M1510 to M1519
3	M1520 to M1529	M1520 to M1529
4	M1530 to M1539	M1530 to M1539
5	M1540 to M1549	
6	M1550 to M1559	
7	M1560 to M1569	
8	M1570 to M1579	

Signal Name

	Signal Name	Fetch Cycle	Refresh Cycle	Signal Direction
0	Temporary stop command	3.5ms		SCPU → PCPU
1	Optional program stop	At start		
2	Optional block skip			
3	Single block			
4	Restart	3.5ms		
5	Override valid/invalid			
6	Unusable	–		
7	Unusable			
8	Single block mode (*1)			
9	Single block start (*1)			

(*1) The single block mode and single block start are not axis statuses. They are used with the first axis (M1508, M1509) only. The user cannot use them for other than the first axis.

• Axis command signals

Axis No.	A172SHCPUN Device Number	A171SHCPUN Device Number
1	M1800 to M1819	M1800 to M1819
2	M1820 to M1839	M1820 to M1839
3	M1840 to M1859	M1840 to M1859
4	M1860 to M1879	M1860 to M1879
5	M1880 to M1899	
6	M1900 to M1919	
7	M1920 to M1939	
8	M1940 to M1959	

Signal Name

	Signal Name	Fetch Cycle	Refresh Cycle	Signal Direction
0	Stop command	3.5ms		SCPU → PCPU
1	Rapid stop command			
2	Forward rotation JOG command	10ms		
3	Reverse rotation JOG command			
4	Completion signal OFF command			
5	Unusable	–		
6	Limit switch output enable	3.5ms		
7	Error reset	10ms		
8	Servo error reset			
9	Start-time stop input invalid	At start		
10	Unusable	–		
11	Unusable			
12	Unusable			
13	Unusable			
14	Unusable			
15	Servo OFF	3.5ms		
16	Unusable	–		
17	Unusable			
18	Unusable			
19	FIN signal	3.5ms		

(5) Axis command signals

• Axis command signals

Axis No.	A273UHCPU (32 axis feature) A173UHCPU (S1) Device No.
1	M3200 to M3219
2	M3220 to M3239
3	M3240 to M3259
4	M3260 to M3279
5	M3280 to M3299
6	M3300 to M3319
7	M3320 to M3339
8	M3340 to M3359
9	M3360 to M3379
10	M3380 to M3399
11	M3400 to M3419
12	M3420 to M3439
13	M3440 to M3459
14	M3460 to M3479
15	M3480 to M3499
16	M3500 to M3519
17	M3520 to M3539
18	M3540 to M3559
19	M3560 to M3579
20	M3580 to M3599
21	M3600 to M3619
22	M3620 to M3639
23	M3640 to M3659
24	M3660 to M3679
25	M3680 to M3699
26	M3700 to M3719
27	M3720 to M3739
28	M3740 to M3759
29	M3760 to M3779
30	M3780 to M3799
31	M3800 to M3819
32	M3820 to M3839

Single name

	Signal name	Refresh cycle			Fetch cycle			Signal direction
		Set number of axis			Set number of axis			
	SV43 A173UHCPU	1 to 12	13 to 24	25 to 32	1 to 12	13 to 24	25 to 32	
	SV43 A273UHCPU	1 to 8	9 to 18	19 to 32	1 to 8	9 to 18	19 to 32	
0	Stop command				3.5ms	7.1ms	14.2ms	SCPU → PCPU
1	Rapid stop command							
2	Forward rotation JOG command				10ms	20ms		
3	Reverse rotation JOG command							
4	Completion signal OFF command							
5	Unusable				3.5ms	7.1ms	14.2ms	
6	Limit switch output enable							
7	Error reset				10ms	20ms		
8	Servo error reset							
9	Start-time stop input invalid				At start			
10	Unusable				–			
11	Unusable							
12	Present feed value update request command				At start			
13	Unusable				–			
14	Unusable							
15	Servo OFF				3.5ms	7.1ms	14.2ms	
16	Unusable				–			
17	Unusable							
18	Unusable							
19	FIN signal				3.5ms	7.1ms	14.2ms	

(5) Axis command signals

• Axis command signals for SV43

Axis No.	A273UHCPU (32 axis feature) A173UHCPU (S1) Device No.	Single name
1	M4400 to M4409	
2	M4410 to M4419	
3	M4420 to M4429	
4	M4430 to M4439	
5	M4440 to M4449	
6	M4450 to M4459	
7	M4460 to M4469	
8	M4470 to M4479	
9	M4480 to M4489	
10	M4490 to M4499	
11	M4500 to M4509	
12	M4510 to M4519	
13	M4520 to M4529	
14	M4530 to M4539	
15	M4540 to M4549	
16	M4550 to M4559	
17	M4560 to M4569	
18	M4570 to M4579	
19	M4580 to M4589	
20	M4590 to M4599	
21	M4600 to M4609	
22	M4610 to M4619	
23	M4620 to M4629	
24	M4630 to M4639	
25	M4640 to M4649	
26	M4650 to M4659	
27	M4660 to M4669	
28	M4670 to M4679	
29	M4680 to M4689	
30	M4690 to M4699	
31	M4700 to M4709	
32	M4710 to M4719	

