MITSUBISHI

PROGRAMMABLE CONTROLLER

MELSEC-A

Reference Manual

type A73CPU

MITSUBISHI
ELECTRIC

REVISIONS

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

Print Date	*Manual Number	Revision
Apr., 1990	IB (NA) 66233-A	First edition

INTRODUCTION

Thank you for choosing the Mitsubishi MELSEC-A Series of General Purpose Programmable Controllers. Please read this manual carefully so that the equipment is used to its optimum. A copy of this manual should be forwarded to the end User.

CONTENTS

[POSITIONING CONTROL]

[SEQUENCE CONTROL]

POSITIONING CONTROL

This section explains the data settings, devices, positioning programs, etc. necessary to execute positioning control with the A73CPU.

1. INTRODUCTION

This section (Positioning Control) explains the positioning control parameters, positioning devices, positioning modes, etc. necessary to control positioning with the A73CPU multi-axis positioning module (to be referred to as the A73CPU in this manual).

1.1 Positioning Control with A73CPU

The A73CPU positioning module can control the positioning of up to eight axes and the sequence control equivalent of the A3NCPU using the multi-axis positioning control CPU (to be referred to as the PCPU in this manual) and the sequence control CPU (to be referred to as the SCPU in this manual).

(1) SCPU control

(a) Sequence control

Controls I/O modules and special function modules using sequence programs. Sequence programs are executed in the same manner as with the A3NCPU.

(b) Initiation of positioning with a sequence program and positioning data setting

1) Gives the servo program execution request with the DSFRP instruction.

2) Changes present position data and speed with the DSFLP instruction.

3) Executes the jog operation.

4) Sets the data necessary for manual pulse generator (MPG) operation.

(2) PCPU control

(a) Executes the servo program after an execution request is given by the DSFRP instruction in the sequence program and controls the requested positioning.

Positioning control parameters and the positioning data set in the servo program are used for positioning control.

(b) Changes the current position data and positioning speed in the servo controller to the actual position data and positioning speed set by the DSFLP instruction in the sequence program.

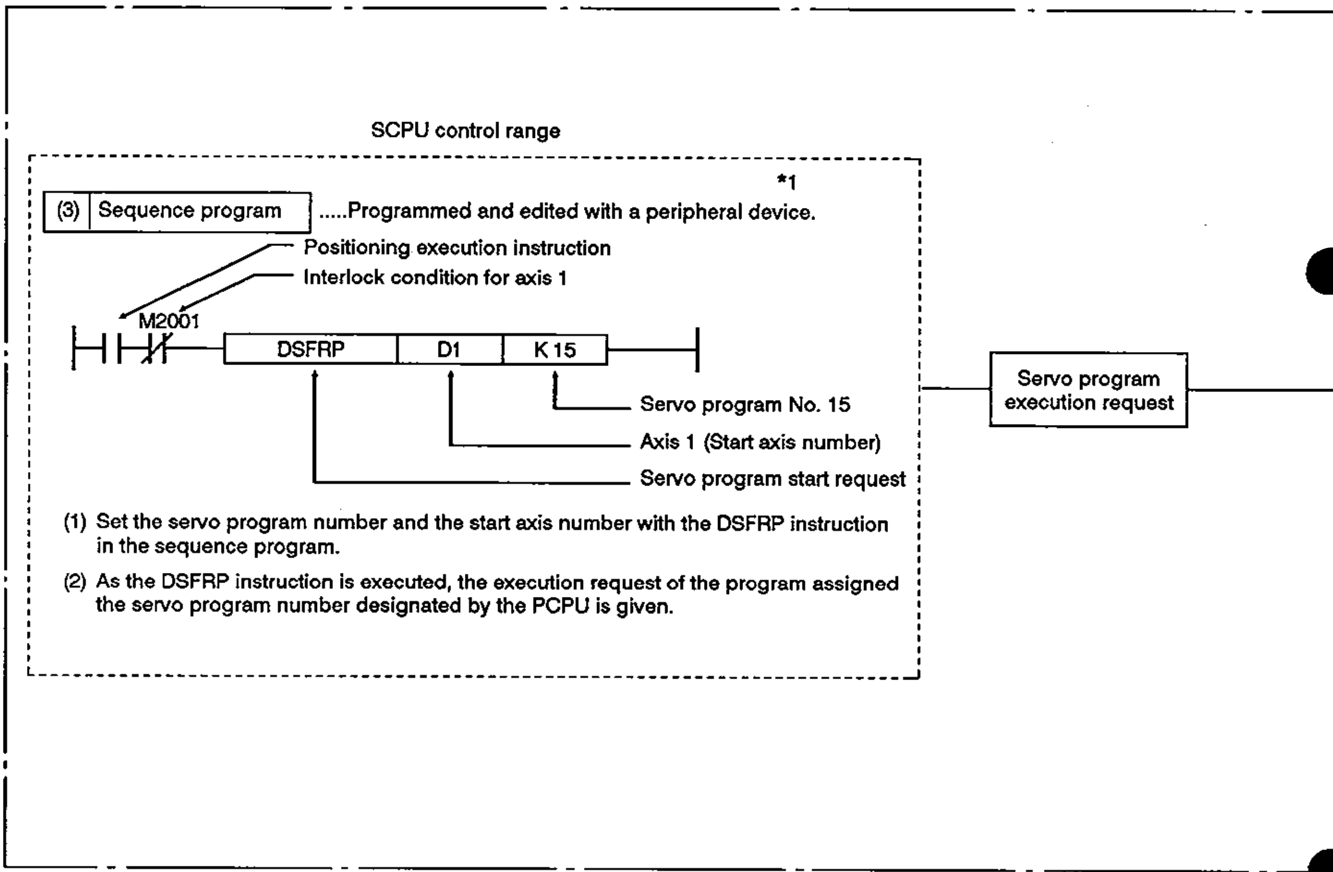
(c) Executes positioning by the manual pulse generator.

[Positioning control with the A73CPU]

The A73CPU controls positioning based on the servo program designated by the SCPU sequence program.

The diagram below outlines the positioning control mode.

A73CPU System



(1) Program the servo program and set the positioning control parameters with a peripheral device.

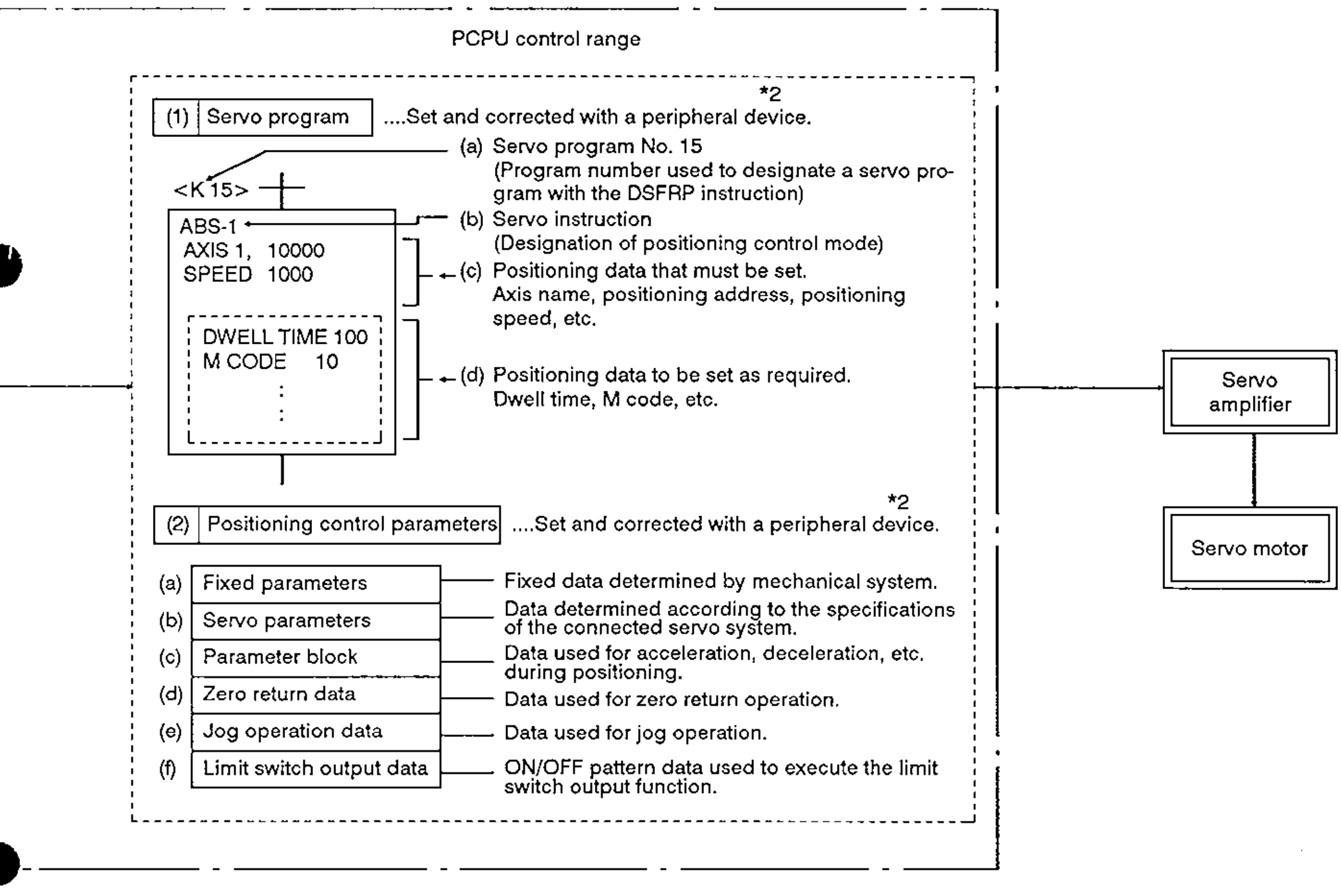
(2) Positioning is initiated by the sequence program (DSFRP instruction).

   (a) Designate the servo program number and the start axis number with the DSFRP instruction.

      1) Servo program number designation:
         Direct or indirect designation is allowed.

      2) Start axis designation:
         Direct designation only.

(3) Required positioning is executed using the designated servo program.



REMARKS

*1: Peripheral devices which can be used to program a sequence program:

* A7PU
* A6GPP or A6PHP booted with SW[ ]GP-GPPA.
* A6HGP booted with SW[ ]-HGPA.

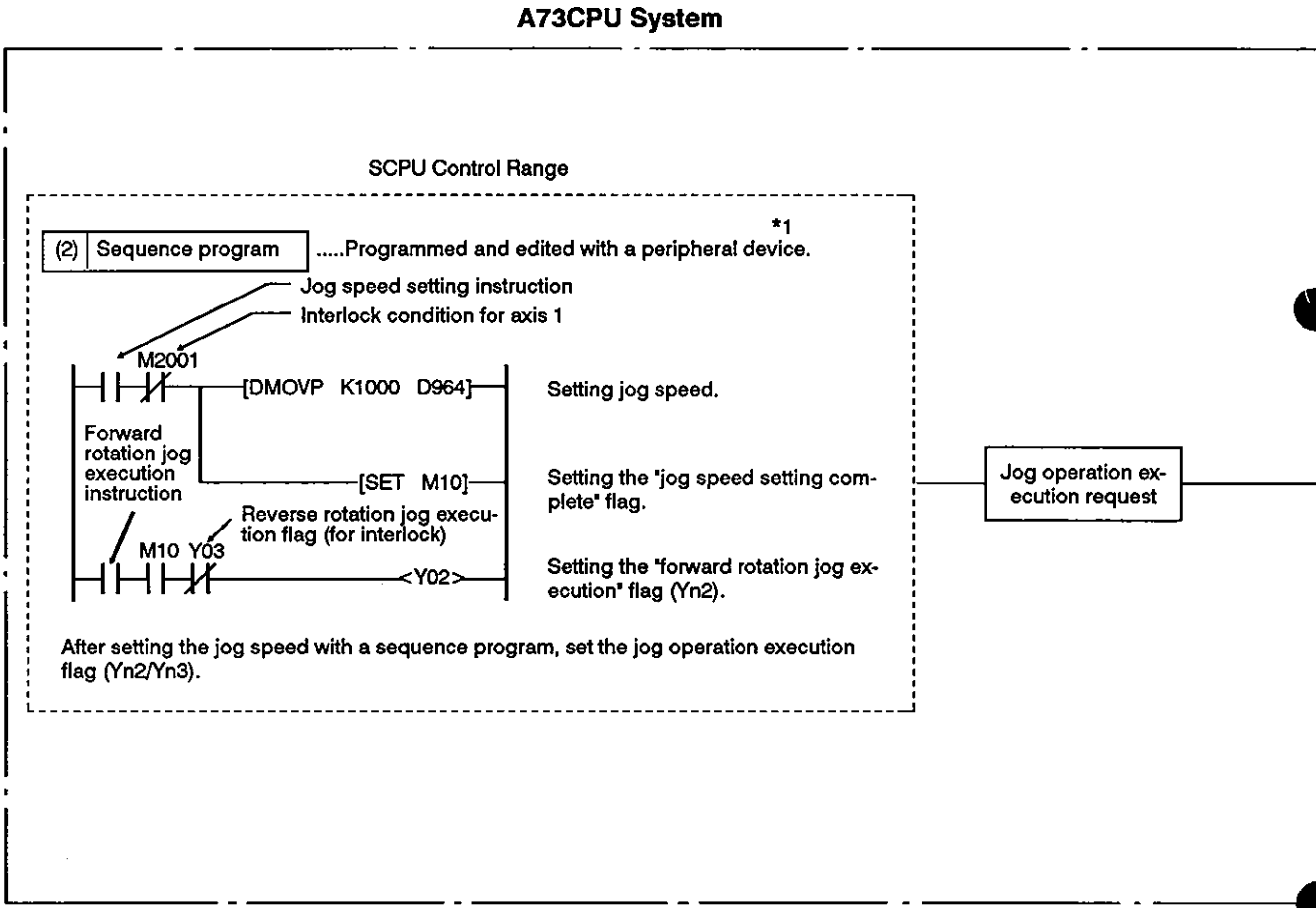
*2: Peripheral devices which can be used to program a servo program or set positioning parameters:

* A6MD
* A6GPP or A6PHP booted with SW[ ]GP-A73P.

[Jog operation with the A73CPU]

The A73CPU can execute jog operation for the required axis using a sequence program.

The diagram below outlines jog operation.



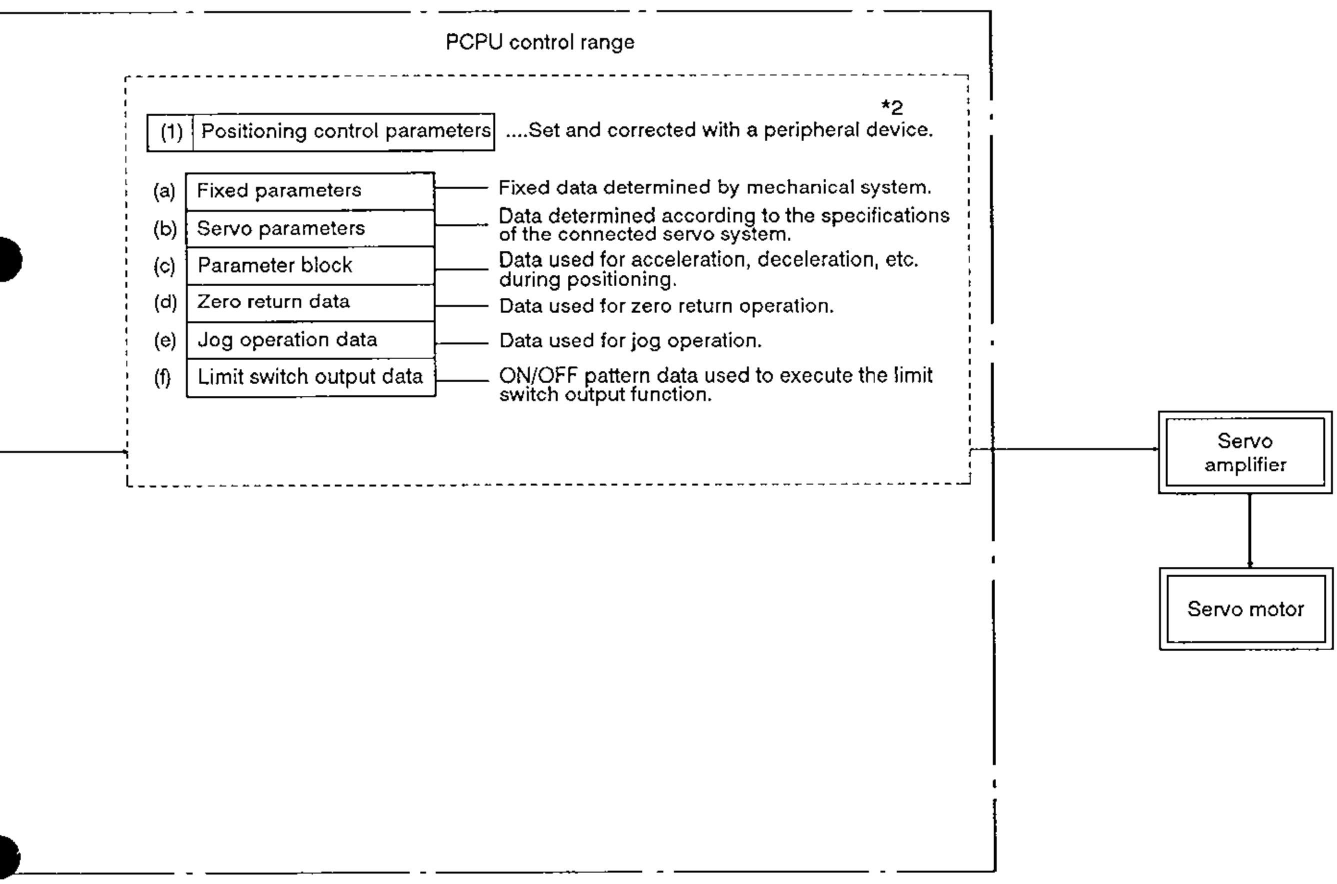
(1) Set the positioning control parameters (1) with a peripheral device.

(2) Set the jog speed in the jog speed setting register for each axis with the sequence program.

(3) Jog operation is executed while the jog operation execution flag (Yn2 or Yn3)*3 is set from the sequence program.

1) Yn2: Forward rotation jog execution

2) Yn3: Reverse rotation jog execution



REMARKS

*1: Peripheral devices which can be used to program a sequence program:

- A7PU
- A6GPP or A6PHP booted with SW[ ]GP-GPPA.
- A6HGP booted with SW[ ]-HGPA.

*2: Peripheral devices which can be used to program a servo program or set positioning parameters:

- A6MD
- A6GPP or A6PHP booted with SW[ ]GP-A73P.

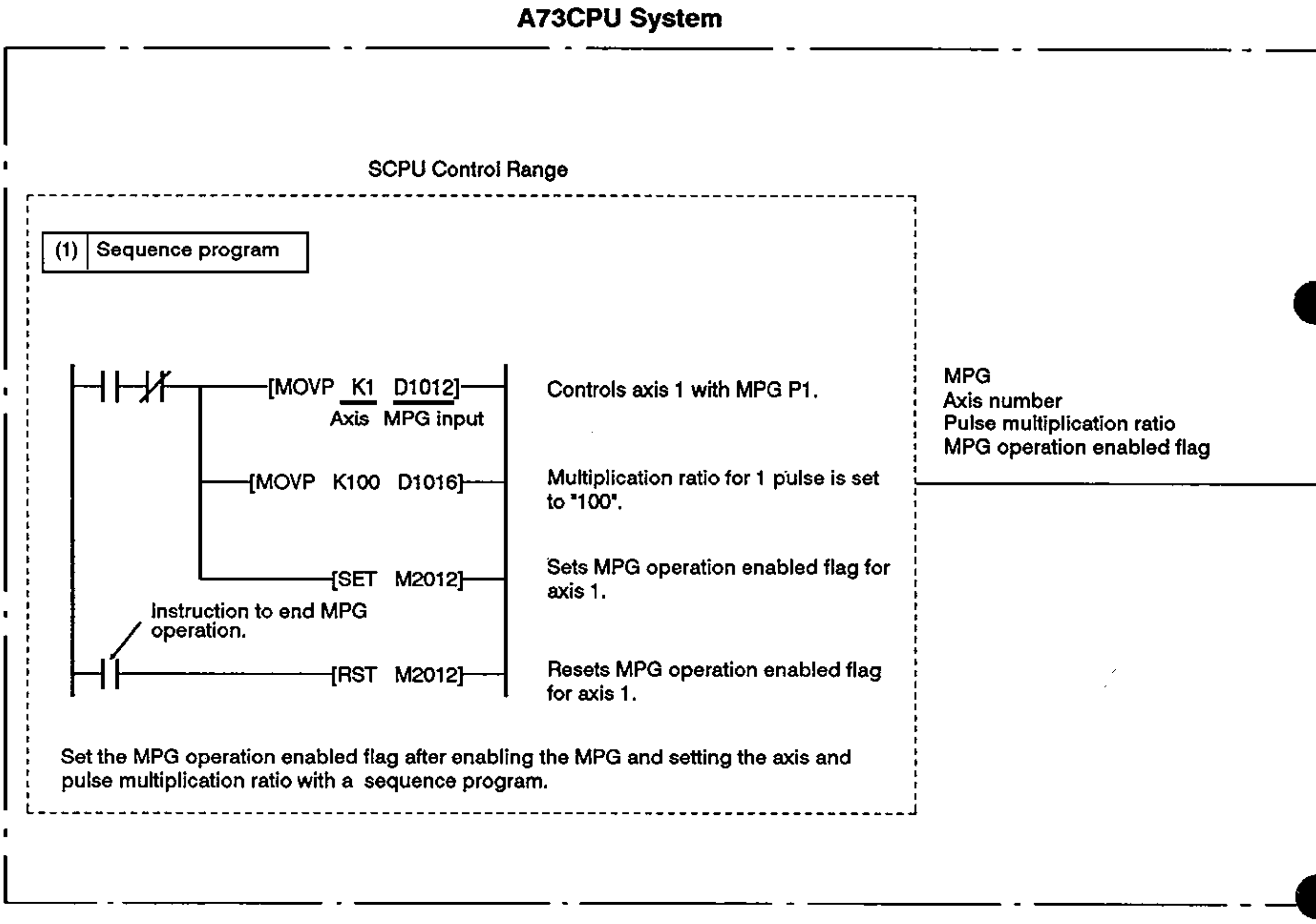
*3: For "n" in Yn2 and Yn3, set the axis number (1 to 8).

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

[Manual pulse generator operation with the A73CPU]

When executing positioning control with the manual pulse generator, connected to the A70SF, it is necessary to set the A73CPU in the manual pulse generator (to be abbreviated to MPG in this manual) operation enabled state with a sequence program.

The diagram below outlines positioning control using the MPG.



(1) Set the MPG to be used, the number of the axis to be operated,and the pulse multiplication ratio with a sequence program.

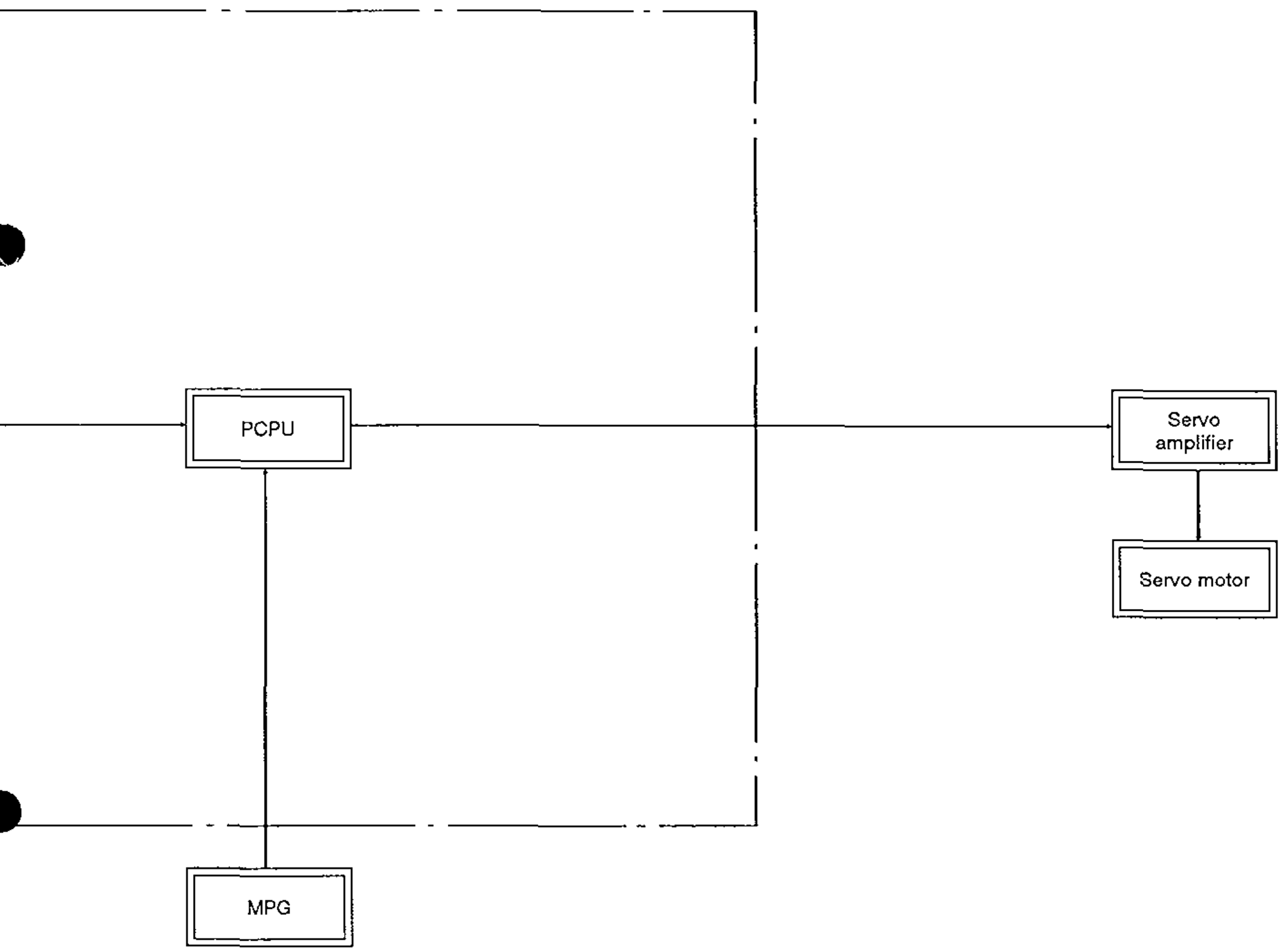
(2) Set the MPG operation enabled flag with a sequence program.

MPG operation is now enabled.

(3) Rotate the MPG to execute positioning.

(4) Reset the MPG operation enabled flag with a sequence program.

MPG operation is completed.


PCPU
Servo amplifier
Servo motor
MPG

(1) Positioning control parameters

There are six types of positioning control parameters as described below.

Parameter data can be set and edited interactively with a peripheral device.

(a) Fixed parameters

Fixed parameters are determined according to the mechanical system and should be set for each individual axis.

These parameters are used to calculate target position data when executing positioning control.

For details, see Section 4.1.

1) Positioning control units
2) Number of pulses per motor rotation
3) Axis travel distance per motor rotation
4) Unit multiplication ratio
5) Backlash compensation data
6) Upper/lower stroke limits
7) Command in-position width
8) Limit switch output used/not-used

(b) Servo parameters

Servo parameters are determined according to the connected servo motor and should be set for each individual axis.

These parameters are used to control the servo motor when executing positioning control.

For details, see Section 4.2.

1) System setting
2) Regenerative resistance used/not-used
3) Motor type
4) Motor capacity
5) Motor rpm
6) Position loop gain
7) Velocity loop gain
8) Velocity integration compensation
9) In-position range
10) Number of feedback pulses per motor rotation
11) Rotating direction

(c) Zero return data

Zero return data is used when executing the zero return operation and should be set for each individual axis.

For details, see Section 7.17.1.

1) Zero return direction
2) Zero return method
3) Zero point address
4) Zero return speed
5) Creep speed
6) Axis travel beyond the near-zero point dog
7) Parameter block number

(d) Jog operation data

Jog operation data is used when executing positioning in the jog mode and should be set for each individual axis.

For details, see Section 7.16.1.

1) Jog speed limit
2) Parameter block number

(e) Parameter blocks

Parameter blocks are the data indicated below.

Up to 16 blocks can be set.

These parameter blocks are provided to facilitate setting changes such as acceleration/deceleration processing (acceleration/deceleration time, speed limits) during positioning control.

For details, see Section 4.3.

1) Interpolation control units
2) Speed limit
3) Acceleration time
4) Deceleration time
5) Deceleration time for immediate stop
6) Torque limit
7) Deceleration processing when the STOP signal is input.
8) Allowable circular interpolation error

(f) Limit switch output data

Limit switch output data should be set when the limit switch output (fixed parameter) is "used".

Limit switch output data should be set for each individual axis.

The preset ON/OFF pattern is output externally when positioning control of the axis for which the limit switch output data is set is carried out.

For details, see Section 4.4.

1) Limit switch ON/OFF pattern

(2) Servo program

The servo program is used for positioning control and its execution is requested by the sequence program.

The servo program consists of a program number, servo instructions, and positioning data.

For details, see Section 6.

(a) Program number

The program number is used to identify the program and is designated to call the required servo program from the sequence program.

(b) Servo instruction

The servo instruction indicates the positioning mode.

(c) Positioning data

The positioning data is the data necessary to execute the servo instructions. For each servo instruction, necessary data is determined.

(3) Sequence program

The sequence program is used to execute positioning control or jog mode operation based on the servo program and for permission to feed an axis using the MPG.

For details, see Section 5.

2. SPECIFICATIONS

Table 2.1 PCPU Specifications

<table>
<tr><th colspan="2">Item</th><th colspan="5">PCPU Specifications</th></tr>
<tr><td colspan="2">Number of controlled axes</td><td colspan="5">8 axes (simultaneous 2 axes or 3 axes, independent 8 axes)</td></tr>
<tr><td colspan="2">Interpolation function</td><td colspan="5">Linear interpolation (2, 3 axes), circular interpolation (2 axes)</td></tr>
<tr><td colspan="2">Control mode</td><td colspan="5">PTP (point to point), velocity control, velocity/position switchover control, incremental feed, constant velocity control</td></tr>
<tr><td colspan="2">Control units</td><td colspan="5">mm, inches, degrees, pulses</td></tr>
<tr><td rowspan="4">Program</td><td>Language</td><td colspan="5">Special instructions</td></tr>
<tr><td>Capacity</td><td colspan="5">13K steps (13312 steps)</td></tr>
<tr><td>Positioning points</td><td colspan="5">Approx. 400 points / axis (varies according to program)
Indirect designation of positioning data is possible</td></tr>
<tr><td>Setting method</td><td colspan="5">Can be set with the A6GPPE or A6PHPE booted with SW0GHP-A73P or the A6MD.</td></tr>
<tr><td rowspan="10">Positioning</td><td>Mode</td><td colspan="5">PTP: Absolute/incremental
Velocity, velocity/position switchover control, incremental feed: Incremental
Constant velocity control: Absolute/incremental (combined)</td></tr>
<tr><td rowspan="7">Positioning command</td><td colspan="5">Set the following data for each axis.</td></tr>
<tr><td rowspan="2">Control units</td><td rowspan="2">Command units</td><td colspan="2">Not in the absolute system</td><td>Absolute system</td></tr>
<tr><td>Setting range</td><td>Max. value</td><td>Setting range</td></tr>
<tr><td>mm</td><td>x 10^{-1} μm</td><td rowspan="4">-2^{31} to $(2^{31}-1)$</td><td>429 m</td><td>−196596000 to 196596000</td></tr>
<tr><td>inch</td><td>x 10^{-5} inch</td><td>42949 inch</td><td>−196596000 to 196596000</td></tr>
<tr><td>degree</td><td>x 10^{-5} degree</td><td>360 degree</td><td>0 to 35999999</td></tr>
<tr><td>pulse</td><td>pulse</td><td>4294967296 PLS</td><td>−196596000 to 196596000</td></tr>
<tr><td>Velocity command (control units)</td><td colspan="5">0.01 to 6000000.00 (mm/min)
0.001 to 600000.000 (inches/min)
0.001 to 600000.000 (degrees/min)
1 to 1000000 (pulses/sec)</td></tr>
<tr><td>Acceleration/ deceleration processing</td><td colspan="5">Automatic trapezoidal acceleration/deceleration
Acceleration time .... 1 to 65535 (ms)
Deceleration time............ 1 to 65535 (ms)</td></tr>
<tr><td rowspan="2">Compensation</td><td>Electronic gears</td><td colspan="5">(0 to 65535) x (position command units)
[0 to 255 pulses when the unit is pulses]</td></tr>
<tr><td>Backlash compensation</td><td colspan="5">The function to compensate for errors between the commanded value and actual axis travel distance</td></tr>
<tr><td colspan="2">Zero return function</td><td colspan="5">When the system is not the absolute system, it is possible to select either the near-zero point dog or count.
For the absolute system, the data set is used.</td></tr>
<tr><td colspan="2">Jog operation function</td><td colspan="5">Supported</td></tr>
<tr><td colspan="2">MPG operation function</td><td colspan="5">Up to three MPGs can be connected to the system.
The axis to be controlled using the MPG is selected by designating the axis number in the sequence program.</td></tr>
<tr><td colspan="2">M function</td><td colspan="5">The M code output function is supported.</td></tr>
<tr><td colspan="2">Limit switch output function</td><td colspan="5">8 points/axis Up to 10 ON/OFF points can be set.</td></tr>
<tr><td colspan="2">Absolute system</td><td colspan="5">Possible (option)</td></tr>
</table>

3. POSITIONING SIGNALS

The A73CPU internal signals and the external signals sent to the A73CPU are used as positioning signals.

(1) Internal signals

For A73CPU internal signals, the following six A73CPU devices are used.

(a) Input (X)..................................... X0 to X7F (128 points)
(b) Output (Y) .... Y0 to Y7F (128 points)
(c) Internal relay (M) .... M2000 to M2047 (48 points)
(d) Special relay (SP. M).............. M9073 to M9079 (7 points)
(e) Data register (D)...................... D800 to D1023 (224 points)
(f) Special register (SP. D)......... D9180 to D9199 (20 points)

(2) External signals

The following signals are provided for external signals to be input to the A73CPU.

(a) Input from the upper/lower stroke end limit switches

The signals which determine the upper and lower stroke ends of the positioning range.

(b) Stop signal

This is the stop signal for velocity control.

(c) Near-zero point dog / velocity to position switchover signal

In zero return operation, the signal is the near-zero point dog signal.

In the velocity/position changeover control (to be referred to as V/P control in this manual), the signal is used to change the control from velocity to position.

(d) MPG input

The signal input from the MPG.

(e) Emergency stop input

The signal used to forcibly interrupt the positioning control.

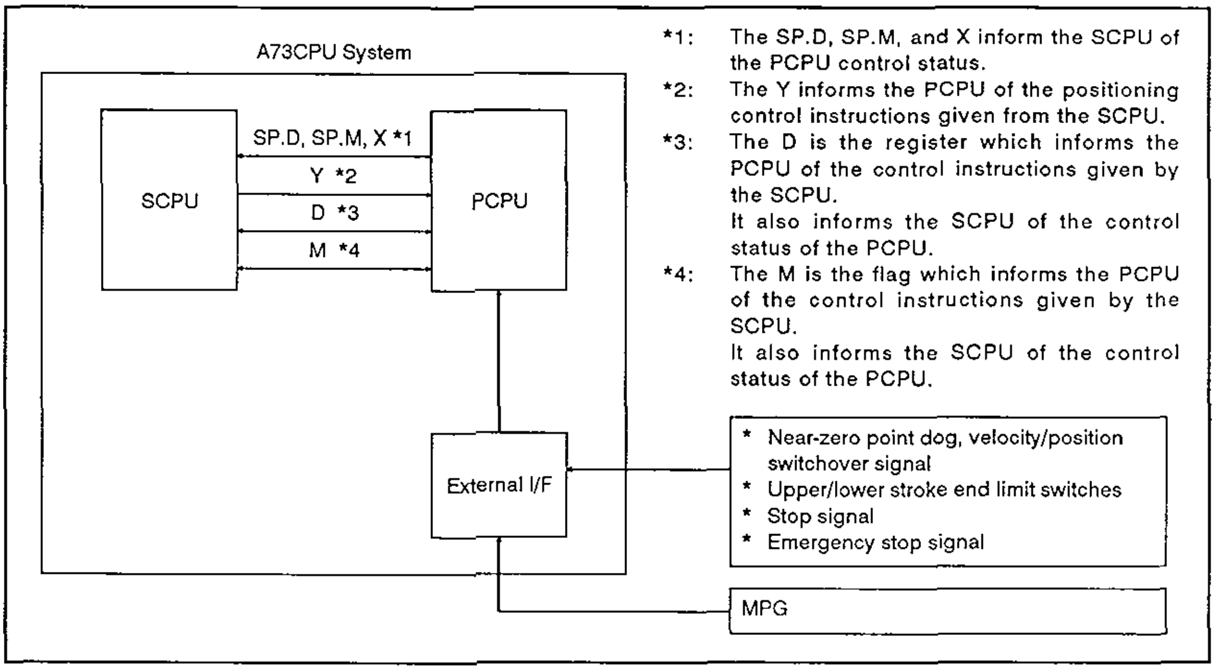


Fig. 3.1 Positioning Signal Flow

3.1 Input/Output (X/Y)

The A73CPU has 2048 I/O points ranging from X/Y0 to X/Y3FF. Among these, X/Y0 to X/Y7F are used for data transfer between the SCPU and PCPU for each axis. The signal name and I/O numbers for each axis are determined as indicated in Table 3.1.

(1) Input (X0 to X7F)

The input signals are set by the PCPU and used by the sequence program to confirm the control status of the axis and next positioning indication.

(2) Output (Y0 to Y7F)

The output signals are set by the sequence program to give positioning instructions to the PCPU.

Table 3.1 Input/output List

Device No.	Signal Name	Device No.	Signal Name
Xn0	Positioning start complete	Yn0	Immediate stop instruction
Xn1	Positioning complete	Yn1	Forward jog start
Xn2	In-position	Yn2	Reverse jog start
Xn3	Command in-position	Yn3	Complete signal OFF instruction
Xn4	Velocity control ON	Yn4	Velocity/position control switchover enable/disable
Xn5	Velocity/position switchover latch	Yn5	Limit switch output enable/disable
Xn6	Zero point passing	Yn6	Error reset
Xn7	Error detection	Yn7	Servo error reset
Xn8	Servo error detection	Yn8	External STOP input effective/ineffective at start of positioning
Xn9	Zero return request	Yn9	Not usable
XnA	Zero return complete	YnA	
XnB	FLS signal	YnB	
XnC	RLS signal	YnC	
XnD	STOP signal	YnD	
XnE	DOG/CHANGE signal	YnE	
XnF	Servo READY	YnF	Servo OFF

REMARKS

1. In Table 3.1, "n" indicates a number (0 to 7) corresponding to the axis number.

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

POINT

When the PC ready flag (M2000) is set after the power is turned on, the ON/OFF data in X0 to X7F remains off until the PCPU ready complete flag (M9074) is set. After the M9074 is set, X0 to X7F show the normal input status.

3.1.1 Positioning start complete signal (Xn0)

(a) The signal is turned on when the positioning start is complete for the axis designated by the DSFRP instruction in the sequence program.

This signal is not turned on when positioning is started for zero return, jog operation, MPG operation, and velocity control mode positioning.

The signal can be used to read an M code at the start of positioning.

(b) The positioning start complete signal is turned off at the leading edge of the complete signal OFF instruction or at the completion of positioning.

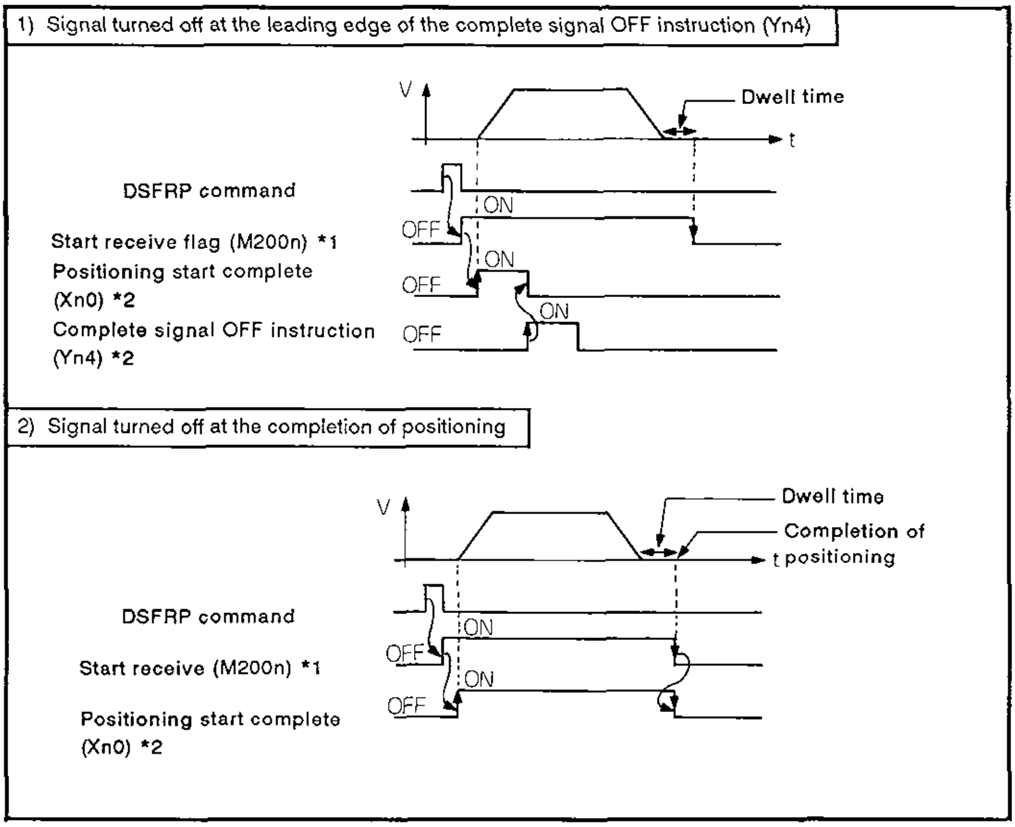


Fig. 3.2 Positioning Start Complete Signal ON/OFF Timing

REMARKS

*1: The "n" in M200n indicates an axis number (1 to 8).

*2: The "n" in Xn0 and Yn4 indicates a number (0 to 7) which corresponds to the axis number.

3.1.2 Positioning complete signal (Xn1)

(a) The positioning complete signal is turned on at the completion of positioning control for the axis which is designated by the DSFRP command in the sequence program.

The signal is not turned on when positioning control is started for zero return, jog operation, MPG operation, and velocity control mode, or if positioning control is stopped before completion.

(b) The positioning complete signal is turned off at the leading edge of the complete signal OFF instruction or at the start completion of positioning.

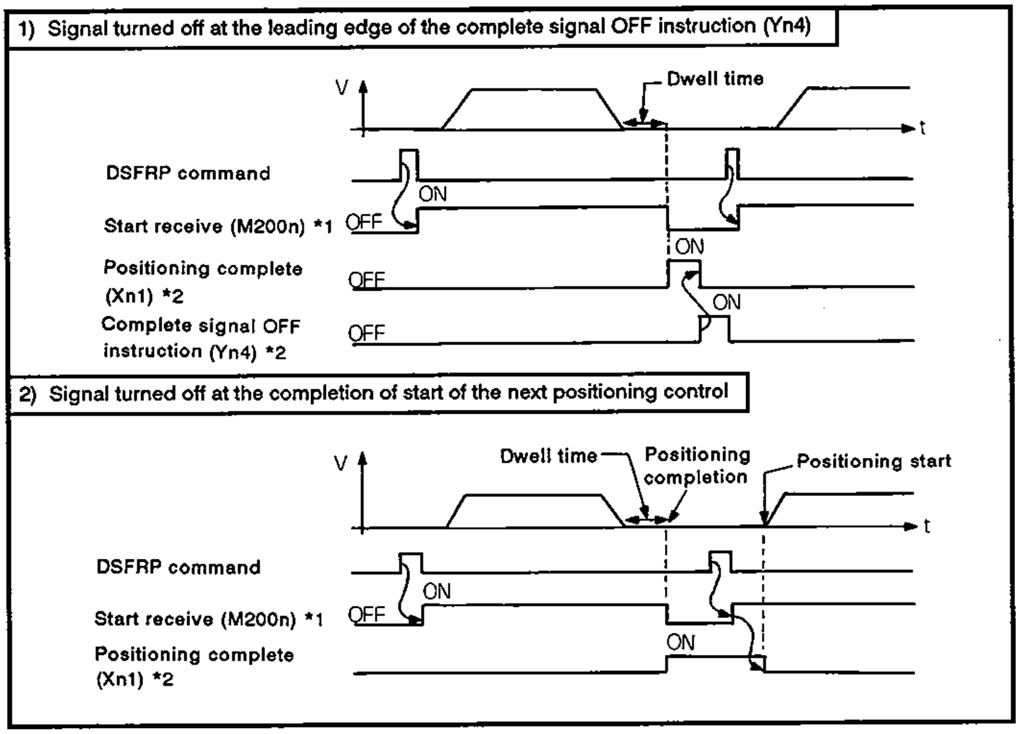


Fig. 3.3 ON/OFF Timing of Positioning Complete Signal

REMARKS

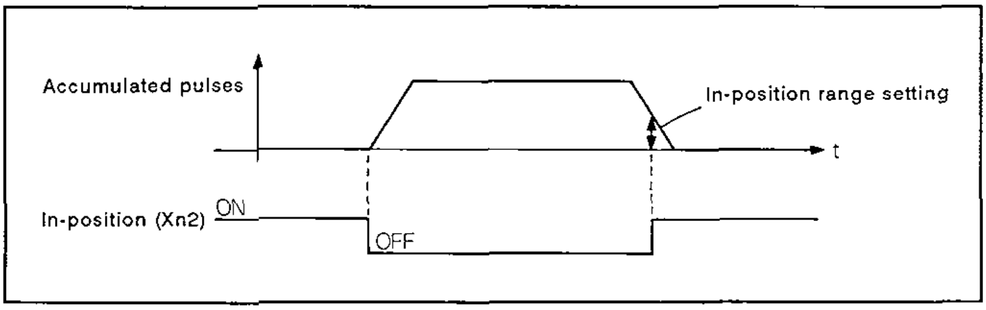
*1: The "n" in M200n indicates an axis number (1 to 8).