	Signal name		Refresh cycle			Fetch cycle			Signal direction
			Set number of axis			Set number of axis			
		SV43 A173UHCPU	1 to 12	13 to 24	25 to 32	1 to 12	13 to 24	25 to 32	
		SV43 A273UHCPU	1 to 8	9 to 18	19 to 32	1 to 8	9 to 18	19 to 32	
0	Temporary stop command		3.5ms	7.1ms	14.2ms				SCPU → PCPU
1	Optional program stop		At start						
2	Optional block skip								
3	Single block								
4	Restart		3.5ms	7.1ms	14.2ms				
5	Override valid/invalid								
6	Unusable		–						
7	Unusable								
8	Single block mode (*1)								
9	Single block start (*1)								

(*1) The single block mode and single block start are not axis statuses. They are used with the first axis (M4408, M4409) only. The user cannot use them for other than the first axis.

(6) Axis monitor devices

Axis No.	A172SHCPUN Device No.	A171SHCPUN Device No.
1	D600 to D619	D600 to D619
2	D620 to D639	D620 to D639
3	D640 to D659	D640 to D659
4	D660 to D679	D660 to D679
5	D680 to D699	
6	D700 to D719	
7	D720 to D739	
8	D740 to D759	

Signal name

	Signal name	Refresh cycle	Fetch cycle	Unit	Signal direction
0 1	Current value	END		Command unit	SCPU← PCPU
2	Execution sequence No. (main)			–	
3	Execution block No. (main)			–	
4	Execution program No. (sub)			–	
5	Execution sequence No. (sub)			–	
6	Execution block No. (sub)			–	
7	Unusable	–		–	
8	G43/44 command	END		–	
9	Tool length offset data No.			–	
10 11	Tool length offset			Command unit	
12	Unusable	–		–	
13	Unusable			–	
14	Unusable			–	
15	Unusable			–	
16	Unusable			–	
17	Unusable			–	
18	Unusable			–	
19	Unusable			–	

Axis No.	A172SHCPUN Device No.	A171SHCPUN Device No.
1	D800 to D819	D800 to D819
2	D820 to D839	D820 to D839
3	D840 to D859	D840 to D859
4	D860 to D879	D860 to D879
5	D880 to D899	
6	D900 to D919	
7	D920 to D939	
8	D940 to D959	

Signal name

	Signal name	Refresh cycle	Fetch cycle	Unit	Signal direction
0 1	Machine value	3.5ms		Command unit	SCPU← PCPU
2 3	Actual current value			Command unit	
4 5	Deviation counter value			PLS	
6	Minor error code	Immedi-ately		–	
7	Major error code			–	
8	Servo error code	10ms		–	
9 10	Travel after DOG/CHANGE ON	END		Command unit	
11	Home position return second travel	3.5ms		PLS	
12	Execution program No.			–	
13	M code			–	
14	Torque limit value			%	
15	Unusable	–		–	
16	Unusable			–	
17 18	Actual present value at STOP input	END		Command unit	
19	Unusable	–		–	

* The entry "END" in the Refresh Cycle column indicates 80ms or a longer sequence program scan time.

(6) Axis monitor device

• Axis monitor device

Axis No.	A273UHCPU (32 axis feature)/ A173UHCPU(S1) Device No.
1	D0 to D19
2	D20 to D39
3	D40 to D59
4	D60 to D79
5	D80 to D99
6	D100 to D119
7	D120 to D139
8	D140 to D159
9	D160 to D179
10	D180 to D199
11	D200 to D219
12	D220 to D239
13	D240 to D259
14	D260 to D279
15	D280 to D299
16	D300 to D319
17	D320 to D339
18	D340 to D359
19	D360 to D379
20	D380 to D399
21	D400 to D419
22	D420 to D439
23	D440 to D459
24	D460 to D479
25	D480 to D499
26	D500 to D519
27	D520 to D539
28	D540 to D559
29	D560 to D579
30	D580 to D599
31	D600 to D619
32	D620 to D639

Signal name

	Signal name		Refresh cycle			Fetch cycle			Unit	Signal direction
			Set No. of axis			Set No. of axis				
	SV43	A173UHCPU	1 to 12	13 to 24	25 to32	1 to 12	13 to 24	25 to32		
		A273UHCPU	1 to 8	9 to 18	19 to 32	1 to 8	9 to 18	19 to 32		
0 1	Machine value		3.5ms	7.1ms	14.2ms				Command Unit	SCPU ← PCPU
2 3	Actual current value								Command Unit	
4 5	Deviation counter value								PLS	
6	Minor error code		Immediately						–	
7	Major error code								–	
8	Servo error code		10ms	20ms					–	
9	Home position return second Travel		3.5ms	7.1ms	14.2ms				PLS	
10 11	Travel after DOG/CHANGE ON		END						Command unit	
12	Execution program No.		At start						–	
13	M code		3.5ms	7.1ms	14.2ms				–	
14	Torque limit value								%	
15	Unusable		–						–	
16	Unusable								–	
17	Unusable								–	
18 19	Actual present value at stop input		END						Command unit	

*"END" in Refresh Cycle indicates a longer one of "50ms" and "sequence program scan time".

(6) Axis monitor device

• Axis monitor device for SV43

Axis No.	A273UHCPU (32 axis feature)/ A173UHCPU(S1) Device No.
1	D800 to D819
2	D820 to D839
3	D840 to D859
4	D860 to D879
5	D880 to D899
6	D900 to D919
7	D920 to D939
8	D940 to D959
9	D960 to D979
10	D980 to D999
11	D1000 to D1019
12	D1020 to D1039
13	D1040 to D1059
14	D1060 to D1079
15	D1080 to D1099
16	D1100 to D1119
17	D1120 to D1139
18	D1140 to D1159
19	D1160 to D1179
20	D1180 to D1199
21	D1200 to D1219
22	D1220 to D1239
23	D1240 to D1259
24	D1260 to D1279
25	D1280 to D1299
26	D1300 to D1319
27	D1320 to D1339
28	D1340 to D1359
29	D1360 to D1379
30	D1380 to D1399
31	D1400 to D1419
32	D1420 to D1439

Signal name

	Signal name		Refresh cycle			Fetch cycle			Unit	Signal direction
			Set No. of axis			Set No. of axis				
	SV43	A173UHCPU	1 to 12	13 to 24	25 to32	1 to 12	13 to 24	25 to32		
		A273UHCPU	1 to 8	9 to 18	19 to 32	1 to 8	9 to 18	19 to 32		
0 1	Current value		END						Command Unit	SCPU ← PCPU
2	Execution sequence No. (main)								–	
3	Execution block No. (main)								–	
4	Execution program No. (sub)								–	
5	Execution sequence No. (sub)								–	
6	Execution block No. (sub)								–	
7	Unusable		–						–	
8	G43/G44 command		END						–	
9	Tool length offset data No.								–	
10 11	Tool length offset								Command unit	
12	Unusable		–						–	
13	Unusable								–	
14	Unusable								–	
15	Unusable								–	
16	Unusable								–	
17	Unusable								–	
18	Unusable								–	
19	Unusable								–	

*"END" in Refresh Cycle indicates a longer one of "50ms" and "sequence program scan time".

(7) Control change register

Axis No.	A172SHCPUN Device No.	A171SHCPUN Device No.	Signal name
1	D500 to D505	D500 to D505	(see table below)
2	D506 to D511	D506 to D511	
3	D512 to D517	D512 to D517	
4	D518 to D523	D518 to D523	
5	D524 to D529		
6	D530 to D535		
7	D536 to D541		
8	D542 to D547		
	D548 to D559	D524 to D559	Unusable

	Signal name	Refresh cycle	Fetch cycle	Unit	Signal direction
0	Override ratio setting register		3.5ms	%	SCPU → PCPU
1	Unusable		–	–	
2	Unusable			–	
3	Unusable			–	
4	Unusable			–	
5	Unusable			–	

Axis No.	A172SHCPUN Device No.	A171SHCPUN Device No.	Signal name
1	D960 to D965	D960 to D965	(see table below)
2	D966 to D971	D966 to D971	
3	D972 To D977	D972 To D977	
4	D78 to D983	D78 to D983	
5	D984 to D989		
6	D990 to D995		
7	D996 to D1001		
8	D1002 to D1007		

	Signal name	Refresh cycle	Fetch cycle	Unit	Signal direction
0	Unusable			–	SCPU → PCPU
1	Unusable			–	
2	Speed change flag		At DSFLP execution	Command unit	
3					
4	JOG speed setting register *1		At start	Command unit	
5					

(*1) ☐ indicates the backup register.

(7) Control change register

• Control change register

Axis No.	A273UHCPU (32 axis feature)/ A173UHCPU(S1) 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			Fetch cycle			Unit	Signal direction
				Set No. of axis			Set No. of axis				
		SV43	A173UHCPU	1 to 12	13 to 24	25 to32	1 to 12	13 to 24	25 to32		
			A273UHCPU	1 to 8	9 to 18	19 to 32	1 to 8	9 to 18	19 to 32		
0 1	JOG speed setting register						At start			Command unit	SCPU → PCPU

(7) Control change register

• Control change register for SV43

Axis No.	A273UHCPU (32 axis feature)/ A173UHCPU(S1) Device No.	Signal name
1	D1440 to D1445	
2	D1446 to D1451	
3	D1452 to D1457	
4	D1458 to D1463	
5	D1464 to D1469	
6	D1470 to D1475	
7	D1476 to D1481	
8	D1482 to D1487	
9	D1488 to D1493	
10	D1494 to D1499	
11	D1500 to D1505	
12	D1506 to D1511	
13	D1512 to D1517	
14	D1518 to D1523	
15	D1524 to D1529	
16	D1530 to D1535	
17	D1536 to D1541	
18	D1542 to D1547	
19	D1548 to D1553	
20	D1554 to D1559	
21	D1560 to D1565	
22	D1566 to D1571	
23	D1572 to D1577	
24	D1578 to D1583	
25	D1584 to D1589	
26	D1590 to D1595	
27	D1596 to D1601	
28	D1602 to D1607	
29	D1608 to D1613	
30	D1614 to D1619	
31	D1620 to D1625	
32	D1626 to D1631	