*2: The "n" in Xn1 and Yn4 indicates a number (0 to 7) which corresponds to the axis number.

3.1.3 In-position signal (Xn2)

(a) The in-position signal is turned on when the number of accumulated pulses in the error counter becomes less than the "in-position width" set by the servo parameter.

The in-position signal is turned off at the start of positioning control.



(b) An in-position check is conducted in the following cases:

1) When the servo power is turned on.
2) After the start of automatic deceleration during positioning control.
3) After the start of deceleration triggered by the jog start signal being turned off.
4) During MPG operation
5) After the near-zero point dog signal is turned on during zero return operation.
6) After the start of deceleration triggered by the stop instruction.

3.1.4 Command in-position signal (Xn3)

(a) The command in-position signal is turned on when the absolute value of the difference between the command position and present position becomes smaller than the "command in-position width" set by a parameter.

The command in-position signal is turned off in the following cases:

1) Start of position control

2) Zero return

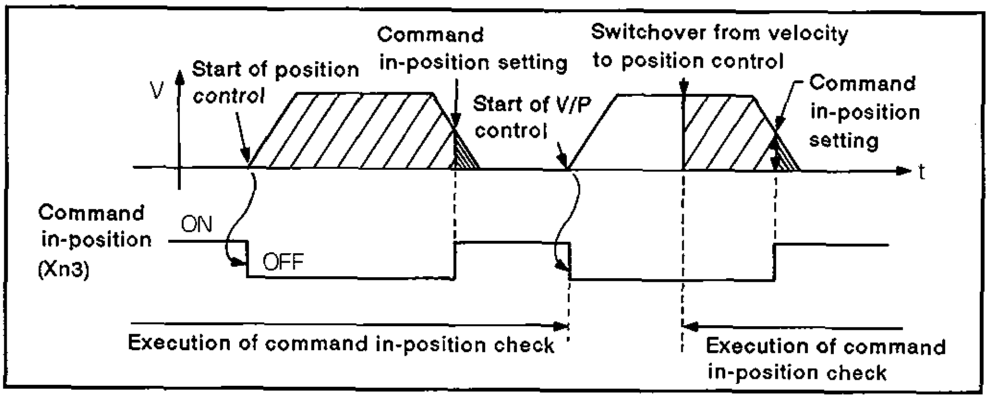
3) Velocity control

4) Jog operation

5) MPG operation

(b) The command in-position check is always effective during position control.

It is not conducted during velocity control or during velocity control in the velocity/position switchover (V/P) control mode.



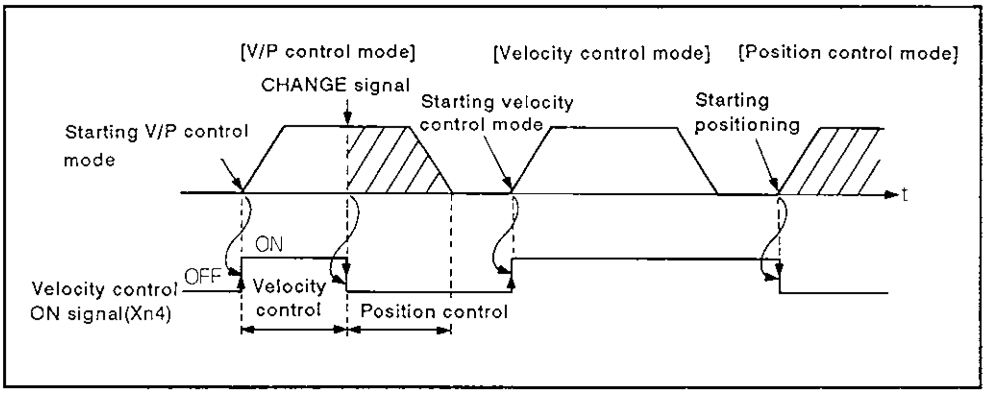
IB (NA) 66233-A

3.1.5 Velocity control ON signal (Xn4)

(a) The velocity control ON signal is turned on during the velocity control mode and used to determine whether the present control mode is position control or velocity control.

During the V/P control mode, the velocity control ON signal remains ON until the control mode is changed from the velocity control mode to the position control mode by an external CHANGE signal.

(b) The velocity control ON signal is off when the power is turned on and when in the position control mode.

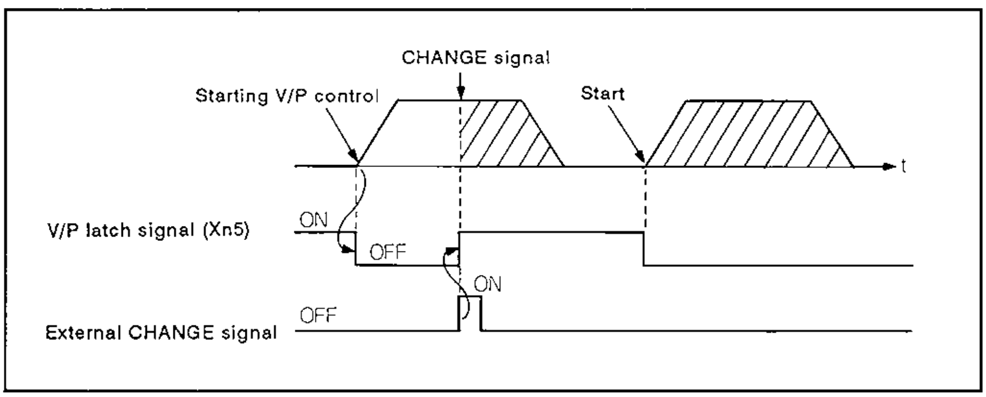


3.1.6 Velocity/position control switchover latch signal (Xn5)

(a) The velocity/position control switchover latch signal is turned on when the control mode is changed from velocity control to position control. The signal is used as the interlock signal to determine whether or not axis travel distance should be changed in the position control mode.

(b) The signal is turned off when the following control or operation starts:

1) Position control
2) V/P control
3) Velocity control
4) Jog operation
5) MPG operation



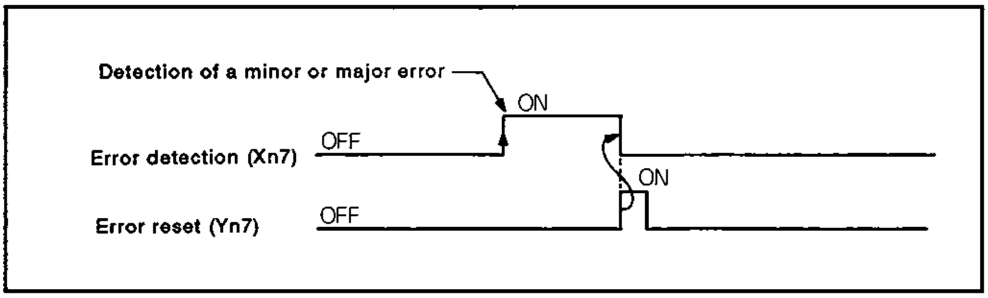
3.1.7 Zero point passing signal (Xn6)

(a) When the MR-SB is used, the zero point passing signal is turned on when the zero point is passed after the power is turned on. This signal is not turned on when the axis passes through the zero point if a general purpose servo is used.

3.1.8 Error detection signal (Xn7)

(a) The error detection signal is turned on if a minor or major error is detected. It is used to determine the presence of an error.

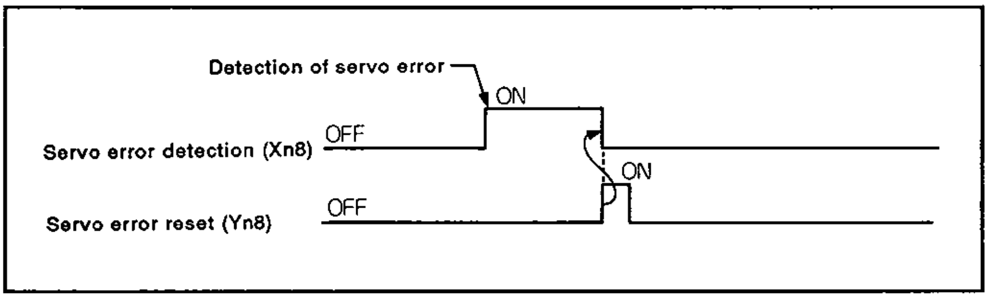
(b) The signal is turned off when the error reset signal (Xn7) is turned on.



3.1.9 Servo error detection signal (Xn8)

(a) The servo error detection signal is turned on when an error is detected by the servo amplifier and is used to determine the presence of a servo error.

(b) The signal is turned off when the servo error reset signal (Yn8) is turned on or when the servo power is turned on again.



3.1.10 Zero return request signal (Xn9)

The zero return request signal is turned on if it is necessary to confirm the zero point address when the power is turned on or during positioning control.

(a) Not in the absolute system

1) The zero return request signal is turned on in the following cases:

i) When the power is turned on or the A73CPU is reset.

ii) During zero return

2) The zero return request signal is turned off when zero return is completed.

(b) Absolute systems

1) The zero return request signal is turned on in the following cases:

i) During zero return

ii) Sum check error is detected when power is turned on in the back-up data (reference value)

2) The zero return request signal is turned off when zero return is completed.

3.1.11 Zero return complete signal (XnA)

(a) The zero return complete signal is turned on when the execution of the servo program based near-zero point dog zero return operation is completed without any errors or problems.

(b) The signal is turned off when the following operation starts: positioning, jog operation, MPG operation, or count zero return.

(c) If the near-zero point dog zero return is executed by the servo program while the zero return complete signal is on, a "continuous zero return start" error occurs and zero return cannot be started.

3.1.12 FLS signal (XnB)

The FLS signal is turned on and off by the ON/OFF status of the input from the upper stroke end limit switch (FLS).

(a) Input of limit switch OFF............. OFF
(b) Input of limit switch ON .... ON

3.1.13 RLS signal (XnC)

The RLS signal is turned on and off by the ON/OFF status of the input from the lower stroke end limit switch (RLS).

(a) Input of limit switch OFF............. OFF
(b) Input of limit switch ON .... ON

3.1.14 STOP signal (XnD)

The STOP signal is turned on and off by the ON/OFF status of the external stop signal (STOP).

(a) Stop signal OFF............................ OFF
(b) Stop signal ON .... ON

3.1.15 DOG/CHANGE signal (XnE)

The DOG/CHANGE signal is turned on and off by the ON/OFF status of the near-zero point dog or position change (DOG/CHANGE) signal.

(a) Near-zero point dog/position change input OFF .... OFF
(b) Near-zero point dog/position change input ON............ ON

3.1.16 Servo READY signal (XnF)

(a) The servo READY signal will be on as long as communications between the A73CPU and the servo amplifier connected to each axis are being conducted correctly. This signal is used to determine whether the servo amplifier is operating normally or not.

(b) The signal is turned off in the following cases:

1) Communications with the servo amplifier are incorrect.

2) The servo parameters are not set.

3) When the MR-SB is used, the emergency stop signal is input to the A70SF.

POINT

When the general purpose servo amplifier is used, the external emergency stop signal input to the A70SF is ignored and, therefore, the servo READY signal is not turned off.

3.1.17 Stop instruction (Yn0)

The stop instruction signal is used to stop an axis being controlled at the leading edge of this external signal. The axis for which the stop instruction signal is on cannot be started.

It is also used as the stop instruction for the axis for which velocity control is being conducted.

(a) Stop processing

<table>
<tr><th rowspan="2">Present Control Mode</th><th colspan="2">Processing when Stop Instruction is Turned ON</th></tr>
<tr><th>During Execution of Control</th><th>During Deceleration/Stop Processing Execution</th></tr>
<tr><td>Position control</td><td rowspan="3">An axis stops after deceleration.
Deceleration time is set by the parameter block or servo program.</td><td rowspan="3">The stop instruction is ignored and the presently executed deceleration/stop processing is continued.</td></tr>
<tr><td>Velocity control</td></tr>
<tr><td>Jog operation</td></tr>
<tr><td>MPG operation</td><td>An axis stops immediately without deceleration processing.</td><td>—</td></tr>
<tr><td>Zero return</td><td colspan="2">1) An axis stops after deceleration. Deceleration time is set by the parameter block.
2) If a zero return mid-stop error occurs, error code (201) is stored in the minor error storage area of the corresponding axis.</td></tr>
</table>

POINT

After stopping an axis with the stop instruction (Yn0) during zero return, execute zero return operation again.

If the stop instruction is turned on after a near-zero point dog signal has been output, the axis must be returned to a point preceding near-zero point dog signal output point in the jog or positioning operation. After that, execute the zero return operation again.

3.1.18 Immediate stop instruction (Yn1)

The immediate stop instruction signal is used to immediately stop an axis being fed at the leading edge of this external signal. The axis for which the immediate stop instruction signal is on cannot be fed.

(a) Stop processing

Present Control Mode	Processing when Stop Instruction is Turned ON	
	During Execution of Control	During Deceleration/Stop Processing Execution
Position control Velocity control Jog operation	An axis stops after deceleration. Deceleration time is set by the parameter block or servo program.	The stop instruction is ignored and the presently executed deceleration/stop processing is continued.
MPG operation	An axis stops immediately without deceleration processing.	—
Zero return	1) An axis stops after deceleration. Deceleration time is set by the parameter block. 2) If a zero return mid-stop error occurs, error code (201) is stored in the minor error storage area of the corresponding axis.	

POINT

After stopping an axis immediately using the stop instruction (Yn1) during zero return, execute zero return operation again.

If the stop instruction is turned on after a near-zero point dog signal has been output, the axis must be returned to a point preceding near-zero point dog signal output point in the jog or positioning operation. After that, execute the zero return operation again.

3.1.19 Forward jog start (Yn2) / reverse jog start (Yn3)

(a) Forward jog operation is executed (addresses increasing) when Yn2 is set by the sequence program. When Yn2 is turned off, the axis stops after deceleration (deceleration time set by the parameter block).

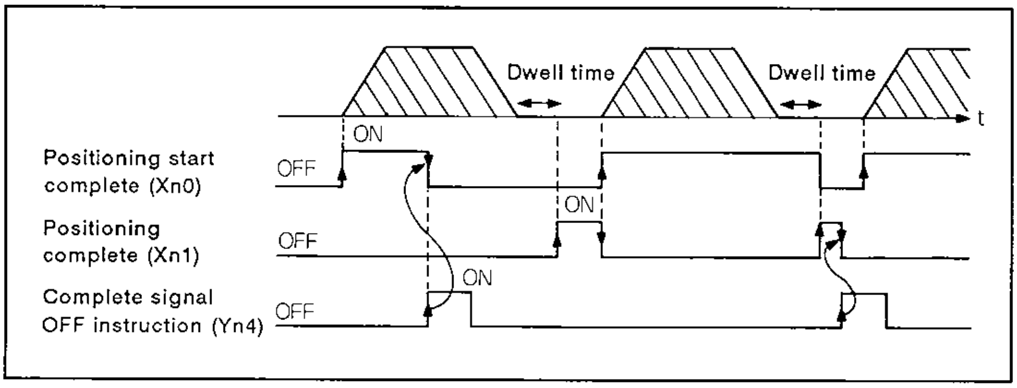
(b) Reverse jog operation is executed (addresses decreasing) when Yn3 is set by the sequence program. When Yn3 is turned off, the axis stops after deceleration (deceleration time set by the parameter block).

POINT

Provide an interlock in the sequence program so that the forward jog start (Yn2) and the reverse jog start (Yn3) will not be turned on simultaneously.

3.1.20 Complete signal OFF instruction (Yn4)

(a) The complete signal OFF instruction is used to turn off the positioning start complete signal (Xn0) and the positioning complete signal (Xn1).



POINT

Do not turn on the complete signal OFF instruction with the PLS command.

If it is turned on with the PLS command, the positioning start complete signal (Xn0) and the positioning complete signal (Xn1) cannot be turned off.

3.1.21 Velocity/position control switchover enable/disable (Yn5)

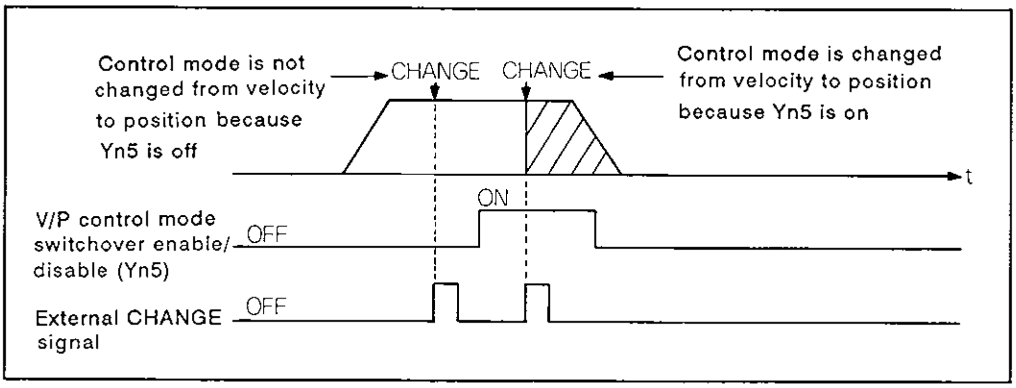
(a) The velocity/position control switchover enble/disable signal is used to make the CHANGE signal (velocity control mode to position control mode switchover signal) effective or ineffective.

1) Yn5 is ON:

The control mode is changed from the velocity control mode to the position control mode when the CHANGE signal is turned on.

2) Yn5 is OFF:

The control mode is not changed from the velocity control mode to the position control mode when the CHANGE signal is turned on.



3.1.22 Limit switch output enable/disable (Yn6)

(a) The limit switch output enable/disable signal is used to set enable/disable state of the limit switch output, described in Section 8.1.

1) Yn6: ON

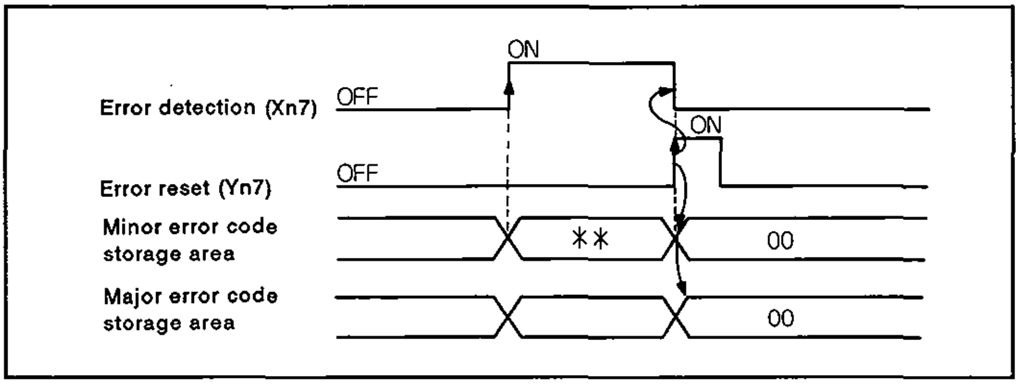
The limit switch output ON/OFF pattern is output from the AY42.

2) Yn6: OFF

The limit switch output from AY42 remains off.

3.1.23 Error reset (Yn7)

The error reset signal clears the error code stored in the corresponding axis error code storage area as well as the error detection signal (Xn7).

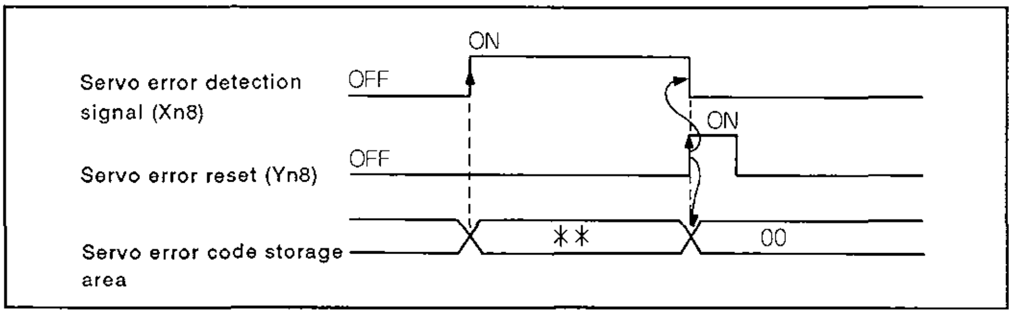


REMARKS

For details on the major and minor error code storage area, refer to Section 4.4.

3.1.24 Servo error reset (Yn8)

The servo error reset signal clears the servo error code stored in the corresponding axis servo error storage area and resets the servo error detection signal (Xn8).



POINT

Do not turn on the error reset signal (Yn7) and servo error reset signal (Yn8) with the PLS command.

If they are turned on with the PLS command, error reset or servo error reset might be impossible.

3.1.25 External STOP signal input effective/ineffective at start of positioning (Yn9)

This sets the external STOP signal input effective/ineffective.

(a) ON

The external STOP input is ineffective; positioning of the axis, for which the STOP input is on, can be started.

(b) OFF

The external STOP input is effective; positioning of the axis, for which the STOP input is on, cannot be started.

POINT

To stop an axis which has been started by turning on the Yn9 signal, turn on the STOP signal (OFF → ON). If the STOP signal is already on when the axis is started, turn off the STOP signal and then back on again one time (ON → OFF → ON).

3.1.26 Servo OFF (YnF)

The servo OFF signal is used to turn the servo system on and off.

(a) YnF: OFF .... Servo system ON
(b) YnF: ON .... Servo system OFF (free-run state)

IMPORTANT

If the servo motor is rotated manually with YnF on (free- run state) and the servo power on, the motor will rotate quickly the number of turns done manually when YnF is turned off.

Therefore, turn off the servo power before rotating the motor manually.

3.2 Internal Relay (M)

The A73CPU has 2048 internal relay points and latch relay points (M/L0 to M/L2047).

Among these internal and latch relays, internal relays M2000 to M2047 are used for positioning control. Their functions are indicated below.

Table 3.2 Internal Relay List

Device No.	Signal Name	Signal Direction
M2000	PC ready flag	SCPU → PCPU
M2001	Axis 1 start receive flag	PCPU → SCPU
M2002	Axis 2 start receive flag	
M2003	Axis 3 start receive flag	
M2004	Axis 4 start receive flag	
M2005	Axis 5 start receive flag	
M2006	Axis 6 start receive flag	
M2007	Axis 7 start receive flag	
M2008	Axis 8 start receive flag	
M2009 : M2011	Not released to users	—
M2012	MPG 1 enabled flag	SCPU → PCPU
M2013	MPG 2 enabled flag	
M2014	MPG 3 enabled flag	
M2015	Simultaneous jog operation start instruction flag	
M2016 : M2019	Not released to users	—
M2020	Start buffer full	PCPU → SCPU
M2021	Axis 1 velocity changing flag	PCPU → SCPU
M2022	Axis 2 velocity changing flag	
M2023	Axis 3 velocity changing flag	
M2024	Axis 4 velocity changing flag	
M2025	Axis 5 velocity changing flag	
M2026	Axis 6 velocity changing flag	
M2027	Axis 7 velocity changing flag	
M2028	Axis 8 velocity changing flag	
M2029 : M2047	Not released to users	—

POINT

M2000 to M2047 are not latched even if they are contained in the set latch range.

On the GPP screen, however, they are displayed as M, L, or S depending on the set latch range.

3.2.1 PC ready flag (M2000) [SCPU → PCPU]

(a) This signal informs the PCPU that the SCPU is operating correctly; the signal is turned on and off by a sequence program.

1) While the M2000 is ON, positioning control or zero return by the servo program designated by the sequence program and jog operation or MPG operation by the sequence program are possible.

2) The control indicated in step 1) will not be executed if M2000 is off or it is turned on [test mode flag (M9075) ON] with the peripheral device during a test mode.

(b) The following can be changed by a peripheral device only while the M2000 is off.

* Fixed parameters
* Servo parameters
* Limit switch output data

Any attempt to change them while M2000 is on will result in an error.

(c) The following processing is executed when M2000 is turned on (OFF → ON).

1) Processing contents

i) Servo parameters are transmitted to the MR-SB servo amplifier.

ii) The M code storage areas of all axes are cleared.

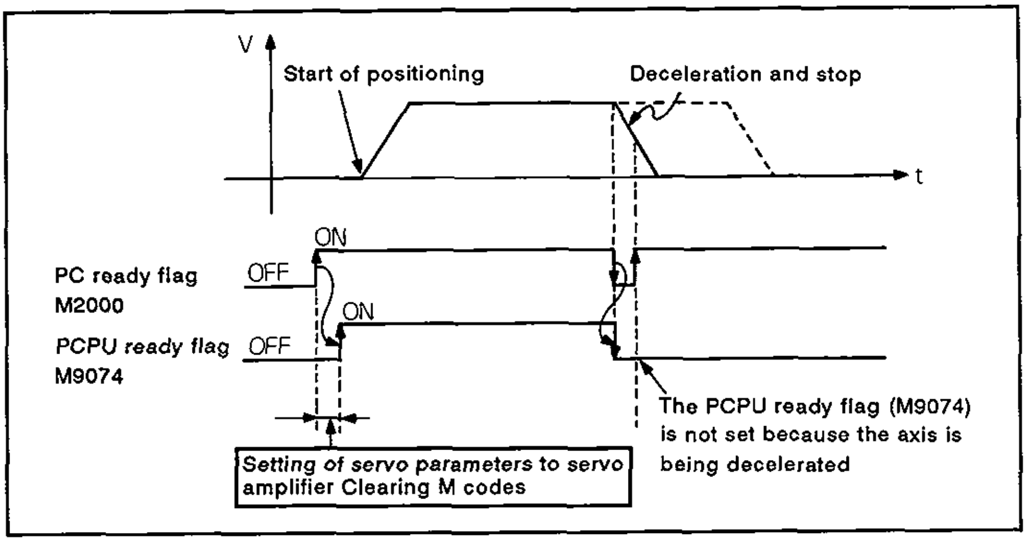
iii) When the MR-SB servo amplifier is used, the torque limit default value (300%) is set in the storage area. (see Section 3.4.1.)

iv) The PCPU ready flag (M9074) is set.

2) An error occurs if there is an axis which has already been started. In this case, the processing in step 1) is not executed.

3) In the test mode, the processing in step 1) is not executed.

If M2000 is on when the test mode is cleared, the processing in step 1) is executed.



(d) The following processing is executed when M2000 is turned off (ON → OFF).

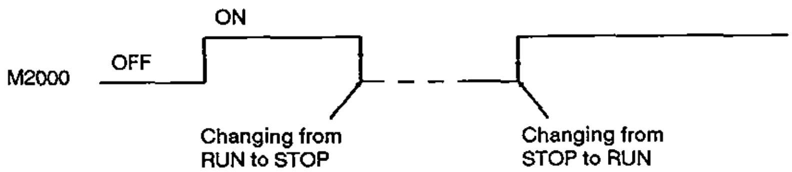
1) Processing contents

i) The PCPU ready flag (M9074) is reset.

ii) The axis which has been started is decelerated and stopped.

POINT

The PC ready flag (M2000) is turned off when the A73CPU is in the STOP state. It is returned to the previous state when the A73CPU is set in the RUN state again.

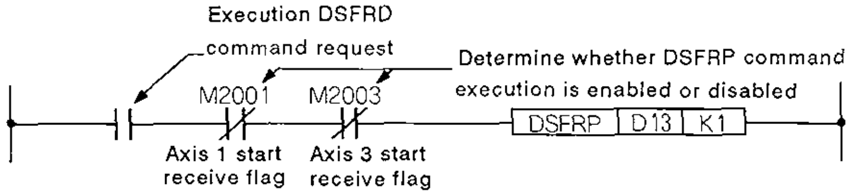


3.2.2 Start receive flag (M2001 to M2008) [PCPU → SCPU]

(a) The start receive flag is set when the positioning start command (DSFRP) in the sequence program is executed. Use this signal as the interlock for enabling or disabling the execution of the DSFRP command.

Example:

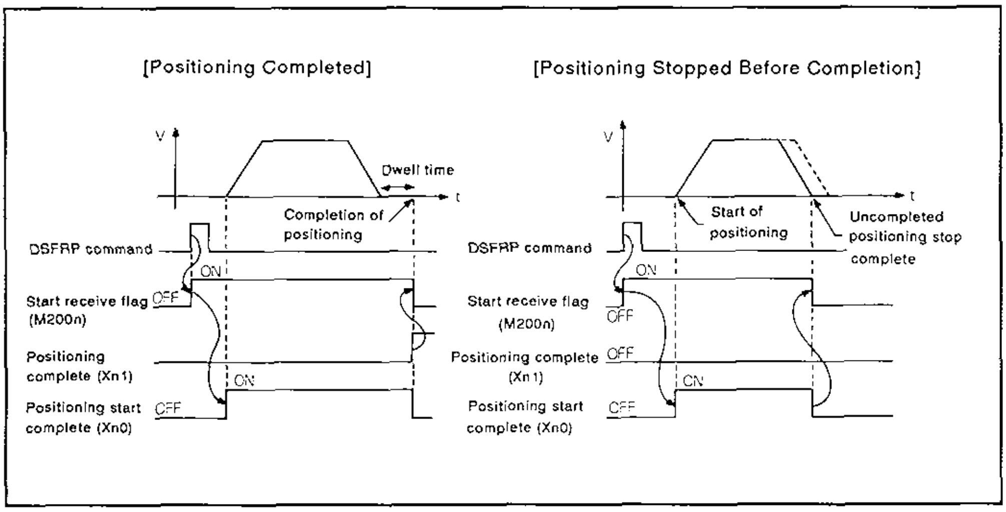
When requesting execution of a servo program to execute the positioning of axis 1 and axis 3, use the start receive flag as described below.



(b) Start receive flag processing is described below.

1) The start receive flag corresponding to the designated axis is turned on by the DSFRP command in a sequence program and turned off when positioning is complete.

   The flag is also turned off when positioning is stopped before completion.



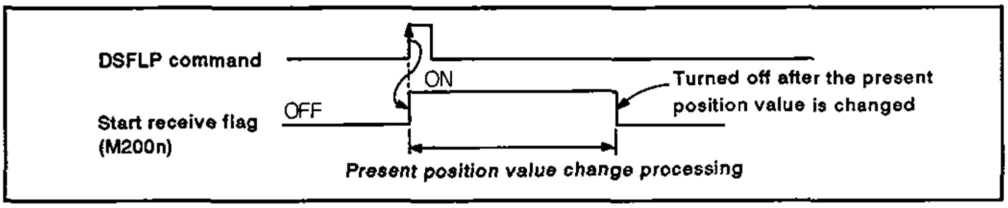
2) In the positioning control mode called by the turning on of the jog operation instruction (Yn2 or Yn3), the flag is reset when positioning is stopped due to the jog operation instruction being turned off.

3) The flag stays on while in the MPG operation enabled state (M2015 ON).

   It is turned off when MPG operation is disabled (M2015: OFF).

4) The flag is on while the present position value is changed by the DSFLP command in the PC program.

   It is reset after the present position value has been changed.



POINT

Do not forcibly turn the start receive flag on and off.

a) *If the flag is forcibly turned off by a sequence program or a peripheral device, correct positioning operation cannot be guaranteed even though an error does not occur.*

b) If the flag is forcibly turned on by a sequence program or a peripheral device, the "start receive already on error" occurs when the next positioning should be started aborting the start of positioning. An error does not occur, however, until this time.

3.2.3 MPG operation enable flag (M2012, M2013, M2014) [SCPU → PCPU]

(a) The MPG operation enable flag sets the enabled/disabled state for positioning using the MPG connected to the terminals P1 to P3 (*) of the A70SF.

1) MPG operation enable flag: ON

   Positioning control by the pulses generated with the MPG is permitted.

2) MPG operation enable flag: OFF

   Positioning control by the pulses generated with the MPG is not possible. These pulses are ignored.

3.2.4 Simultaneous jog operation start instruction (M2015) [SCPU → PCPU]

(a) When M2015 is turned on, simultaneous jog operation of the axes, set as the jog operation axis (axis 1 to axis 8) in the simultaneous jog operation start axis area (D1015), starts.

(b) When M2015 is turned off, the axes controlled in the jog operation decelerate and stop.

3.2.5 Start buffer full (M2020) [PCPU → SCPU]

(a) The start buffer full signal is turned on if more than 15 requests are given to the PCPU from the DSFRP and DSFLP commands in the sequence program. Processing by the PCPU impossible.

(b) Reset M2020 with a sequence program.

3.2.6 Velocity changing flag (M2021 to M2028) [PCPU → SCPU]

The velocity changing flag is set while the velocity is being changed with the DSFLP command in a sequence program. Use this flag as the interlock for execution of the velocity changing program.

REMARKS

*: For details of terminals P1 to P3 of the A70SF, refer to the A73CPU User's Manual.

3.3 Special Relay (SP.M)

The A73CPU has 256 special relay points (M9000 to M9255).

Among these special relays, the following seven special relay points are used for positioning control. Their functions are determined as described below.

Table 3.3 Special Relay List

Device No.	Signal Name	Signal Direction
M9073	WDT error flag	PCPU → SCPU
M9074	PCPU ready flag	
M9075	Test mode flag	
M9076	External emergency stop input flag	
M9077	MPG axis setting error flag	
M9078	Test mode request error flag	
M9079	Servo program setting error flag	

3.3.1 WDT error flag (M9073) [PCPU → SCPU]

The WDT error flag is turned on when a watch dog timer error is detected by the SCPU self-diagnosis function.

If the WDT error flag is set, reset the A73CPU with the reset key switch.

If M9073 is set again after the A73CPU is reset, the PCPU is faulty.

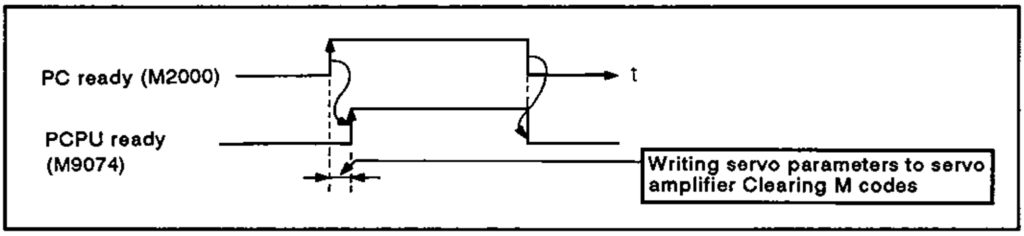
The PCPU stops the axis immediately without deceleration if it detects a WDT error.

3.3.2 PCPU ready flag (M9074) [PCPU → SCPU]

The PCPU ready flag is used to determine PCPU operation status (normal or faulty) with a sequence program.

(a) Data such as the fixed parameters, servo parameters, and limit switch output data are checked when the PC ready flag (M2000) is turned on; the flag is set when there is no error in this check.

(b) The flag is reset when the PC ready flag (M2000) is turned off.



3.3.3 Test mode flag (M9075) [PCPU → SCPU]

(a) The flag is used to determine whether or not the test mode is the present mode with a peripheral device.

Use this flag as an interlock signal to start the servo program with the DSFRP command in the sequence program.

1) OFF : Not in the test mode

2) ON : In the test mode

(b) M9078 is set if the test mode is not established in response to the test mode request signal input by a peripheral device.

3.3.4 External emergency stop input flag (M9076) [PCPU → SCPU]

This flag is used to confirm the status of the external emergency stop signal input to the EMG terminal of the A70SF.

1) OFF : External emergency stop input is ON.

2) ON : External emergency stop input is OFF.

3.3.5 MPG axis setting error flag (M9077) [PCPU → SCPU]

(a) This flag is used to determine whether or not the setting for the MPG axis setting registers (D1012 to D1014) for P1 to P3 is correct.

1) OFF : All settings for D1012 to D1014 are correct.

2) ON : Any one of the settings for D1012 to D1014 is incorrect.

(b) If M9077 is set, the contents of the error are stored in the MPG axis setting error storing register (D9187).

3.3.6 Test mode request error flag (M9078) [PCPU → SCPU]

(a) The flag is set if the test mode is not established when the test mode request is given by a peripheral device.

(b) If M9078 is set, the contents of the error are stored in the test mode request error storing register (D9188).

3.3.7 Servo program setting flag (M9079) [PCPU → SCPU]

(a) This flag is used to determine whether or not the positioning data of the servo program designated by the DSFRP command is correct.

1) OFF : Correct

2) ON : Incorrect

3.4 Data Register (D)

The A73CPU has 1024 data register points (D0 to D1023).

Among these data registers, the following 224 points (D800 to D1023) are used for positioning control. Their functions are determined as described below.

Table 3.4 Data Register List

Device No.	Signal Name
D800 to D819	Monitor data - Axis 1
D820 to D839	Monitor data - Axis 2
D840 to D859	Monitor data - Axis 3
D860 to D879	Monitor data - Axis 4
D880 to D899	Monitor data - Axis 5
D900 to D919	Monitor data - Axis 6
D920 to D939	Monitor data - Axis 7
D940 to D959	Monitor data - Axis 8
D960 to D965	Control mode change data storage area - Axis 1
D966 to D971	Control mode change data storage area - Axis 2
D972 to D977	Control mode change data storage area - Axis 3
D978 to D983	Control mode change data storage area - Axis 4
D984 to D989	Control mode change data storage area - Axis 5
D990 to D995	Control mode change data storage area - Axis 6
D996 to D1001	Control mode change data storage area - Axis 7
D1002 to D1007	Control mode change data storage area - Axis 8
D1008 to D1011	Limit switch output enabled/disabled setting
D1012	Control axis number storing area - MPG 1
D1013	Control axis number storing area - MPG 2
D1014	Control axis number storing area - MPG 3

Monitor data (D800 to D959):

Head register number	Signal	
+0	Command position data	L
1		H
2	Actual position data	L
3		H
4	Error counter value	L
5		H
6	Minor error code	
7	Major error code	
8	Servo error code	
9	Axis travel distance after the near-zero point signal is turned on	L
10		H
11	Zero return correction travel distance	
12	Execution program number	
13	M code	
14	Torque limit	
15	Axis travel change register	L
16		H
17	Actual position data when STOP is input	L
18		H
19	Not released to user	

Control mode change data storage area (D960 to D1007):

Head register number	Signal	
0	Present position data change register	L
1		H
2	Velocity change register	L
3		H
4	Jog speed setting register	L
5		H

Table 3.4 Data Register List (Continued)

Device No.	Signal Name
D1015	Simultaneous jog operation start axis setting area
D1016	Multiplication ratio storing area - MPG for axis 1
D1017	Multiplication ratio storing area - MPG for axis 2
D1018	Multiplication ratio storing area - MPG for axis 3
D1019	Multiplication ratio storing area - MPG for axis 4
D1020	Multiplication ratio storing area - MPG for axis 5
D1021	Multiplication ratio storing area - MPG for axis 6
D1022	Multiplication ratio storing area - MPG for axis 7
D1023	Multiplication ratio storing area - MPG for axis 8

3.4.1 Monitor data area (D800 to D959) [PCPU ↔ SCPU]

The monitor data area stores data such as the command position data, actual position data, and accumulated pulses in the error counter during PCPU positioning control.

The data in the monitor data area is used to confirm the positioning control status with a sequence program.

Users may not write data to the monitor data area with the exception of the axis travel distance change register.

POINT

Data storage timing to the monitor data area, with the exception of M codes and axis travel change registers, is delayed as listed below because of the on/off timing of the positioning device (input, internal relay, special relay).

a) Scan time of a sequence program is less than 80 msec ..... 80 msec

b) Scan time of a sequence program is longer than 80 msec ..... 1 scan time

Table 3.5 Monitor Data Area List

Data Name	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5	Axis 6	Axis 7	Axis 8
Command position data	D801, D800	D821, D820	D841, D840	D861, D860	D881, D880	D901, D900	D921, D920	D941, D940
Actual position data	D803, D802	D823, D822	D843, D842	D863, D862	D883, D882	D903, D902	D923, D922	D943, D942
Error counter value	D805, D804	D825, D824	D845, D844	D865, D864	D885, D884	D905, D904	D925, D924	D945, D944
Minor error code	D806	D826	D846	D866	D886	D906	D926	D946
Major error code	D807	D827	D847	D867	D887	D907	D927	D947
Servo error code	D808	D828	D848	D868	D888	D908	D928	D948
Axis travel after the near-zero point signal is turned on	D8010, D809	D8210, D829	D8410, D849	D8610, D869	D8810, D889	D9010, D909	D9210, D929	D9410, D949
Zero return correction travel distance	D8011	D8211	D8411	D8611	D8811	D9011	D9211	D9411
Execution program number	D8012	D8212	D8412	D8612	D8812	D9012	D9212	D9412
M code	D8013	D8213	D8413	D8613	D8813	D9013	D9213	D9413
Torque limit	D8014	D8214	D8414	D8614	D8814	D9014	D9214	D9414
Axis travel change register	D8016, D8015	D8216, D8215	D8416, D8415	D8616, D8615	D8816, D8815	D9016, D9015	D9216, D9215	D9416, D9415
Actual position data when STOP is input	D8018, D8017	D8218, D8217	D8418, D8417	D8618, D8617	D8818, D8817	D9018, D9017	D9218, D9217	D9418, D9417
Not released to user	D8019	D8219	D8419	D8619	D8819	D9019	D9219	D9419

(1) Command position data [PCPU → SCPU]

(a) The target address output to the servo amplifier, according to the positioning address or axis travel distance designated by the servo program, is stored.

(b) The stroke range is checked based on the command position data.

(2) Actual position data [PCPU → SCPU]

(a) In this area, data on the position where the axis was actually moved (command position data - accumulated pulses in error counter) is stored.

1) In the incremental feed control mode, axis travel distance beginning with "0" is stored.

2) In the V/P control mode, present position data in the address accessed at the start of axis motion is stored.

3) In the velocity control mode, the value "0" is stored.

(b) In the stop state, the "command position data" equals the "actual position data".

(3) Error counter value storing area [PCPU → SCPU]

In this area, difference between the command position data and the actual position data is stored.

(4) Minor error code storage area [PCPU → SCPU]

(a) In this area, the error code corresponding to a minor error is stored if such an error occurs.

If another minor error occurs after one minor error has been stored, it is overwritten by the new error code.

(b) Turn on the error reset signal (Yn7) to clear the error code.

(5) Major error code storage area [PCPU → SCPU]

(a) In this area, the error code corresponding to a major error is stored if such an error occurs.

If another major error occurs after one major error has been stored, it is overwritten by the new error code.

(b) Turn on the error reset signal (Yn7) to clear the error code.

(6) Servo error code storage area [PCPU → SCPU]

(a) In this area, the error code corresponding to a servo error is stored if such an error occurs.

If another servo error occurs after one servo error has been stored, it is overwritten by the new error code.

(b) Turn on the servo error reset signal (Yn8) to clear the error code.

(7) Storage area of axis travel distance after near-zero point signal is turned on [PCPU → SCPU]

(a) The distance through which the axis has moved until the completion of zero return after the near-zero point signal is turned on; this data is stored without a sign.

(b) In the V/P control mode, the distance moved in the position control mode is stored with a sign.

(8) Zero return correction travel distance storage area (only for MR-SB) [PCPU → SCPU]

(a) If the position reached according to the "axis travel distance after near-zero point signal is turned on" setting which is set by a peripheral device (see Section 7.18.1) is not the zero point, the A73CPU moves the axis further to reach the zero point. The distance through which the axis has been moved for this operation is stored with a sign.

With the data set zero return operation, the data remains unchanged.

(b) The data to be stored is "0" when a general-purpose servo system is used.

(9) Execution program number storage area [PCPU → SCPU]

(a) The program number of the servo program being executed when the DSFRP command is executed is stored.

(b) In jog operation or MPG operation, the following value is stored:

1) Jog operation .... FFFF

2) MPG operation .... FFFE

3) When the power is turned on FFF0

(c) Value "FFFD" is stored if the following is being executed in the test mode called by a peripheral device:

1) During zero return

2) During position loop gain check by the servo diagnosis function (only for MR-SB)

(10) M code storage area [PCPU → SCPU]

(a) The M codes set in the servo program to be executed are stored at the start of positioning.

Value "0" is set when no M codes are set in the servo program.

(b) The data in this area changes only when the positioning starts.

(c) At the leading edge of the PC ready flag (M2000), the set data is cleared to "0".

(11) Torque limit storage area [PCPU → SCPU]

When the MR-SB servo is used, the torque limit value given to the servo system is stored in this area.

When the power is turned on or when the PC ready flag (M2000) is turned on, "300%" is set.

(12) Axis travel distance change register [SCPU → PCPU]

This area is used when the axis travel distance in the position control mode should be changed for V/P control mode.

See Section 7.12.

(13) Storage area for the actual position data when STOP is input [PCPU → SCPU]

In this area, the actual position data is stored when the external STOP signal is input.

3.4.2 Control change data storage area (D960 to D1007) [SCPU → PCPU]

In this area, the present position change data, velocity change data, and jog speed change data are stored.

Table 3.6 Control Change Data Storage Area List

Data Name	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5	Axis 6	Axis 7	Axis 8
Present position data change register	D961, D960	D967, D966	D973, D972	D979, D978	D985, D984	D991, D990	D997, D996.	D1003, D1002
Velocity change register	D963, D962	D969, D968	D975, D974	D981, D980	D987, D986	D993, D992	D999, D998	D1005, D1004
Jog speed setting register	D965, D964	D971, D970	D977, D976	D983, D982	D989, D998	D995, D994	D1001, D1000	D1007, D1006

(1) Present position data change register

(a) When changing the command position data of the axis which is stopped, the required command position data is stored in this area.

(b) The setting range for the present position data change register is indicated below.

Item \ Unit	mm		inch		degree		pulse		Remarks
	Setting Range	Unit	Setting Range	Unit	Setting Range	Unit	Setting Range	Unit	
Present value after change	-2^{31} to $2^{31}-1$	x 10^{-1} μm	-2^{31} to $2^{31}-1$	x 10^{-5} inch	-2^{31} to $2^{31}-1$	x 10^{-5} degree	-2^{31} to $2^{31}-1$	PLS	An error does not occur if the setting is outside the stroke.

(c) The command position data is changed to the set data by the positioning control mode change command (DSFLP).

(d) For details, see Section 8.8.

(2) Velocity change register

(a) When changing the velocity of the axis being moved, the required velocity is stored in this area.

(b) The setting range for the velocity change register is indicated below.

Item \ Unit	mm		inch		degree		pulse	
	Setting Range	Unit	Setting Range	Unit	Setting Range	Unit	Setting Range	Unit
Velocity change	1 to 600000000	x 10^{-2} mm/min	1 to 600000000	x 10^{-3} inch/min	1 to 600000000	x 10^{-3} degree/min	1 to 1000000	PLS/sec

(c) Positioning velocity is changed to the set value by designating the positioning control mode change command (DSFLP).

(d) For details, refer to Section 8.7.

(3) Jog speed setting register

(a) Jog speed is stored in this area.

(b) The setting range of the jog speed is indicated below.

Item \ Unit	mm		inch		degree		pulse	
	Setting Range	Unit	Setting Range	Unit	Setting Range	Unit	Setting Range	Unit
Jog speed	1 to 600000000	x 10^{-2} mm/min	1 to 600000000	x 10^{-3} inch/min	1 to 600000000	x 10^{-3} degree/min	1 to 1000000	PLS/sec

(c) The jog feedrate set in the register becomes effective at the leading edge of the jog start signal.

It is not possible to change jog speed if the data is changed during jog operation.

(d) For details, refer to Section 7.16.

3.4.3 Limit switch output disable setting register (D1008 to D1011) [PCPU → SCPU]

(a) *In this area, limit switch output is enabled and disabled in units of points.*

Setting the corresponding bit on disables limit switch output and the output remains off.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
D1008	Y0F	Y0E	Y0D	Y0C	Y0B	Y0A	Y09	Y08	Y07	Y06	Y05	Y04	Y03	Y02	Y01	Y00
	For axis 2								For axis 1							
D1009	Y1F	Y1E	Y1D	Y1C	Y1B	Y1A	Y19	Y18	Y17	Y16	Y15	Y14	Y13	Y12	Y11	Y10
	For axis 4								For axis 3							
D1010	Y2F	Y2E	Y2D	Y2C	Y2B	Y2A	Y29	Y28	Y27	Y26	Y25	Y24	Y23	Y22	Y21	Y20
	For axis 6								For axis 5							
D1011	Y3F	Y3E	Y3D	Y3C	Y3B	Y3A	Y39	Y38	Y37	Y36	Y35	Y34	Y33	Y32	Y31	Y30
	For axis 8								For axis 7							

Setting of each bit is "1" or "0".

1: Disabled .... Limit switch output remains off.
0: Enabled .... Limit switch output is turned on and off according to the set data.

3.4.4 Storage area for axis numbers to be controlled by MPG (D1012, D1013, D1014) [SCPU → PCPU]

(a) In this area, the numbers of the axes to be controlled by MPGs (P1 to P3) is stored.

(b) For details, see Section 7.17.

3.4.5 Simultaneous jog operation start axes setting area (D1015) [SCPU → PCPU]

(a) In this area, the numbers of the axes to be started simultaneously for jog operation are stored; jog operation direction is stored as well.

	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
	Reverse jog start axis								Forward jog start axis							

Setting of each bit is "1" or "0".

1: Simultaneous jog operation executed
0: Simultaneous jog operation not executed

(b) For details, see Section 7.16.3.

3.4.6 Pulse multiplication ratio storage area (D1016 to D1023) [SCPU → PCPU]

(a) In this area, the multiplication ratio (1 to 100) for one pulse output from the MPG is set.

(b) For details, see Section 7.17.

IB (NA) 66233-A

3.5 Special Register (D)

The A73CPU has 256 data register points (D9000 to D9255).

Among these data registers, the following 20 points (D9180 to D9199) are used for positioning control. Their functions are determined as described below.

Table 3.7 Special Register List

Device number	Signal Name
D9180	Limit switch output status storing area - Axis 1 and 2
D9181	Limit switch output status storing area - Axis 3 and 4
D9182	Limit switch output status storing area - Axis 5 and 6
D9183	Limit switch output status storing area - Axis 7 and 8
D9184	PCPU error factor
D9185	Servo amplifier installation information
D9186	Not used
D9187	MPG axis setting error
D9188	Test mode request error
D9189	Error program number
D9190	Error item information
D9191	Servo amplifier type
D9192 to D9199	Not used

3.5.1 Limit switch output status storing area (D9180 to D9183) [PCPU → SCPU]

(a) The output status (ON/OFF) of the limit switch output to the AY42 is stored as "1" or "0".

1) 1: ON
2) 0: OFF

(b) The data stored in these areas is used to output the limit switch output data externally with a sequence program.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
D9180	Y0F	Y0E	Y0D	Y0C	Y0B	Y0A	Y09	Y08	Y07	Y06	Y05	Y04	Y03	Y02	Y01	Y00
	For axis 2								For axis 1							
D9181	Y1F	Y1E	Y1D	Y1C	Y1B	Y1A	Y19	Y18	Y17	Y16	Y15	Y14	Y13	Y12	Y11	Y10
	For axis 4								For axis 3							
D9182	Y2F	Y2E	Y2D	Y2C	Y2B	Y2A	Y29	Y28	Y27	Y26	Y25	Y24	Y23	Y22	Y21	Y20
	For axis 6								For axis 5							
D9183	Y3F	Y3E	Y3D	Y3C	Y3B	Y3A	Y39	Y38	Y37	Y36	Y35	Y34	Y33	Y32	Y31	Y30
	For axis 8								For axis 7							

For each bit (D9180 to D9183), "1" or "0" is stored.

1: ON
0: OFF

3.5.2 PCPU error factor (D9184) [PCPU → SCPU]

(a) The register is used to determine the contents of the PCPU error with a sequence program.

1) 0 .... Normal
2) 1 .... A73PCPU hardware faulty
3) 2 .... PCPU error
4) 10, 11 .... A70AF error
5) 12 .... A70MDF error
6) 13 .... AY42 error

IB (NA) 66233-A

3.5.3 Servo amplifier installation information (D9185) [PCPU → SCPU]

(a) The servo amplifier installation status is checked when the power is turned on to the A73CPU. The results of the check are stored in the least significant eight bits of the register.

When a servo amplifier is installed after the power is turned on, the servo amplifier installation status is changed from "not installed" to "installed". However, if the servo amplifier is removed after the power is turned on, the servo installation status of the axis remains "installed".

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

b15 to b8: All 0s

b7 to b0: Installed/not-installed information

Installed 1
Not installed 0

1) Installed status

The MR-SB servo amplifier or a general-purpose servo amplifier is operating correctly.

2) Not-installed status

A servo amplifier is not installed.

The servo amplifier is off.

Communications with the servo amplifier are incorrect, possibly due to a faulty cable.

3.5.4 MPG axis setting error (D9187) [PCPU → SCPU]

(a) MPG axis setting error contents are stored in this area if the MPG axis setting error flag (M9077) is set.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
D9187	Axis 8	Axis 7	Axis 6	Axis 5	Axis 4	Axis 3	Axis 2	Axis 1	0	0	0	0	0	P3	P2	P1

b2 to b0: MPG axis setting error for the MPGs connected to terminals P1 to P3 of the A70SF is stored

0: Correct
1: Setting error (Setting is not 1 to 8)

b7 to b3: Not used

b15 to b8: Pulse multiplication ratio setting error is stored

0: Correct
1: Setting error (Setting is not 1 to 100)

3.5.5 Test mode request error (D9188) [PCPU → SCPU]

(a) The number of the axis which has been started when the test mode request error flag (M9078) is set is stored.

	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

b15 to b8: Not used

b7 to b0: Axis state (moving/stopped) is stored
0: Stopped
1: Moving

3.5.6 Error program number (D9189) [PCPU → SCPU]

(a) The servo program number (0 to 4095) of the program causing an error is stored if the servo program setting error flag (M9079) is set.

3.5.7 Error item information (D9190) [PCPU → SCPU]

(a) The error code corresponding to the error item is stored if the servo program setting error flag (M9079) is set.

Error Code	Contents
900	The servo program designated with the DSFRP command is not found.
901	The axis number designated with the DSFRP command and the axis number designated in the servo program do not agree.
902	The command code cannot be decoded (a command which should not be present is read.).
Error item data	The setting item designated with the DSFRP command in the servo program contains an error. The error item data, explained in Section 6.3, is stored.

3.5.8 Servo amplifier (D9191) [PCPU → SCPU]

(a) The number of the axes to which the general-purpose servo amplifier is connected is stored; this check is made when the power is turned on.

("0" for the axis to which MR-SB is connected or no servo amplifier is connected.)

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
D9191	0	0	0	0	0	0	0	0	Axis 8	Axis 7	Axis 6	Axis 5	Axis 4	Axis 3	Axis 2	Axis 1

b15 to b8: All 0s

b7 to b0: "1" for the axis to which the general servo amplifier is connected

4. POSITIONING CONTROL PARAMETERS

The A73CPU uses the following six types of positioning control parameters.

A default is set for all of the positioning control parameters. When the setting should be changed from the default, set the required value with a peripheral device.

(1) Fixed parameters

The fixed parameters are determined according to the mechanical system and should be set for each individual axis.

They are used to calculate the command position when executing positioning control.

For details, see Section 4.1.

(2) Servo parameters

The servo parameters include the servo name determined according to the connected servo motor and the motor type, and should be set for each individual axis.

They are used to control the servo motor when executing positioning control.

For details, see Section 4.2.

(3) Zero return data

The zero return data includes the zero return direction, mode, and velocity, and should be set for each individual axis.

They are used when zero return is executed.

For details, see Section 7.18.1.

(4) Jog operation data

The jog operation data includes jog velocity limit and parameter block number data. It should be set for each individual axis.

It is used for jog mode positioning control.

For details, see Section 7.16.1.

(5) Parameter block

The parameter block includes the acceleration/deceleration time and velocity limit. Up to 16 blocks can be set.

The parameter block is designated by the servo program, jog operation data, or zero return data; it facilitates the changes in acceleration/deceleration processing (acceleration time, deceleration time, velocity limit) during positioning control.

For details, see Section 4.3.

(6) Limit switch output data

The limit switch output data is the ON/OFF pattern which is output if the setting for the "limit switch output" (fixed parameter) is "used".

The axis for which the limit switch output data and the set ON/OFF pattern is output in positioning control.

For details, see Section 8.1.

4.1 Fixed Parameters

(1) The fixed parameters are the fixed data which is determined according to the mechanical system; the parameters are set for each axis.

(2) The default is set before shipment. Set the required data if it is necessary to change the setting from the default.

(3) The fixed parameters are summarized in Table 4.1.

Table 4.1 Fixed Parameter List

No.	Item		Setting Range								Default		Remarks
			mm		inch		degree		pulse		Setting	Unit	
			Setting range	Unit	Setting range	Unit	Setting range	Unit	Setting range	Unit			
1	Unit setting		0	—	1	—	2	—	3	—	3	—	Set the command unit for positioning control.
2	Travel distance per pulse (A)	*1	1 to 65535 PLS								20000	PLS	Set the number of feedback pulses per one rotation of the motor. See Section 4.1.1.
3		*2	1 to 65535	$\times 10^{-1}$ μm	1 to 65535	$\times 10^{-5}$ inch	1 to 65535	$\times 10^{-5}$ degree	1 to 65535	PLS	20000	PLS	Set axis travel distance per one rotation of the motor. See Section 4.1.1.
4		*3	1: x 1, 10: x10, 100: x100, 1000: x1000						—	—	1	—	Set the multiplication ratio for one pulse. See Section 4.1.1.
5	Backlash compensation amount		0 to 65535	$\times 10^{-1}$ μm	0 to 65535	$\times 10^{-5}$ inch	0 to 65535	$\times 10^{-5}$ degree	0 to 255	PLS	0	—	Set the backlash amount in the mechanical system. Backlash is compensated for each time positioning direction is changed. See Section 8.3.
6	Upper stroke limit		-2^{31} to $2^{31}-1$	$\times 10^{-1}$ μm	-2^{31} to $2^{31}-1$	$\times 10^{-5}$ inch	-2^{31} to $2^{31}-1$	$\times 10^{-5}$ degree	-2^{31} to $2^{31}-1$	PLS	$2^{31}-1$	PLS	Set the upper limit to define the axis travel distance range. See Section 4.1.2.
7	Lower stroke limit		-2^{31} to $2^{31}-1$	$\times 10^{-1}$ μm	-2^{31} to $2^{31}-1$	$\times 10^{-5}$ inch	-2^{31} to $2^{31}-1$	$\times 10^{-5}$ degree	-2^{31} to $2^{31}-1$	PLS	0	PLS	Set the lower limit to define the axis travel distance range. See Section 4.1.2.
8	Command in-position width		1 to 32767000	$\times 10^{-1}$ μm	1 to 32767000	$\times 10^{-5}$ inch	1 to 32767000	$\times 10^{-5}$ degree	1 to 32767	PLS	100	PLS	Set the position where the command in-position signal (Xn3) is turned on; (positioning address) – (present position data) See Section 4.1.3.
9	Limit switch output used/not-used		0: Not used 1: Used								0	—	Set whether or not the limit switch output function is used. See Section 8.1.

*1: Number of pulses per rotation (Ap)
*2: Travel distance per rotation (AL)
*3: Unit multiplication ratio (AM)

4.1.1 Travel distance per pulse

Axis travel distance per pulse is determined by the mechanical system. It is calculated by the PCPU according to the following data set with a peripheral device: number of pulses per motor rotation, travel distance per motor rotation, and unit multiplication ratio.

(1) Calculation of axis travel distance per pulse

(a) Mechanical specification

For the calculation of the axis travel distance per pulse, feed screw lead (pitch), the number of gear teeth on the motor output shaft, and the number of slits in the pulse generator are necessary.

1) Feed screw lead .... Pit (mm/rev)
2) Number of gear teeth (motor shaft) Z1
3) Number of gear teeth (feed screw) Z2
4) Number of slits in pulse generator................ n (PLS/rev)

(b) In this case, the number of pulses generated per motor rotation, axis travel distance per motor rotation, and unit multiplication ratio are as indicated below.

1) Number of pulses per motor rotation = n

2) Axis travel distance per motor rotation = Pit x (Z1/Z2) x 10^4 (10^{-1} μm)

3) Unit multiplication ratio = 1

(c) Axis travel distance per pulse (A)

The axis travel distance per pulse is calculated using the following formula.

$$A = \frac{\text{Axis travel distance per rotation}}{\text{Number of pulses per rotation}} \times \text{Unit multiplication ratio}$$

$$= \frac{\text{Pit} \times (Z1 / Z2) \times 10^4}{n} \times 1 \ (10^{-1}\ \mu\text{m /PLS})$$

(2) Error compensation

When positioning is carried out based on the set "axis travel distance per pulse", an error (mechanical error) will be generated between the "command travel distance" and the "actual travel distance".

The A73CPU compensates for these errors by changing the number of pulses per rotation, axis travel distance per rotation, and the unit multiplication ratio.

The error compensation method used by the A73CPU is described below:

(a) Executes positioning by setting "command travel distance".

(b) Measures "distance actually travelled" after positioning is completed.

(c) According to the "command travel distance" and "actual travel distance", the number of pulses per rotation and axis travel distance per rotation are calculated using the following formula:

1) Number of pulses per rotation = n x (command travel distance)

2) Axis travel distance per rotation = Pit x (Z1 / Z2) x 10^4 x (actual travel distance)

(d) The ratio between the "number of pulses per rotation" and the "travel distance per rotation" is approximated and reduced to a simple fraction.

$$\frac{\text{Number of pulses per rotation}}{\text{Axis travel distance per rotation}} \longrightarrow \text{Reduced to a simple fraction}$$

(e) The number of pulses per motor rotation and axis travel distance per motor rotation set as the fixed parameter are adjusted to the calculated values.

(f) Positioning control is executed using the new values.

Example:

Command travel distance = 100 mm
Actual travel distance = 101 mm

$$\frac{\text{Number of pulses per rotation}}{\text{Axis travel distance per rotation}}$$

$$= \frac{n \times 100}{\text{Pit} \times \frac{Z1}{Z2} \times 10^4 \times 101}$$

$$= \frac{n}{\text{Pit} \times \frac{Z1}{Z2} \times 10100} \longrightarrow \text{Reduced to a simple fraction}$$

4.1.2 Upper/lower stroke limit values

These values set the upper and lower limits of axis travel range.

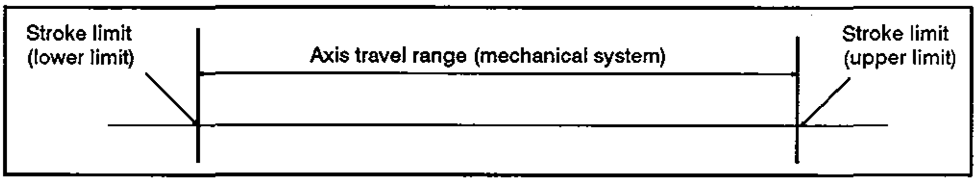


Fig. 4.1 Axis Travel Range Defined by Upper/Lower Stroke Limits

(1) Stroke limit check

The stroke limit check is executed at or during the start of positioning.

Control Mode	Check	Description
Positioning control	Checked	1) At the start of positioning, the command position data is checked to determine if it is within the travel range. If it is outside the set travel range, an error (error code: 106) occurs and positioning is not executed. 2) If the interpolation path exceeds the set travel range during the start of circular interpolation, an error (error code: 207, 208) occurs and the axes decelerate and stop.
Incremental feed	Checked	—
Velocity control	Not checked	The present position data is cleared to "0"; the axis keeps moving until an external limit signal (FLS, RLS, STOP) is received.
V/P control mode (includes re-start)	Checked	Checked after the control mode is changed to the position control.
Jog operation	Not checked	The axis keeps moving until the jog signal is turned off or an external limit signal is received even if the present position data is outside the travel range.
Velocity change control	Checked	—
Constant velocity control	Checked	—

POINT

The travel range is set using the external limit signals (FLS, RLS) in addition to the upper/lower stroke limit settings with fixed parameters.

a) The axis decelerates and stops when the external limit signal is received.

b) Deceleration/stop time can be set with the "deceleration time" and "immediate stop deceleration time" with a servo program.

4.1.3 Command in-position width

The command in-position indicates the difference between the positioning address (command position) and the present position data.

If the command in-position width is set, the command in-position signal (Xn3) is turned on when the difference between the command value and present position data becomes less than the set value [(command position 1) − (present position data 1) $\leq$ (command in-position width)].

For details on the command in-position signal, see Section 3.1. The command in-position check is constantly executed during positioning control.

4.2 Servo Parameters

(1) The servo parameters are determined according to the specification of the servo motor controlled by the parameters. They are set for each axis.

(2) The default is set before shipment. Set the required data if it is necessary to change the setting from the default.

POINT

1) After changing the setting of the data indicated by an asterisk (*1), reset the A73CPU with the RUN key switch or set the PC ready flag (M2000). Turn on the servo power.
2) When a general-purpose servo is used, set only the items indicated by (*2).
3) After setting the parameters with a peripheral device, carry out a "cross-check" and execute positioning control only after a "no error" state is confirmed.

(3) Servo parameters are summarized in Table 4.2.

Table 4.2 Servo Parameter List

<table>
<tr><th rowspan="3">No.</th><th rowspan="3">Item</th><th colspan="8">Setting Range</th><th colspan="2">Default</th><th rowspan="3">Description</th></tr>
<tr><th colspan="2">mm</th><th colspan="2">inch</th><th colspan="2">degree</th><th colspan="2">pulse</th><th rowspan="2">Setting</th><th rowspan="2">Unit</th></tr>
<tr><th>Setting range</th><th>Unit</th><th>Setting range</th><th>Unit</th><th>Setting range</th><th>Unit</th><th>Setting range</th><th>Unit</th></tr>
<tr><td>1*1*2</td><td>System setting</td><td colspan="8">0: MR-SB servo amplifier
1: MR-SB servo amplifier (absolute)
2: General-purpose servo amplifier</td><td>0</td><td></td><td>Set the servo amplifier to be used.</td></tr>
<tr><td>2*1</td><td>Regenerative resistance</td><td colspan="8">0: External regenerative resistor not used
1: External regenerative resistor used (MR-RB30)
2: External regenerative resistor used (MR-RB50/MR-RB51)
3: External regenerative resistor used (MR-RB100/101)</td><td>0</td><td></td><td>Set the name of regenerative resistor installed to the AC servo motor.</td></tr>
<tr><td>3*1</td><td>Motor type</td><td colspan="8">0: Standard
1: Low inertia L
2: Flat U</td><td>0</td><td></td><td>Set the type of AC servo motor.</td></tr>
<tr><td>4*1</td><td>Motor capacity</td><td colspan="8">Set the value (motor capacity x 10)
<table>
<tr><th>Motor Output (KW)</th><th>Setting</th><th>Motor Output (KW)</th><th>Setting</th></tr>
<tr><td>0.2</td><td>2</td><td>2.0</td><td>20</td></tr>
<tr><td>0.3</td><td>3</td><td>3.0 (U, L only)</td><td>30</td></tr>
<tr><td>0.5</td><td>5</td><td>3.5</td><td>35</td></tr>
<tr><td>1.0</td><td>10</td><td>5.0</td><td>50</td></tr>
<tr><td>1.5</td><td>15</td><td>7.0</td><td>70</td></tr>
</table></td><td>0</td><td></td><td>Set the output capacity (kW) of the AC servo motor.
If the setting is "0", access to the servo is not executed.</td></tr>
<tr><td>5*1*2</td><td>Motor rpm (R)</td><td colspan="8">
<table>
<tr><th>Motor Rated Speed (rpm)</th><th>Setting</th></tr>
<tr><td>2000</td><td>2</td></tr>
<tr><td>3000</td><td>3</td></tr>
</table></td><td>2</td><td></td><td>Set the rated speed (rpm) of the AC servo motor.</td></tr>
<tr><td>6</td><td>Position loop gain</td><td colspan="8">1 to 999 rad/sec</td><td>25</td><td>rad/sec</td><td>The ratio of the command pulse frequency to the accumulated pulses in the error counter.
See Section 4.2.1.</td></tr>
<tr><td>7</td><td>Velocity loop gain</td><td colspan="8">1 to 9999</td><td>100</td><td></td><td>Set the gain needed to improve frequency response for larger load inertia ratio (GDL2/GDH2).
See Section 4.2.2.</td></tr>
<tr><td>8</td><td>Velocity integration compensation</td><td colspan="8">1 to 9999 ms</td><td>20</td><td>msec</td><td>Set the data needed to improve transient characteristics by improving the frequency response of the velocity control system.
See Section 4.2.3.</td></tr>
<tr><td>9</td><td>In-position range</td><td>1 to 32767000</td><td>$\times 10^{-1}$ µm</td><td>1 to 32767000</td><td>$\times 10^{-5}$ inch</td><td>1 to 32767000</td><td>$\times 10^{-5}$ degree</td><td>1 to 32767</td><td>PLS</td><td>100</td><td>PLS</td><td>Set the number of accumulated pulses in the error counter.
The in-position signal (Xn2) is turned on when the number of pulses in the error counter equals the set number.
See Section 4.2.4.</td></tr>
<tr><td>10*2</td><td>Number of feedback pulses per motor rotation (N)</td><td colspan="8">1 to 65535 PLS</td><td>12000</td><td>PLS</td><td>Set the number of feedback pulses per motor rotation.</td></tr>
<tr><td>11*1*2</td><td>Rotating direction</td><td colspan="8">0: Forward rotation (CCW) as positioning addresses increase
1: Reverse rotation (CW) as positioning addresses increase</td><td>0</td><td>—</td><td>Set the rotating direction as viewed from the load side.
Forward:

Reverse:
</td></tr>
</table>

4.2.1 Position loop gain

(a) The position loop gain indicates the control responsiveness during the position control mode. It is related to the number of accumulated pulses in the error counter during operation and stop operation.

The number of accumulated pulses (ε) is calculated with the following formula.

$$\varepsilon = \frac{f}{kp}$$

ε : Number of accumulated pulses
f : Command pulse frequency (pps)
kp : Position loop gain (rad/sec)

1) When the position loop gain is low, the number of accumulated pulses increases, elongating the stabilizing period when the axis stops.

2) If the position loop gain is too high, the overshoot occurring when axis stops will be excessively large. While an axis is stopped it has a tendency to vibrate.

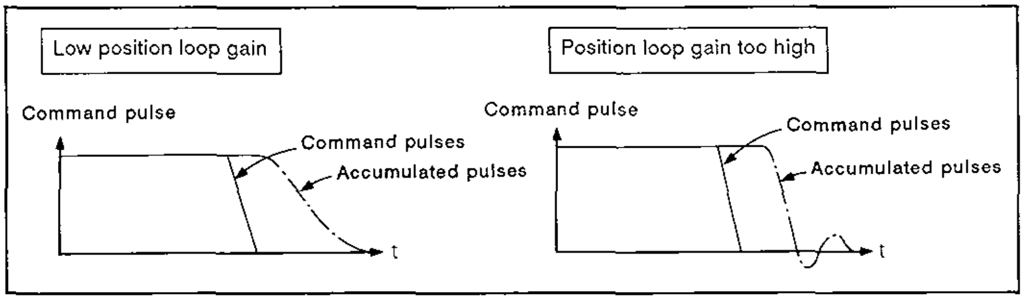


Fig. 4.2 Stop Mode Varies According to Position Loop Gain

(b) Reference values for setting the position loop gain are shown in Table 4.3.

Table 4.3 Position Loop Gain (Reference Values)

Load Inertia Ratio (GDL2/GDH2)		0	1	3	5	Description
Setting (rad/sec)	Standard	35	35	25	15	Setting range: 1 to 999
	Maximum	100	80	40	25	

POINT

1) If the position loop gain is too low, the number of accumulated pulses in the error counter increases resulting in a servo error (excessive position error) during high-speed operation.

2) When the MR-SB is used, the set position loop gain can be checked with a peripheral device.

For the procedure for checking the position loop gain with a peripheral device, refer to the peripheral device operating manual.

4.2.2 Velocity loop gain

(a) Velocity loop gain indicates the control responsiveness during the velocity control mode.

Increase the velocity loop gain if the velocity loop gain becomes too low deteriorating responsiveness as the load inertia ratio (GDL2/GDH2) increases.

(b) Reference values for setting the velocity loop gain are shown in Table 4.4.

Table 4.4 Velocity Loop Gain (Reference Values)

Load Inertia Ratio (GDL2/GDH2)	0	1	3	5	Description
Setting value	100	100	200	300	Setting range: 1 to 9999

POINT

1) If the velocity loop gain is too low, overshoot becomes excessive and vibration (motor noise) is generated while an axis is stopping.
2) When the MR-SB is used, the set velocity loop gain can be checked with a peripheral device.

   For the procedure for checking the velocity loop gain with a peripheral device, refer to the peripheral device operating manual.

4.2.3 Velocity integration compensation

(a) Use this parameter to improve the transient characteristics by increasing the frequency response during velocity control.

(b) If overshoot cannot be reduced during acceleration or deceleration after adjusting the velocity loop gain, increase the velocity integration compensation setting.

(c) Reference values for the velocity loop gain setting are shown in Table 4.5.

Table 4.5 Velocity Integration Compensation (Reference Values)

Load Inertia Ratio (GDL2/GDH2)	0	1	3	5	Description
Setting value (ms)	20	20	30	40	Setting range: 0 to 9999

4.2.4 In-position range

(a) The in-position range indicates the number of accumulated pulses in the error counter.

(b) When the difference between the command position data and the servo motor position feedback data is reduced to a value within the set range, the in-position signal (Xn2) is turned on.

4.3 Parameter Block

(1) The parameter block is provided to facilitate the setting or changing of the data used for acceleration/deceleration control during positioning processing.

(2) Up to 16 parameter blocks can be set.

(3) Default values are set before shipment. Set the required data to change the setting from the default values.

(4) Parameter blocks are summarized in Table 4.6.

Table 4.6 Parameter Block List

No.	Item	Setting Range								Default		Description
		mm		inch		degree		pulse		Setting	Unit	
		Setting range	Unit	Setting range	Unit	Setting range	Unit	Setting range	Unit			
1	Interpolation control unit	0	—	1	—	2	—	3	—	3	—	Set the units used for interpolation control. The setting is also used for the unit of command velocity and circular interpolation radius to be set in the servo program. See Section 7.1.4.
2	Velocity limit	0.01 to 6000000.00	mm/min	0.001 to 600000.000	inch/min	0.001 to 600000.000	degree/min	1 to 1000000	PLS/sec	200000	PLS/sec	Set the maximum velocity to be used for positioning or zero return operation. If the positioning velocity setting or zero return velocity setting is greater than the velocity limit, operation is controlled by replacing the set velocity with the velocity limit value. See Section 4.3.1.
3	Acceleration time	1 to 65535 msec								1000	ms	Set the time in which the velocity reaches the velocity limit after the start of positioning. See Section 4.3.1.
4	Deceleration time	1 to 65535 msec								1000	ms	Set the time in which the axis being moved at the set limit velocity is stopped when an immediate stop is executed. See Section 4.3.1.
5	Immediate stop deceleration time	1 to 65535 msec								1000	ms	Set the time in which the axis being moved at the set limit velocity is stopped when an immediate stop is executed. See Section 4.3.1.
6	Torque limit	1 to 500%								300	%	Set the servo program torque limit. The setting is not effective when a general-purpose servo amplifier is used.
7	Deceleration processing when the STOP signal is input	0: The axis decelerates and stops based on the set deceleration time. 1: The axis decelerates and stops based on the set immediate stop deceleration time.								0	—	Set the deceleration processing in response to the input of external signal (STOP, FLS, RLS).
8	Allowable circular interpolation error limits	0 to 100000	$\times 10^{-1}$ μm	0 to 100000	$\times 10^{-5}$ inch	1 to 100000	$\times 10^{-5}$ degree	0 to 100000	PLS	100	PLS	Set the allowable error limit between the circular path generated by the circular interpolation function and the set end point coordinate value. See Section 4.3.2.

POINT

1) Designate the parameter block with the zero return data, jog operation data, and servo program.
2) Each parameter block data can be changed with the servo program.

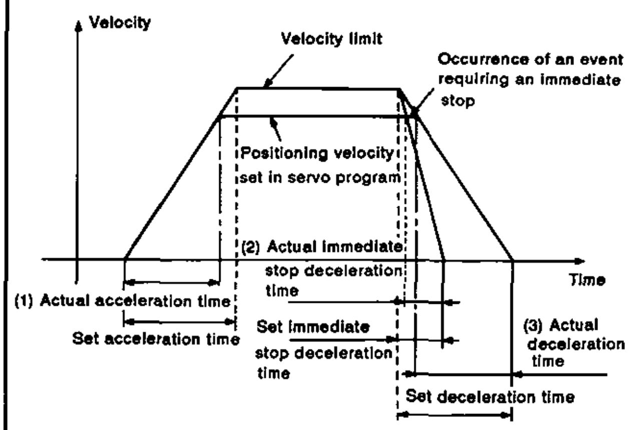
4.3.1 The relationship among velocity limit, acceleration time, deceleration time, and immediate stop deceleration time

The velocity limit sets the maximum velocity allowed for positioning and zero return operation.

Acceleration time sets the time in which the velocity reaches the set velocity limit after the start of axis travelling.

Deceleration time and immediate stop deceleration time indicate the time in which an axis being moved at the set limit velocity stops.

Therefore, the actual acceleration/deceleration time and the immediate stop deceleration time are shorter than the settings because the actual positioning velocity is lower than the set velocity limit value.



(1) Actual acceleration time
The time in which the axis reaches the positioning velocity set by the servo program.

(2) Actual immediate stop deceleration time
The time in which the axis being moved at the positioning velocity set by the servo program stops immediately.

(3) Actual deceleration time
The time in which the axis being moved at the positioning velocity set by the servo program stops.

Fig. 4.3 Relationship among Velocity Limit, Acceleration Time, Deceleration Time, and Immediate Stop Deceleration Time

4.3.2 Allowable circular interpolation error limits

In circular interpolation, executed by designating the center point, there may be cases where the path calculated with the start point address and center point address does not match the set end point address.

This parameter sets the allowable error between the calculated path and the set end point address.

If the error is within the set limit, circular interpolation is executed to the set end point address while compensating for the error through spiral interpolation.

If the error is greater than the set limit, an error occurs when axis travelling starts. The corresponding error code is stored to the minor error code area for that axis.

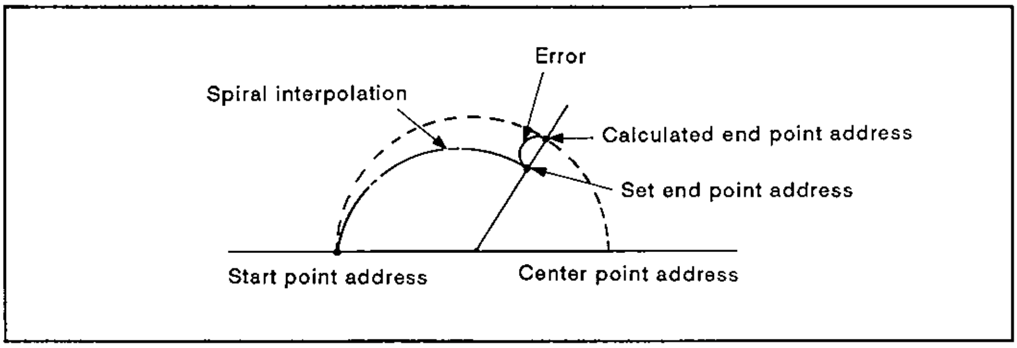


Fig. 4.4 Spiral Interpolation

5. SEQUENCE PROGRAM

5.1 Cautions on Programming Sequence Programs

(1) Positioning control commands

Servo program start command (DSFRP) and present position data and velocity change command (DSFLP) are used for the positioning control commands.

(2) Commands that cannot be used

The following commands cannot be used:

DSFL (one word shift to left)

DSFR (one word shift to right)

If the DSFL or DSFR command is attempted an operation error will occur and following error processing is executed.

(a) The operation error flag (M9010, M9011) is set.

(b) Error data (50 (OPERATION ERROR)) is stored in the self-diagnosis error code register (D9008).

(c) The step containing the DSFL or DSFR command is stored in the error step register (D9010, D9011).

Use the BMOV command to execute word data shift. See Appendix 4.

(3) Special devices for the PCPU

Among the devices which are used by the A73CPU, the use of those indicated in Table 5.1 is determined by the PCPU. Therefore, use such devices in the sequence program after confirming their functions. For details, see Section 3.

Table 5.1 Special Devices for the PCPU

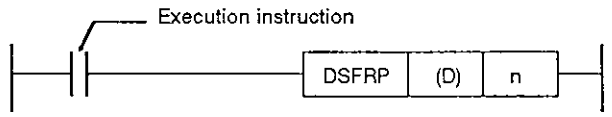
Device Name	Device Number
Input	X0 to X7F
Output	Y0 to Y7F
Internal relay	M2000 to M2047
Data register	D800 to D1023
Special relay	M9073 to M9079
Special register	D9180 to D9189

The internal relays (M2000 to M2047) and the data registers (D800 to D1023) are not latched when they are included in the set latch range. The device symbols for M2000 to M2047 are displayed as M, L, or S on the GPP according to the parameter settings.

5.2 Servo Program Start Request Command (DSFRP)

	Devices																				*4	*5	*6	*7	*8	*9	
	Bit Device							Word (16 Bit) Device									*1		*2		*3							
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K	H	P	I	N					*10	*11	*12
(D)										o													7		x		o	o
N																	o	o										

*1 Constant
*2 Pointer
*3 Level
*4 Column Designation
*5 Number of Steps
*6 Sub-set
*7 Index
*8 Carry Flag
*9 Error Flag
*10 M9012
*11 M9010
*12 M9011



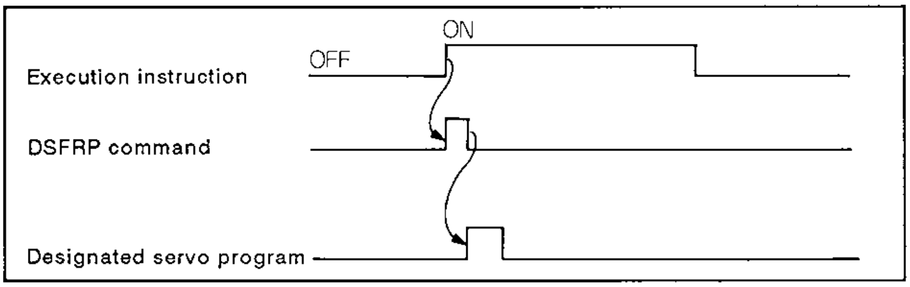
Setting data

(D)	The number of the starting axis
n	The number of the servo program to be executed

(1) The following processing is executed at the leading edge (OFF →ON) of the DSFRP command execution instruction.

(a) The start receive flag (M200n) designated by (D) is set. See Section 3.2.2.

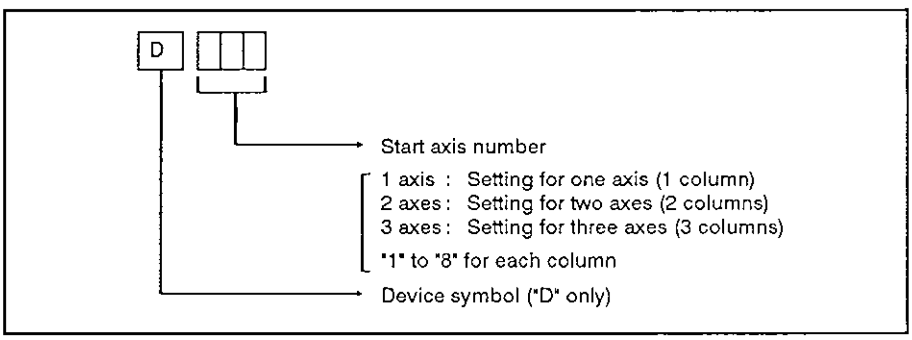
(b) The start request for the servo program designated by "n" is given.



[Data setting]

(1) Start axis

Start axis to be set for (D):



Example:

Designation of start axes:

(a) Axis 4 : D4

(b) Axes 4 and 5 : D45

(c) Axes 4, 5, and 6: D456

POINT

Set either of the axes to be started simultaneously in the servo program for simultaneous start of multiple axes.

(a) If linear interpolation between axis 2 and axis 3, and circular interpolation between axis 4 and axis 5 are started simultaneously, set axis 2 or 3 and axis 4 or 5 in the sequence program.

(2) Servo program number

A servo program number can be set directly or indirectly.

(a) Direct setting

A servo program number (0 to 4095) is set directly with a number.

Example:

To set servo program number 50:

(a) When K is used: K50

(b) Indirect setting

A servo program number is set using the setting in the data register.

The data registers which can be used are registers D0 to D799. The setting is indicated below.

1) K [3][0] [ ][ ][ ]

Data register number (000 to 799):
To be set in three digits
Ex. 50 050

Data register designation

2) It is possible to use a hexadecimal number (H7530 to H784F) to designate a data register number.

Example:

To designate the number of the start servo program with the data in the data register (D50):

(a) When K is used:

K30 050

D
50 → D50

[Error contents]

(1) In the following cases, an operation error will occur and the DSFRP command will not be executed.

(a) 4 digits are used to set (D).

(b) The setting for a digit is outside the range (1 to 8).

(c) The identical axis number is set for (D).

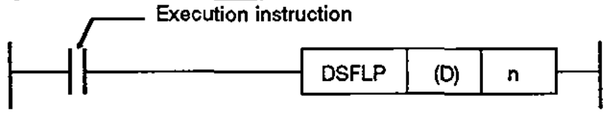
(d) Value "n" is not between 0 and 4095 or 30000 and 30799.

(e) (D) and "n" are set indirectly by the index register (Z, V).

5.3 Present Position Data and Velocity Change Command (DSFLP)

	Devices																					*4	*5	*6	*7	*8	*9	
	Bit Device							Word (16 Bit) Device								*1		*2		*3								
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K	H	P	I	N					*10	*11	*12
(D)										o													7		x		o	o
N																	o	o										

*1 Constant
*2 Pointer
*3 Level
*4 Column Designation
*5 Number of Steps
*6 Sub-set
*7 Index
*8 Carry Flag
*9 Error Flag
*10 M9012
*11 M9010
*12 M9011



(D)	Axis number for which the present position data or velocity is changed.
n	Required present position data or velocity

(1) The following processing is executed at the leading edge of the execution instruction of the DSFLP command.

(a) Changing present position data

The present position data is changed with the following procedure when the DSFLP command is executed.

1) The start receive flag corresponding to the axis designated by (D) is set.

2) The present position data is changed with the data in the present position data change register which corresponds to the axis designated with (D).

3) The start receive flag is reset after the present position data has been changed.

(b) Changing the velocity

The velocity is changed with the following procedure when the DSFLP command is executed.

1) The velocity change flag, corresponding to the axis designated with (D) is set.

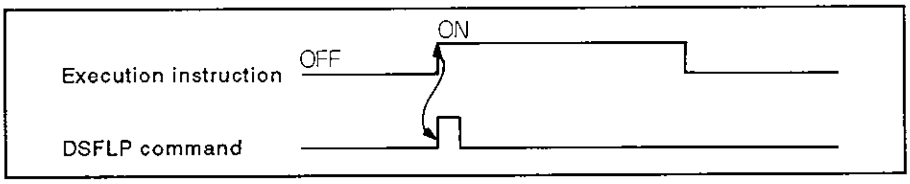
2) The instruction to change the present velocity to the value in the velocity change register which corresponds to the axis designated with (D) is output.

3) The velocity change flag is reset.

(2) The present position data change register numbers and the velocity change register numbers are indicated below. For details, see Section 3.4.2.

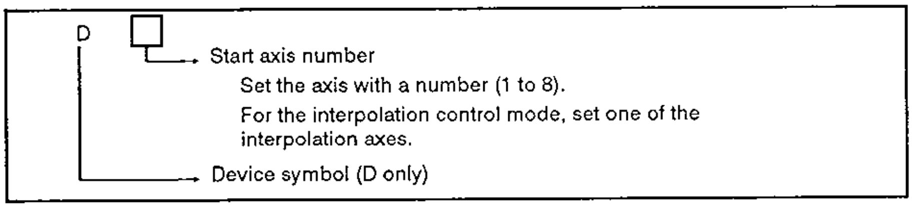
Axis No.	Present Position Data Change Register		Velocity Change Register	
	Upper Digit	Lower Digit	Upper Digit	Lower Digit
Axis 1	D961	D960	D963	D962
Axis 2	D967	D966	D969	D968
Axis 3	D973	D972	D975	D974
Axis 4	D979	D978	D981	D980
Axis 5	D985	D984	D987	D986
Axis 6	D991	D990	D993	D992
Axis 7	D997	D996	D999	D998
Axis 8	D1003	D1002	D1005	D1004

[Operation timing]



[Data setting]

(1) Setting the axis to change present position data or velocity



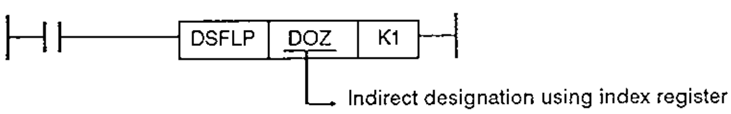
Example:

(1) Designation of start axes:
- (a) Axis 4 : D4
- (b) Interpolation between axis 4 and axis 5 : D4 or D5

(2) Changing present position data or velocity
- (a) Changing present position data : K0 or H0
- (b) Changing velocity : K1 or H1

POINT

With the DSFLP command, indirect designation of (D) and "n" with the index register (Z, V) is not allowed.



Indirect designation of this type causes an operation error and the DSFLP command will not be executed.

[Error contents]

(1) An operation error will occur in the following cases, causing the DSFLP command to not be executed.

(a) Setting for (D) is not 1 to 8.

(b) Setting for "n" is not 1 or 0.

(c) (D) and "n" are set indirectly with the index register (Z, V).

(2) A minor error (error in changing control mode) occurs in the following cases and present position data and velocity are not changed.

In these cases, the error detection flag (Xn7) is set and the error code is stored in the minor error code area of the corresponding axis.

(a) The axis designated with (D) has already started when the present position data is executed.

(b) The axis designated with (D) has already started zero return or circular interpolation when the velocity change is executed.

(c) The axis designated with (D) is decelerating when velocity change is executed.

(d) The velocity designated with "n" is not "1" to the "velocity limit" when the velocity change is executed.

[Program example]

(1) The program for changing the present position data of axis 2 with the position data set using the 8-bit digital switch is indicated below.

(a) Conditions

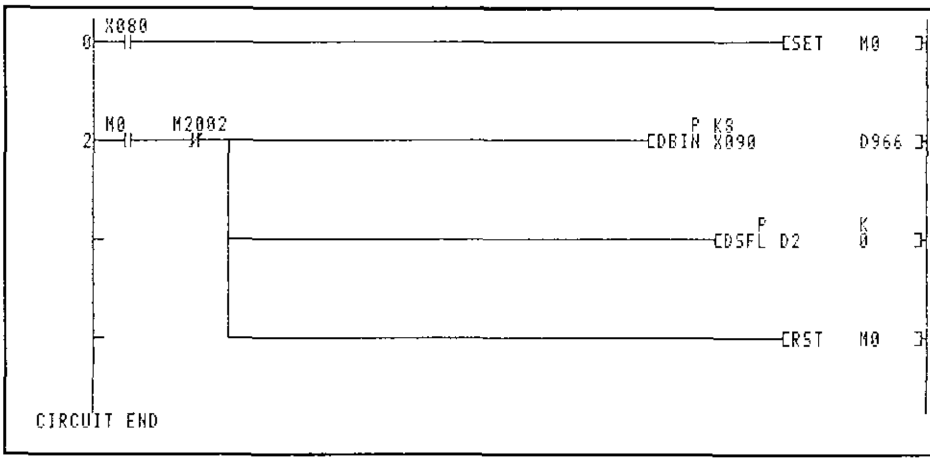
1) Input number (digital switch):
X90-XAF

2) Present position data change instruction:
Leading edge of X80 (OFF→ON)

3) Present position data change:
M0 execution flag

4) Axis 2 start receive (Axis 2 stop/start confirmation) flag:
M2002 (Axis 2 start receive flag)

(b) Program example



M0 is set when X80 is turned on.

The value set with digital switch is converted into binary data and stored in the present position data change register (D966, D967).

Present position data change execution request for Axis 2.

M0 is reset.

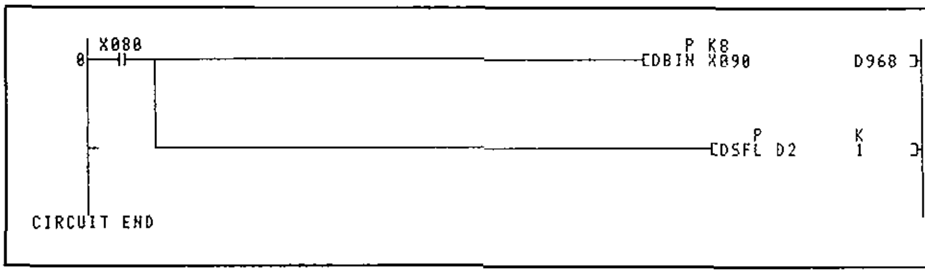
(2) The program to change the positioning velocity of axis 2 with the value set with the 8-bit digital switch is indicated below.