	Signal name		Refresh cycle			Fetch cycle			Unit	Signal direction
			Set No. of axis			Set No. of axis				
	SV43	A173UHCPU	1 to 12	13 to 24	25 to32	1 to 12	13 to 24	25 to32		
		A273UHCPU	1 to 8	9 to 18	19 to 32	1 to 8	9 to 18	19 to 32		
0	Override ratio setting register					3.5ms	7.1ms	14.2ms	%	SCPU → PCPU
1	Unusable					–			–	
2	Unusable					–			–	
3	Unusable					–			–	
4	Unusable					–			–	
5	Unusable					–			–	

(8) Common devices

A172SHCPUN

Device Number	Signal Name		Fetch Cycle	Refresh Cycle	Signal Direction
M1960	Unusable (40 points)		–	–	–
M1961					
M1962					
M1963					
M1964					
M1965					
M1966					
M1967					
M1968					
M1969					
M1970					
M1971					
M1972					
M1973					
M1974					
M1975					
M1976					
M1977					
M1978					
M1979					
M1980					
M1981					
M1982					
M1983					
M1984					
M1985					
M1986					
M1987					
M1988					
M1989					
M1990					
M1991					
M1992					
M1993					
M1994					
M1995					
M1996					
M1997					
M1998					
M1999					
M2000	PC READY flag		10ms		SCPU→PCPU
M2001	Axis 1	START accept flag (8 points)		10ms	SCPU←PCPU
M2002	Axis 2				
M2003	Axis 3				
M2004	Axis 4				
M2005	Axis 5				
M2006	Axis 6				
M2007	Axis 7				
M2008	Axis 8				
M2009	All-axes servo ON accept flag				
M2010	Unusable (2 points)		–	–	–
M2011					
M2012	Manual pulse generator enable flag		10ms		SCPU→PCPU
M2013	Unusable (2 points)		–	–	–
M2014					
M2015	JOG simultaneous start command		10ms		SCPU→PCPU
M2016	Unusable (4 points)		–	–	–
M2017					
M2018					
M2019					
M2020	Start buffer full			END	SCPU←PCPU
M2021	Axis 1	Speed change flag (8 points)			
M2022	Axis 2				
M2023	Axis 3				
M2024	Axis 4				
M2025	Axis 5				
M2026	Axis 6				
M2027	Axis 7				
M2028	Axis 8				
M2029	Unusable (5 points)		–	–	–
M2030					
M2031					
M2032					
M2033					
M2034	PC link communication error flag			END	SCPU←PCPU
M2035	Unusable (6 points)		–	–	–
M2036					
M2037					
M2038					
M2039					
M2040					
M2041	System setting error flag			END	SCPU←PCPU
M2042	All-axes servo ON command		3.5ms		SCPU→PCPU
M2043	Unusable (4 points)		–	–	–
M2044					
M2045					
M2046					
M2047	Motion slot module error detection flag			END	SCPU←PCPU

A172SHCPUN

Device Number	Signal Name		Fetch Cycle	Refresh Cycle	Signal Direction
M1960	Unusable (40 points)		–	–	–
M1961					
M1962					
M1963					
M1964					
M1965					
M1966					
M1967					
M1968					
M1969					
M1970					
M1971					
M1972					
M1973					
M1974					
M1975					
M1976					
M1977					
M1978					
M1979					
M1980					
M1981					
M1982					
M1983					
M1984					
M1985					
M1986					
M1987					
M1988					
M1989					
M1990					
M1991					
M1992					
M1993					
M1994					
M1995					
M1996					
M1997					
M1998					
M1999					
M2000	PC READY flag		10ms		SCPU→PCPU
M2001	Axis 1	START accept flag (4 points)		10ms	SCPU←PCPU
M2002	Axis 2				
M2003	Axis 3				
M2004	Axis 4				
M2005	Unusable (4 points)		–	–	–
M2006					
M2007					
M2008					
M2009	All-axes servo ON accept flag			10ms	SCPU←PCPU
M2010	Unusable (2 points)		–	–	–
M2011					
M2012	Manual pulse generator enable flag		10ms		SCPU→PCPU
M2013	Unusable (2 points)		–	–	–
M2014					
M2015	JOG simultaneous start command		10ms		SCPU→PCPU
M2016	Unusable (4 points)		–	–	–
M2017					
M2018					
M2019					
M2020	Start buffer full			END	SCPU←PCPU
M2021	Axis 1	Speed change flag (4 points)			
M2022	Axis 2				
M2023	Axis 3				
M2024	Axis 4				
M2025	Unusable (9 points)		–	–	–
M2026					
M2027					
M2028					
M2029					
M2030					
M2031					
M2032					
M2033					
M2034	PC link communication error flag			END	SCPU←PCPU
M2035	Unusable (6 points)		–	–	–
M2036					
M2037					
M2038					
M2039					
M2040					
M2041	System setting error flag			END	SCPU←PCPU
M2042	All-axes servo ON command		3.5ms		SCPU→PCPU
M2043	Unusable (4 points)		–	–	–
M2044					
M2045					
M2046					
M2047	Motion slot module error detection flag			END	SCPU←PCPU

* The entry "END" in the Refresh Cycle column indicates 80ms or a longer sequence program scan time.

(8) Common devices (A273UHCPU(32 axis feature)/A173UHCPU(S1))

Device No.	Signal name (SV43)		Refresh cycle, Set No. of axis: A173UHCPU 1 to 12 / A273UHCPU 1 to 8	A173UHCPU 13 to 24 / A273UHCPU 9 to 18	A173UHCPU 25 to32 / A273UHCPU 19 to 32	Fetch cycle, Set No. of axis: A173UHCPU 1 to 12 / A273UHCPU 1 to 8	A173UHCPU 13 to 24 / A273UHCPU 9 to 18	A173UHCPU 25 to32 / A273UHCPU 19 to 32	Signal direction
M2000	PLC READY flag					10ms	20ms		SCPU → PCPU
M2001	Axis1	Start accept flag	10ms						SCPU ← PCPU
M2002	Axis2								
M2003	Axis3								
M2004	Axis4								
M2005	Axis5								
M2006	Axis6								
M2007	Axis7								
M2008	Axis8								
M2009	Axis9								
M2010	Axis10								
M2011	Axis11								
M2012	Axis12								
M2013	Axis13								
M2014	Axis14								
M2015	Axis15								
M2016	Axis16								
M2017	Axis17								
M2018	Axis18								
M2019	Axis19								
M2020	Axis20								
M2021	Axis21								
M2022	Axis22								
M2023	Axis23								
M2024	Axis24								
M2025	Axis25								
M2026	Axis26								
M2027	Axis27								
M2028	Axis28								
M2029	Axis29								
M2030	Axis30								
M2031	Axis31								
M2032	Axis32								
M2033	Unusable		–			–			–
M2034	PC link communication error flag		10ms						SCPU ← PCPU
M2035	Unusable (6 points)		–			–			–
M2036									
M2037									
M2038									
M2039									
M2040									
M2041	System setting error flag		10ms						SCPU ← PCPU
M2042	All axes servo ON command					3.5ms	7.1ms	14.2ms	SCPU → PCPU
M2043	Unusable (4 points)		–			–			–
M2044									
M2045									
M2046									
M2047	Motion slot module error detection flag		10ms						SCPU ← PCPU
M2048	JOG simultaneous start command					10ms	20ms		SCPU → PCPU
M2049	All axes servo ON accept flag		END						SCPU ← PCPU
M2050	Start buffer full								
M2051	Manual pulse generator 1 enable flag					10ms	20ms		SCPU → PCPU
M2052	Manual pulse generator 2 enable flag								
M2053	Manual pulse generator 3 enable flag								
M2054	Unusable (7 points)		–			–			–
M2055									
M2056									
M2057									
M2058									
M2059									
M2060									
M2061	Axis1	Speed change flag	END						SCPU ← PCPU
M2062	Axis2								
M2063	Axis3								
M2064	Axis4								
M2065	Axis5								
M2066	Axis6								
M2067	Axis7								
M2068	Axis8								
M2069	Axis9								
M2070	Axis10								
M2071	Axis11								
M2072	Axis12								
M2073	Axis13								
M2074	Axis14								
M2075	Axis15								
M2076	Axis16								
M2077	Axis17								
M2078	Axis18								
M2079	Axis19								
M2080	Axis20	Speed change flag	END						SCPU ← PCPU
M2081	Axis21								
M2082	Axis22								
M2083	Axis23								
M2084	Axis24								
M2085	Axis25								
M2086	Axis26								
M2087	Axis27								
M2088	Axis28								
M2089	Axis29								
M2090	Axis30								
M2091	Axis31								
M2092	Axis32								
M2093	Unusable (35 points)		–			–			–
M2094									
M2095									
M2096									
M2097									
M2098									
M2099									
M2100									
M2101									
M2102									
M2103									
M2104									
M2105									
M2106									
M2107									
M2108									
M2109									
M2110									
M2111									
M2112									
M2113									
M2114									
M2115									
M2116									
M2117									
M2118									
M2119									
M2120									
M2121									
M2122									
M2123									
M2124									
M2125									
M2126									
M2127									
M2128	Axis1	Automatically decelerating flag	3.5ms	7.1ms	14.2ms				SCPU ← PCPU
M2129	Axis2								
M2130	Axis3								
M2131	Axis4								
M2132	Axis5								
M2133	Axis6								
M2134	Axis7								
M2135	Axis8								
M2136	Axis9								
M2137	Axis10								
M2138	Axis11								
M2139	Axis12								
M2140	Axis13								
M2141	Axis14								
M2142	Axis15								
M2143	Axis16								
M2144	Axis17								
M2145	Axis18								
M2146	Axis19								
M2147	Axis20								
M2148	Axis21								
M2149	Axis22								
M2150	Axis23								
M2151	Axis24								
M2152	Axis25								
M2153	Axis26								
M2154	Axis27								
M2155	Axis28								
M2156	Axis29								
M2157	Axis30								
M2158	Axis31								
M2159	Axis32								

* The entry "END" in the Refresh Cycle column indicates 50ms or a longer sequence program scan time.