(a) Conditions

1) Input number (digital switch):
X90-XAF

2) Velocity change instruction:
Leading edge of X80 (OFF→ON)

(b) Program example



The following processing is executed when X80 is turned on.

The value set with the digital switch is converted into binary data and stored in the velocity change register (D968, D969).

Velocity change execution request for Axis 2.

MEMO

6. POSITIONING CONTROL SERVO PROGRAM

The servo program is used by the A73CPU to designate the positioning control mode and positioning data necessary for executing positioning control by the A73CPU. The servo program structure and designation method are explained in this section.

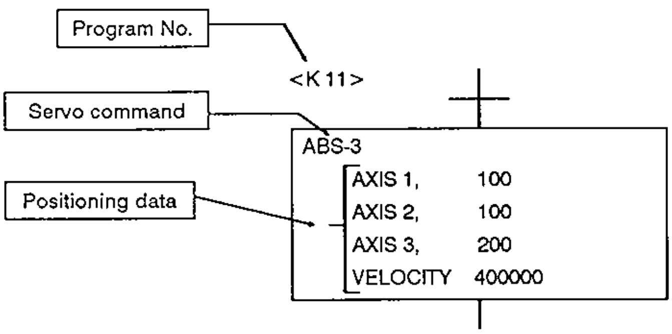
For details on servo programs, see Section 7.

6.1 Servo Program Structure and Storing Area

6.1.1 Servo program structure

A servo program consists of a program number, servo commands, and positioning data.

It is possible to set the positioning data necessary to execute the designated servo command. Use a peripheral device to designate the program number and the servo program in question.



[Description of program]

K11 : Program No. 11

ABS-3 : Linear interpolation of 3 axes in the absolute mode

AXIS 1, 100
AXIS 2, 100
AXIS 3, 200
— Positioning Address

Axis	Positioning Address
1	100
2	100
3	200

VELOCITY : Command velocity (400000)

Fig. 6.1 Servo Program Structure Example

(1) Program number:

Used to designate sequence programs. Selectable between 0 and 4095.

(2) Servo command:

Indicates the positioning control mode.

For details, see Section 6.2.

(3) Positioning data:

The data necessary to execute the servo command.

For each servo instruction, necessary data is determined.

For details, see Section 6.3.

6.1.2 Servo program storage area

(1) Servo program storage area

Servo programs are stored in the A73CPU internal memory.

The servo program storage area is the E^2ROM.

(2) Servo program storage area capacity

The servo program storage area has a capacity of 13K steps (13312 steps).

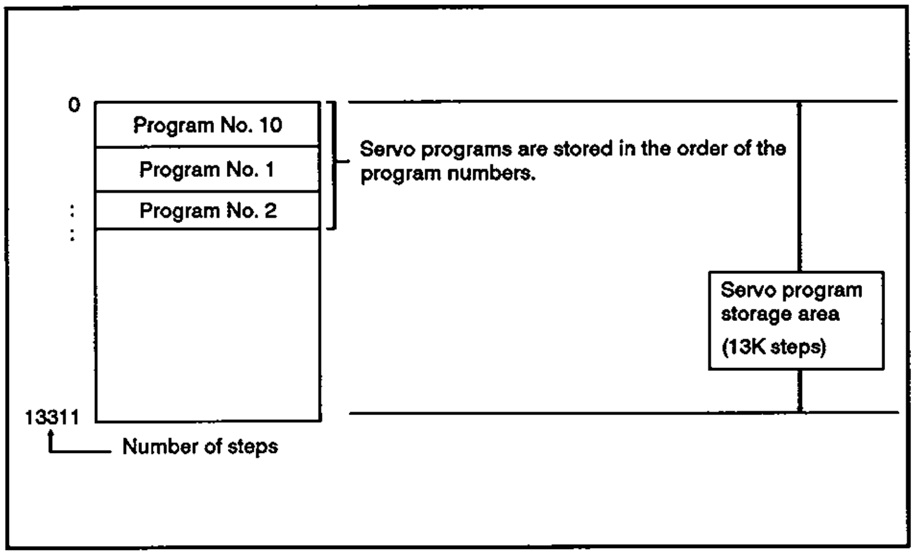


Fig. 6.2 Servo Program Storage Area

POINT

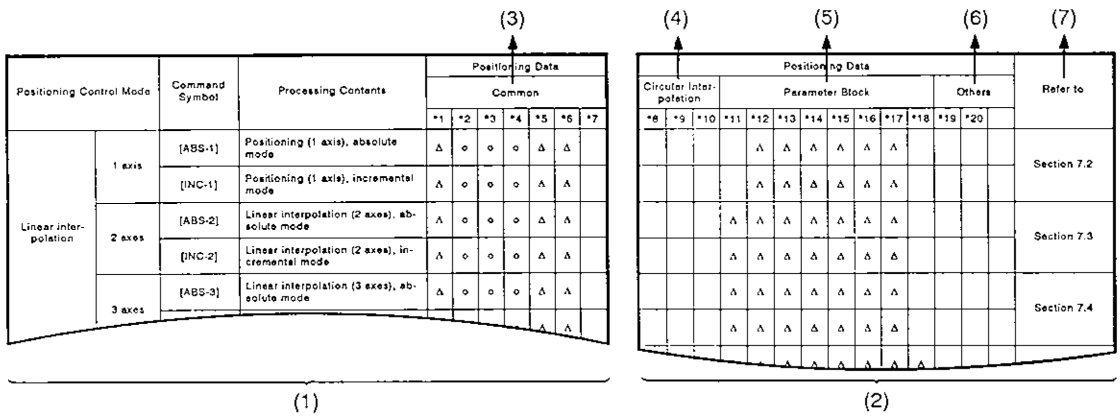
(1) If the servo program storage area is not sufficient for storing the necessary programs, change the positioning data setting method, used by the servo program, from direct to indirect. More than one positioning operation can now be executed by a single program. For details on the indirect setting, see Section 6.4.2.

(2) Servo programs in the servo program storage area can be rewritten up to 10,000 times.

6.2 Servo Commands

(1) How to read the servo command list

Table 6.1 How to Read the Servo Command List



No.	Contents	
(1)	Command symbol	Indicates the servo commands that can be used in the servo program.
	Processing contents	Indicates the processing outline of servo command.
(2)	(a) Indicates the positioning data that can be set with a servo command. 1) o : Items that must be set 2) Δ : Items to be set as required	
	(b) Direct or indirect setting is possible (not for axis number). 1) Direct setting : Set with a number. 2) Indirect setting : Set with a word device (D, W). Servo program execution is controlled by the contents of the set word device. 1-word data or 2-word data is used depending on set item. For 2-word data, set the head device number.	
	(c) Number of steps The number of command steps increases as the number of items to be set increases. The used number of steps is displayed during programming. (One Δ indicated item occupies one step.)	
(3)	Items common to servo commands.	
(4)	Items to be set in the circular interpolation start servo program.	
(5)	Set when positioning is controlled by changing the parameter block data set with the servo program or default values. The data in the parameter block is not changed.	
(6)	Setting items other than common, circular interpolation, and parameter block. Items to be set varies depending on the servo command.	
(7)	The section to be referenced.	

(2) Servo command list

The servo commands that can be used in the servo program and the positioning data to be set with the servo command are summarized in Table 6.2.

For the positioning data to be set with the servo command, see Section 6.3.

Table 6.2 Servo Command List

Positioning Control Mode		Command Symbol	Processing Contents	Positioning Data: Common *1	*2	*3	*4	*5	*6	*7
Linear interpolation	1 axis	[ABS-1]	Positioning (1 axis), absolute mode	Δ	o	o	o	Δ	Δ	
		[INC-1]	Positioning (1 axis), incremental mode	Δ	o	o	o	Δ	Δ	
	2 axes	[ABS-2]	Linear interpolation (2 axes), absolute mode	Δ	o	o	o	Δ	Δ	
		[INC-2]	Linear interpolation (2 axes), incremental mode	Δ	o	o	o	Δ	Δ	
	3 axes	[ABS-3]	Linear interpolation (3 axes), absolute mode	Δ	o	o	o	Δ	Δ	
		[INC-3]	Linear interpolation (3 axes), incremental mode	Δ	o	o	o	Δ	Δ	
Circular interpolation	Assist-point designation	[ABS ⌒]	Circular interpolation, absolute mode	Δ	o	o	o	Δ	Δ	
		[INC ⌒]	Circular interpolation, incremental mode	Δ	o	o	o	Δ	Δ	
	Radius designation	[ABS ↷]	Circular interpolation, absolute mode, CW, 180° or less	Δ	o	o	o	Δ	Δ	
		[ABS ↻]	Circular interpolation by radius designation, absolute mode, CW, 180° or greater	Δ	o	o	o	Δ	Δ	
		[ABS ↶]	Circular interpolation, absolute mode, CCW, 180° or less	Δ	o	o	o	Δ	Δ	
		[ABS ↺]	Circular interpolation, absolute mode, CCW, 180° or greater	Δ	o	o	o	Δ	Δ	
		[INC ↷]	Circular interpolation, incremental mode, CW, 180° or less	Δ	o	o	o	Δ	Δ	
		[INC ↻]	Circular interpolation, incremental mode, CW, 180° or greater	Δ	o	o	o	Δ	Δ	
		[INC ↶]	Circular interpolation, incremental mode, CCW, 180° or less	Δ	o	o	o	Δ	Δ	
		[INC ↺]	Circular interpolation, incremental mode, CCW, 180° or greater	Δ	o	o	o	Δ	Δ	

o : Items to be set always
Δ : Items to be set as needed

Positioning Data														Refer to
Circular Interpolation			Parameter Block								Others			
*8	*9	*10	*11	*12	*13	*14	*15	*16	*17	*18	*19	*20		
				Δ	Δ	Δ	Δ	Δ	Δ					Section 7.2
				Δ	Δ	Δ	Δ	Δ	Δ					
			Δ	Δ	Δ	Δ	Δ	Δ	Δ					Section 7.3
			Δ	Δ	Δ	Δ	Δ	Δ	Δ					
			Δ	Δ	Δ	Δ	Δ	Δ	Δ					Section 7.4
			Δ	Δ	Δ	Δ	Δ	Δ	Δ					
o			Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ				Section 7.5
o			Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ				
	o		Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ				Section 7.6
	o		Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ				
	o		Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ				
	o		Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ				
	o		Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ				
	o		Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ				
	o		Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ				
	o		Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ				

*1 : Parameter block number
*2 : Axis
*3 : Address/travel distance
*4 : Command velocity
*5 : Dwell time
*6 : M code
*7 : Torque limit
*8 : Assist-point
*9 : Radius
*10 : Center
*11 : Control units
*12 : Velocity limit
*13 : Acceleration time
*14 : Deceleration time
*15 : Immediate stop deceleration time
*16 : Torque limit
*17 : Deceleration processing when the STOP signal is input
*18 : Allowable circular interpolation error limits
*19 : Repeat conditions
*20 : Program No.

IB (NA) 66233-A

Table 6.2 Servo Command List (Continued)

Positioning Control Mode		Command Symbol	Processing Contents	Positioning Data						
				Common						
				*1	*2	*3	*4	*5	*6	*7
Circular inter-polation	Center point designa-tion	[ABS ↻]	Circular interpolation, absolute mode, CW	Δ	o	o	o	Δ	Δ	
		[ABS ↺]	Circular interpolation, absolute mode, CCW	Δ	o	o	o	Δ	Δ	
		[INC ↻]	Circular interpolation, incremental mode, CW	Δ	o	o	o	Δ	Δ	
		[INC ↺]	Circular interpolation, incremental mode, CCW	Δ	o	o	o	Δ	Δ	
Incremental feed	1 axis	[FEED-1]	Start of incremental feed (1 axis)	Δ	o	o	o	Δ	Δ	
	2 axes	[FEED-2]	Start of incremental feed (2 axes, linear interpolation)	Δ	o	o	o	Δ	Δ	
	3 axes	[FEED-3]	Start of incremental feed (3 axes, linear interpolation)	Δ	o	o	o	Δ	Δ	
Velocity control	Forward rotation	[VF]	Start of forward rotation (velocity control mode)	Δ	o		o		Δ	
	Reverse rotation	[VR]	Start of reverse rotation (velocity control mode)	Δ	o		o		Δ	
V/P control	Forward rotation	[VPF]	Start of forward rotation (V/P control mode)	Δ	o	o	o	Δ	Δ	Δ
	Reverse rotation	[VPR]	Start of reverse rotation (V/P control mode)	Δ	o	o	o	Δ	Δ	Δ
	Re-start	[VPSTART]	Re-start of V/P control mode		o					
Velocity change control		[VSTART]	Start of velocity change control	Δ						
		[VEND]	End of velocity change control							
		[VABS]	Designation of velocity change point in absolute value			o	o		Δ	Δ
		[VINC]	Designation of velocity change point in incremental value			o	o		Δ	Δ

o : Items to be set always
Δ : Items to be set as needed

IB (NA) 66233-A

Positioning Data														Refer to
Circular Inter-polation			Parameter Block								Others			
*8	*9	*10	*11	*12	*13	*14	*15	*16	*17	*18	*19	*20		
		○	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ				Section 7.7
		○	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ				
		○	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ				
		○	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ				
				Δ	Δ	Δ	Δ	Δ	Δ					Section 7.8
			Δ	Δ	Δ	Δ	Δ	Δ	Δ					Section 7.9
			Δ	Δ	Δ	Δ	Δ	Δ	Δ					Section 7.10
			Δ	Δ	Δ	Δ	Δ	Δ	Δ					Section 7.11
			Δ	Δ	Δ	Δ	Δ	Δ	Δ					
			Δ	Δ	Δ	Δ	Δ	Δ	Δ					Section 7.12.1
			Δ	Δ	Δ	Δ	Δ	Δ	Δ					
														Section 7.12.2
			Δ	Δ	Δ	Δ	Δ	Δ	Δ					Section 7.13.1

*1 : Parameter block number
*2 : Axis
*3 : Address/travel distance
*4 : Command velocity
*5 : Dwell time
*6 : M code
*7 : Torque limit
*8 : Assist-point
*9 : Radius
*10 : Center
*11 : Control units
*12 : Velocity limit
*13 : Acceleration time
*14 : Deceleration time
*15 : Immediate stop deceleration time
*16 : Torque limit
*17 : Deceleration processing when the STOP signal is input
*18 : Allowable circular interpolation error limits
*19 : Repeat conditions
*20 : Program No.

Table 6.2 Servo Command List (Continued)

Positioning Control Mode	Command Symbol	Processing Contents	Positioning Data						
			Common						
			*1	*2	*3	*4	*5	*6	*7
Constant velocity control	[CPSTART2]	Start of 2-axis constant velocity control	Δ	o		o			
	[CPSTART3]	Start of 3-axis constant velocity control	Δ	o		o			
	[CPEND]	End of constant velocity control					Δ		
Repetition of the same control (used for velocity change control and constant velocity control)	[FOR-TIMES]	Setting of the head address for the repeat range							
	[FOR-ON]								
	[FOR-OFF]								
	[NEXT]	Setting of the end address for the repeat range							
Simultaneous start	[START]	Simultaneous start of servo programs							
Zero return	[ZERO]	Start of zero return		o					

o : Items to be set always
Δ : Items to be set as needed

Positioning Data														Refer to
Circular Interpolation			Parameter Block								Others			
*8	*9	*10	*11	*12	*13	*14	*15	*16	*17	*18	*19	*20		
			Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ				Section 7.14.1
			Δ	Δ	Δ	Δ	Δ	Δ	Δ					
											o			Section 7.13.2 Section 7.14.2
												o		Section 7.15
														Section 7.18

*1 : Parameter block number
*2 : Axis
*3 : Address/travel distance
*4 : Command velocity
*5 : Dwell time
*6 : M code
*7 : Torque limit
*8 : Assist-point
*9 : Radius
*10 : Center
*11 : Control units
*12 : Velocity limit
*13 : Acceleration time
*14 : Deceleration time
*15 : Immediate stop deceleration time
*16 : Torque limit
*17 : Deceleration processing when the STOP signal is input
*18 : Allowable circular interpolation error limits
*19 : Repeat conditions
*20 : Program No.

6.3 Positioning Data

The positioning data to be set by the servo program is summarized in Table 6.3.

Table 6.3 Positioning Data

Name			Contents		Default
Common	Parameter block number		Set the parameter blocks to be used for acceleration and deceleration processing, STOP signal input processing, etc. for each positioning operation.		1
Common	Axis		Set the number of the axis to start. For interpolation control, set the axis number of the axes for interpolation.		—
Common	Address/ travel distance	Absolute mode	Address	When positioning is controlled in the absolute mode, set the positioning address in the absolute address.	—
Common	Address/ travel distance	Incremental mode	Axis travel distance	When positioning is controlled in the incremental mode, set the positioning address in the axis travel distance. The direction of axis travel is indicated by a plus or minus sign. In the V/P switchover control mode, only the plus sign is allowed. Plus : Positive direction (addresses increasing direction) Minus : Negative direction (addresses decreasing direction)	—
Common	Designated velocity		Set the positioning velocity. Velocity is set in the unit set as the control unit in the parameter block. At the start of interpolation, the velocity is the vector sum of the designated velocity for each axis.		—
Common	Dwell time		Set the delay time between the completion of positioning at the positioning address and the output of the positioning complete signal (xo[ ]1).		0 (ms)
Common	M code		Set an M code. In the velocity change control mode or constant velocity control mode, setting is possible for each point. At the start and at each designated point, the M code is updated.		0
Common	Torque limit		Set the torque limit value. At the start of positioning, torque is limited according to the data set in the parameter block. In the velocity change control mode or the constant velocity control mode, setting is possible at each point; torque is limited to the set value at the designated point.		Torque limit setting in the parameter block (%)
Arc	Assist-point	Absolute mode	Set circular interpolation by assist-point designation.		—
Arc	Assist-point	Incremental mode			
Arc	Radius	Absolute mode	Set circular interpolation by radius designation. The setting range is indicated in the right column.		—
Arc	Radius	Incremental mode			
Arc	Center point	Absolute mode	Set circular interpolation by center point designation.		—
Arc	Center point	Incremental mode			

Setting Range				(1)		(4)		
mm	inch	degree	PULSE	(2)	(3)	(5)	(6)	(7)
1 to 16				o	1	1	o	
1 to 8				x	—	—		o
-2^{31} to $2^{31}-1$ ($\times 10^{-1}$ μm)	-2^{31} to $2^{31}-1$ ($\times 10^{-5}$ inch)	0 to 35999999 ($\times 10^{-5}$ degree)	-2^{31} to $2^{31}-1$ (PLS)	o	2	*1 n03		
Other than V/P control								
0 to $\pm(2^{31}-1)$	0 to $\pm(2^{31}-1)$	0 to $\pm(2^{31}-1)$	0 to $\pm(2^{31}-1)$					
V/P control						—		
0 to $2^{31}-1$ ($\times 10^{-1}$ μm)	0 to $2^{31}-1$ ($\times 10^{-5}$ inch)	0 to $2^{31}-1$ ($\times 10^{-5}$ degree)	0 to $2^{31}-1$ (PLS)					
0.01 to 6000000.00 (mm/min)	0.001 to 600000.000 (inch/min)	0.001 to 600000.000 (degree/min)	1 to 1000000 (PLS/sec)	o	2	4	*2 o	*3 o
0 to 5000 (ms)				o	1	5	o	
0 to 255				o	1	6	o	
1 to 500(%)				o	1	7	o	
-2^{31} to $2^{31}-1$ ($\times 10^{-1}$ μm)	-2^{31} to $2^{31}-1$ ($\times 10^{-5}$ inch)	0 to 35999999 ($\times 10^{-5}$ degree)	-2^{31} to $2^{31}-1$ (PLS)	o	2 x 2	*1 n08		o
0 to $\pm(2^{31}-1)$	0 to $\pm(2^{31}-1)$	0 to $\pm(2^{31}-1)$	0 to $\pm(2^{31}-1)$					
1 to $2^{32}-1$ ($\times 10^{-1}$ μm)	1 to $2^{32}-1$ ($\times 10^{-5}$ inch)	0 to 35999999 ($\times 10^{-5}$ degree) / 1 to 231 -1 ($\times 10^{-5}$ degree)	1 to $2^{32}-1$ (PLS)	o	2	9		
-2^{31} to $2^{31}-1$	-2^{31} to $2^{31}-1$ ($\times 10^{-5}$ inch)	0 to 35999999 (x10-5 degree)	-2^{31} to $2^{31}-1$ (PLS)	o	2 x 2	*1 n10		
0 to $\pm(2^{31}-1)$	0 to $\pm(2^{31}-1)$	0 to $\pm(2^{31}-1)$	0 to $\pm(2^{31}-1)$					

(1) : Indirect Setting
(2) : Yes/No
(3) : Number of Words Used
(4) : Processing at the Occurrence of a Setting Error
(5) : Error Item Data (stored at D9190)
(6) : Control by Default
(7) : Start Disabled

Table 6.3 Positioning Data (Continued)

Name		Contents	Default
Parameter block	Control units	It is only possible to designate the items to be changed in the data set in the designated parameter block. For details on the data, see Section 4.3.	3
	Velocity limit		200.000 (PLS/sec)
	Acceleration time		1000 (ms)
	Deceleration time		1000 (ms)
	immediate stop deceleration time		1000 (ms)
	Torque limit		300 (%)
	Deceleration processing when the STOP signal is input		0
	Allowable circular interpolation		100 (PLS)
Others	Repeat conditions	Set the repeat conditions for FOR-TIMES command and NEXT command.	—
	Program No.	Set the program number of programs to be started simultaneously.	—

REMARKS

(1) *1: Letter "n" in "n03", "n08", and "n10" indicates the axis number (1 to 8).

(2) *2: *The error caused by a designated velocity larger than the set limit; clamped* by the limit velocity.

(3) *3: Designated velocity is "0".

(4) *4: If more than one error occurs in the same program, the error item data of the latest error is stored.

Setting Range				(1)		(4)		
mm	inch	degree	PULSE	(2)	(3)	(5)	(6)	(7)
0	1	2	3	o	1	11		
0.01 to 6000000.00 (mm/min)	0.001 to 600000.000 (inch/min)	0.001 to 600000.000 (degree/min)	1 to 1000000 (PLS/sec)	o	2	12		
1 to 65535 (ms)				o	1	13		
1 to 65535 (ms)				o	1	14	o	
1 to 65535 (ms)				o	1	15		
1 to 500(%)				o	1	16		
0: Decelerates and stops according to the set deceleration time. 1: Decelerates and stops according to the set immediate deceleration time.				o	1	—		
0 to 100000	0 to 100000	0 to 100000	0 to 100000	o	2	17		
					—	18		
0 to 4095					—	19		o

(1) : Indirect Setting
(2) : Yes/No
(3) : Number of Words Used
(4) : Processing at the Occurrence of a Setting Error
(5) : Error Item Data (stored at D9190)
(6) : Control by Default
(7) : Start Disabled

6.4 Positioning Data Setting Method

The method to set the positioning data used by the servo program is described below.

Positioning data can be set in either of the following two methods.

(a) Data setting by designating a number:
See Section 6.4.1.

(b) Indirect data setting using a word device:
See Section 6.4.2.

Both of these methods can be used in combination in one servo program.

6.4.1 Setting by designating a number

Positioning data is set directly by a number; the set data becomes the fixed data.

Setting and correcting data is only possible with a peripheral device.

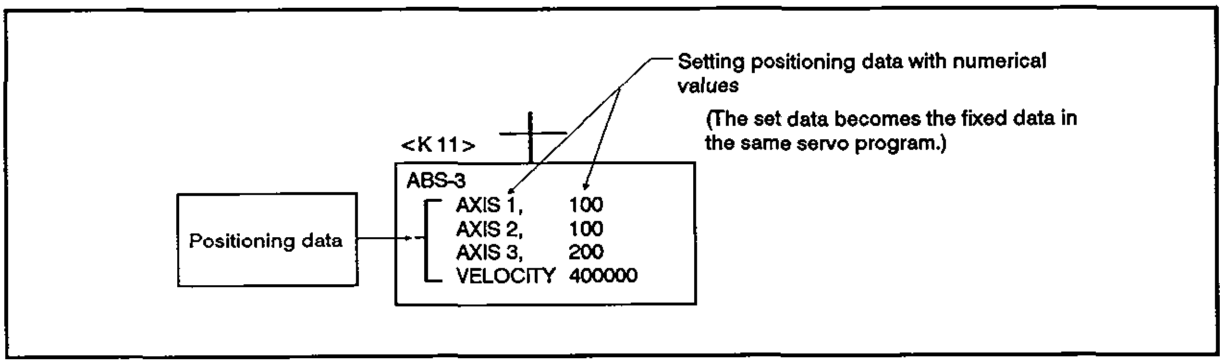


Fig. 6.3 Example of Positioning Data Setting by Numerical Designation

6.4.2 Indirect setting with word device (D, W)

The word device (D, W) number is set for the positioning data designated by the servo program.

By changing the data assigned to the word device with a sequence program, more than one positioning control is possible with one servo program.

(1) Indirect data setting device

The devices which can be used for indirect setting are the data registers (D) and link registers (W).

(a) Data registers:

D0 to D799 (800 points) can be designated.

Because data registers from D800 to D1023 are used by the PCPU OS, they cannot be used as the indirect setting device.

(b) Link registers:

W0 to W3FF (1024 points) can be designated.

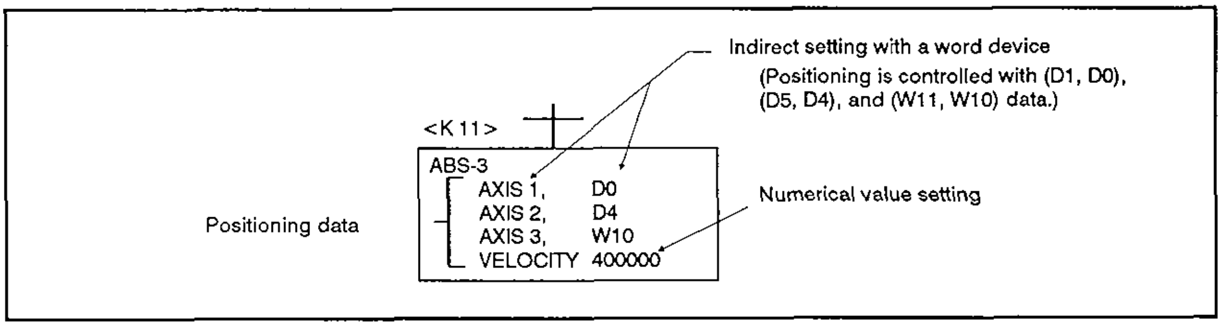


Fig. 6.4 Indirect Setting of Positioning Data Using Word Devices

(2) Reading positioning data

With indirect setting with word devices, the designated word device data is read when the PCPU executes the servo program.

Therefore, to carry out positioning control, the servo program start request may only be given after setting the data for the devices used for indirect setting.

POINT

(1) Indirect setting of an axis number using a word device is not possible in the servo program.

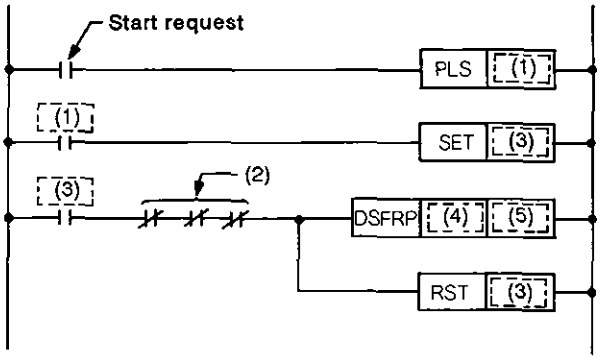
(2) Interlock must be taken so that the designated device data for indirect setting will not be changed until the designated axis receives the start signal; use the start receive signal (M2001 to M2008) to take interlocks.

If the data is changed before the start signal is received, positioning control will not be executed correctly.

6.5 Programming Sequence Program for Starting the Servo Program

6.5.1 To execute servo program only once

The basic programming concept of a sequence program to execute the servo program only once by outputting the start request signal is described below.



If the designated axis is not moving when the start request is turned on, the execution request for the DSFRP command is output.

The servo program start request is given with the DSFRP command.

After the completion of DSFRP command, the device designated in step (1) is reset.

(1) to (5):

(1) : Start request leading edge detection flag (usable device: Y, M, L, S, B, F)

(2) : Start receive flag (to be set by the number of axes designated by the servo program)

(3) : DSFRP command execution request flag (usable device: Y, M, L, S, B, F)

(4) : Axis number designated by the servo program

(5) : Servo program number to be started (K0 to K4095)

Example:

(1) The sequence program to give the start request for servo program number 10, which calls circular interpolation with axis 3 and axis 4, when the X80 is turned on.

(a) Conditions

The flags to be used by the sequence program are indicated below.

Flag Name		Device No.
Start request registration flag		M10
DSFRP command execution request flag		M11
Start receive flag	Axis 3	M2003
	Axis 4	M2004

(b) Sequence program example

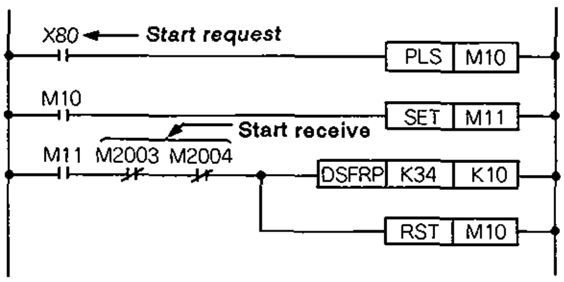
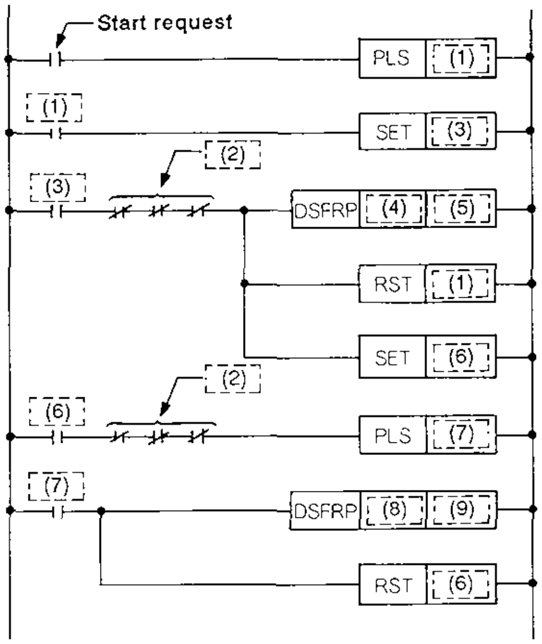


Fig. 6.5 Servo Program Example

6.5.2 To execute different programs continuously

The basic concept of programming the sequence program that can execute two different servo programs continuously is described below.



If the designated axis has not started when the start request is turned on, the request to execute the DSFRP command is output.

The servo program start request with the DSFRP command is given.

After the completion of the DSFRP command, the device designated in (1) is reset.

The execution request flag (6) for the next DSFRP command is set.

After all axes have stopped, the execution request for the next DSFRP command is given.

The servo program start request with the DSFRP command is given.

After the completion of the DSFRP command, the device designated in (6) is reset.

(1) to (9):

(1) : Start request leading edge detection flag (usable device: Y, M, L, S, B, F)

(2) : Start receive flag (to be set by the number of axes designated by the servo program)

(3), (7) : DSFRP command execution request flag (usable device: Y, M, L, S, B, F)

(4), (8) : Axis number designated by the servo program

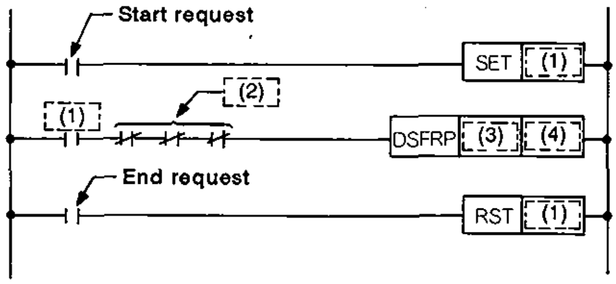
(5), (9) : Servo program number to be started (K0 to K4095)

(6) : Start request registration flag (Y, M, L, S, B, F)

Fig. 6.6 Servp Program Start Sequence Program

6.5.3 To execute the same servo program repeatedly

The basic concept of programming a sequence program that executes positioning repeatedly called by the same servo program is described below.



If the designated axis has not started when the start request is turned on, the request to execute the DSFRP command is output.

The servo program start request with the DSFRP command is given.

The device designated in (1) is reset when the end request is turned on.

(1) to (4):

(1) : Start request leading edge detection flag (usable device: Y, M, L, S, B, F)

(2) : Start receive flag (to be set by the number of axes designated by the servo program)

(3) : Axis number designated by the servo program

(4) : Servo program number to be started (K0 to K4095)

Fig. 6.7 Servo Program Start Sequence Program

7. POSITIONING CONTROL

7.1 Positioning Control Basics

7.1.1 Positioning velocity

The positioning velocity is set by the servo program.

For details on the servo program, refer to Section 4.

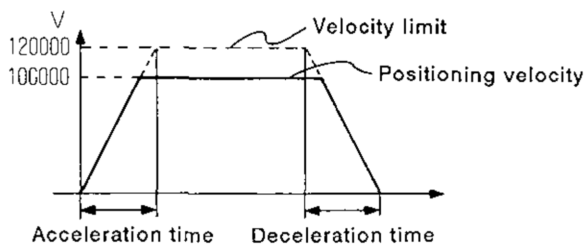
The actual velocity used for positioning control is determined according to the positioning velocity designated by the servo program and the setting for the parameter (velocity limit).

(a) If the positioning velocity designated by the servo program is smaller than the velocity limit, positioning is executed at the designated velocity.

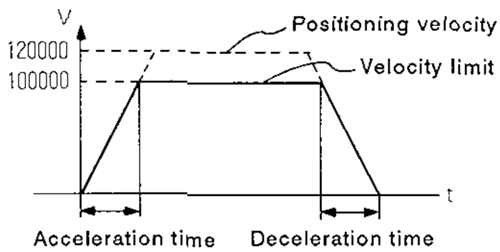
(b) If the positioning velocity designated by the servo program is larger than the velocity limit, positioning is executed at the set velocity limit value.

Example:

(1) When "100000 mm/min" is designated by the servo program and the set velocity limit is "120000 mm/min", positioning is executed as indicated below.



(2) When "120000 mm/min" is designated by the servo program and the set velocity limit is "100000 mm/min", positioning is executed as indicated below.



7.1.2 Positioning velocity in the interpolation control mode

The A73CPU sets the positioning velocity as the axis movement velocity of the axis in question.

(1) Positioning of one axis

For positioning control with one axis, the designated positioning velocity for the designated axis is used.

(2) Linear interpolation control

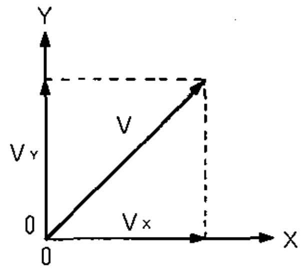
In the linear interpolation mode, the velocity of the control objective is controlled at the designated velocity.

The velocity of each axis is calculated by the A73CPU from the required travel distance of each axis.

Therefore, if the positioning velocity setting is fixed, the control objective is moved at the same velocity even when the travel direction of the control objective changes.

Example:

Linear interpolation of 2 axes

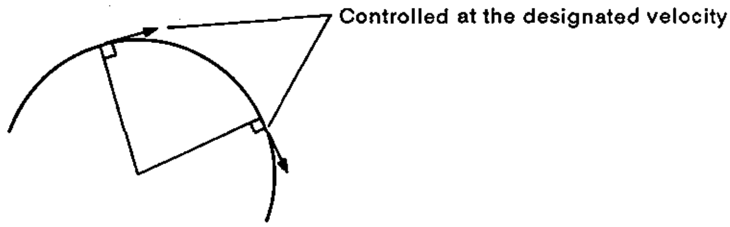


V: Positioning velocity of control objective (to be set by the user)

Vx, Vy: Positioning velocity of each axis (calculated by the A73CPU)

(3) Circular interpolation

In the circular interpolation mode, the velocity of the axes is controlled so that the angular velocity will be the designated velocity.



7.1.3 Control unit for one-axis positioning control

When positioning control is executed for one axis, its control unit is the same as that designated with the fixed parameter.

(In this control mode, the control unit designated by the parameter block is ignored.)

7.1.4 Control unit for interpolation mode

(1) The control units set for the interpolation mode by the parameter block are compared to the control units set by the fixed parameter.

If the control units set by the parameter block and the control units set by the fixed parameter are different, control starts in the following manner.

	Parameter Block Interpolation Control Units				Starting of Interpolation
	mm	inch	degree	PULSE	
Conditions for starting interpolation control	There are axes for which control units, set by the fixed parameter, are either mm or inches.		There are axes for which control units, set by the fixed parameter, are degrees.	There are axes for which control units, set by the fixed parameter, are pulses.	Interpolation control starts in the interpolation control unit set by the parameter block.
Conditions for a unit mismatch error (error code: 40)	The control units set for all axes do not agree with the interpolation control units set by parameter block.				If the control units of the axes for which interpolation control is executed are the same, the interpolation control is executed in the set control units. If the control units of the axes for which interpolation control is carried out are different, interpolation control is executed in order of unit priority. [Priority order: PLS > degree > inches > mm] [Example]: For the axes for which 1000 PLS and 10.000 inches are designated, 10.000 inches are assumed to be 10000 PLS and interpolation control is executed in units of PLS.

(2) Control unit combinations are indicated below.

	mm	inch	degree	PULSE
mm	(1)	(2)	(3)	(3)
inch	(2)	(1)	(3)	(3)
degree	(3)	(3)	(1)	(3)
PULSE	(3)	(3)	(3)	(1)

REMARKS

(1): Same units

(2): Combination of mm and inches

(3): Unit mismatch

(a) Same units (1)

The positioning command value is calculated based on the set address/axis travel distance, positioning velocity, and electronic gear. Positioning control is executed according to the calculated result.

(b) Combination of mm and inches (2)

1) If the interpolation control unit is "mm", the data set in "inches" is converted into "mm" [(inches) x 25.4] and positioning command value is calculated based on the converted set address/axis travel distance, positioning velocity, and electronic gear. Positioning control is executed according to the calculated result.

2) If the interpolation control unit is "inches", the data set in "mm" is converted into "inches" [(mm) ÷ 25.4] and positioning command value is calculated based on the converted set address/axis travel distance, positioning velocity, and electronic gear. Positioning control is executed according to the calculated result.

(c) Unit mismatch (3)

1) If a "unit mismatch" occurs, the axis travel distance and positioning velocity are calculated for each axis.

i) Axis travel distance is converted to a value in units of PLS based on the electronic gear of the axis in question.

ii) Positioning velocity is converted in a value in units of PLS/sec based on the electronic gear of the axis which agrees with the interpolation control units.

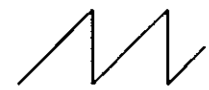
Positioning command value is calculated based on the axis travel distance in PLS, velocity in PLS/sec, and electronic gear and positioning control is executed according to the calculated result.

2) For the linear interpolation control for three axes, if there are two axes that have the same control unit as the interpolation control unit, positioning velocity is calculated based on the electronic gear of the axis assigned with a smaller axis number of these two axes.

7.1.5 When control unit is "degrees"

(1) Present value address

When control unit is "degrees", the present position address is the ring address, ranging from 0° to 360°.



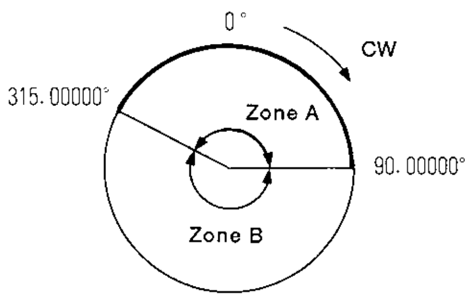
(2) Stroke limit setting

The upper and lower limits of the stroke are in the range of 0° and 359.99999°.

(a) Settings to make stroke limit effective

When the stroke limits are made effective, set the upper and lower stroke limits in the manner as indicated below.

Note that the travel range is set in the CW direction from the lower limit to the upper limit.



1) To set the travel range in zone A:

i) Lower limit ..... 315.00000°

ii) Upper limit ..... 90.00000°

2) To set the travel range in zone B:

i) Lower limit ..... 90.00000°

ii) Upper limit ..... 315.00000°

(b) Settings to make stroke limit ineffective

To make the stroke limit ineffective, set the same value for both the upper and lower stroke limits.

Control is possible regardless of the set stroke limits.

POINT

Circular interpolation including the axis, for which the stroke limit is made effective, is not possible.

(3) Positioning control

(a) In the absolute mode (ABS[ ] command)

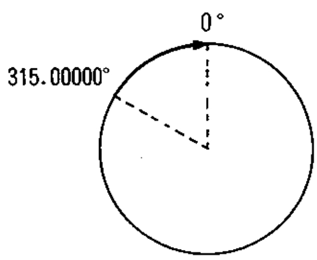
In the absolute mode, positioning is executed from the present position to the designated address selecting a shorter path.

Example:

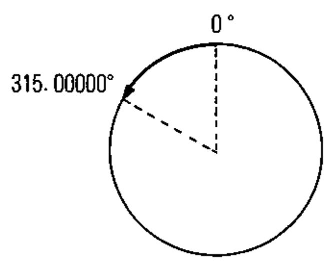
(1) Present position: 315.00000° Designated address: 0°
Positioning is executed in the CW direction.

(2) Present position: 0° Designated address: 315.00000°
Positioning is executed in the CCW direction.

[315.0000° → 0°]



[0° → 315.00000°]

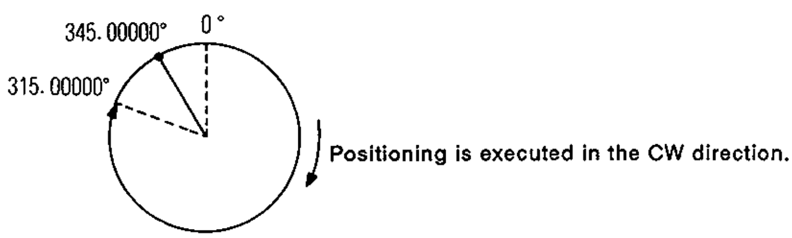


POINT

(1) In the absolute mode, positioning direction is influenced by the set stroke limits. Therefore, positioning is not always executed in the shorter path direction.

Example:

Present position: 0° Designated address: 315.00000°
Lower limit: 0° Upper limit: 345.00000°
Positioning is executed in the CW direction.



(2) Positioning address: 0° to 360°
Positioning which requires more than one rotation must be programmed in the incremental mode.

(b) In the incremental mode (INC[ ] command)

In the incremental mode, positioning direction is determined by the plus or minus sign preceding the axis travel distance.

1) Plus .... CW

2) Minus ..... CCW

POINT

Positioning at a point exceeding 360° is possible.

7.1.6 Stop processing and re-starting after a stop

(1) If a factor to stop positioning occurs during positioning, stop processing is executed.

Stop processings are classified into the following three types:

(a) Processing 1

Deceleration and stop controlled by the "stop deceleration time" set in the parameter block.

(b) Processing 2

Deceleration and stop controlled by the "immediate stop deceleration time" set in the parameter block.

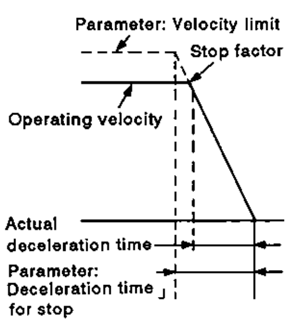
(c) Processing 3

Immediate stop without deceleration.

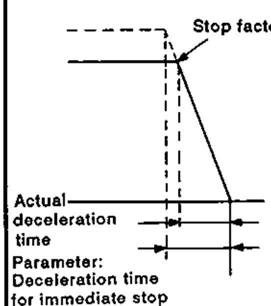
(2) The stop instruction and stop factor are active to either for individual axes or for all axes.

However, stop processing is executed for all interpolation axes if a stop instruction active to a single axis is input during control.

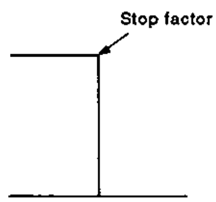
No.	Stop Factor	Axis	Stop Processing	Error	Remark
1	External STOP signal ON	Individual axes	Processing 1 or processing 2 according to the "stop processing when STOP signal is input" setting in the parameter block.	Major error during zero return operation.	(1) Numbers 1 to 4 are used as the stop instruction for the velocity control mode operation. (2) During MPG operation, an axis will stop immediately without decelerating (processing 3). (3) Stop factor which requires stop processing 1 is ignored if it occurs during deceleration. (4) If stop factor requiring stop processing 2 is input after deceleration has started during positioning control or due to the jog start signal being turned off, deceleration and stop control are switched to the control according to processing 2. (5) Processing priority (processing 1) < (processing 2) < (processing 3) (6) Processing 3 * is an emergency stop controlled by hardware.
2	Stop instruction YnD ON		Processing 1	Minor error during zero return operation	
3	Immediate stop instruction Yn1 ON		Processing 2		
4	External input FLS OFF		Processing 1 or processing 2 according to the "stop processing when STOP signal is input" setting in the parameter block.	Major error during zero return operation.	
5	External input RLS OFF				
6	Servo error detection Yn8 ON		Processing 3*		
7	PC ready M2000 OFF	All axes	Processing 1	Minor error during zero return operation	
8	Emergency stop signal from a peripheral device [BREAK] key input		Processing 2		
9	A73CPU STOP		Processing 1		
10	A73CPU reset		Processing 3*	—	
11	PCPU WDT error		Processing 3*	M9073 (WDT error) ON	
12	SCPU WDT error		Processing 1	—	
13	A73CPU power OFF		Processing 3*	—	
14	Servo amplifier power OFF	Individual axes	Processing 3*	Major error during positioning (servo not installed)	



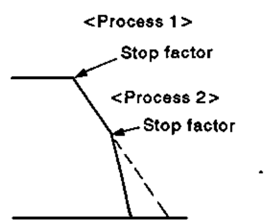
<Processing 1>



<Processing 2>



<Processing 3>



After an occurrence of stop factor for <process 2> during <process 1>, stop processing is executed according to <process 2>

7.2 One-axis Straight Positioning Control

In this control mode, positioning of the designated axis is executed from the present position to the designated position.

Positioning control is executed either in the absolute mode or incremental mode called by the servo command (ABS-1 or INC-1).

Servo Command			[ABS-1]	[INC-1]
Positioning Control Mode			Absolute	Incremental
Number of axes to be controlled			1	
Processing Contents	Common	Parameter block No.	Δ	
		Axis	O	
		Address/travel distance	O	
		Command velocity	O	
		Dwell time	Δ	
		M code	Δ	
		Torque limit		
	Circular interpolation	Assist-point		
		Radius		
		Center		
	Parameter block	Control units	Δ	
		Velocity limit	Δ	
		Acceleration time	Δ	
		Deceleration time	Δ	
		Immediate stop deceleration time	Δ	
		Torque limit	Δ	
		STOP input deceleration processing	Δ	
		Circular interpolation allowable error range		
	Others			
Velocity Change			Possible	

O: Items to be set always

Δ: Items to be set as needed

[Control contents]

Absolute mode control (ABS-1)

(1) Positioning control is executed from the present stop address (address given before positioning instruction) to the designated address. In the absolute mode, addresses are referenced to the zero point.

(2) Positioning direction is determined by the relationship between the present and designated addresses.

Example:

Present address : 1000

Designated address : 8000

Positioning is executed as illustrated below.

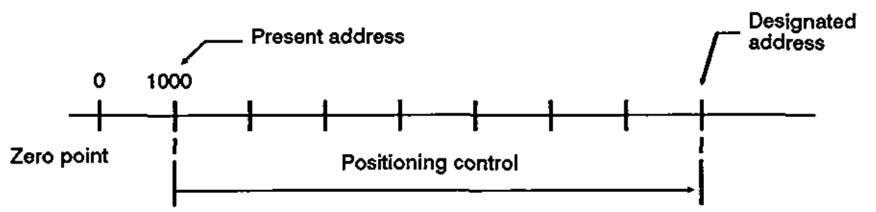


Fig. 7.1 Positioning in the Absolute Mode

Incremental mode positioning (INC-1)

(1) Positioning control is carried from the present address to a point defined by the designated axis travel distance.

(2) Positioning direction is determined by a plus or minus sign preceding the designated axis travel distance.

(a) Plus (+)

Positioning in the forward direction (addresses increasing)

(b) Minus (–)

Positioning in the reverse direction (addresses decreasing)

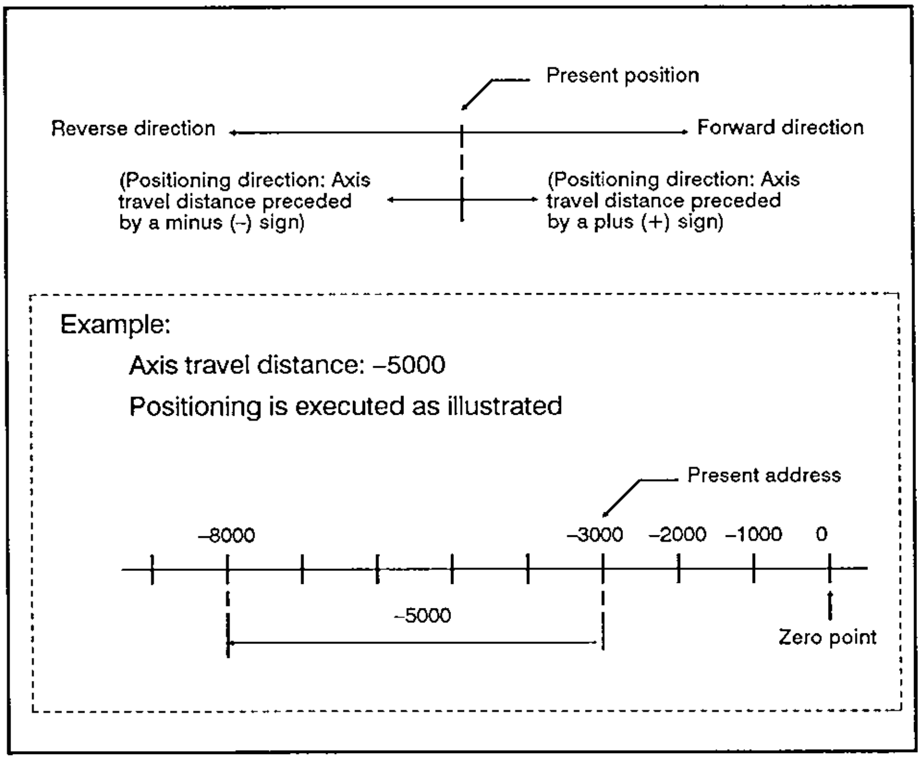
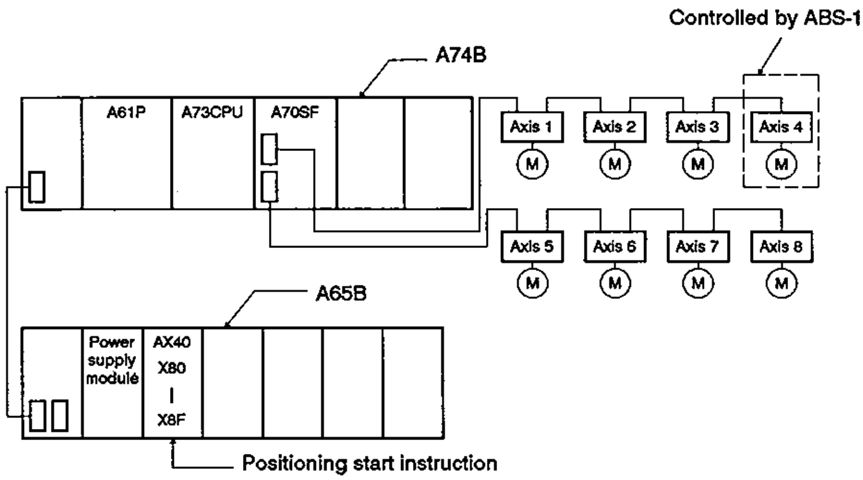


Fig. 7.2 Positioning in the Incremental Mode

[Program example]

The program for executing positioning control using the servo program No. 0 is explained below.

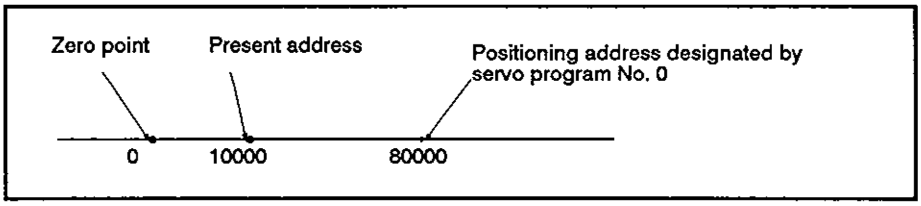
(1) System configuration



(2) Positioning addresses and controlled axis

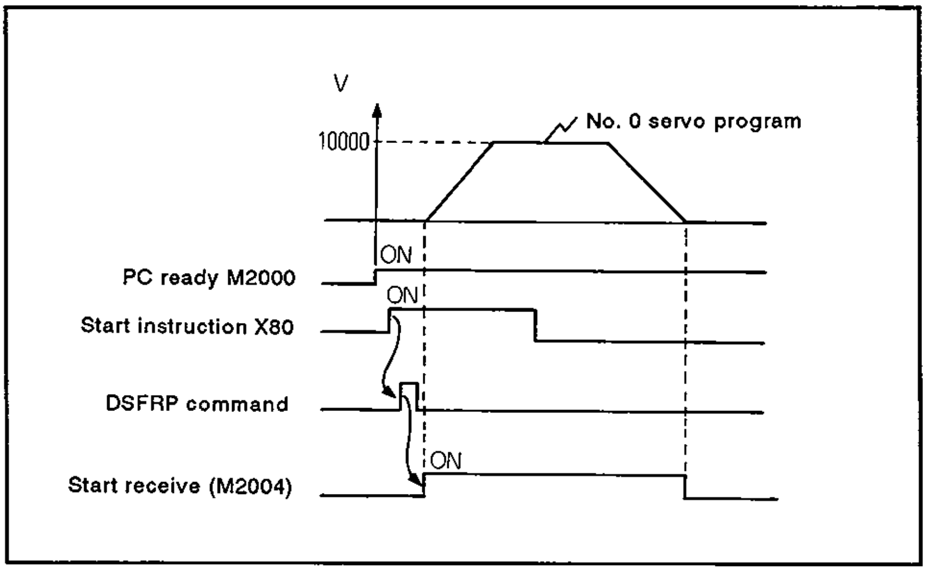
Positioning control executed by calling the servo program No. 0 is illustrated below.

Controlled axis: Axis 4



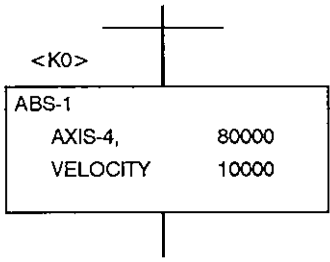
(3) Timing chart

The timing chart of positioning control called by the servo program No. 0 is indicated below.



(4) Servo program example

Servo program No. 0, which calls the positioning control, is indicated below.

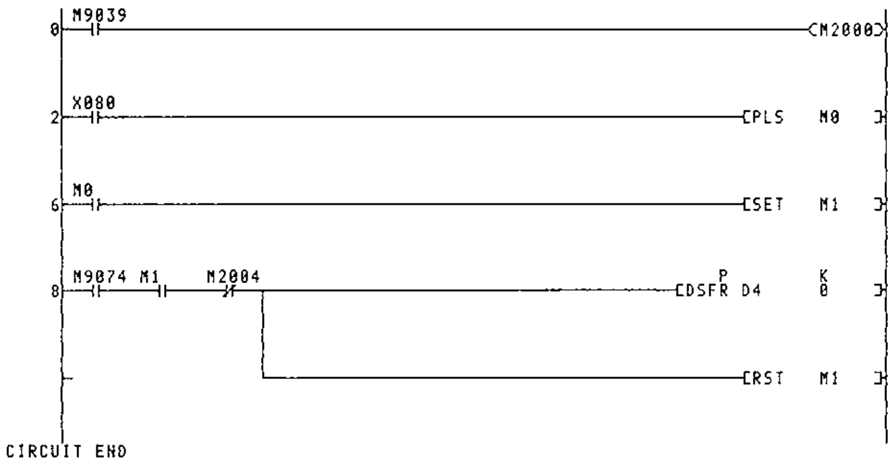


1-axis straight positioning control

Axis controlled	:	Axis 4
Positioning address	:	80000
Positioning velocity	:	10000

(5) Sequence program example

The sequence program used to execute the servo program is indicated below.



PC ready ON

Servo program No. 0 start instruction flag (M1) is set at the leading edge of X80

Servo program No. 0 execution request

M1 is reset after the completion of the execution request servo program No. 0.

7.3 Two-axis Linear Interpolation Control

In this control mode, linear interpolation control is executed from the present position to the designated position with the axes designated by the sequence program positioning instruction.

Linear interpolation control for the two axes is executed in either the absolute mode or incremental mode, called by the servo command (ABS-2 or INC-2).

<table>
<tr><td colspan="3">Servo Command</td><td>[ABS-2]</td><td>[INC-2]</td></tr>
<tr><td colspan="3">Positioning Control Mode</td><td>Absolute</td><td>Incremental</td></tr>
<tr><td colspan="3">Number of axes to be controlled</td><td colspan="2">2</td></tr>
<tr><td rowspan="21">Processing Contents</td><td rowspan="7">Common</td><td>Parameter block No.</td><td colspan="2">Δ</td></tr>
<tr><td>Axis</td><td colspan="2">O</td></tr>
<tr><td>Address/travel amount</td><td colspan="2">O</td></tr>
<tr><td>Command velocity</td><td colspan="2">O</td></tr>
<tr><td>Dwell time</td><td colspan="2">Δ</td></tr>
<tr><td>M code</td><td colspan="2">Δ</td></tr>
<tr><td>Torque limit</td><td colspan="2"></td></tr>
<tr><td rowspan="3">Circular interpola-tion</td><td>Assist-point</td><td colspan="2"></td></tr>
<tr><td>Radius</td><td colspan="2"></td></tr>
<tr><td>Center</td><td colspan="2"></td></tr>
<tr><td rowspan="8">Parameter block</td><td>Control units</td><td colspan="2">Δ</td></tr>
<tr><td>Velocity limit</td><td colspan="2">Δ</td></tr>
<tr><td>Acceleration time</td><td colspan="2">Δ</td></tr>
<tr><td>Deceleration time</td><td colspan="2">Δ</td></tr>
<tr><td>Immediate stop deceleration time</td><td colspan="2">Δ</td></tr>
<tr><td>Torque limit</td><td colspan="2">Δ</td></tr>
<tr><td>STOP input deceleration processing</td><td colspan="2">Δ</td></tr>
<tr><td>Circular interpolation allowable error range</td><td colspan="2"></td></tr>
<tr><td rowspan="3">Others</td><td></td><td colspan="2"></td></tr>
<tr><td></td><td colspan="2"></td></tr>
<tr><td></td><td colspan="2"></td></tr>
<tr><td colspan="3">Velocity Change</td><td colspan="2">Possible</td></tr>
</table>

O: Items to be set always

Δ: Items to be set as needed

Absolute mode control (ABS-2)

(1) Linear interpolation is executed by the two axes from the present address (X1, Y1) to the designated address (X2, Y2); the addresses (X1, Y1) and (X2, Y2) are referenced to the zero point.

(2) Positioning direction is determined by the relationship between the present and designated addresses of each axis.

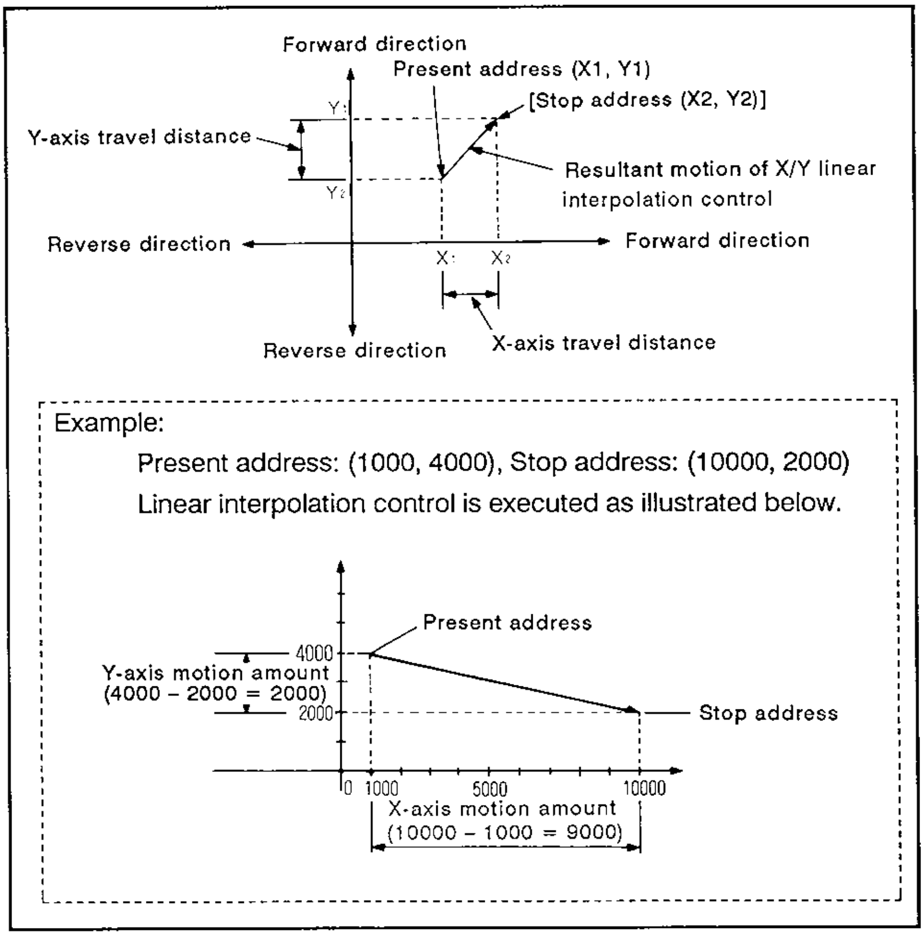


Fig. 7.3 Linear Interpolation in the Absolute Mode

Incremental mode control (INC-2)

(1) Linear interpolation control is executed from the present address to the position, defined by the axis travel distance and direction in the servo program.

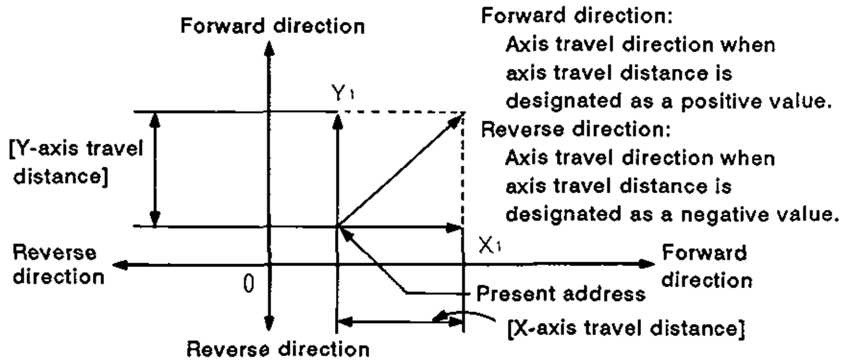
(2) Positioning direction is determined by the plus or minus sign preceding the designated travel distance of each axis.

(a) Plus (+)

Positioning in the forward direction (addresses increasing)

(b) Minus (–)

Positioning in the reverse direction (addresses decreasing)



Example:

X-axis travel distance: 6000, Y-axis travel distance: -2000

Positioning control is executed as illustrated below:.

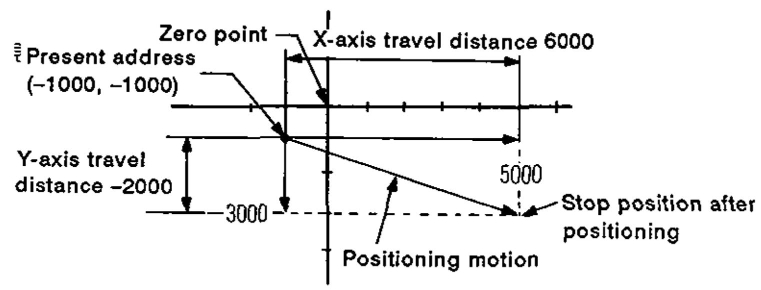
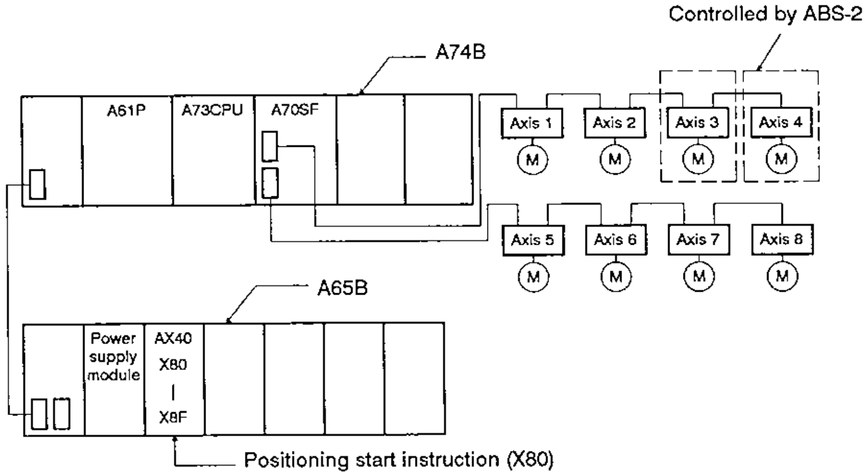


Fig. 7.4 Linear Interpolation in the Incremental Mode

[Program example]

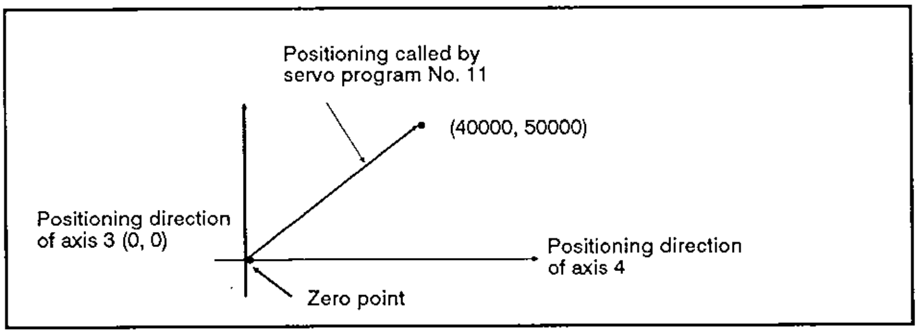
The program for executing two-axis linear interpolation control is explained below.

(1) System configuration



(2) Positioning addresses and controlled axis

Positioning control executed by servo motors for axis 3 and axis 4 is illustrated below.



(3) Positioning conditions

(a) Positioning conditions are indicated below.

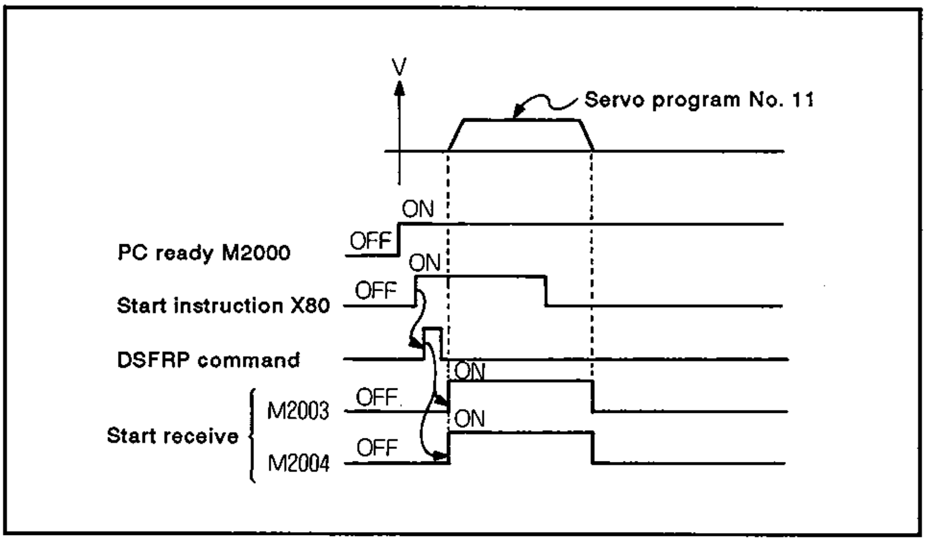
Item	Servo Program No.
	No. 11
Positioning velocity	30000

(b) Positioning start

Leading edge (OFF → ON) of X80

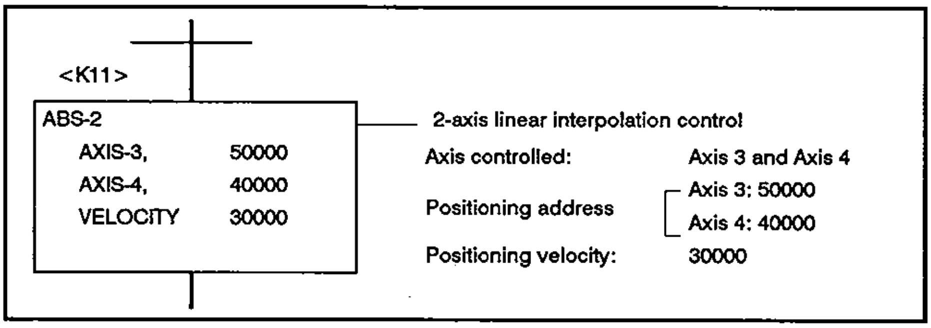
(4) Timing chart

The timing chart for two-axis linear interpolation control called by the servo program No. 11 is indicated below.



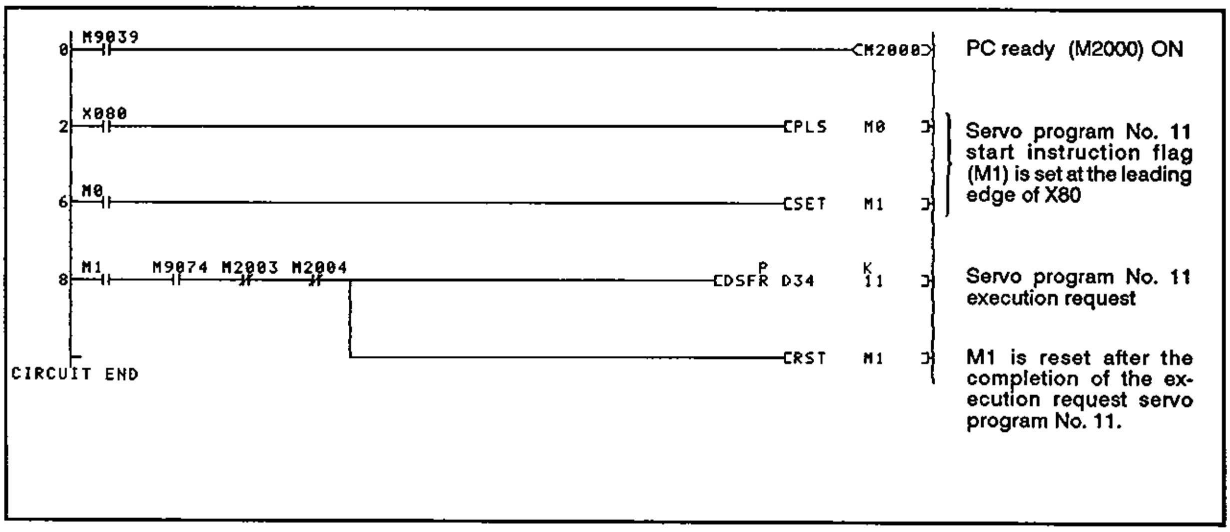
(5) Servo program

Servo program No. 11, which calls the positioning control, is indicated below.



(6) Sequence program example

The sequence program used to execute the servo program is indicated below.



7.4 Three-axis Linear Interpolation Control

In this control mode, linear interpolation control is executed from the present position to the designated position with the three axes designated by the sequence program positioning instruction.

Linear interpolation control for the three axes is executed in either the absolute mode or the incremental mode, called by the servo command (ABS-3 or INC-3).

Servo Command			[ABS-3]	[INC-3]
Positioning Control Mode			Absolute	Incremental
Number of axes to be controlled			3	
Processing Contents	Common	Parameter block No.	Δ	
		Axis	O	
		Address/travel distance	O	
		Command velocity	O	
		Dwell time	Δ	
		M code	Δ	
		Torque limit		
	Circular interpolation	Assist-point		
		Radius		
		Center		
	Parameter block	Control units	Δ	
		Velocity limit	Δ	
		Acceleration time	Δ	
		Deceleration time	Δ	
		Immediate stop deceleration time	Δ	
		Torque limit	Δ	
		STOP input deceleration processing	Δ	
		Circular interpolation allowable error range		
	Others			
Velocity Change			Possible	

O: Items to be set always

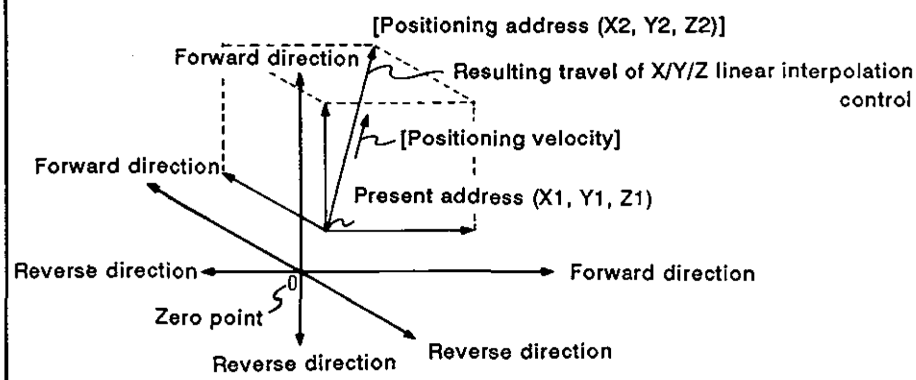
Δ: Items to be set as needed

[Control contents]

Absolute mode control (ABS-3)

(1) Linear interpolation is executed by the three axes from the present address (X1, Y1, Z1) to the designated address (X2, Y2, Z3); addresses (X1, Y1, Z1) and (X2, Y2, Z2) are referenced to the zero point.

(2) Positioning direction is determined by the relationship between the present and designated addresses of each axis.



Example:

Present address: (1000, 2000, 1000)

Positioning address: (4000, 8000, 4000)

Linear interpolation control is executed as illustrated below.

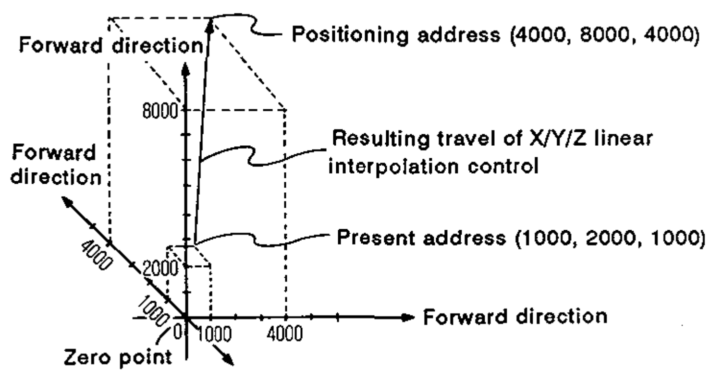


Fig. 7.5 Linear Interpolation in the Absolute Mode

Incremental mode control (INC-3)

(1) Linear interpolation control is executed from the present address to the position, defined by the axis travel distance and direction in the servo program.

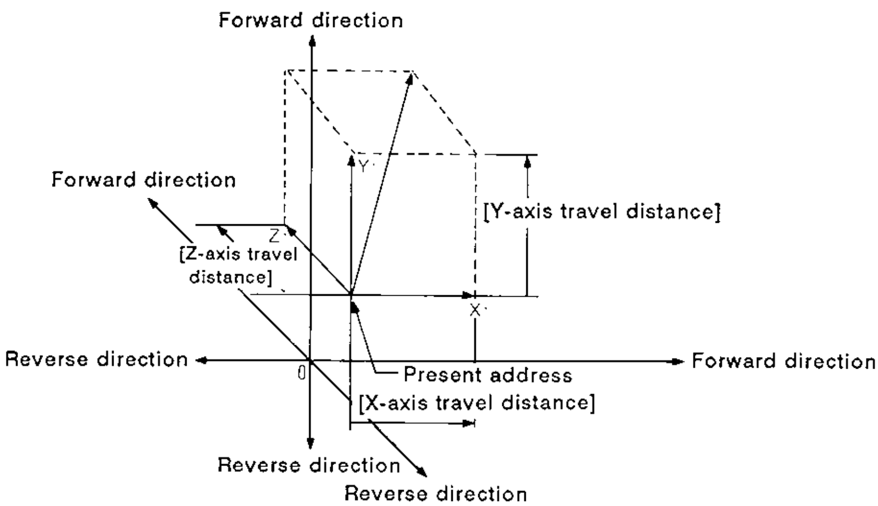
(2) The positioning direction is determined by the plus or minus sign preceding the designated travel distance of each axis.

(a) Plus (+)

Positioning in the forward direction (addresses increasing)

(b) Minus (–)

Positioning in the reverse direction (addresses decreasing)



Example:

X-axis travel distance: 10000, Y-axis travel distance: 5000

Z-axis travel distance: 6000

Positioning control is executed as illustrated below:

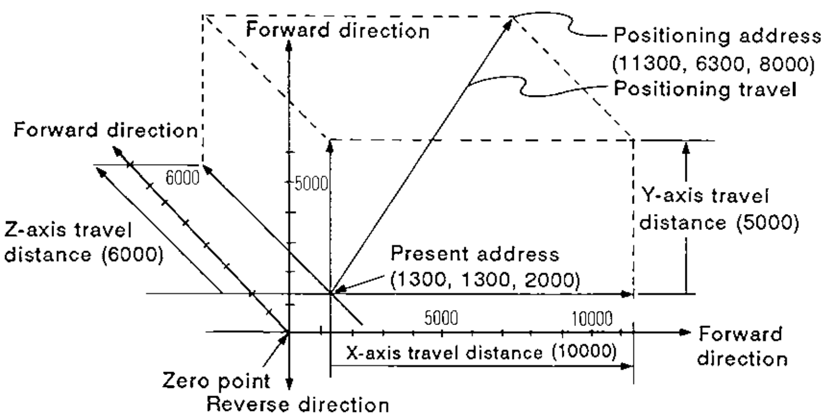
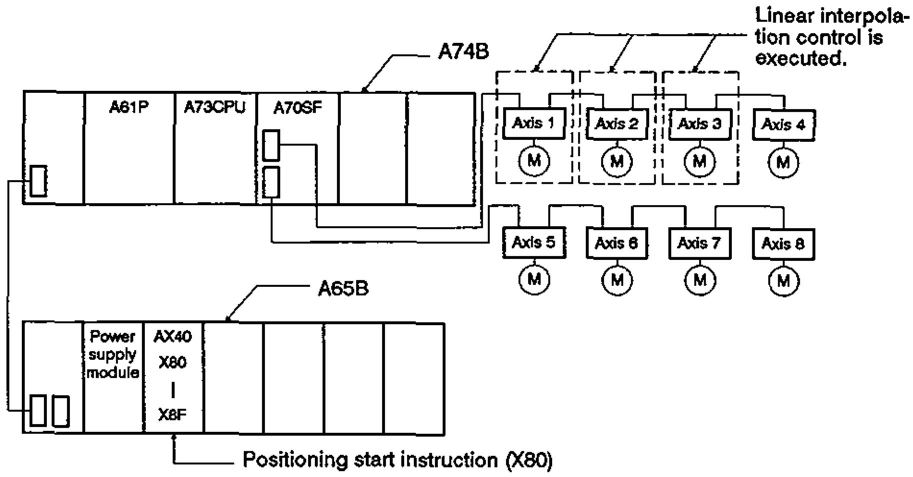


Fig. 7.5 Linear Interpolation in the Incremental Mode

[Program example]

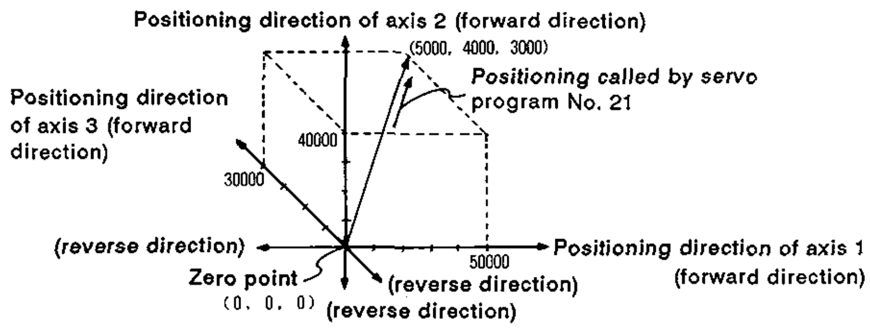
The program for executing three-axis linear interpolation control is explained below.



(1) System configuration

(2) Positioning addresses and controlled axes

Positioning control executed with the servo motors for axis 1, axis 2, and axis 3 is illustrated below.



(3) Positioning conditions

(a) The positioning conditions are indicated below.

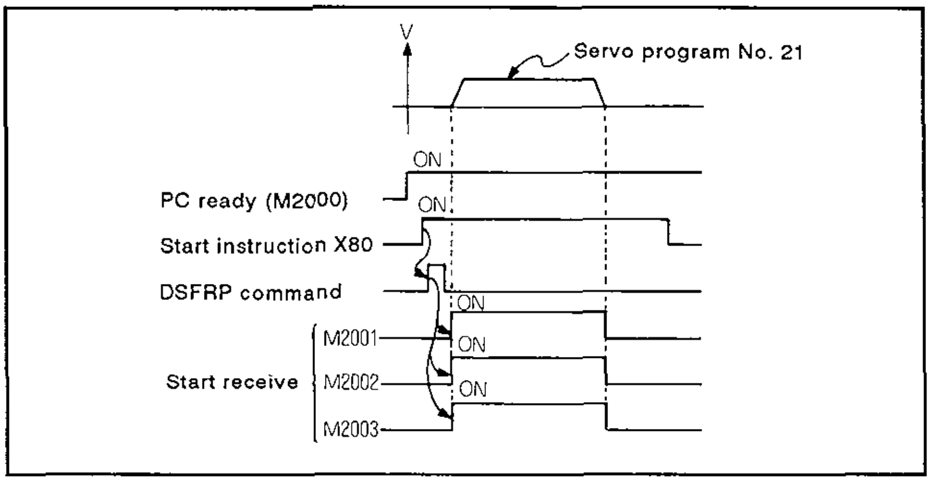
Item	Servo Program No.
	No. 21
Positioning mode	Absolute
Positioning velocity	1000

(b) Positioning start

Leading edge (OFF → ON) of X80

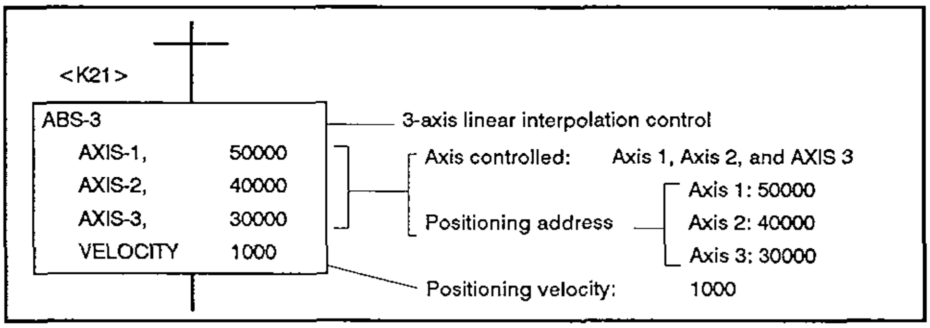
(4) Timing chart

The timing chart for three-axis linear interpolation control called by servo program No. 21 is indicated below.



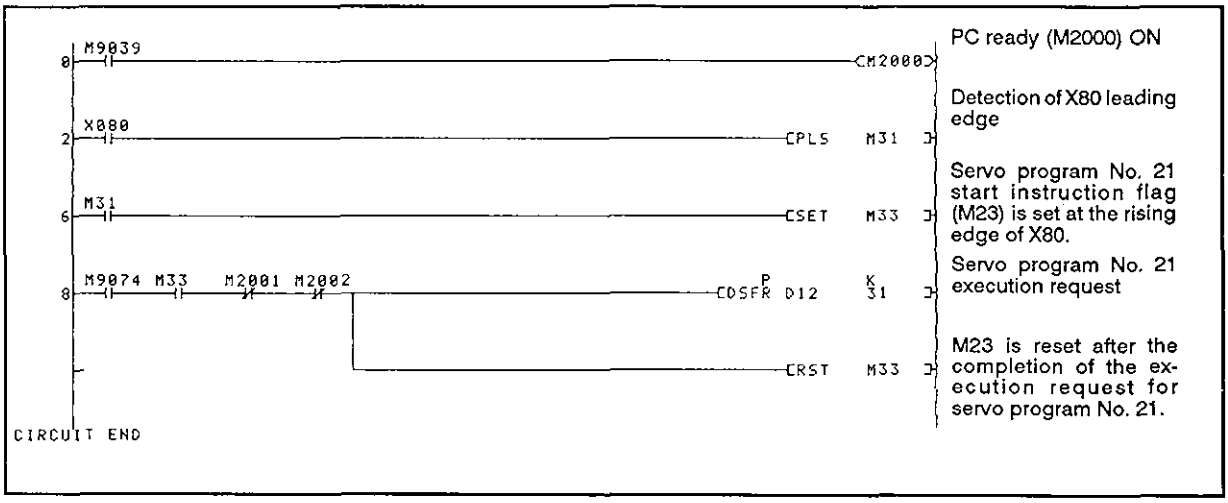
(5) Servo program

The servo program No. 21, which calls the positioning control, is indicated below.



(6) Sequence program example

The sequence program used to execute the servo program is indicated below.



7.5 Circular Interpolation Control by Designating on Assist-Point

In this control mode, circular interpolation control is executed by designating an end point address and an assist-point address.

Circular interpolation control by designating an assist-point is executed in the absolute mode or incremental mode called by the servo command (ABS ⌒ or INC ⌒).

<table>
<tr><td colspan="3">Servo Command</td><td>[ABS- ⌒]</td><td>[INC- ⌒]</td></tr>
<tr><td colspan="3">Positioning Control Mode</td><td>Absolute</td><td>Incremental</td></tr>
<tr><td colspan="3">Number of axes to be controlled</td><td colspan="2">2</td></tr>
<tr><td rowspan="21">Processing Contents</td><td rowspan="7">Common</td><td>Parameter block No.</td><td colspan="2">Δ</td></tr>
<tr><td>Axis</td><td colspan="2">O</td></tr>
<tr><td>Address/travel distance</td><td colspan="2">O</td></tr>
<tr><td>Command velocity</td><td colspan="2">O</td></tr>
<tr><td>Dwell time</td><td colspan="2">Δ</td></tr>
<tr><td>M code</td><td colspan="2">Δ</td></tr>
<tr><td>Torque limit</td><td colspan="2"></td></tr>
<tr><td rowspan="3">Circular interpola-tion</td><td>Assist-point</td><td colspan="2">O</td></tr>
<tr><td>Radius</td><td colspan="2"></td></tr>
<tr><td>Center</td><td colspan="2"></td></tr>
<tr><td rowspan="8">Parameter block</td><td>Control units</td><td colspan="2">Δ</td></tr>
<tr><td>Velocity limit</td><td colspan="2">Δ</td></tr>
<tr><td>Acceleration time</td><td colspan="2">Δ</td></tr>
<tr><td>Deceleration time</td><td colspan="2">Δ</td></tr>
<tr><td>Immediate stop deceleration time</td><td colspan="2">Δ</td></tr>
<tr><td>Torque limit</td><td colspan="2">Δ</td></tr>
<tr><td>STOP input deceleration processing</td><td colspan="2">Δ</td></tr>
<tr><td>Circular interpolation allowable error range</td><td colspan="2">Δ</td></tr>
<tr><td rowspan="3">Others</td><td></td><td colspan="2"></td></tr>
<tr><td></td><td colspan="2"></td></tr>
<tr><td></td><td colspan="2"></td></tr>
<tr><td colspan="3">Velocity Change</td><td colspan="2">Impossible</td></tr>
</table>

O: Items to be set always

Δ: Items to be set as needed

[Control contents]

Absolute mode control (ABS ⌐)

(1) Circular interpolation is executed from the present address to the end point address, passing the designated assist-point address.

(2) The center of the arc is defined at the intersection point between the two perpendicular bisectors: one bisects the linear distance between the start point (present) address and the assist-point address and the other between the assist-point address and the end point address.

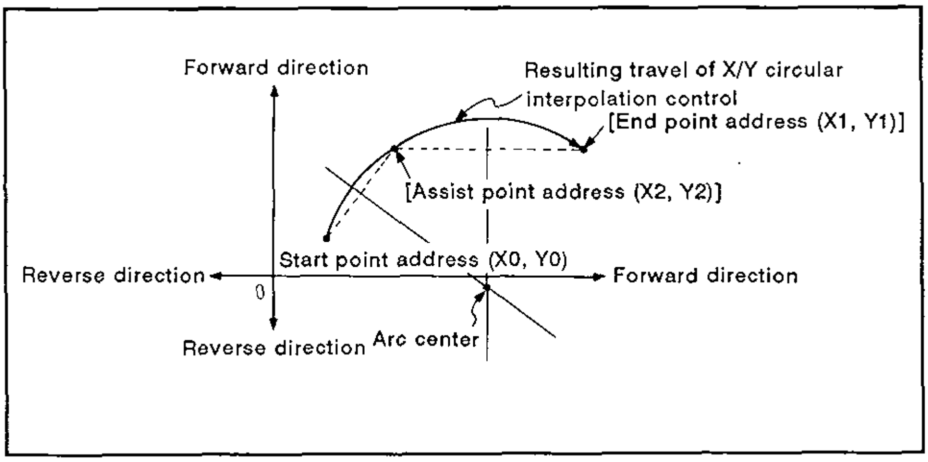


Fig. 7.7 Circular Interpolation Control in the Absolute Mode

(3) Setting range for end point and assist-point addresses: -2^{23} to 2^{23}

(4) Maximum radius: 2^{23}

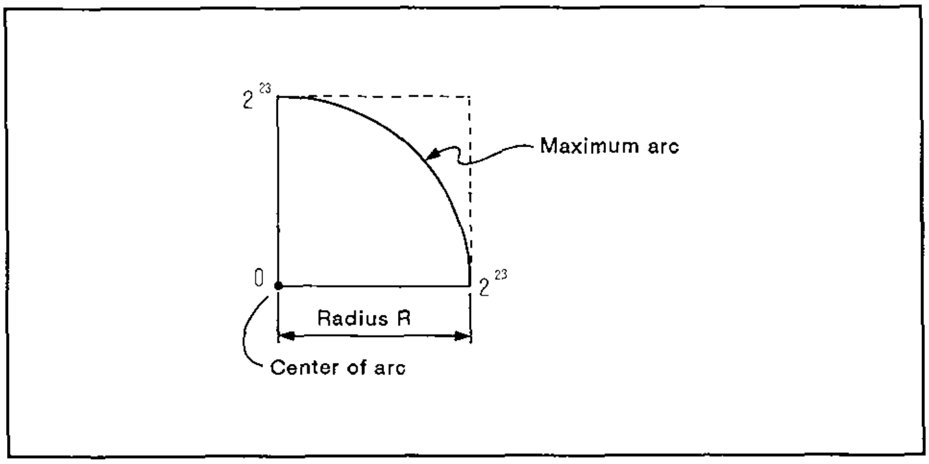


Fig. 7.8 Maximum Arc

Incremental mode control (INC ↗)

(1) Circular interpolation is executed from the present address to the end point address, passing the designated assist-point address.

(2) The center of the arc is defined at the intersection point between the two perpendicular bisectors: one bisects the distance between the start point (present) and the assist-point and the other between the assist-point and end point.