(8) Common devices (A273UHCPU(32 axis feature)/A173UHCPU(S1))

Device No.	Signal name (SV43)	Refresh cycle, Set No. of axis: A173UHCPU 1 to 12 / A273UHCPU 1 to 8	A173UHCPU 13 to 24 / A273UHCPU 9 to 18	A173UHCPU 25 to32 / A273UHCPU 19 to 32	Fetch cycle, Set No. of axis: A173UHCPU 1 to 12 / A273UHCPU 1 to 8	A173UHCPU 13 to 24 / A273UHCPU 9 to 18	A173UHCPU 25 to32 / A273UHCPU 19 to 32	Signal direction
M2160								
M2161								
M2162								
M2163								
M2164								
M2165								
M2166								
M2167								
M2168								
M2169								
M2170								
M2171								
M2172								
M2173								
M2174								
M2175								
M2176								
M2177								
M2178								
M2179								
M2180								
M2181								
M2182								
M2183								
M2184								
M2185								
M2186								
M2187								
M2188								
M2189								
M2190								
M2191								
M2192								
M2193								
M2194								
M2195								
M2196								
M2197								
M2198								
M2199	Unusable							
M2200	(80 points)		–			–		–
M2201								
M2202								
M2203								
M2204								
M2205								
M2206								
M2207								
M2208								
M2209								
M2210								
M2211								
M2212								
M2213								
M2214								
M2215								
M2216								
M2217								
M2218								
M2219								
M2220								
M2221								
M2222								
M2223								
M2224								
M2225								
M2226								
M2227								
M2228								
M2229								
M2230								
M2231								
M2232								
M2233								
M2234								
M2235								
M2236								
M2237								
M2238								
M2239								

Device No.	Axis	Signal name (SV43)	Refresh cycle, Set No. of axis: A173UHCPU 1 to 12 / A273UHCPU 1 to 8	A173UHCPU 13 to 24 / A273UHCPU 9 to 18	A173UHCPU 25 to32 / A273UHCPU 19 to 32	Fetch cycle, Set No. of axis: A173UHCPU 1 to 12 / A273UHCPU 1 to 8	A173UHCPU 13 to 24 / A273UHCPU 9 to 18	A173UHCPU 25 to32 / A273UHCPU 19 to 32	Signal direction
M2240	Axis1								
M2241	Axis2								
M2242	Axis3								
M2243	Axis4								
M2244	Axis5								
M2245	Axis6								
M2246	Axis7								
M2247	Axis8								
M2248	Axis9								
M2249	Axis10								
M2250	Axis11								
M2251	Axis12								
M2252	Axis13								
M2253	Axis14								
M2254	Axis15								
M2255	Axis16	Speed change accepting							
M2256	Axis17	flag "0"	3.5ms	7.1ms	14.2ms				SCPU ← PCPU
M2257	Axis18								
M2258	Axis19								
M2259	Axis20								
M2260	Axis21								
M2261	Axis22								
M2262	Axis23								
M2263	Axis24								
M2264	Axis25								
M2265	Axis26								
M2266	Axis27								
M2267	Axis28								
M2268	Axis29								
M2269	Axis30								
M2270	Axis31								
M2271	Axis32								
M2272									
M2273									
M2274									
M2275									
M2276									
M2277									
M2278									
M2279									
M2280									
M2281									
M2282									
M2283									
M2284									
M2285									
M2286									
M2287									
M2288									
M2289									
M2290									
M2291									
M2292									
M2293									
M2294									
M2295	Unusable								
M2296	(48 points)			–			–		–
M2297									
M2298									
M2299									
M2300									
M2301									
M2302									
M2303									
M2304									
M2305									
M2306									
M2307									
M2308									
M2309									
M2310									
M2311									
M2312									
M2313									
M2314									
M2315									
M2316									
M2317									
M2318									
M2319									

* The entry "END" in the Refresh Cycle column indicates 50ms or a longer sequence program scan time.

(8) Common devices

A273UHCPU(32 axis feature)/A173UHCPU(S1)

Device No.	Signal name			Refresh cycle			Fetch cycle			Signal direction
				Set number of axes			Set number of axes			
		SV43	A173UHCPU	1 to 12	13 to 24	25 to 32	1 to 12	13 to 24	25 to 32	
			A273UHCPU	1 to 8	9 to 18	19 to 32	1 to 8	9 to 18	19 to 32	
D9180 D9181	Unusable			–			–			–
D9182 D9183	Test mode request error information			When test mode is requested						SCPU ← PCPU
D9184	PCPU WDT error cause			When PCPU WDT error occurs						
D9185 D9186 D9187	Manual pulse generator axis setting error information			When manual pulse generator operation is enabled						
D9188	Unusable			–			–			–
D9189	Error program No.			At start						SCPU ← PCPU
D9190	Error item information									
D9191 D9192	Servo amplifier loading information			At power-on and 10ms	20ms					
D9193 D9194 D9195	Unusable			–			–			–
D9186	Personal computer link communication error code			3.5ms	7.1ms	14.2ms				SCPU ← PCPU
D9187 D9198 D9199	Unusable			–			–			–