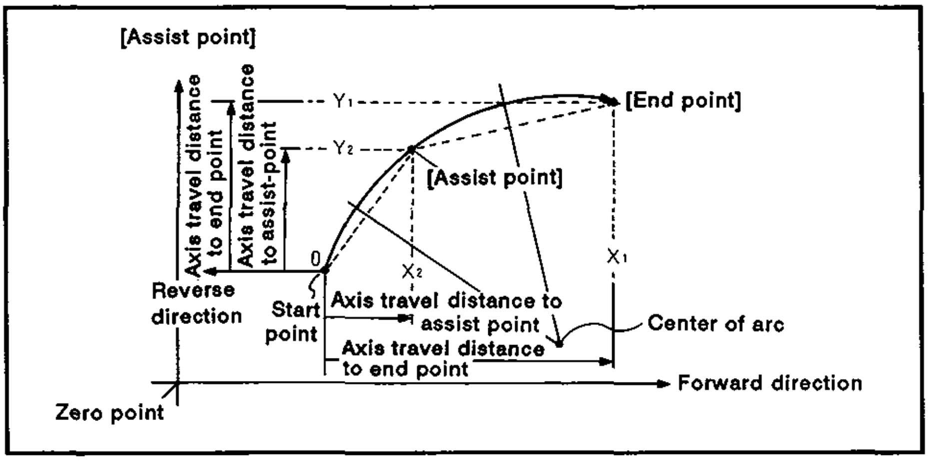
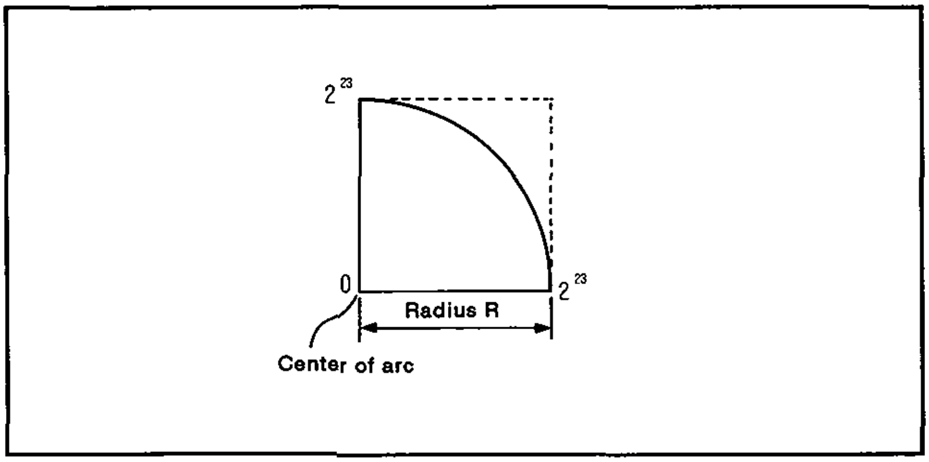


Fig. 7.9 Circular Interpolation in the Incremental Mode

(3) Setting range for travel distance to end or assist-points: 0 to $\pm 2^{23}$

(4) Maximum radius: 2^{23}

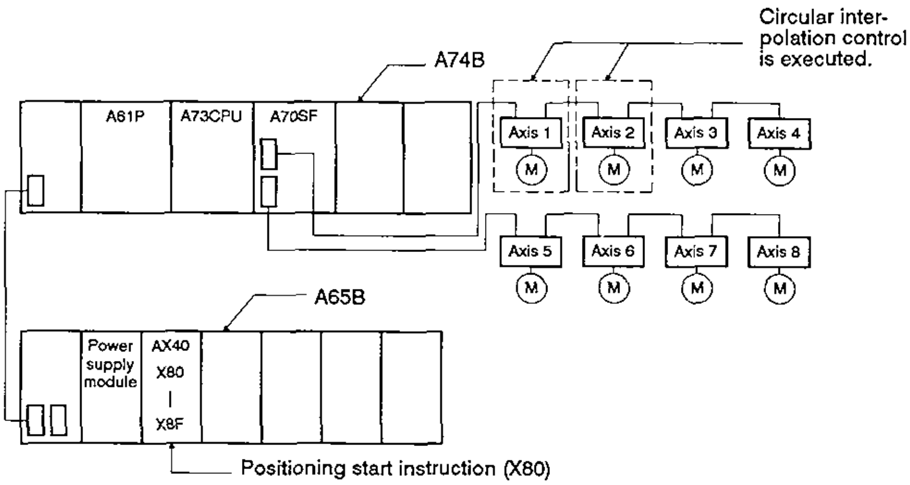
If a designated end point or assist-point causes the radius to become greater than 2^{23}, an error will occur at the start of interpolation; the error code (107) is stored in the data register.



[Program example]

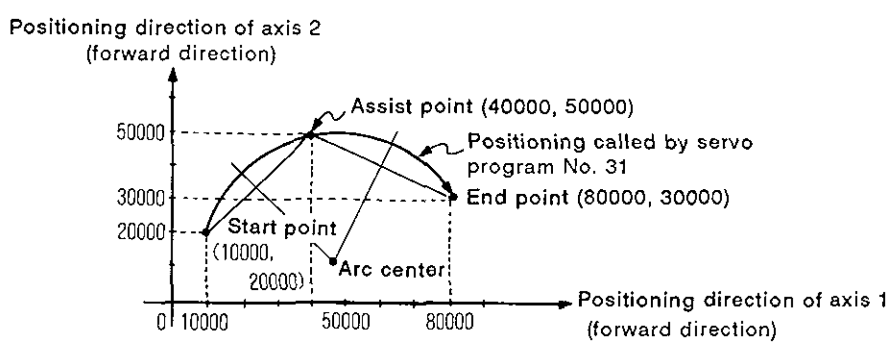
The program for carrying out circular interpolation control is explained below.

(1) System configuration



(2) Positioning addresses and controlled axes

Positioning control executed by the servo motors for axis 1 and axis 2 is illustrated below.



(3) Positioning conditions

(a) The positioning conditions are indicated below.

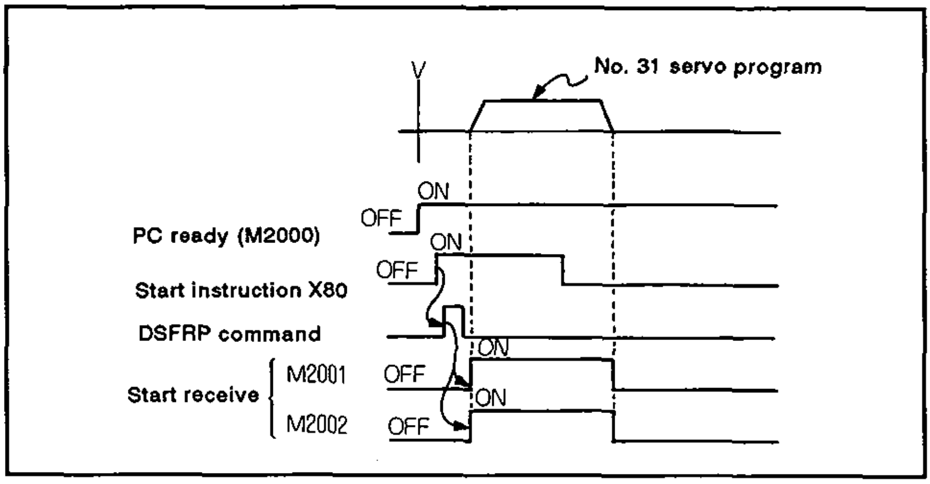
Item	Servo Program No.
	No. 31
Positioning mode	Absolute
Positioning velocity	1000

(b) Positioning start

Leading edge (OFF → ON) of X80

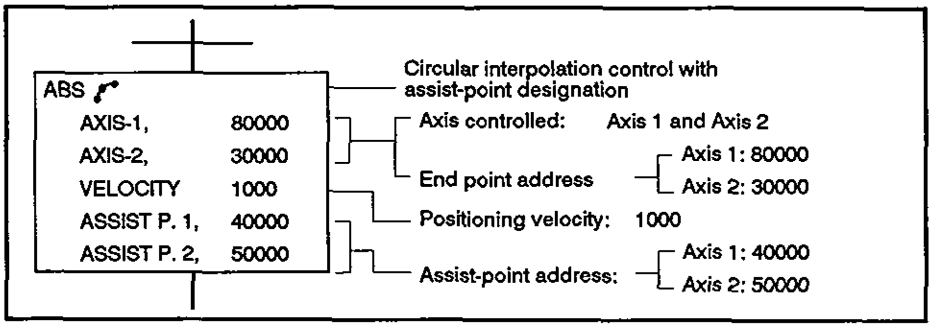
(4) Timing chart

The timing chart for circular interpolation control called by No. 31 servo program is indicated below.



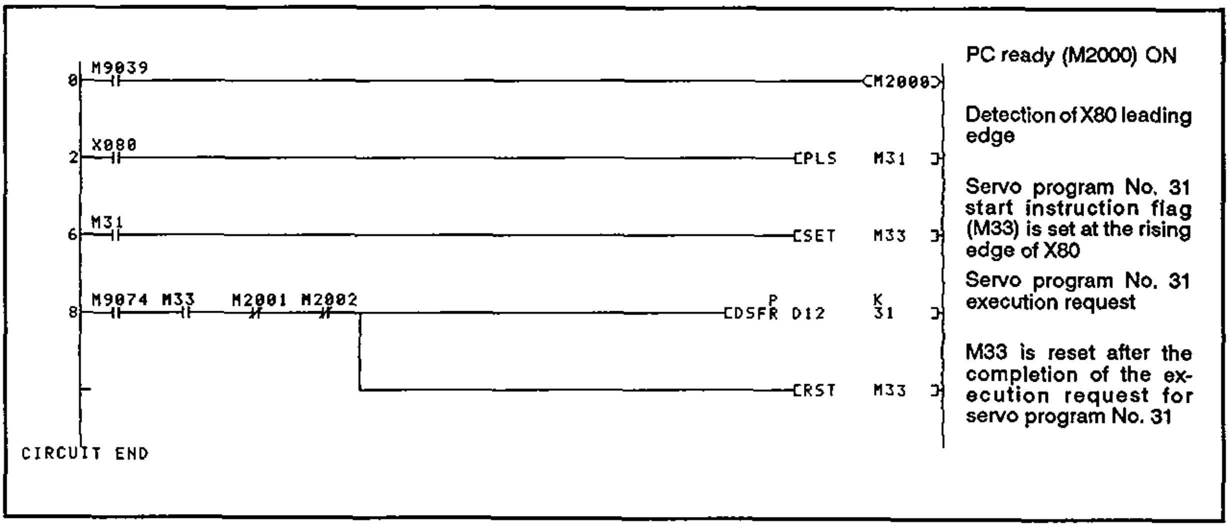
(5) Servo program

Servo program No. 31, which calls the circular interpolation control, is indicated below.



(6) Sequence program example

The sequence program used to execute the servo program is indicated below.Servo program No. 31 start instruction flag (M33) is set at the leading edge of X80.



7.6 Circular Interpolation Control by Designating the Radius

In this control mode, circular interpolation control is executed by designating the end point address and radius.

Circular interpolation control by designating the radius is executed in the absolute mode or incremental mode, called by a servo command (ABS , ABS , ABS , ABS , INC , INC , INC , INC).

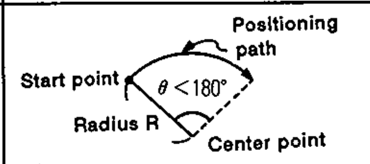
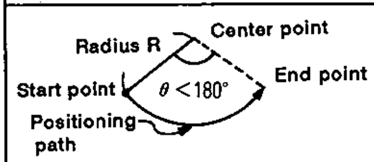
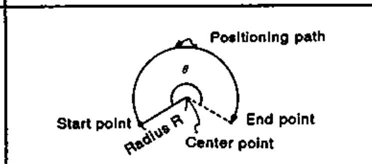
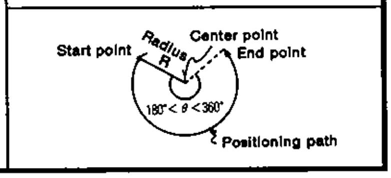
Servo Command			[ABS	[ABS	[ABS	[ABS	[INC	[INC	[INC	[INC
Positioning Control Mode			Absolute				Incremental			
Number of axes to be controlled			2							
Processing Contents	Common	Parameter block No.	Δ							
		Axis	O							
		Address/travel distance	O							
		Command velocity	O							
		Dwell time	Δ							
		M code	Δ							
		Torque limit								
	Circular interpola-tion	Assist-point								
		Radius	O							
		Center								
	Parameter block	Control units	Δ							
		Velocity limit	Δ							
		Acceleration time	Δ							
		Deceleration time	Δ							
		Immediate stop deceleration time	Δ							
		Torque limit	Δ							
		STOP input deceleration processing	Δ							
		Circular interpolation allowable error range	Δ							
	Others									
Velocity Change			Impossible							

O: Items to be set always

Δ: Items to be set as needed

[Control details]

The servo commands and the related control contents are summarized in the table below:

Command	Servo Motor Rotation Direction	Center Angle	Positioning Path
ABS	CW	$0° < \theta < 180°$	
INC			
ABS	CCW		
INC			
ABS	CW	$180° \leq \theta < 360°$	
INC			
ABS	CCW		
INC			

Absolute mode control (ABS ↷ , ABS ↶ , ABS ↷ , ABS ↶)

(1) Circular interpolation is executed from the present address to the end point address along an arc of the designated radius.

(2) The center point of the arc is the intersection point between the perpendicular bisector of the line defined by the start and end points and the arc of the designated radius R drawn around the start point.

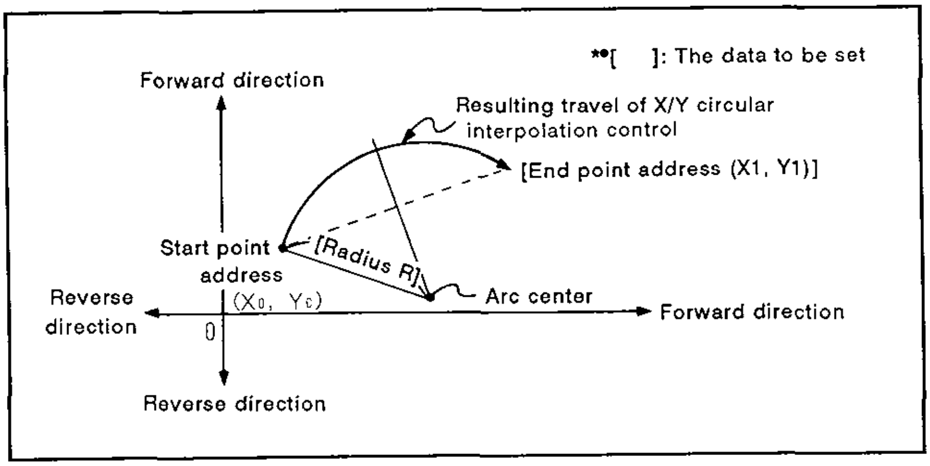


Fig. 7.10 Circular Interpolation in the Absolute Mode

(3) Setting range of end point address: -2^{23} to 2^{23}

(4) Maximum radius: 2^{23}

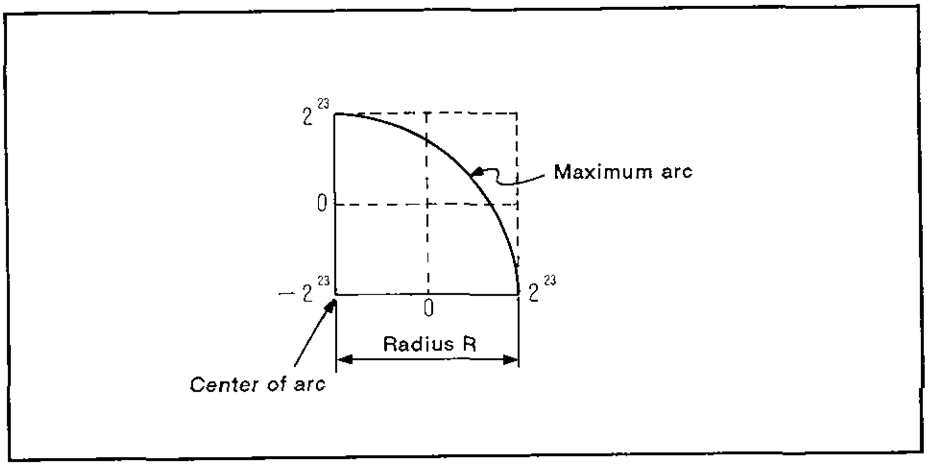
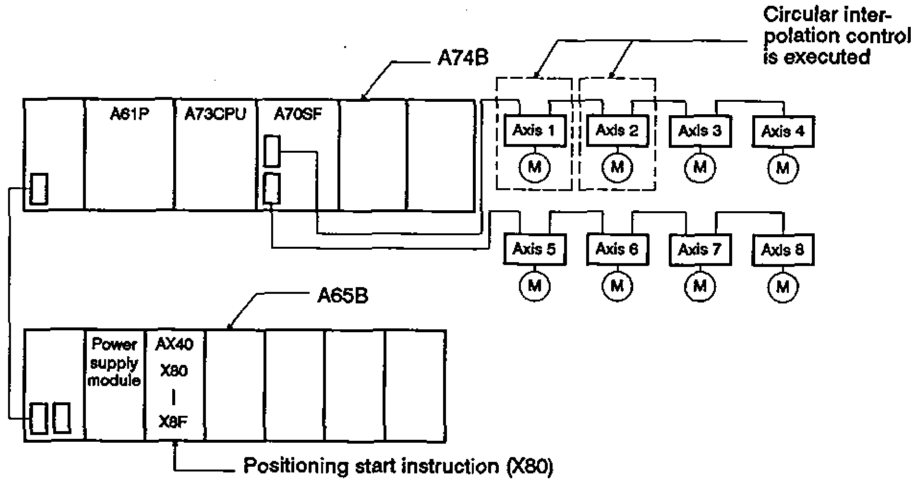


Fig. 7.11 Maximum Arc

[Program example]

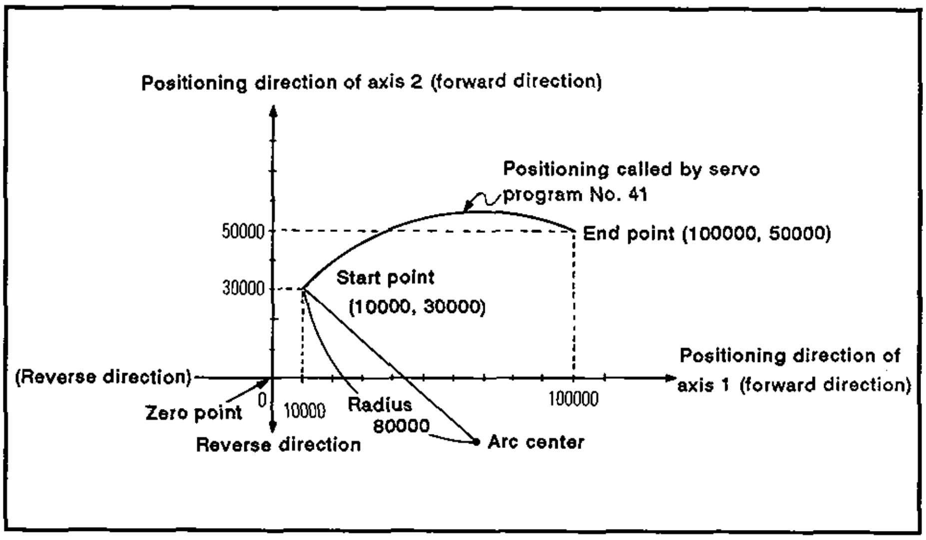
The program for executing circular interpolation control by designating the radius is explained below.

(1) System configuration



(2) Positioning addresses and controlled axes

Positioning control executed by the servo motors for axis 1 and axis 2 is illustrated below.



(3) Positioning conditions

(a) The positioning conditions are indicated below.

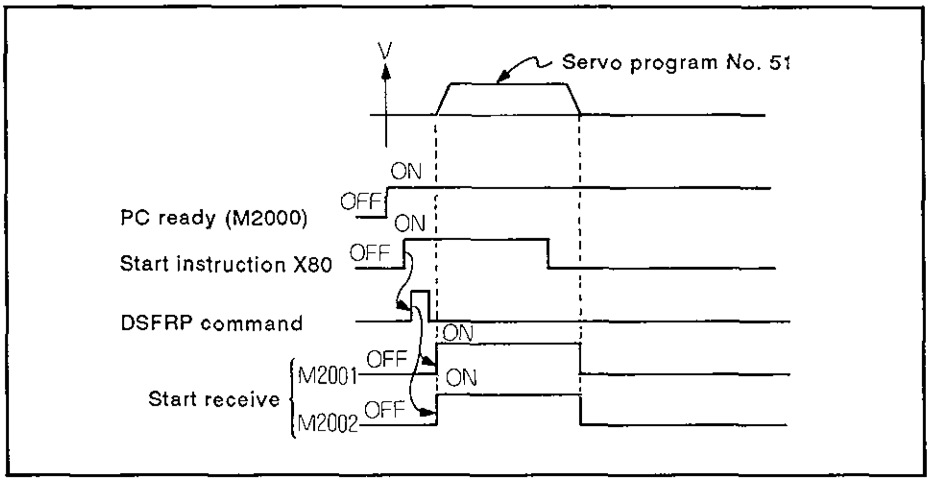
Item	Servo Program No.
	No. 41
Positioning mode	Absolute
Positioning velocity	1000

(b) Positioning start

Leading edge (OFF → ON) of X80

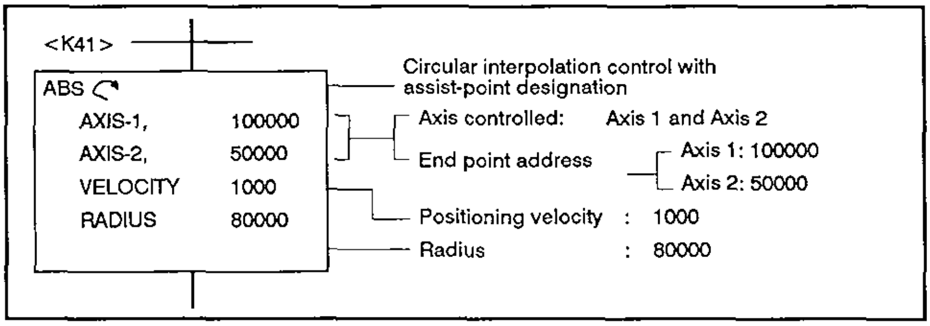
(4) Timing chart

The timing chart for circular interpolation control called by the No. 41 servo program is indicated below.



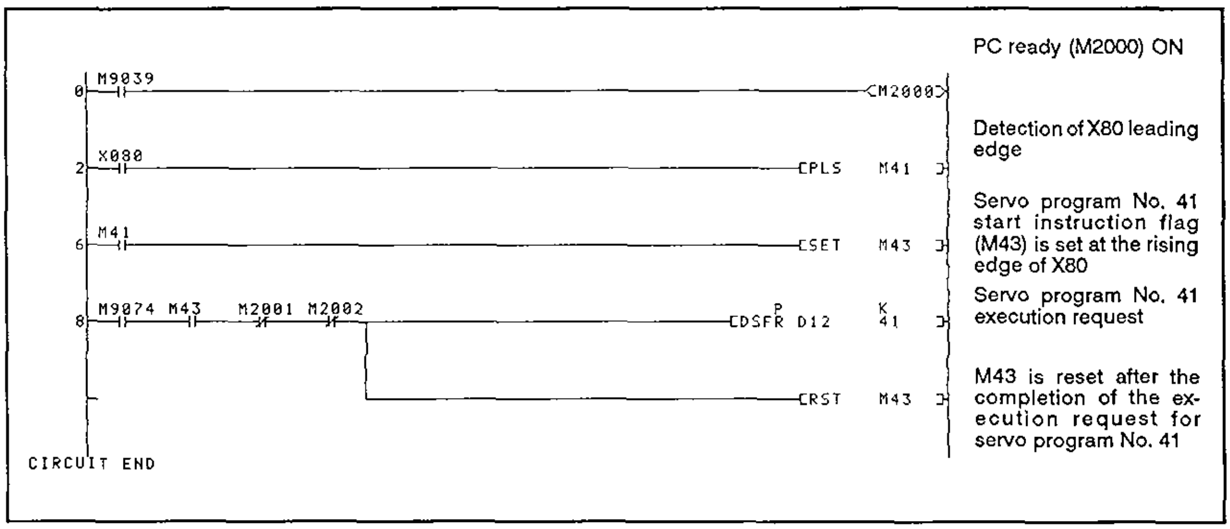
(5) Servo program

Servo program No. 41, which calls the circular interpolation control, is indicated below.



(6) Sequence program example

The sequence program used to execute the servo program is indicated below.



7.7 Circular Interpolation Control by Designating the Center

In this control mode, circular interpolation control is executed by designating the end point address and the arc center.

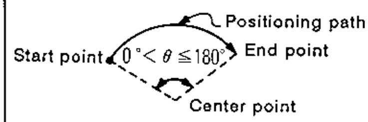
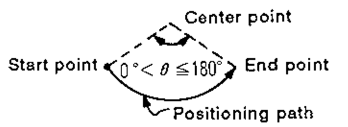
Circular interpolation control by designating the arc center is executed in the absolute mode or incremental mode called by the servo command
(ABS , ABS , ABS , ABS , INS , INS , INS , INS).

Servo Command			[ABS-]	[ABS-]	[INC-]	[INC-]
Positioning Control Mode			Absolute		Incremental	
Number of axes to be controlled			2			
Processing Contents	Common	Parameter block No.	Δ			
		Axis	O			
		Address/travel distance	O			
		Command velocity	O			
		Dwell time	Δ			
		M code	Δ			
		Torque limit				
	Circular interpolation	Assist-point				
		Radius				
		Center	O			
	Parameter block	Control unit	Δ			
		Velocity limit	Δ			
		Acceleration time	Δ			
		Deceleration time	Δ			
		Immediate stop deceleration time	Δ			
		Torque limit	Δ			
		STOP input deceleration processing	Δ			
		Circular interpolation allowable error range	Δ			
	Others					
Velocity Change			Impossible			

O: Items to be set always
Δ: Items to be set as needed

[Control details]

The servo commands and the related control contents are summarized in the table below:

Command	Servo Motor Rotation Direction	Center Angle	Positioning Path
ABS	CW	$0° < \theta < 360°$	
INC			
ABS	CCW		
INC			

Absolute mode control (ABS , ABS)

(1) Circular interpolation is executed from the start point address to the end point address along the arc which has its radius defined by the distance between the start point address and the designated center point address.

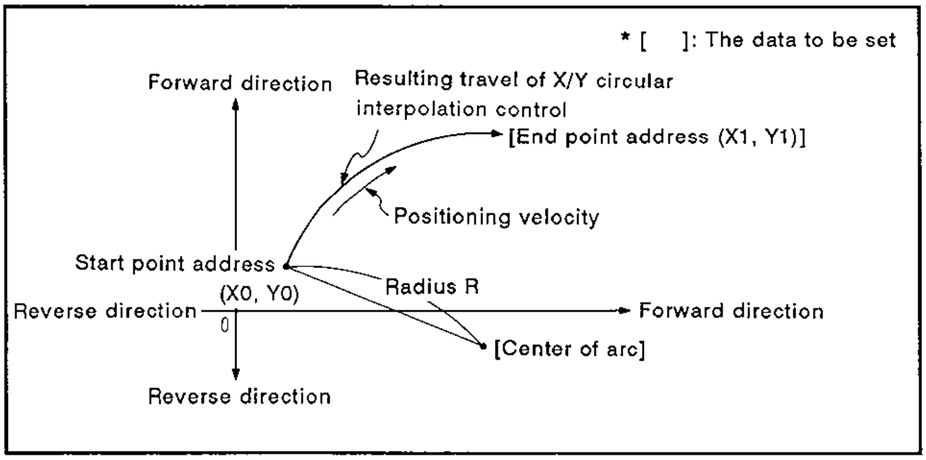


Fig. 7.12 Circular Interpolation in the Absolute Mode

(2) Positioning along a full circle is possible by designating the same address for the end point as for the start point.

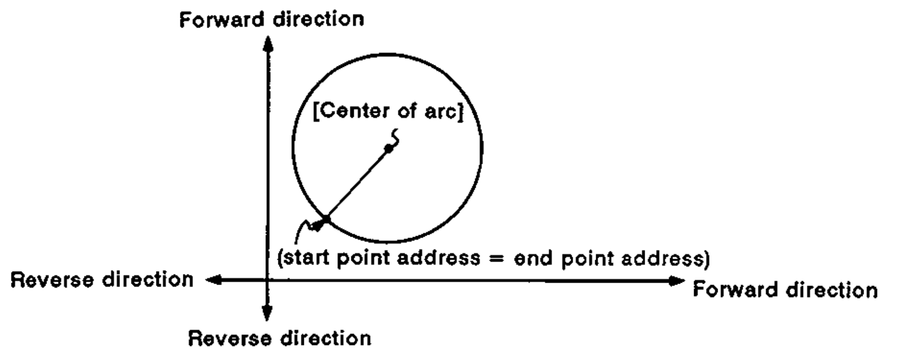


Fig. 7.13 Positioning Along a Full Circle

(3) Setting range of end point address and arc center: -2^{23} to 2^{23}

(4) Maximum radius: 2^{23}

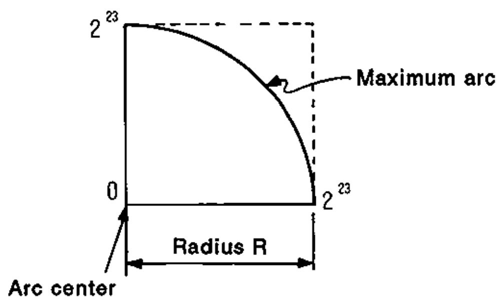


Fig. 7.14 Maximum Arc

Incremental mode control (INC ↻, INC ↺)

(1) Circular interpolation is executed from the start address to the end point along an arc; the radius of the arc is defined by the distance between the start point and the designated center point.

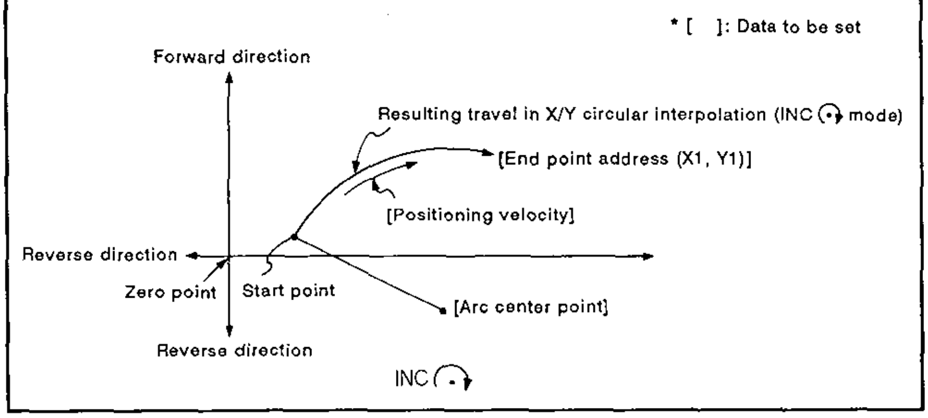


Fig. 7.15 Circular Interpolation in the Incremental Mode (INC ↻)

(2) Positioning along a full circle is possible by setting (0, 0) for the end point address.

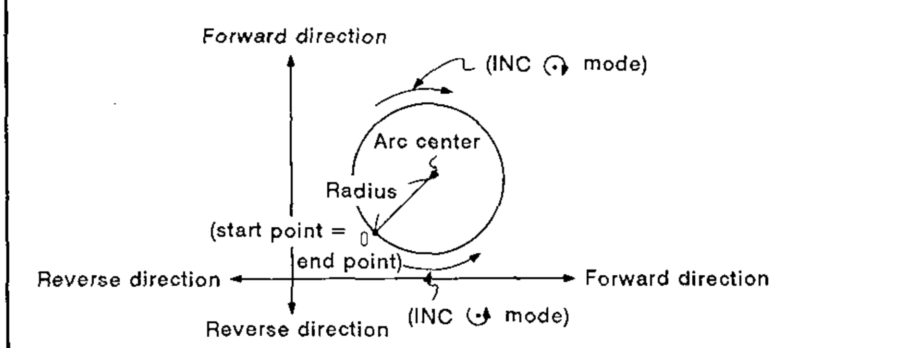


Fig. 7.16 Positioning Along a Full Circle

(3) Setting range for arc center and axis travel distance to end point: 0 to $+/-2^{23}$

(4) Maximum radius: 2^{23}

If a designated end point or arc center causes the radius to become greater than 2^{23}, an error will occur at the start of interpolation; the error code (107) is stored in the data register.

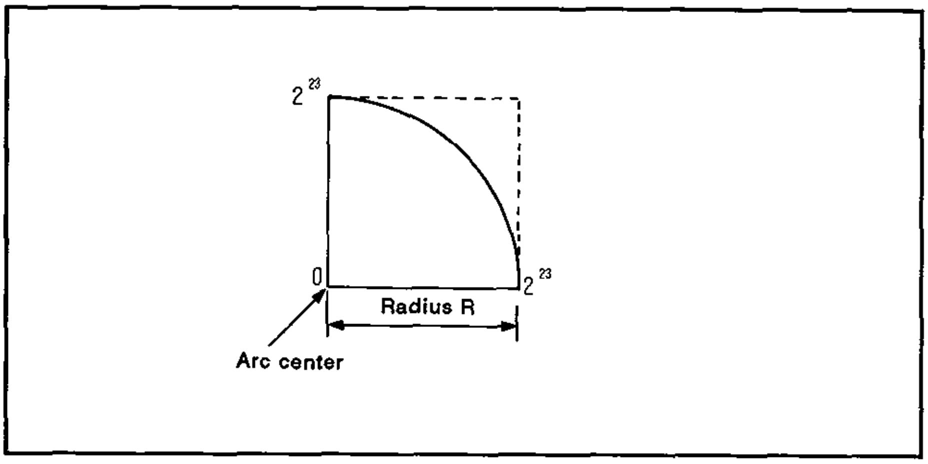
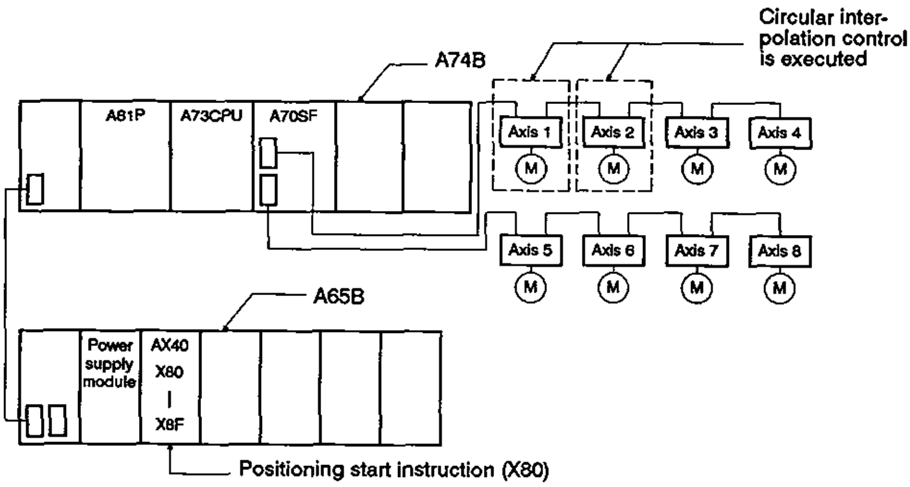


Fig. 7.17 Maximum Arc

[Program example]

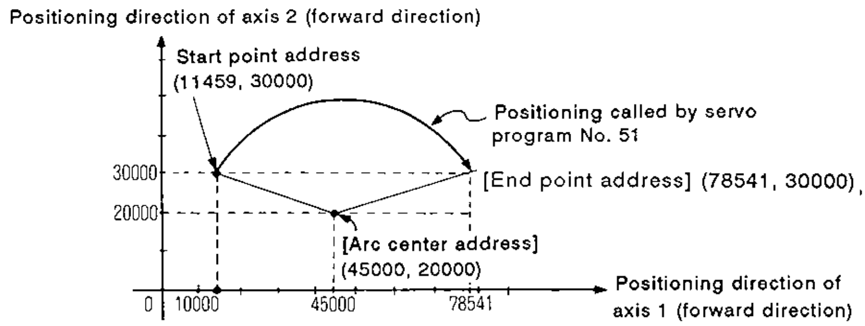
The program for carrying out circular interpolation control by designating the arc center is explained below.

(1) System configuration



(2) Positioning addresses and controlled axes

Positioning control executed by the servo motors for axis 1 and axis 2 is illustrated below.



(3) Positioning conditions

(a) The positioning conditions are indicated below.

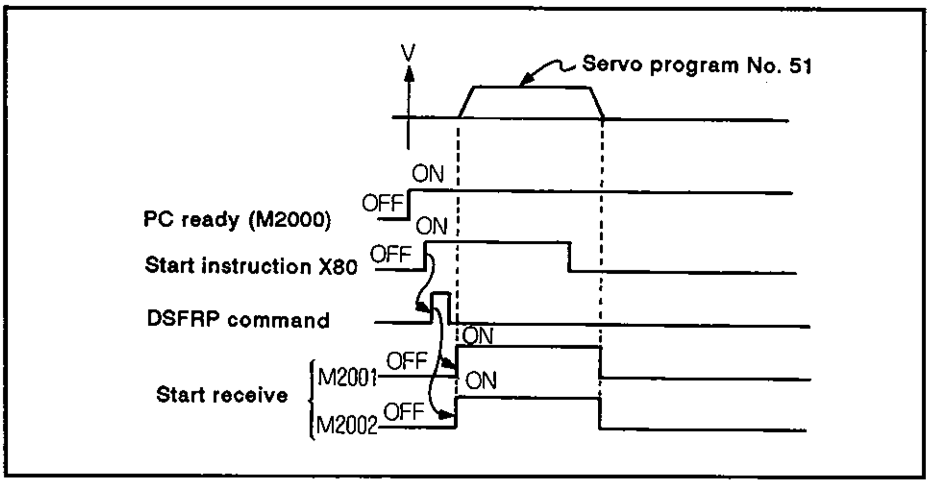
Item	Servo Program No.
	No. 51
Positioning mode	Absolute
Positioning velocity	1000

(b) Positioning start

Leading edge (OFF → ON) of X80

(4) Timing chart

The timing chart for circular interpolation control called by No. 51 servo program is indicated below.



(5) Servo program

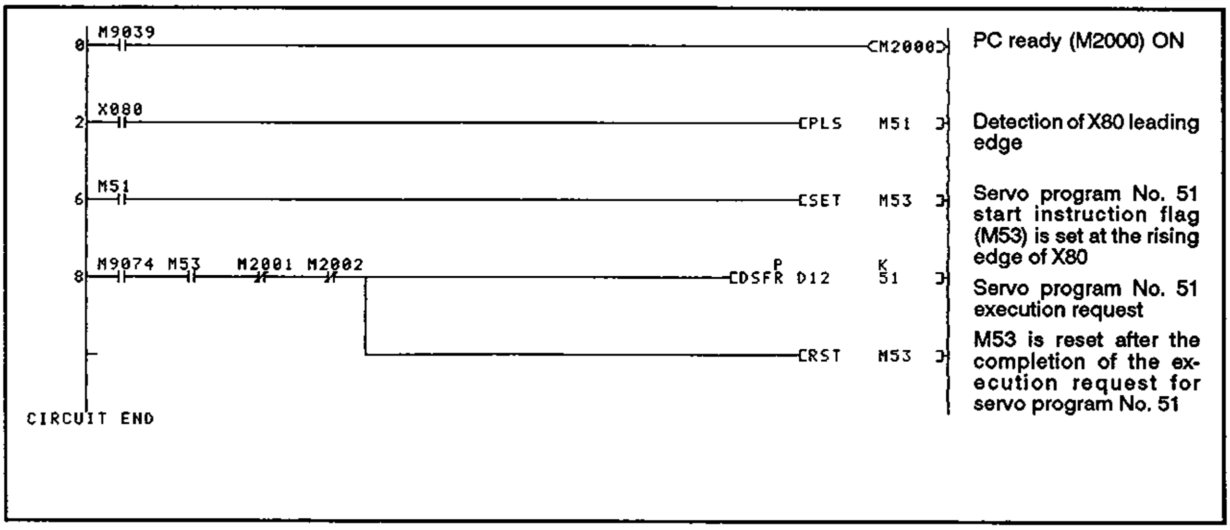
Servo program No. 51, which calls the circular interpolation control, is indicated below.

```
<K51>
ABS
  AXIS-1,      78541
  AXIS-2,      30000
  VELOCITY     1000
  CENTER 1     45000
  CENTER 2     20000
```

- Circular interpolation control with arc center
- Axis controlled: Axis 1 and Axis 2
- End point address: Axis 1: 78541, Axis 2: 30000
- Positioning velocity: 1000
- Arc center address: Axis 1: 45000, Axis 2: 20000

(6) Sequence program example

The sequence program used to execute the servo program is indicated below.



7.8 One-axis Incremental Feed Control

In the incremental feed control mode, positioning control is executed to feed the axis designated by the sequence program positioning instruction the designated travel distance from the present position.

The incremental feed control mode is called by servo command FEED-1.

Servo Command			[FEED-1]
Positioning Control Mode			Incremental
Number of axes to be controlled?			1
Processing Contents	Common	Parameter block No.	Δ
		Axis	O
		Address/travel distance	O
		Command velocity	O
		Dwell time	Δ
		M code	Δ
		Torque limit	
	Circular interpolation	Assist-point	
		Radius	
		Center	
	Parameter block	Control units	Δ
		Velocity limit	Δ
		Acceleration time	Δ
		Deceleration time	Δ
		Immediate stop deceleration time	Δ
		Torque limit	Δ
		STOP input deceleration processing	Δ
		Circular interpolation allowable error range	
	Others		
Velocity Change			Possible

O: Items to be set always

Δ: Items to be set as needed

[Control contents]

(1) Taking the present position as the reference position (0), positioning control is executed to feed the axis the designated distance.

(2) The positioning direction is determined by the plus or minus sign preceding the designated travel distance.

(a) Plus (+)

Positioning in the forward direction (addresses increasing)

(b) Minus (–)

Positioning in the reverse direction (addresses decreasing)

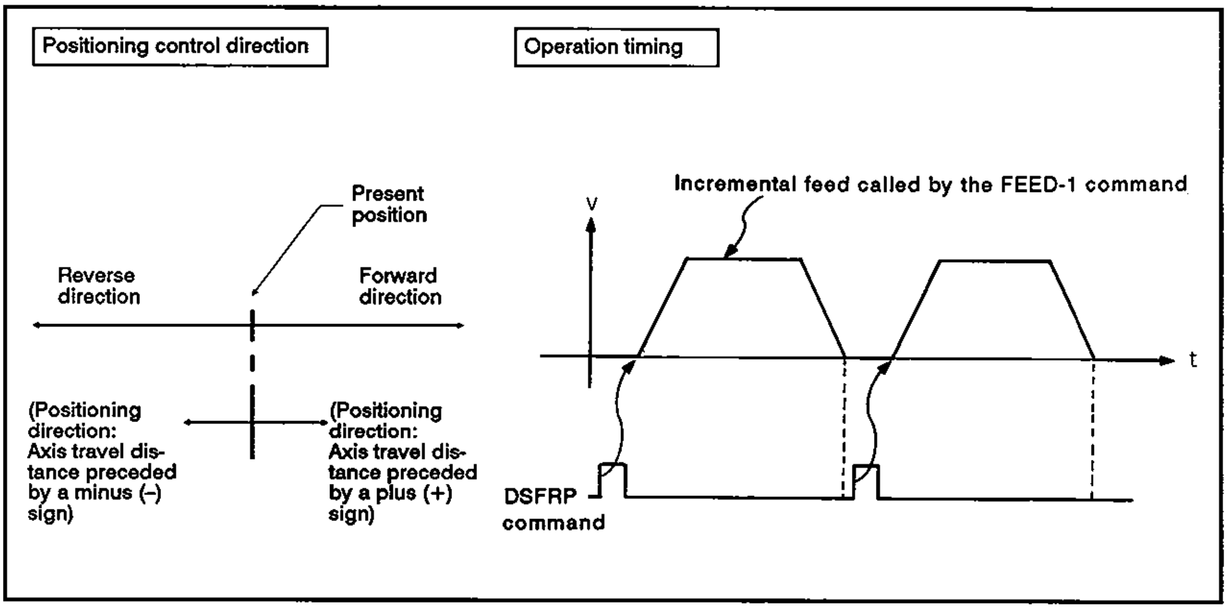


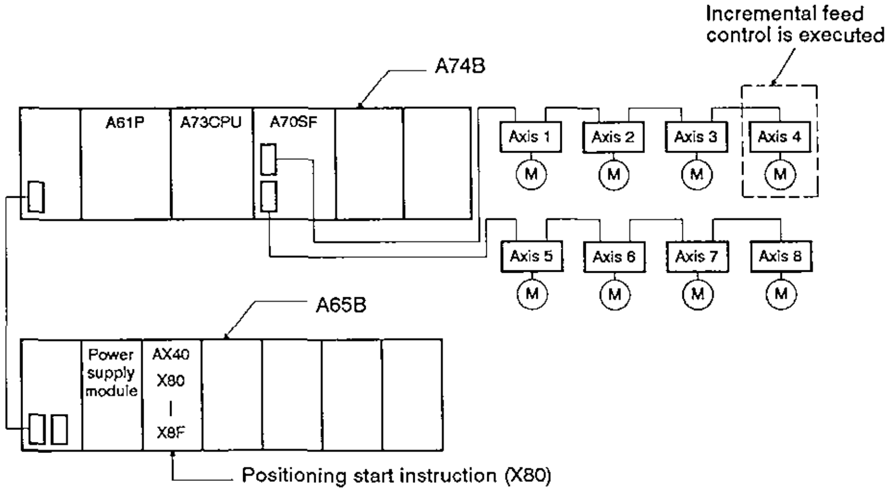
Fig. 7.18 One-axis Incremental Feed Control

POINT
Do not set "0" for the travel distance in incremental feed. If "0" is set for incremental feed distance, the incremental feed is completed before incremental feed is executed.

[Program example]

The program used to execute one-axis incremental feed control repeatedly is explained below.

(1) System configuration



(2) Incremental feed control conditions

(a) The incremental feed control conditions are indicated below.