(8) Common devices

A172SHCPUN

Device No.	Signal Name		Fetch Cycle	Refresh Cycle	Signal Direction
D1008	Limit switch output disable setting register (4 points)		3.5ms		SCPU →PCPU
D1009					
D1010					
D1011					
D1012	Setting Register for a axis number controlled with manual pulse generator 1		Manual pulse generator operation enabled		
D1013	Unusable (2 points)		–	–	–
D1014					
D1015	JOG operation simultaneous start axis setting register		At driving		SCPU →PCPU
D1016	Axis 1	1 pulse input modification setting register for manual pulse generators (8 points)	Manual pulse generator operation enabled		
D1017	Axis 2				
D1018	Axis 3				
D1019	Axis 4				
D1020	Axis 5				
D1021	Axis 6				
D1022	Axis 7				
D1023	Axis 8				

A171SHCPUN

Device No.	Signal Name		Fetch Cycle	Refresh Cycle	Signal Direction
D1008	Limit switch output disable setting register (4 points)		3.5ms		SCPU →PCPU
D1009					
D1010					
D1011					
D1012	Setting Register for a axis number controlled with manual pulse generator 1		Manual pulse generator operation enabled		
D1013	Unusable (2 points)		–	–	–
D1014					
D1015	JOG operation simultaneous start axis setting register		At driving		SCPU →PCPU
D1016	Axis 1	1 pulse input modification setting register for manual pulse generator (4 points)	Manual pulse generator operation enabled		
D1017	Axis 2				
D1018	Axis 3				
D1019	Axis 4				
D1020	Unusable (4 points)		–	–	–
D1021					
D1022					
D1023					

(8) Common devices
A273UHCPU (32 axis feature) / A173UHCPU (S1)

<table>
<tr><th rowspan="4">Device No.</th><th colspan="3" rowspan="2">Signal name</th><th colspan="3">Refresh cycle</th><th colspan="3">Fetch cycle</th><th rowspan="4">Signal direction</th></tr>
<tr><th colspan="3">Set No. of axis</th><th colspan="3">Set No. of axis</th></tr>
<tr><th rowspan="2"></th><th rowspan="2">SV43</th><th>A173UHCPU</th><th>1 to 12</th><th>13 to 24</th><th>25 to 32</th><th>1 to 12</th><th>13 to 24</th><th>25 to 32</th></tr>
<tr><th>A273UHCPU</th><th>1 to 8</th><th>9 to 18</th><th>19 to 32</th><th>1 to 8</th><th>9 to 18</th><th>19 to 32</th></tr>
<tr><td>D704</td><td colspan="3" rowspan="6">Unusable (6 points)</td><td colspan="3" rowspan="6">–</td><td colspan="3" rowspan="6">–</td><td rowspan="6">–</td></tr>
<tr><td>D705</td></tr>
<tr><td>D706</td></tr>
<tr><td>D707</td></tr>
<tr><td>D708</td></tr>
<tr><td>D709</td></tr>
<tr><td>D710</td><td colspan="3" rowspan="4">JOG simultaneous start axis setting register</td><td colspan="3" rowspan="45"></td><td colspan="3" rowspan="4">At start</td><td rowspan="45">SCPU → PCPU</td></tr>
<tr><td>D711</td></tr>
<tr><td>D712</td></tr>
<tr><td>D713</td></tr>
<tr><td>D714</td><td colspan="3" rowspan="2">Manual pulse generator 1 axis No. setting register</td><td colspan="3" rowspan="41">When manual pulse generator enable</td></tr>
<tr><td>D715</td></tr>
<tr><td>D716</td><td colspan="3" rowspan="2">Manual pulse generator 2 axis No. setting register</td></tr>
<tr><td>D717</td></tr>
<tr><td>D718</td><td colspan="3" rowspan="2">Manual pulse generator 3 axis No. setting register</td></tr>
<tr><td>D719</td></tr>
<tr><td>D720</td><td>Axis 1</td><td colspan="2" rowspan="32">Manual pulse generator 1-pulse input magnification setting register</td></tr>
<tr><td>D721</td><td>Axis 2</td></tr>
<tr><td>D722</td><td>Axis 3</td></tr>
<tr><td>D723</td><td>Axis 4</td></tr>
<tr><td>D724</td><td>Axis 5</td></tr>
<tr><td>D725</td><td>Axis 6</td></tr>
<tr><td>D726</td><td>Axis 7</td></tr>
<tr><td>D727</td><td>Axis 8</td></tr>
<tr><td>D728</td><td>Axis 9</td></tr>
<tr><td>D729</td><td>Axis 10</td></tr>
<tr><td>D730</td><td>Axis 11</td></tr>
<tr><td>D731</td><td>Axis 12</td></tr>
<tr><td>D732</td><td>Axis 13</td></tr>
<tr><td>D733</td><td>Axis 14</td></tr>
<tr><td>D734</td><td>Axis 15</td></tr>
<tr><td>D735</td><td>Axis 16</td></tr>
<tr><td>D736</td><td>Axis 17</td></tr>
<tr><td>D737</td><td>Axis 18</td></tr>
<tr><td>D738</td><td>Axis 19</td></tr>
<tr><td>D739</td><td>Axis 20</td></tr>
<tr><td>D740</td><td>Axis 21</td></tr>
<tr><td>D741</td><td>Axis 22</td></tr>
<tr><td>D742</td><td>Axis 23</td></tr>
<tr><td>D743</td><td>Axis 24</td></tr>
<tr><td>D744</td><td>Axis 25</td></tr>
<tr><td>D745</td><td>Axis 26</td></tr>
<tr><td>D746</td><td>Axis 27</td></tr>
<tr><td>D747</td><td>Axis 28</td></tr>
<tr><td>D748</td><td>Axis 29</td></tr>
<tr><td>D749</td><td>Axis 30</td></tr>
<tr><td>D750</td><td>Axis 31</td></tr>
<tr><td>D751</td><td>Axis 32</td></tr>
<tr><td>D752</td><td colspan="3">Manual pulse generator 1 smoothing magnification setting register</td></tr>
<tr><td>D753</td><td colspan="3">Manual pulse generator 2 smoothing magnification setting register</td></tr>
<tr><td>D754</td><td colspan="3">Manual pulse generator 3 smoothing magnification setting register</td></tr>
<tr><td>D755</td><td colspan="3" rowspan="5">Unusable (5 points)</td><td colspan="3" rowspan="5">–</td><td colspan="3" rowspan="5">–</td><td rowspan="5">–</td></tr>
<tr><td>D756</td></tr>
<tr><td>D757</td></tr>
<tr><td>D758</td></tr>
<tr><td>D759</td></tr>
<tr><td>D760</td><td colspan="3" rowspan="16">Limit switch output disable setting register</td><td colspan="3" rowspan="32"></td><td rowspan="32">3.5ms</td><td rowspan="32">7.1ms</td><td rowspan="32">14.2ms</td><td rowspan="40">SCPU → PCPU</td></tr>
<tr><td>D761</td></tr>
<tr><td>D762</td></tr>
<tr><td>D763</td></tr>
<tr><td>D764</td></tr>
<tr><td>D765</td></tr>
<tr><td>D766</td></tr>
<tr><td>D767</td></tr>
<tr><td>D768</td></tr>
<tr><td>D769</td></tr>
<tr><td>D770</td></tr>
<tr><td>D771</td></tr>
<tr><td>D772</td></tr>
<tr><td>D773</td></tr>
<tr><td>D774</td></tr>
<tr><td>D775</td></tr>
<tr><td>D776</td><td colspan="3" rowspan="16">Limit switch output status storage register</td></tr>
<tr><td>D777</td></tr>
<tr><td>D778</td></tr>
<tr><td>D779</td></tr>
<tr><td>D780</td></tr>
<tr><td>D781</td></tr>
<tr><td>D782</td></tr>
<tr><td>D783</td></tr>
<tr><td>D784</td></tr>
<tr><td>D785</td></tr>
<tr><td>D786</td></tr>
<tr><td>D787</td></tr>
<tr><td>D788</td></tr>
<tr><td>D789</td></tr>
<tr><td>D790</td></tr>
<tr><td>D791</td></tr>
<tr><td>D792</td><td colspan="3" rowspan="8">Servo amplifier type</td><td colspan="3" rowspan="8">At power ON</td><td colspan="3" rowspan="8"></td></tr>
<tr><td>D793</td></tr>
<tr><td>D794</td></tr>
<tr><td>D795</td></tr>
<tr><td>D796</td></tr>
<tr><td>D797</td></tr>
<tr><td>D798</td></tr>
<tr><td>D799</td></tr>
</table>