Item	Setting
Servo program No.	No. 300
Controlled axis	Axis 4
Control velocity	10000
Axis travel distance	100000

(b) Incremental feed start instruction

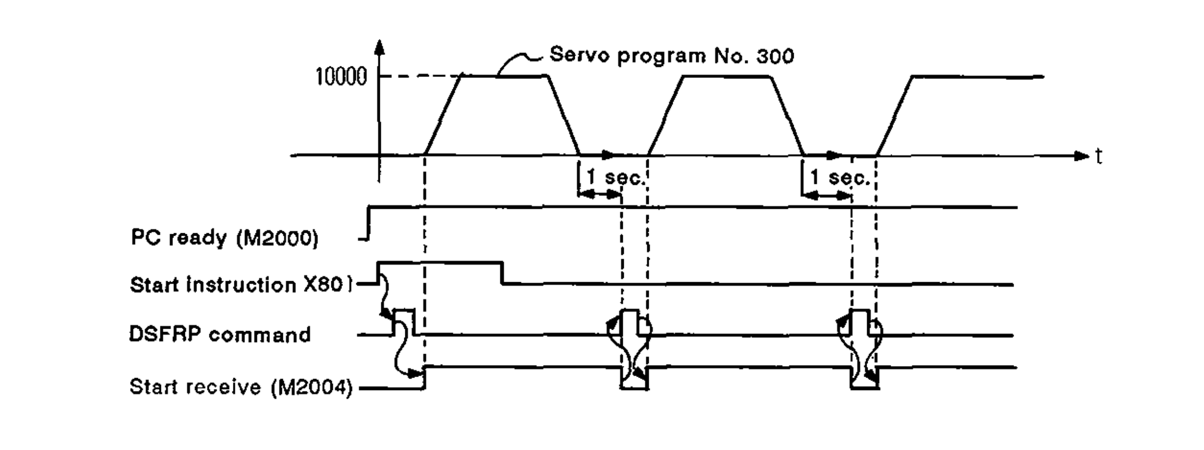
Leading edge (OFF → ON) of X80

(c) Incremental feed end instruction

Trailing edge (OFF → ON) of X81

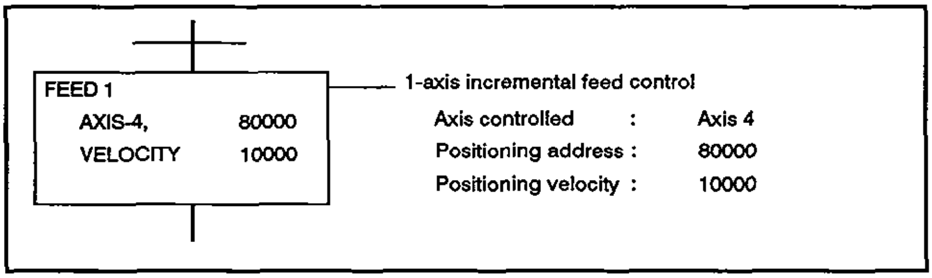
(3) Timing chart

The timing chart of the incremental feed called by servo program No. 300 is illustrated below.



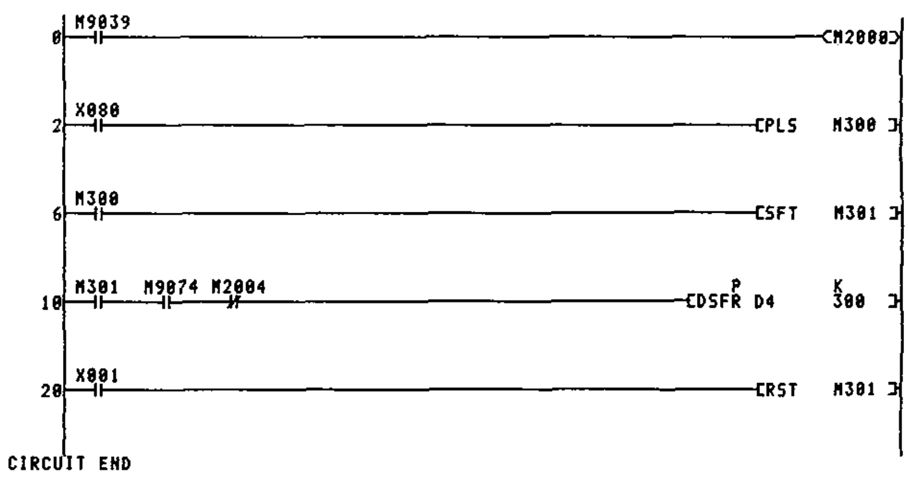
(4) Servo program example

Servo program No. 300, which calls the incremental feed control, is indicated below.



(5) Sequence program example

The sequence program used to execute the servo program is indicated below.



PC ready ON

Servo program No. 300 start instruction flag (M301) is set at the rising edge of X80

M301 is reset after the completion of the execution request for servo program No. 300

7.9 Incremental Feed Control in the Two-axis Linear Interpolation Mode

In this control mode, incremental feed control is executed with the two axes designated by the sequence program positioning instruction from the present position to the designated position; the motion of the two axes is controlled in the two-axis linear interpolation mode.

Incremental feed control in the two-axis linear interpolation mode is called by servo command FEED-2.

Servo Command			[FEED-2]
Positioning Control Mode			Incremental
Number of axes to be controlled			2
Processing Contents	Common	Parameter block No.	Δ
		Axis	O
		Address/travel distance	O
		Command velocity	O
		Dwell time	Δ
		M code	Δ
		Torque limit	
	Circular interpolation	Assist-point	
		Radius	
		Center	
	Parameter block	Control units	Δ
		Velocity limit	Δ
		Acceleration time	Δ
		Deceleration time	Δ
		Immediate stop deceleration time	Δ
		Torque limit	Δ
		STOP input deceleration processing	Δ
		Circular interpolation allowable error range	
	Others		
Velocity Change			Possible

O: Items to be set always

Δ: Items to be set needed

[Control contents]

(1) Positioning control is executed from the present position, which is taken as "0", to the position defined by the travel distance and direction in the servo program.

(2) Positioning direction is determined by the plus or minus sign preceding the designated travel distance of each axis.

(a) Plus (+)

Positioning in the forward direction (addresses increasing)

(b) Minus (-)

Positioning in the reverse direction (addresses decreasing)

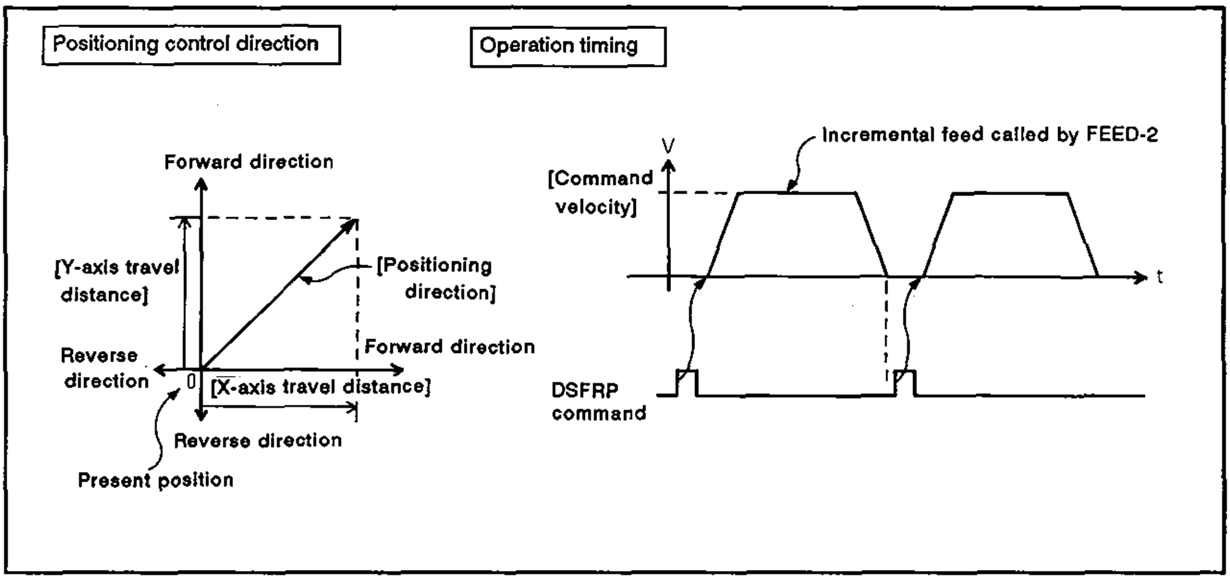


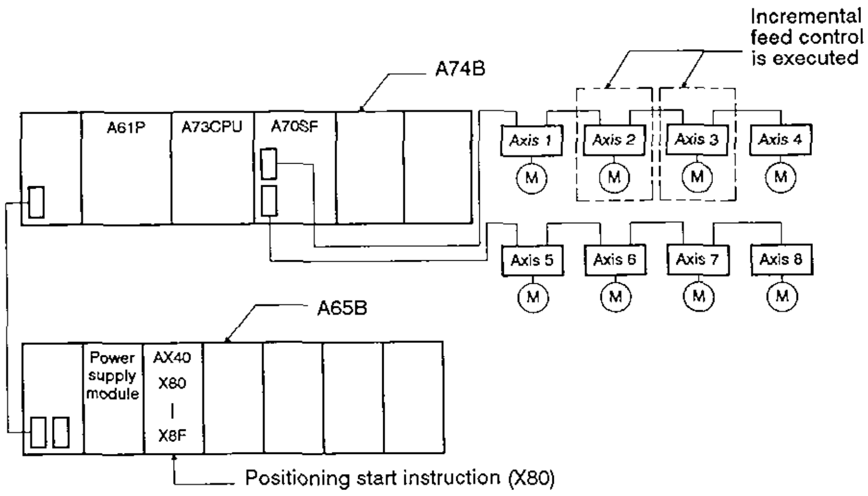
Fig. 7.19 Incremental Feed in 2-axis Linear Interpolation Mode

> **POINT**
>
> Do not set "0" for the travel distance in incremental feed.
>
> If "0" is set for incremental feed distance, the following processing will occur.
>
> (a) If the setting is "0" for both axes, incremental feed is completed before incremental feed is executed.
>
> (b) If "0" is set for either of the axes, the other axis is moved; incremental feed is, however, not carried in the correct positioning velocity.

[Program example]

The program used to execute incremental feed control in the 2-axis linear interpolation mode is explained below.

(1) System configuration



(2) Incremental feed control conditions

(a) The incremental feed control conditions are indicated below.

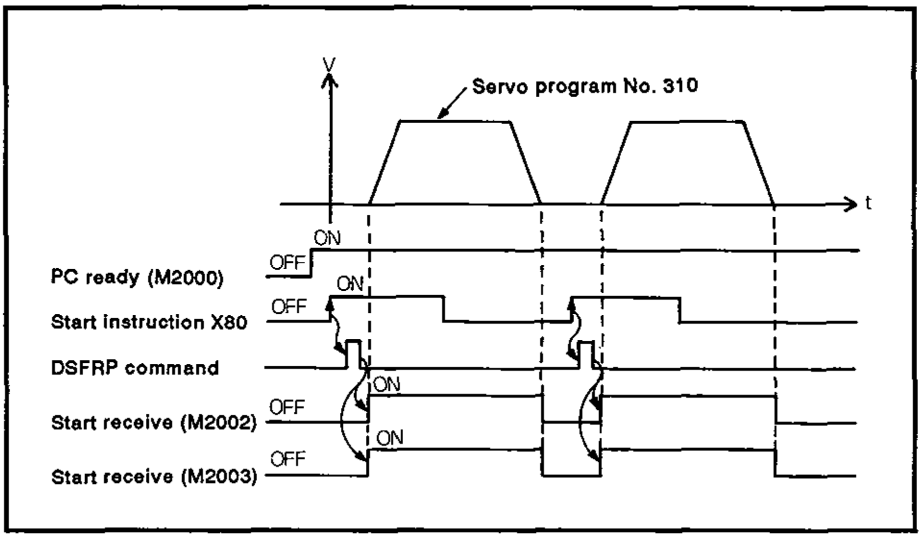
Item	Setting	
Servo program No.	No. 310	
Controlled velocity	10000	
Control axis	Axis 2	Axis 3
Axis travel distance	500000	300000

(b) Incremental feed start instruction

Leading edge (OFF → ON) of X80

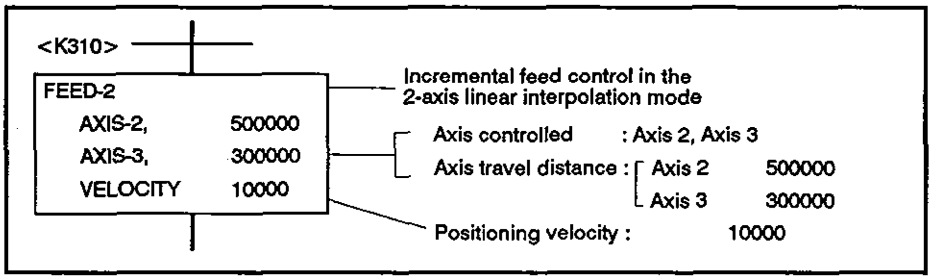
(3) Timing chart

The timing chart for incremental feed in the 2-axis linear interpolation mode called by servo program No. 310 is illustrated below.



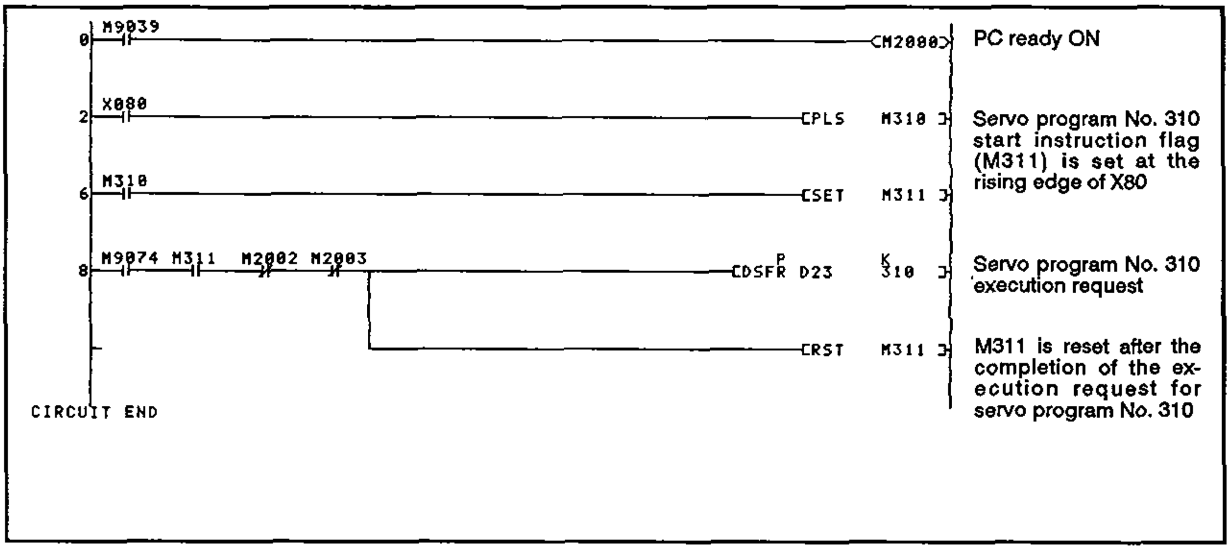
(4) Servo program example

Servo program No. 310, which calls the incremental feed control in the 2-axis linear interpolation mode, is indicated below.



(5) Sequence program example

The sequence program used to execute the servo program is indicated below.



7.10 Incremental Feed Control in Three-Axis Linear Interpolation Mode

In this control mode, incremental feed control is executed with the three axes designated by the sequence program positioning instruction from the present position to the designated position; the travel of the two axes is controlled in the two-axis linear interpolation mode.

Incremental feed control in the three-axis linear interpolation mode is called by servo command FEED-3.

Servo Command			[FEED-3]
Positioning Control Mode			Incremental
Number of axes to be controlled			3
Processing Contents	Common	Parameter block No.	Δ
		Axis	O
		Address/travel distance	O
		Command velocity	O
		Dwell time	Δ
		M code	Δ
		Torque limit	
	Circular interpolation	Assist-point	
		Radius	
		Center	
	Parameter block	Control units	Δ
		Velocity limit	Δ
		Acceleration time	Δ
		Deceleration time	Δ
		Immediate stop deceleration time	Δ
		Torque limit	Δ
		STOP input deceleration processing	Δ
		Circular interpolation allowable error range	
	Others		
Velocity Change			Possible

O: Items to be set always

Δ: Items to be set as needed

(1) Positioning control is executed from the present position, which is taken as "0", to the position defined by the travel distance and direction in the servo program.

(2) Positioning direction is determined by the plus or minus sign preceding the designated travel distance of each axis.

(a) Plus (+)

Positioning in the forward direction (addresses increasing)

(b) Minus (–)

Positioning in the reverse direction (addresses decreasing)

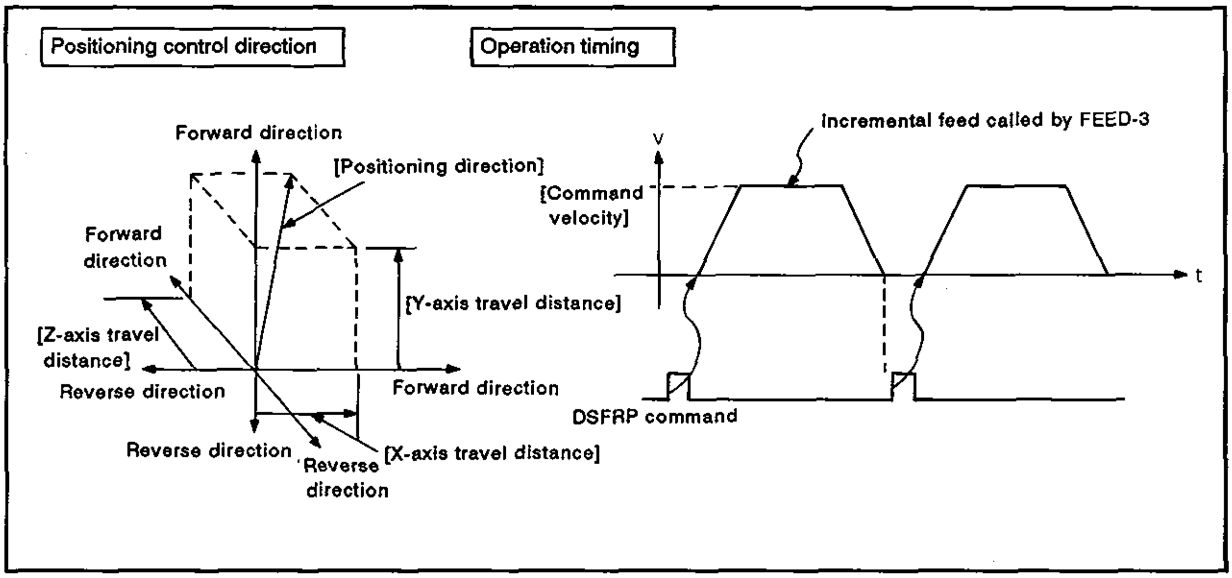


Fig. 7. 20 Incremental Feed in 3-axis Linear Interpolation Mode

POINT

Do not set "0" for the travel distance in incremental feed.

If "0" is set for incremental feed distance, processing is as indicated below:

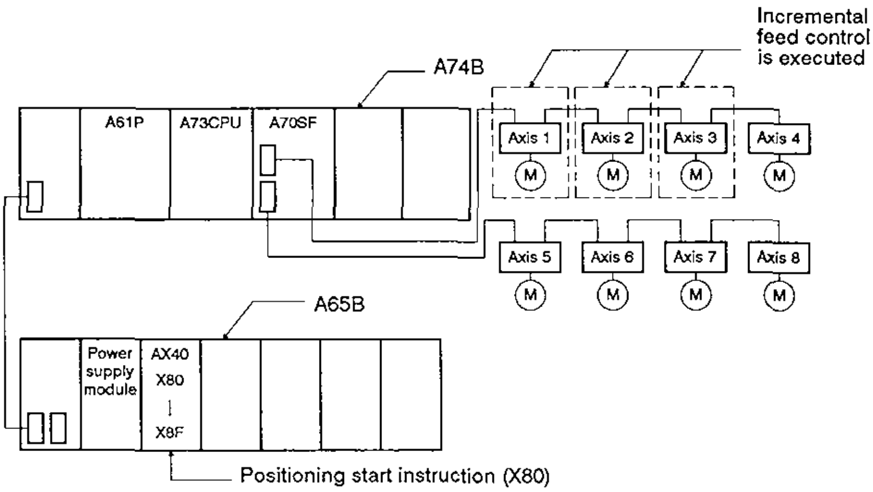
(a) If the setting is "0" for all axes, incremental feed is completed before incremental feed is executed.

(b) If "0" is set for one of the three axes, the other axes are moved; incremental feed is, however, not executed in the correct positioning velocity.

[Program example]

The program used to execute incremental feed control in the 3-axis linear interpolation mode is explained below.

(1) System configuration



(2) Incremental feed control conditions

(a) The incremental feed control conditions are indicated below.

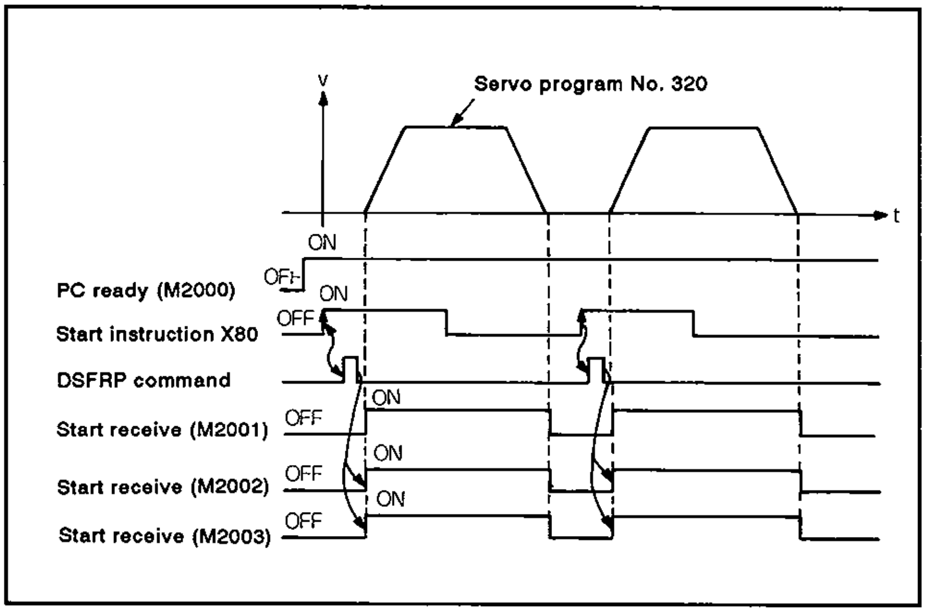
Item	Setting		
Servo program No.	No. 320		
Control velocity	1000		
Controlled axis	Axis 1	Axis 2	Axis 3
Axis travel distance	50000	40000	30000

(b) Incremental feed start instruction

Leading edge (OFF → ON) of X80

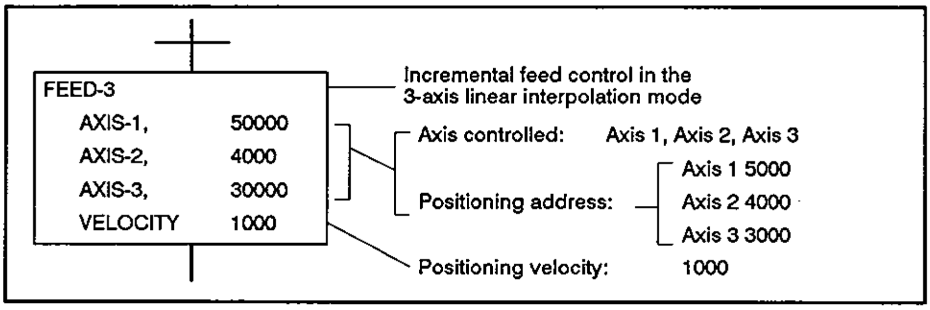
(3) Timing chart

The timing chart for incremental feed in the 3-axis linear interpolation mode called by servo program No. 320 is illustrated below.



(4) Servo program example

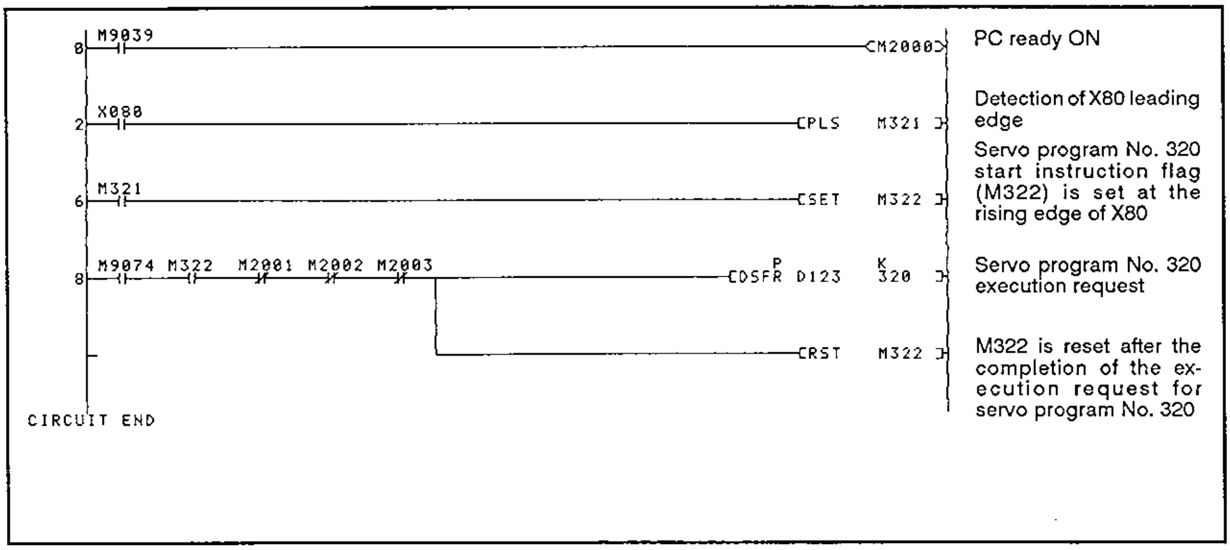
Servo program No. 320, which calls the incremental feed control in the 3-axis linear interpolation mode, is indicated below.



IB (NA) 66233-A

(5) Sequence program example

The sequence program used to execute the servo program is indicated below.



7.11 Velocity Control

The velocity of the axis designated by the sequence program is controlled.

Velocity control is executed by designating the servo command [VF (forward) and VR (reverse)].

Servo Command			[VF]	[VR]
Positioning Control Mode			—	
Number of axes to be controlled			1	
Processing Contents	Common	Parameter block No.	Δ	
		Axis	O	
		Address/travel distance		
		Command velocity	O	
		Dwell time		
		M code	Δ	
		Torque limit		
	Circular interpolation	Assist-point		
		Radius		
		Center		
	Parameter block	Control units	Δ	
		Velocity limit	Δ	
		Acceleration time	Δ	
		Deceleration time	Δ	
		Immediate stop deceleration time	Δ	
		Torque limit	Δ	
		STOP input deceleration processing	Δ	
		Circular interpolation allowable error range		
	Others			
Velocity Change			Possible	

O: Items to be set always

Δ: Items to be set as needed

[Control details]

(1) After motor rotation is started, the motor is controlled at the designated velocity until a stop instruction is input.

(a) VF ..... Start of forward motion

(b) VR ..... Start of reverse motion

(2) The present value remains "0".

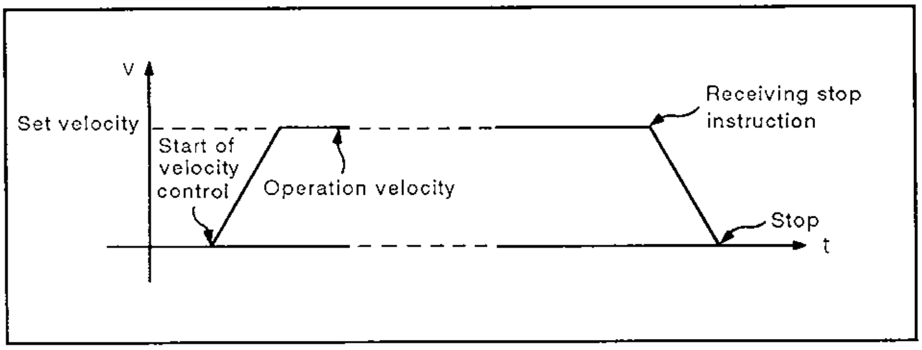


Fig. 7.21 Velocity Control

(3) Stop instruction and stop processing

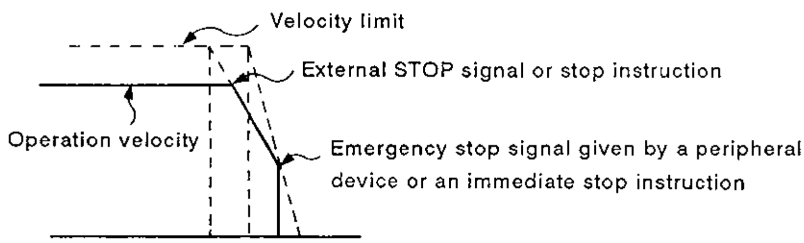
The stop instructions used in the velocity control mode and the corresponding stop processing are summarized in the table below.

Table 7.1 Stop Instructions and Stop Processing

Stop Instruction	Stop Condition	Axis	Stop Processing
External stop signal	OFF → ON	Designated axis	Deceleration and stop controlled by the "stop processing when stop signal is input" set in the parameter block.
Stop instruction (Yn0)			Deceleration and stop controlled by the "stop deceleration time" set in the parameter block.
Immediate stop * instruction (Yn1)			Deceleration and stop controlled by the "immediate stop deceleration time" set in the parameter block.
Emergency stop* signal from a peripheral device	Key input	All axes	Deceleration and stop controlled by the "immediate stop deceleration time" set in the parameter block.

POINT

* These instructions are effective during deceleration due to an external STOP signal or stop instruction (Yn0). The deceleration processing is carried out according to the "immediate stop deceleration time" when the stop condition is received.



[Caution]

(1) With the absolute value system, the command present position data is not reset to "0" after the execution of the velocity control if the following operation is executed.

(a) Resetting with the RUN key switch

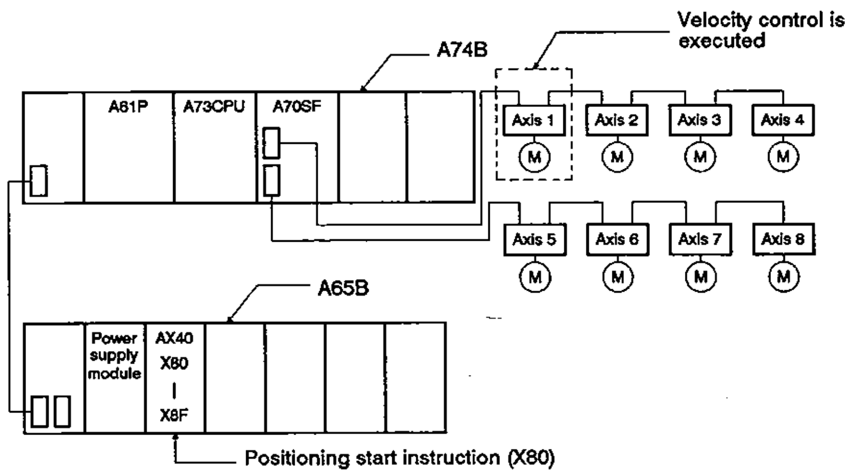
(b) Turning the servo power on

(2) It is not possible to set the dwell time.

[Program example]

The servo program that calls the velocity control mode is explained below.

(1) System configuration



(2) Velocity control conditions

(a) Velocity control conditions are indicated below.

Item	Setting
Servo program No.	No. 91
Control axis	Axis 1
Control velocity	3000
Rotation direction	Forward

(b) Velocity control start instruction

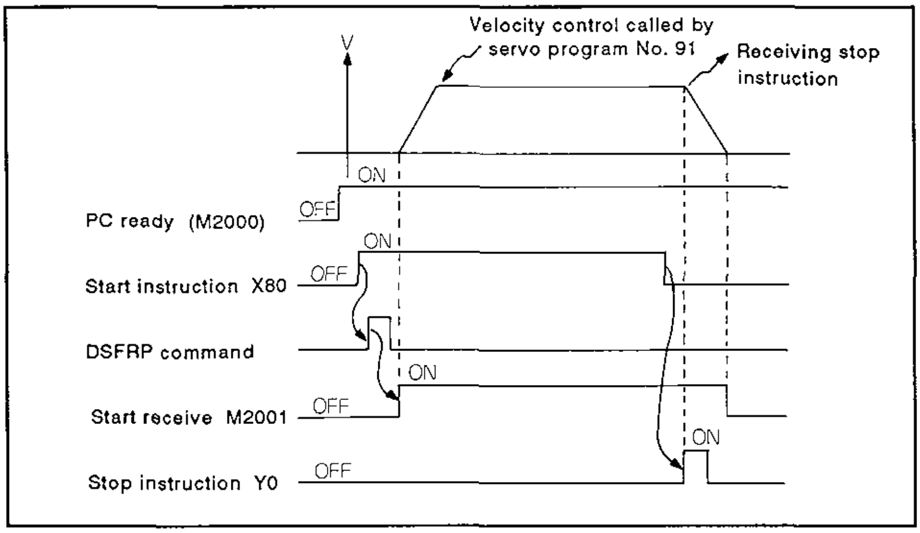
Leading edge (OFF → ON) of X80

(c) Stop instruction

Trailing edge (ON → OFF) of X80

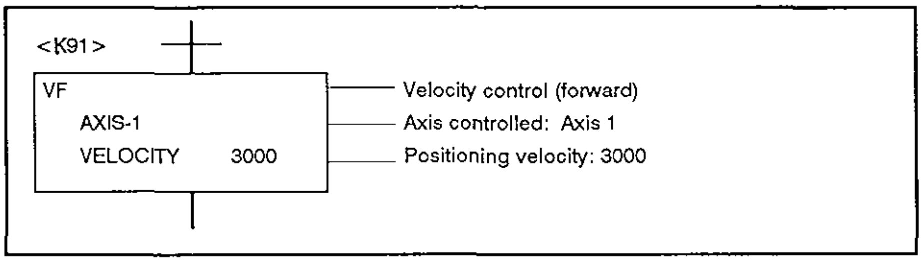
(3) Timing chart

The timing chart for velocity control called by the servo program No. 91 is indicated below.



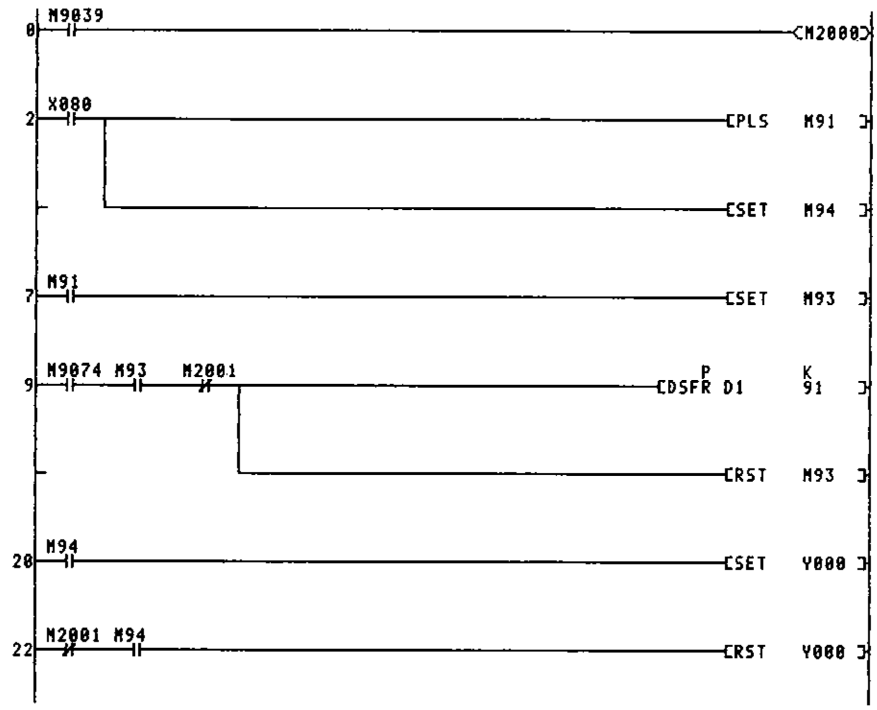
(4) Servo program

Servo program No. 91, which calls the velocity control, is indicated below.



(5) Sequence program

The sequence program used to execute the servo program is indicated below.



PC ready (M2000) ON

Detection of X80 leading edge

Detection of X80 trailing edge

Servo program No. 91 start instruction flag (M93) is set at the leading edge of X80.

Servo program No. 91 execution request

M93 is reset after the completion of the execution request for servo program No. 91.

Stop instruction Y00 (Yn0) is turned on at the trailing edge of X80.

The stop instruction (Y00) is turned off when Axis 1 stops.

7.12 Velocity/Position Switchover (V/P) Control

7.12.1 Starting V/P control

The control mode is changed from velocity control to position control for the axis designated by the sequence program positioning instruction.

V/P control is called by the following servo commands: [VPF (forward rotation), VPR (reverse rotation), VPSTART (re-start)].

Servo Command			[VPF]	[VPR]
Positioning Control Mode			Incremental	
Number of axes to be controlled			1	
Processing Contents	Common	Parameter block No.	Δ	
		Axis	O	
		Address/travel distance	O	
		Command velocity	O	
		Dwell time	Δ	
		M code	Δ	
		Torque limit	Δ	
	Circular interpolation	Assist-point		
		Radius		
		Center		
	Parameter block	Control units		
		Velocity limit	Δ	
		Acceleration time	Δ	
		Deceleration time	Δ	
		Immediate stop deceleration time	Δ	
		Torque limit	Δ	
		STOP input deceleration processing	Δ	
		Circular interpolation allowable error range		
	Others			
Velocity Change			Possible	

O: Items to be set always

Δ: Items to be set as needed

[Control details]

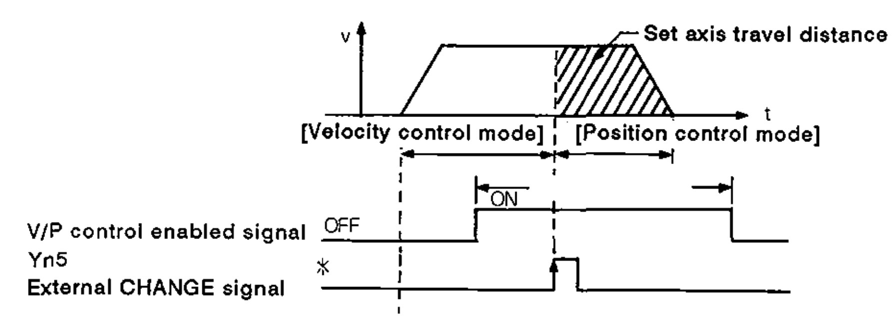
(1) Just after the servo motor starts, the control starts in the velocity control mode; the control mode is then changed to position control when the external CHANGE signal is received. After the control mode is changed to position control, positioning is executed to move the axis the designated distance.

(a) VPF ..... Start in the forward direction (addresses increasing)

(b) VPR ..... Start in the forward direction (addresses decreasing)

(2) The external CHANGE signal is effective while Yn5 (V/P control enabled) is on.

If Yn5 is turned on after the CHANGE signal is turned on, the control mode remains in as velocity control.



(3) The command position data is updated in the velocity control mode.

The command position data is calculated the after the stop as indicated below.

(Command position data after the stop) = (Address before the start of velocity control) + (Axis travel distance during velocity control) + (Axis travel distance during position control)

REMARKS

The external CHANGE signal is the signal input to the DOG/CHANGE terminal on the A70SF. For details, refer to the A73CPU User's Manual.

(4) Changing the set axis travel distance during velocity control

After starting the V/P control mode, it is possible to change the set axis travel distance in the position control mode. Follow the steps below.

(a) The axis travel distance is indirectly set with the servo program by the data registers (2 word register) as indicated below.

Axis No.	Data Register No. for Indirect Designation	Data Register for Changing Axis Travel Distance		Axis No.	Data Register No. for Indirect Designation	Data Register for Changing Axis Travel Distance	
		Upper Digit Data	Lower Digit Data			Upper Digit Data	Lower Digit Data
1	D815	D816	D815	5	D895	D896	D895
2	D835	D836	D835	6	D915	D916	D915
3	D855	D856	D855	7	D935	D936	D935
4	D875	D876	D875	8	D955	D956	D955

Example:

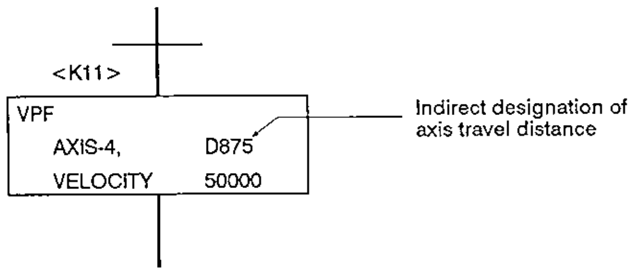
The servo program which calls the following control is shown below:

Axis controlled : Axis 4,

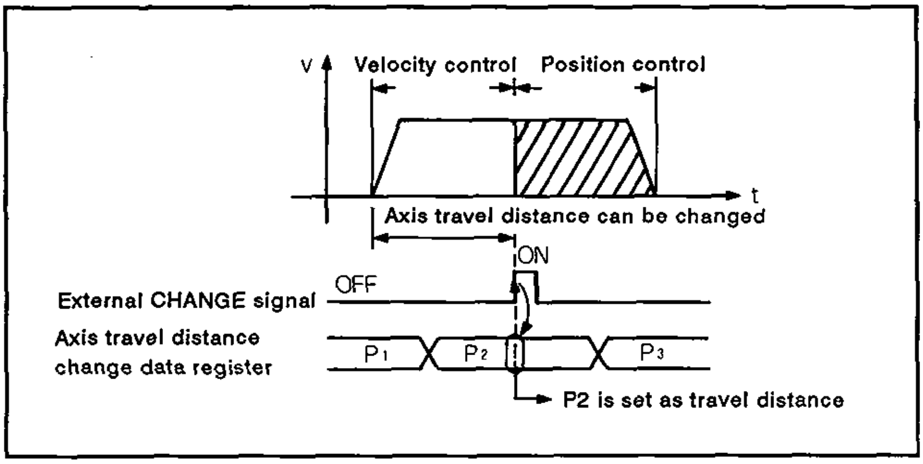
Velocity control : Velocity of 50000 in the forward direction

Position control : Axis travel distances set for D875 and D876

The control starts in the velocity control mode and is changed to the position control when the external CHANGE signal is turned on.



(b) The sequence program stores the axis travel distance to the travel distance change data registers while velocity control is executed. The axis travel distance stored in the designated axis travel distance change data register is set as the axis travel distance for position control when the external CHANGE signal is turned on.



(5) Axis travel distance data storage area

The distance that an axis has moved in the position control mode is stored in the area storing travelling distance after near-zero point signal is turned on. Refer to Section 3.4.1.

[Caution]

(1) Items checked when the external CHANGE signal is turned on

The control mode will only be changed from velocity control to position control when the CHANGE signal is turned on when all of the following conditions are met.

(a) Start receive flag (M200n) is set.

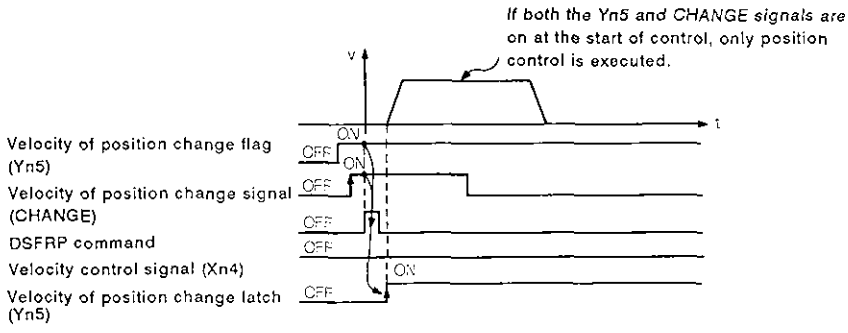
(b) Started in the V/P control mode and the current control is executed in the velocity control mode.

(c) The V/P control enabled signal (Yn5) is on.

(2) Cases where velocity control is skipped

If both the Yn5 and CHANGE signals are on when the control is started, only position control is executed.

In this case, the velocity control signal (Yn4) is not turned on.



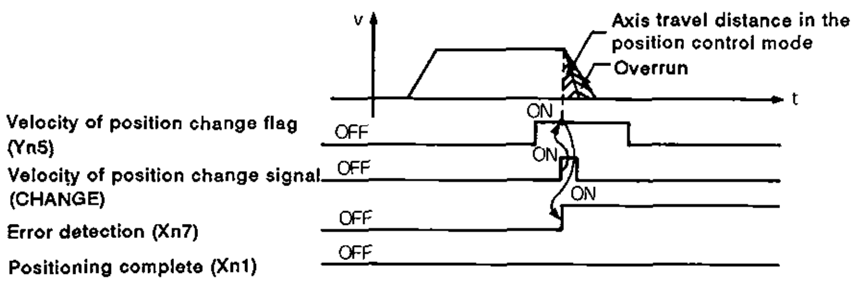
(3) If "set travel distance in position control" "deceleration distance".

(a) If the axis travel distance is smaller than the distance required for deceleration for the current velocity, deceleration processing is executed at the point where the CHANGE signal is input.

(b) The difference between the distance travelled to "decelerate to stop" and the axis travel distance in the position control mode is the overrun distance.

In this case, the error detection signal (Xn7) is turned on and the error code (209) is stored in the data register.

(c) The positioning complete signal (Xn1) is not turned on.