(9) Special Relays

Device No.	Signal Name	Fetch Cycle	Refresh Cycle	Signal Direction
M9073	PCPU WDT error flag		END	PCPU → SCPU
M9074	PCPU REDAY-completed flag			
M9075	In-test-mode flag			
M9076	External emergency stop input flag			
M9077	Manual pulse generator axis setting error flag			
M9078	Test mode request error flag			
M9079	Servo program setting error flag			

* The entry “END” in the Refresh Cycle column indicates 80ms (A172SHCPUN/A171SHCPUN) or 50ms (A273UHCPU (32 axis feature) / A173UHCPU (S1)), or a longer sequence program scan time.

(10) Table 3.2 Special Registers (A172SHCPUN / A171SHCPUN)

A172SH CPUN/ A171SH CPUN Device Number	Signal Name	Refresh Cycle	Fetch Cycle	Signal Direction
D9180	Limit switch output status	3.5ms		SCPU←PCPU
D9181				
D9182				
D9183				
D9184	PCPU WDT error cause	At PCPU WDT error occurrence		
D9185	Servo amplifier type	Power ON		
D9186				
D9187	Manual pulse generator axis setting error information	Manual pulse generator operation enabled		
D9188	Test mode request error information	Test mode request		
D9189	Error program number	At driving		
D9190	Error item information			
D9191	Servo amplifier loading information	Power ON, 10 ms		
D9192	Manual pulse generator 1 smoothing magnification setting register		Manual pulse generator operation enabled	SCPU→PCPU
D9193 D9194 D9195	Unusable	–	–	–
D9196	PC link communication error code	3.5ms		SCPU←PCPU
D9197 D9198 D9199	Unusable	–	–	–

MITSUBISHI ELECTRIC CORPORATION
HEAD OFFICE:MITSUBISHI DENKI BLDG MARUNOUCHI TOKYO 100 TELEX: J24532 CABLE MELCO TOKYO
NAGOYA WORKS : 1-14 , YADA-MINAMI 5 , HIGASHI-KU , NAGOYA , JAPAN

IB (NA) 0300014-A (0002) MEE Printed in Japan Specifications subject to change without notice.