(4) Stroke limit check

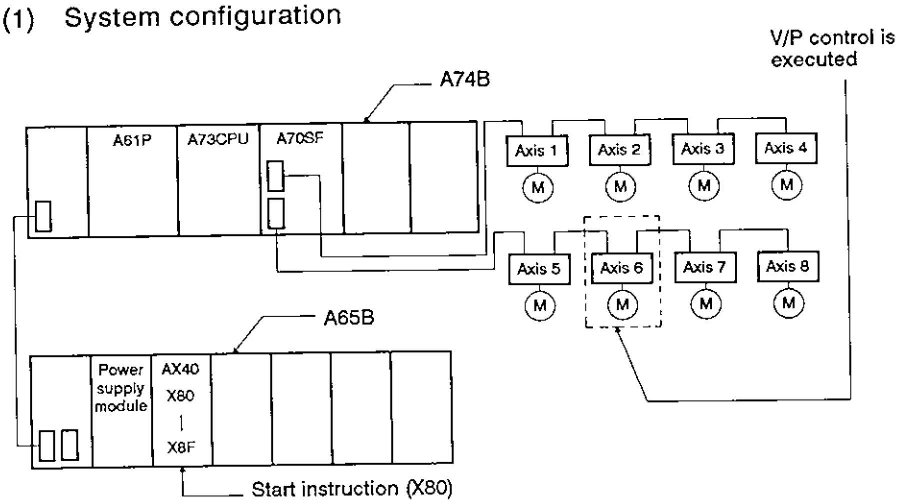
While the control is in the velocity control mode, stroke limit is not checked.

If the axis travel distance exceeds the stroke limit, a minor error (error code: 210) occurs when the control mode is changed to position control and the axis decelerates and stops.

[Program example]

The program which calls the V/P control is explained below.

(1) System configuration



(2) Positioning conditions

(a) The positioning conditions are indicated below.

Item	Setting
Servo program No.	101
Control axis	Axis 6
Axis travel distance in position control mode	40000
Control velocity	1000

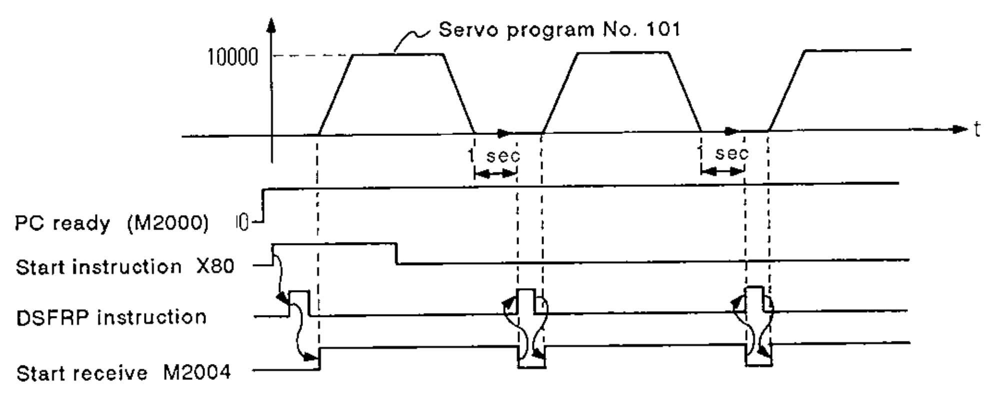
(b) Start instruction of position control

Leading edge (OFF → ON) of X80

(c) Velocity/position control change enabled flag ... Y55

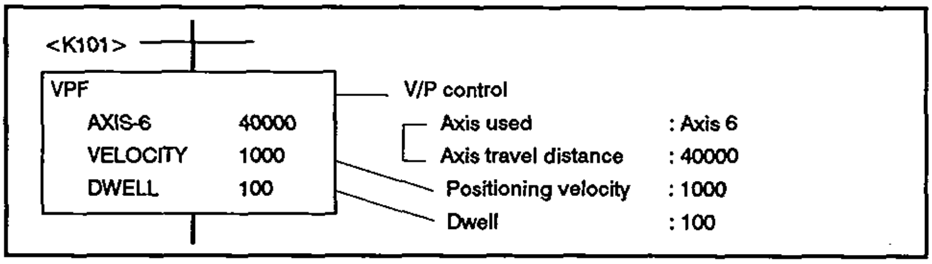
(3) Timing chart

The timing for the control mode to change from velocity to position control is indicated below.



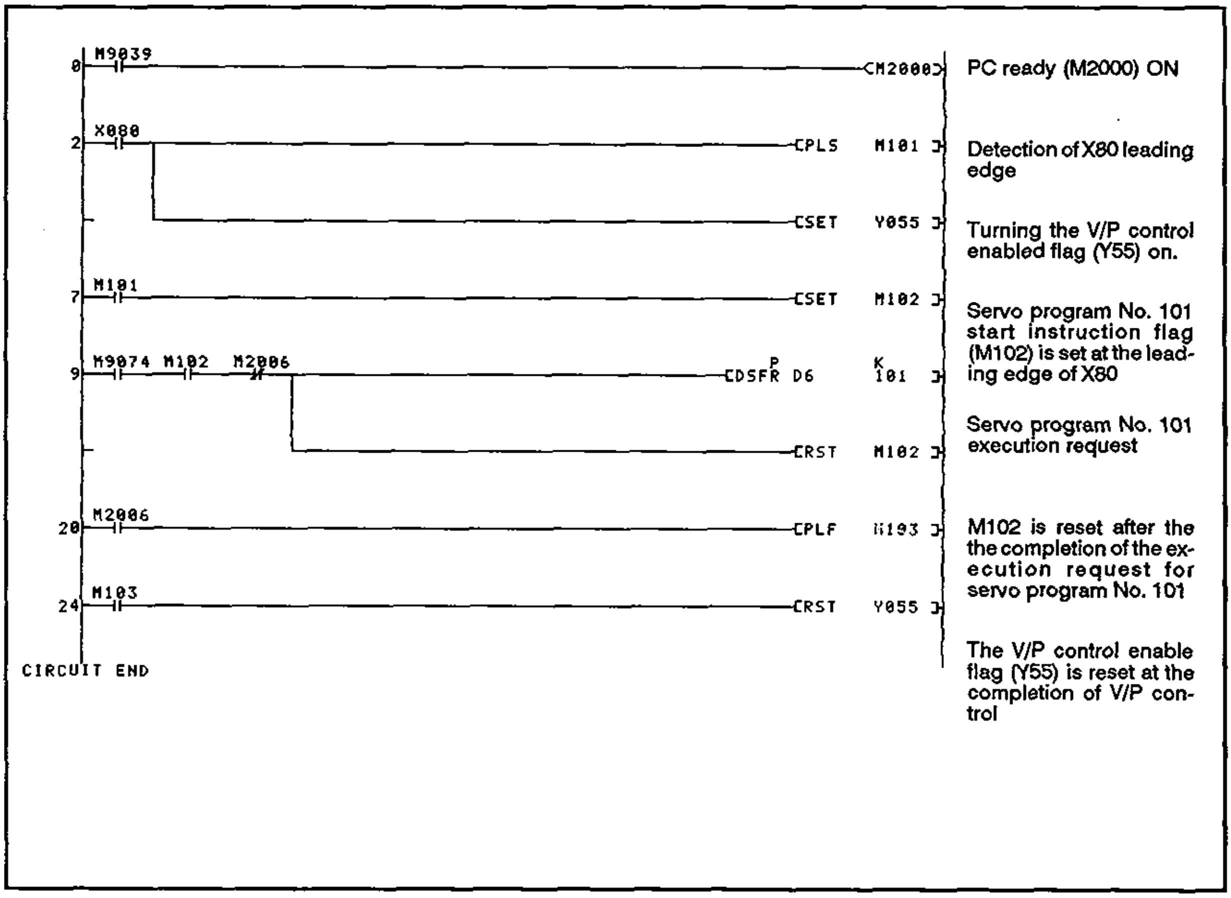
(4) Servo program

Servo program No. 101, which calls the V/P control, is indicated below.



(5) Sequence program

The sequence program used to execute the servo program is indicated below.



7.12.2 Restart after stopping before completion

Restart of the control is possible after stopping an axis with the stop instruction during V/P control.

For restart, servo instruction VPSTART is used.

Servo Command			[VPSTART]
Positioning Control Mode			
Number of axes to be controlled			
Processing Contents	Common	Parameter block No.	
		Axis	O
		Address/travel distance	
		Command velocity	
		Dwell time	
		M code	
		Torque limit	
	Circular interpola-tion	Assist-point	
		Radius	
		Center	
	Parameter block	Control units	
		Velocity limit	
		Acceleration time	
		Deceleration time	
		Immediate stop deceleration time	
		Torque limit	
		STOP input deceleration processing	
		Circular interpolation allowable error range	
	Others		
Velocity change			

O: Items to be set always

Δ: Items to be set as needed

[Control details]

(1) Control stopped before completion of the V/P control can be restarted.

(2) Restart control using the VPSTART command is possible regardless of when the control is stopped, during velocity control or position control.

(a) If the control has been stopped before completion of the velocity control mode, the control is restarted in the velocity control mode; when the external CHANGE signal is turned on, the control mode is changed to position control.

(The control contents after restart are the same as those of the V/P control. For details, refer to Section 7.11.1.)

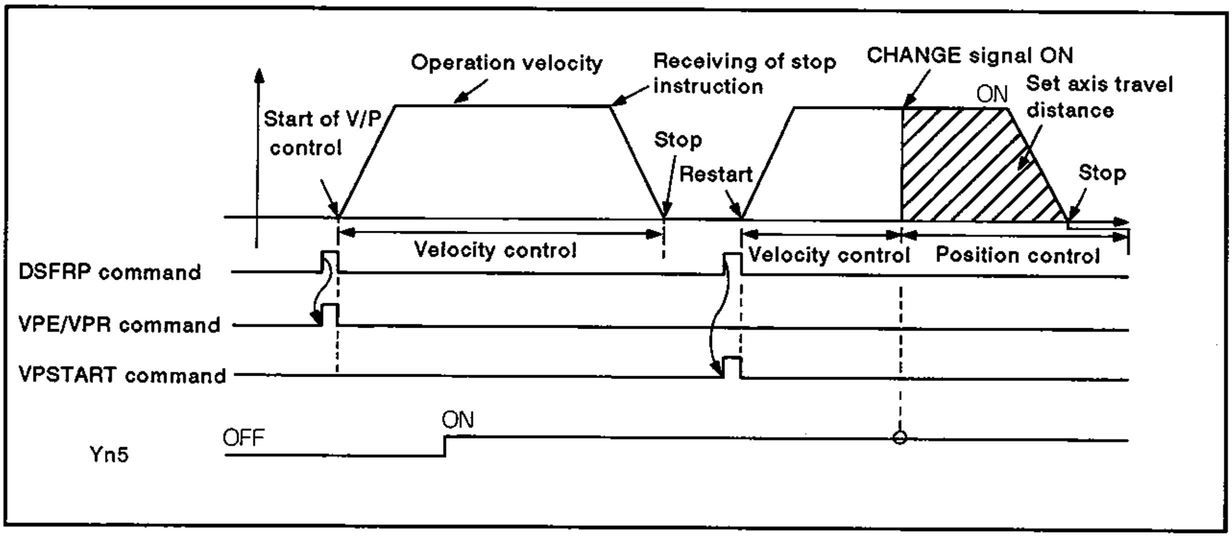


Fig. 7.22 Restart in the Velocity Control Mode

(b) If the control has been stopped before completion in the position control mode, the control is restarted in the position control mode; positioning at the point defined by the set axis travel distance is executed.

Axis travel distance after restart is indicated below:

$$\left[\text{Axis travel distance after restart (P2)}\right] = \left[\text{Set axis travel distance (P)}\right] - \left[\text{Axis travel distance prior to stop (P1)}\right]$$

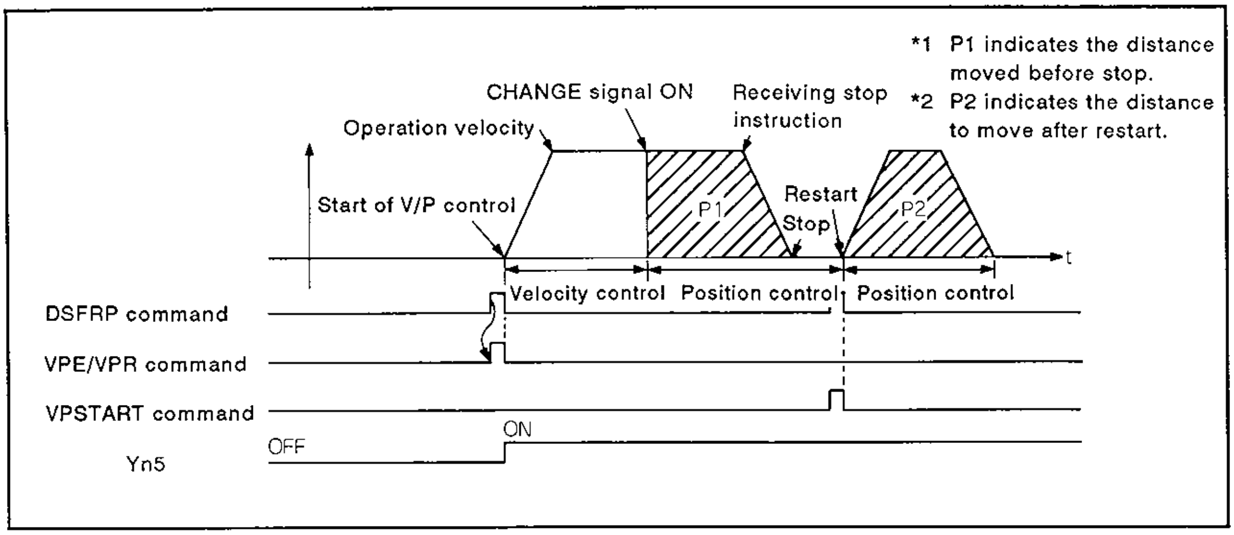


Fig. 7.23 Restart in the Position Control Mode

(3) When the control is restarted, an axis is controlled at the velocity registered when the VPF/VPR command is executed.

Therefore, if the velocity is changed after the start of the V/P control and before the stop, the control is restarted at the velocity registered when the VPF/VPR command is executed.

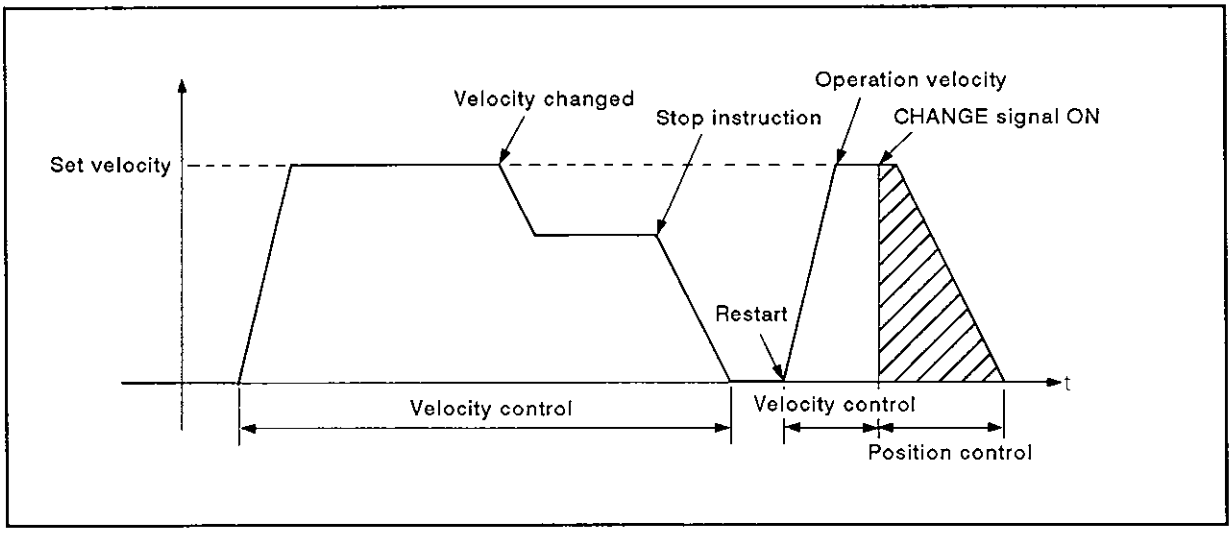
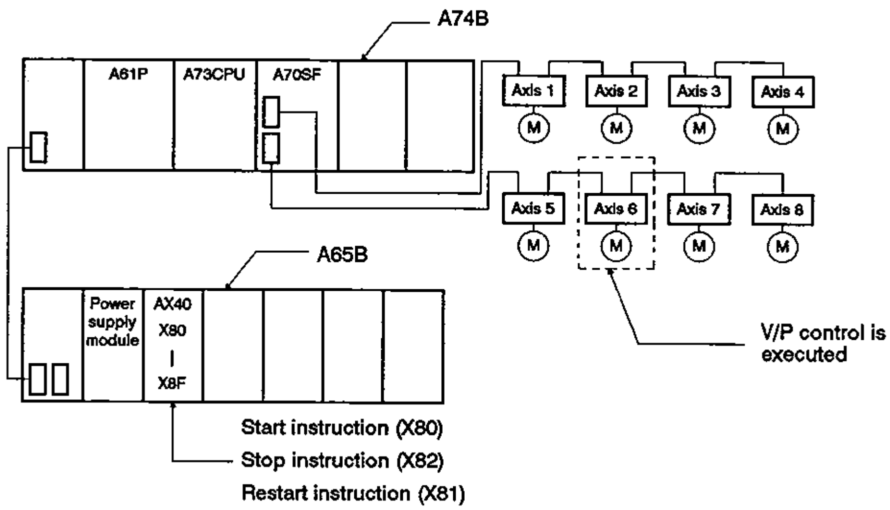


Fig. 7.24 Restart After Changing Velocity

[Program example]

The program that calls the restart after stop in V/P control is explained below.

(1) System configuration



(2) Positioning conditions

(a) Positioning conditions are indicated below.

Item	Positioning Conditions	
	V/P control	Restart
Servo program No.	101	102
Control axis	6	6
Axis travel distance in position control	40000	—
Control velocity	1000	—

(b) Positioning start

Leading edge (OFF → ON) of X80

(c) V/P control enable flag

Y55

(d) Restart

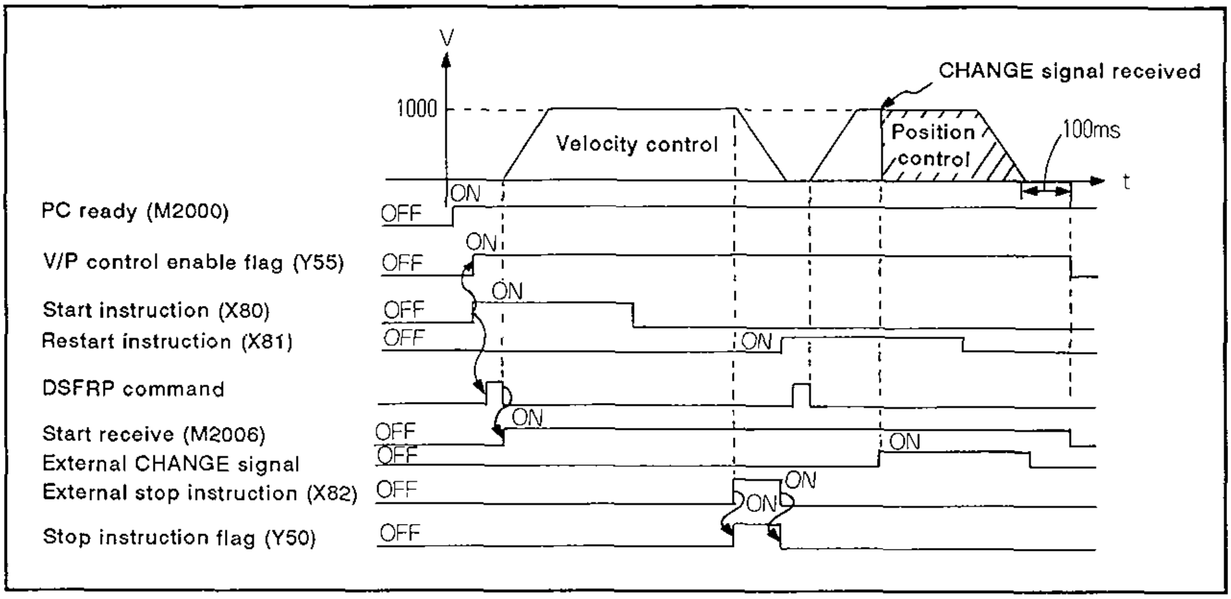
Leading edge (OFF → ON) of X81

(e) Stop instruction

Leading edge (OFF → ON) of X82

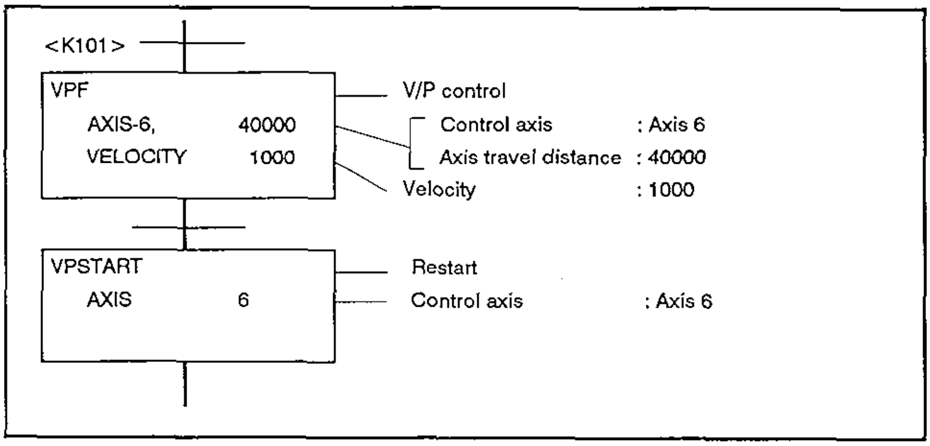
(3) Timing chart

The operation timing of V/P control and restart is illustrated below.



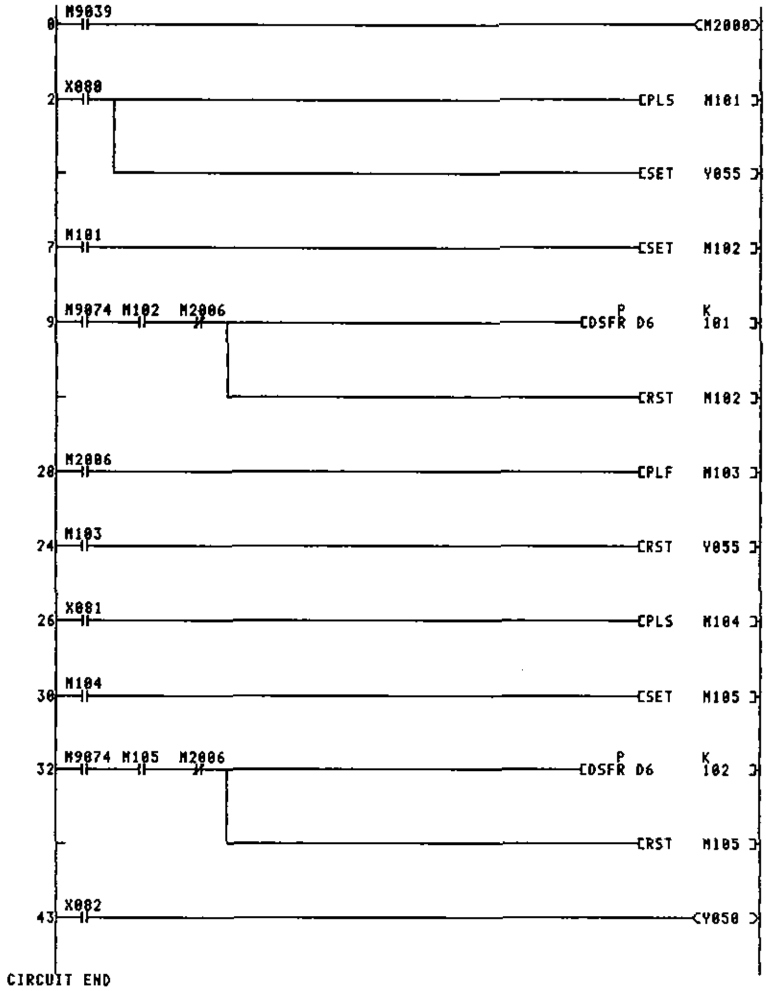
(4) Servo program

Servo programs No. 101 and No. 102, which call the V/P control and restart control respectively, are explained below.



(5) Sequence program

The sequence program used to execute the servo programs is indicated below.



PC ready (M2000) ON

Detection of X80 leading edge

Turning the V/P control enable flag (Y55) on.

Servo program No. 101 start instruction flag (M102) is set at the leading edge of X80.

Servo program No. 101 execution request

M102 is reset after the completion of the execution request for servo program No. 101.

The V/P control enable flag (Y55) is reset at the completion of V/P control.

Detection of X81 leading edge

Servo program No. 102 start instruction flag (M105) is set at the leading edge of X81.

Servo program No. 102 execution request

M105 is reset after the completion of the execution request for servo program No. 102.

Axis 6 stops when external stop instruction is turned on.

7.13 Velocity Change Control

(1) With a single start signal, positioning is executed while changing the velocity at preset points.

(2) The velocity change points and corresponding velocity are set by the servo program.

(3) By using the repeat command, it is possible to repeat the control between the required velocity change points.

(4) The M code and torque limit can be changed at each velocity change point.

7.13.1 Designation of velocity change control start, velocity change points, and end of control

<table>
<tr><th colspan="3" rowspan="2">Servo Command</th><th>Start</th><th>End</th><th colspan="3">End Point Address</th><th colspan="3">Travel Distance to End Point</th><th colspan="2">Velocity Change Point</th></tr>
<tr><th>VSTART</th><th>VEND</th><th>ABS-1</th><th>ABS-2</th><th>ABS-3</th><th>INC-1</th><th>INC-2</th><th>INC-3</th><th>VABS</th><th>VINC</th></tr>
<tr><td colspan="3">Positioning Control Mode</td><td colspan="2">—</td><td colspan="3">Absolute</td><td colspan="3">Incremental</td><td>Ab-solute</td><td>In-cremen-tal</td></tr>
<tr><td colspan="3">Number of axes to be controlled</td><td colspan="2">—</td><td>1</td><td>2</td><td>3</td><td>1</td><td>2</td><td>3</td><td colspan="2">—</td></tr>
<tr><td rowspan="21">Processing Contents</td><td rowspan="7">Common</td><td>Parameter block No.</td><td>Δ</td><td></td><td colspan="6"></td><td colspan="2"></td></tr>
<tr><td>Axis</td><td></td><td></td><td colspan="6">O</td><td colspan="2"></td></tr>
<tr><td>Address/travel distance</td><td></td><td></td><td colspan="6">O</td><td colspan="2">O</td></tr>
<tr><td>Command velocity</td><td></td><td></td><td colspan="6">O</td><td colspan="2">O</td></tr>
<tr><td>Dwell time</td><td></td><td></td><td colspan="6">Δ</td><td colspan="2"></td></tr>
<tr><td>M code</td><td></td><td></td><td colspan="6">Δ</td><td colspan="2">Δ</td></tr>
<tr><td>Torque limit</td><td></td><td></td><td colspan="6">Δ</td><td colspan="2">Δ</td></tr>
<tr><td rowspan="3">Circular interpola-tion</td><td>Assist-point</td><td></td><td></td><td colspan="6"></td><td colspan="2"></td></tr>
<tr><td>Radius</td><td></td><td></td><td colspan="6"></td><td colspan="2"></td></tr>
<tr><td>Center</td><td></td><td></td><td colspan="6"></td><td colspan="2"></td></tr>
<tr><td rowspan="8">Parameter block</td><td>Control units</td><td>Δ</td><td></td><td colspan="6"></td><td colspan="2"></td></tr>
<tr><td>Velocity limit</td><td>Δ</td><td></td><td colspan="6"></td><td colspan="2"></td></tr>
<tr><td>Acceleration time</td><td>Δ</td><td></td><td colspan="6"></td><td colspan="2"></td></tr>
<tr><td>Deceleration time</td><td>Δ</td><td></td><td colspan="6"></td><td colspan="2"></td></tr>
<tr><td>Immediate stop decelera-tion time</td><td>Δ</td><td></td><td colspan="6"></td><td colspan="2"></td></tr>
<tr><td>Torque limit</td><td>Δ</td><td></td><td colspan="6"></td><td colspan="2"></td></tr>
<tr><td>STOP input deceleration processing</td><td>Δ</td><td></td><td colspan="6"></td><td colspan="2"></td></tr>
<tr><td>Circular interpolation al-lowable error range</td><td></td><td></td><td colspan="6"></td><td colspan="2"></td></tr>
<tr><td rowspan="3">Others</td><td></td><td></td><td></td><td colspan="6"></td><td colspan="2"></td></tr>
<tr><td></td><td></td><td></td><td colspan="6"></td><td colspan="2"></td></tr>
<tr><td></td><td></td><td></td><td colspan="6"></td><td colspan="2"></td></tr>
<tr><td colspan="3">Velocity change</td><td colspan="2">—</td><td colspan="6">Possible</td><td colspan="2">—</td></tr>
</table>

O: Items to be set always

Δ: Items to be set as needed

[Control details]

Start and end of velocity change control

The following commands are used to start and end velocity change control.

(1) VSTART

Starts the velocity change control.

(2) VEND

Ends the velocity change control.

Setting end point address and axis travel distance to the end point

The following commands are used to designate the end point address, motion distance to the end point, positioning control mode, and velocity to the end point.

(1) ABS-1/INC-1

Designates the one-axis straight line positioning control.

For details, refer to Section 7.2.

(2) ABS-2/INC-2

Designates the two-axis linear interpolation control.

For details, refer to Section 7.3.

(3) ABS-3/INC-3

Designates the three-axis linear interpolation control.

For details, refer to Section 7.4.

Setting the velocity change point

The following commands are used to designate the velocity change points (address or axis travel distance) and velocity.

(1) VABS

Designates the velocity change points in the absolute mode.

(2) VINC

Designates the velocity change points in the incremental mode.

Programming servo program and operation timing

The servo program which calls the velocity change control and the operation timing are shown in Fig. 7.25.

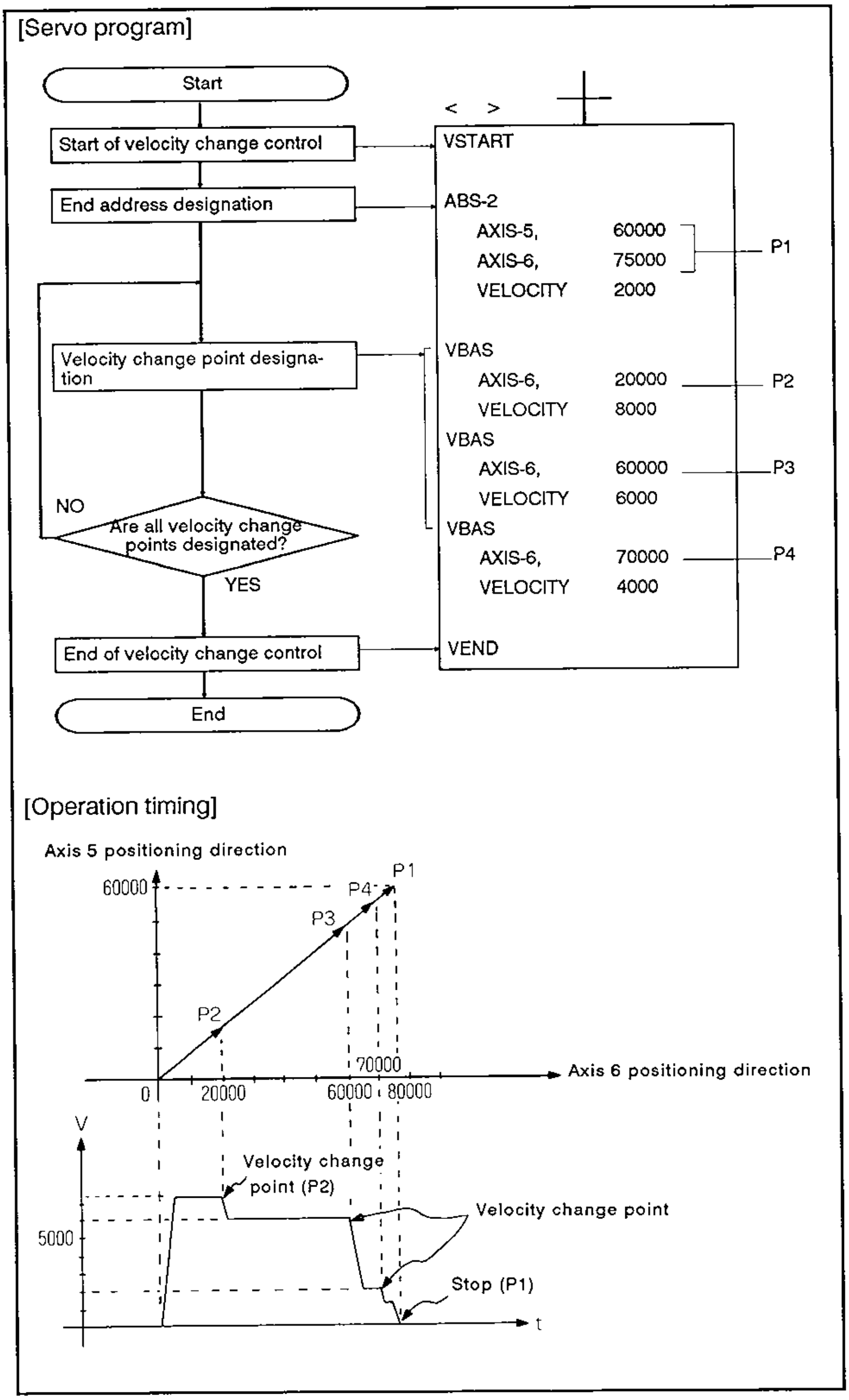


Fig. 7.25 Servo Program for Velocity Change Control and Operation

[Caution]

(1) The number of controlled axes cannot be changed before completion.

(2) For the positioning control mode up to the velocity change point, absolute (ABS[ ]) and incremental (INC[ ]) can be used in combination.

(3) For the velocity change points, it is not possible to designate an address that causes the axis travel direction to change.

If the axis travel direction changes, an error code (215) is registered in the minor error storage register for each axis; the axis decelerates and stops.

(4) The maximum number of steps that can be designated in a single velocity change control program is 768 steps which are approximately 100 points.

(5) At the start of the control, a stroke check is executed.

If the end point address is beyond the stroke limit, an error code (106) is registered in the minor error storage register for each axis; control does not start.

(6) If the distance between the two velocity change points is short and the axis reaches the next velocity change point while the velocity is being changed, the velocity will not be changed.

(7) The M code is modal. Therefore, if an M code is not designated at a velocity change point, the M code effective at the previous point remains effective.

[Program example]

(1) Servo program

Servo program No. 500 which calls the velocity change control is explained below.

```
<K500>
VSTART
ABS-2
    AXIS-2,      100000
    AXIS-3,       50000
    VELOCITY      10000
VBAS
    AXIS-2,       40000
    VELOCITY       8000
VBAS
    AXIS-2,       70000
    VELOCITY       5000
VEND
```

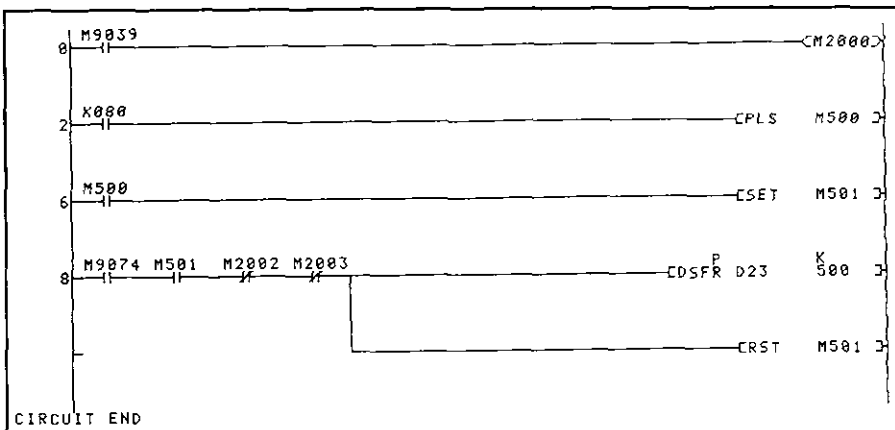
- VSTART: Start of velocity change control
- ABS-2: 2-axis linear interpolation control (absolute)
  - Control axis: Axis 2 and Axis 3
  - End address: Axis 2 100000, Axis 3 50000
- VELOCITY: Positioning velocity
- VBAS: Setting of velocity change point and velocity

Axis No.	Axis 2	
Velocity change point	40000	70000
Velocity up to the velocity change point	8000	5000

- VEND: End of velocity change control

(2) Sequence program

The sequence program used to execute the servo program is indicated below.



PC ready (M2000) ON

Detection of X80 leading edge

Servo program No. 500 start instruction flag (M501) is set at the leading edge of X80.

Servo program No. 500 execution request

M501 is reset after the completion of the execution request for servo program No. 500.

7.13.2 Designating velocity changeover points with a repeat command

Servo Command			[FOR-TIMES]	[FOR-ON]	[FOR-OFF]	[NEXT]
Positioning Control Mode			—			—
Number of axes to be controlled			—			—
Processing Contents	Common	Parameter block No.				
		Axis				
		Address/travel distance				
		Command velocity				
		Dwell time				
		M code				
		Torque limit				
	Circular interpolation	Assist-point				
		Radius				
		Center				
	Parameter block	Control units				
		Velocity limit				
		Acceleration time				
		Deceleration time				
		Immediate stop deceleration time				
		Torque limit				
		STOP input deceleration processing				
		Circular interpolation allowable error range				
	Others	Repeat condition	O			
Velocity Change			—			

O: Items to be set always
Δ: Items to be set as needed

[Control details]

Setting the repeat range head address

The following commands are used to designate the repeat range head address.

(1) FOR-TIMES (setting loop-out count)

(a) The set repeat range is executed repeatedly for the set number of times.

(b) The setting range is between 1 and 32767.

If the set data is outside the allowable setting range (-32768 to 0), it is regarded as "1".

(c) The devices which can be used to set the repeat count are indicated below:

1) Data register (D)

2) Link register (W)   For indirect setting

3) *Decimal constant* (K)

4) Hexadecimal constant (H)

(2) FOR-ON (setting loop-out trigger conditions)

(a) The designated range is executed repeatedly until the designated bit device is turned on.

(b) The devices that can be used as the loop-out trigger condition are indicated below:

1) Input (X)

2) Output (Y)

3) Internal relay (M) / special relay (SP.M)

4) Latch relay (L)

5) Link relay (B)

6) Annunciator (F)

(3) FOR-OFF (setting loop-out trigger conditions)

(a) The designated range is repeatedly executed until the designated bit device is turned off.

(b) The devices which can be used as the loop-out trigger condition are indicated below:

1) Input (X)

2) Output (Y)

3) Internal relay (M) / special relay (SP.M)

4) Latch relay (L)

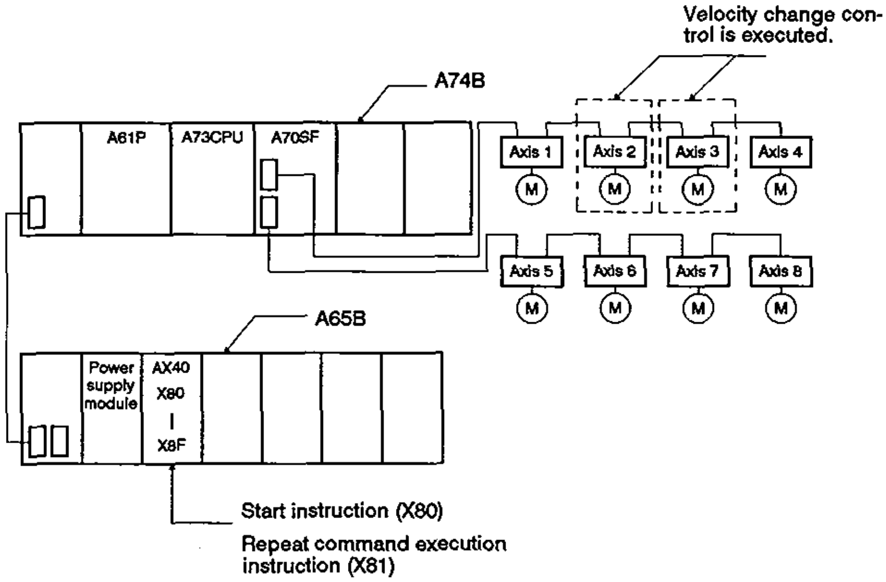
5) Link relay (B)

6) Annunciator (F)

[Program example]

The program for executing the same processing repeatedly in the velocity change control mode is explained below.

(1) System configuration



(2) Positioning conditions

(a) Velocity change conditions are indicated below.

Item	Setting	
Servo program No.	501	
Control axes	Axis 2	Axis 3
Axis travel distance up to stop position	230000	100000

(b) Start of velocity change control

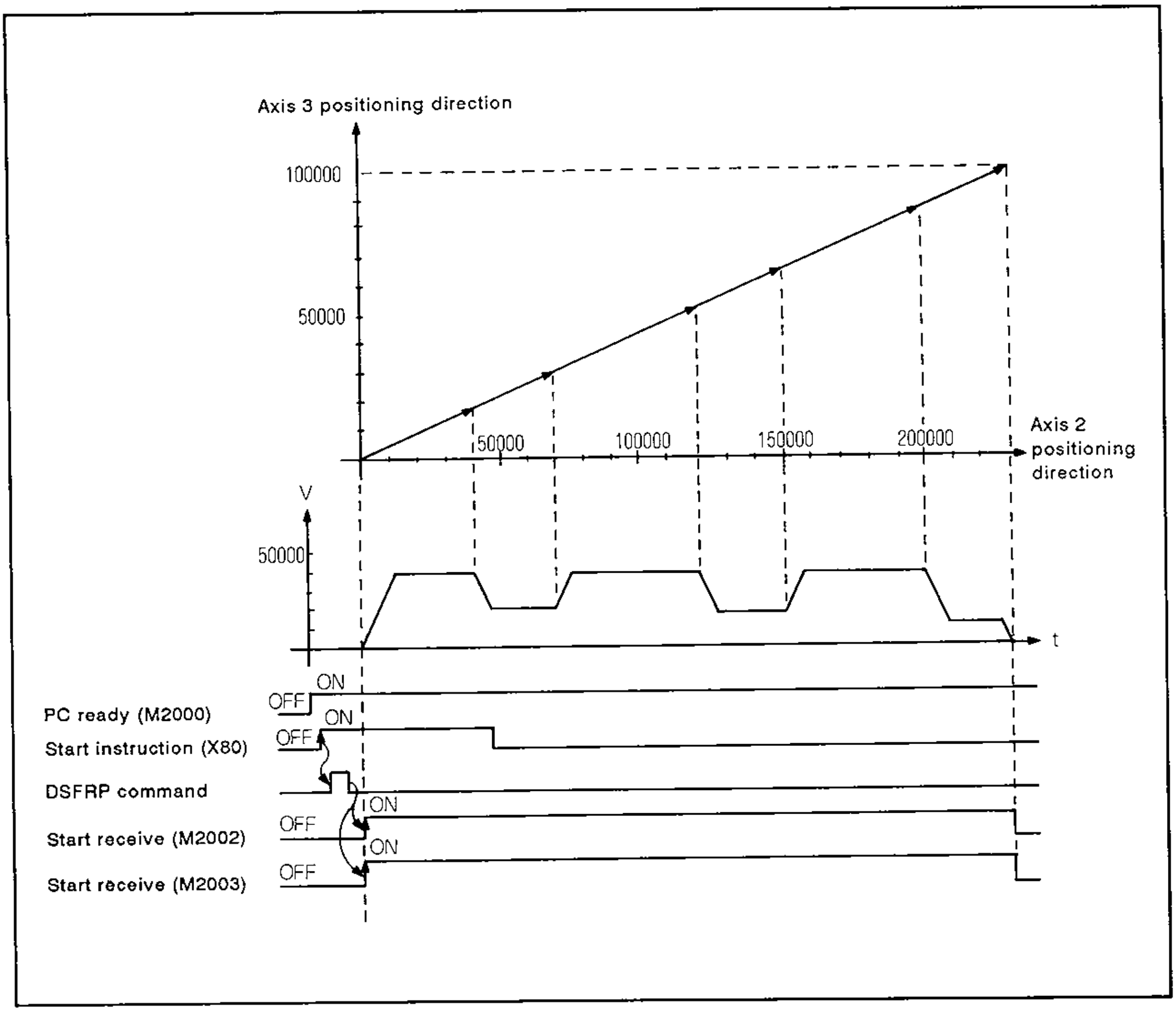
Leading edge (OFF → ON) of X80

(c) Execution of repeat command

X81 (designated range is executed repeatedly while X81 is on.)

(3) Operation timing and velocity change position

The timing chart for velocity change control and the velocity change positions are illustrated below.



(4) Servo program

Servo program No. 501 which calls the velocity change control, is explained below.

```
<K501>
VSTART
INC-2
    AXIS-2,       230000
    AXIS-3,       100000
    VELOCITY       10000
VINC
    AXIS-2,        40000
    VELOCITY       40000
FOR-TIMES
    K2
VINC
    AXIS-2,        30000
    VELOCITY       20000
VINC
    AXIS-2,        50000
    VELOCITY       40000
NEXT
VEND
```

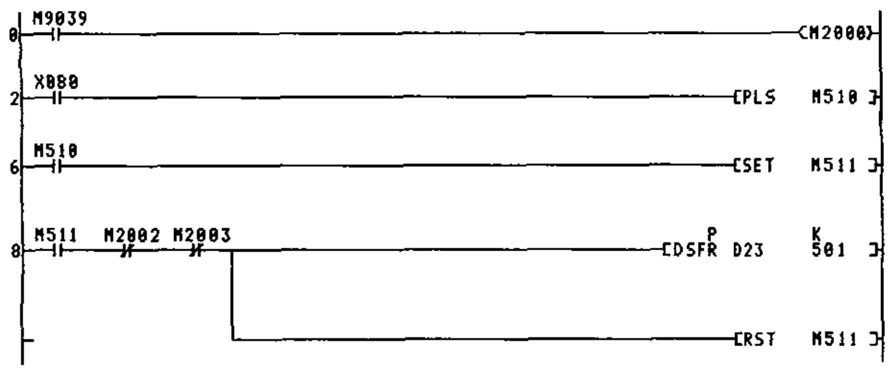
- VSTART: Start of velocity change control
- INC-2: 2-axis linear interpolation control (incremental)
- AXIS-2, AXIS-3:
  - Control axis: Axis 2 and Axis 3
  - Axis travel distance to stop position:
    - Axis 2 230000
    - Axis 3 100000
- VELOCITY 10000: Positioning velocity : 40000
- VINC: Setting of velocity change point and velocity
  - Setting velocity change position and velocity
  - Control axis: Axis 2
  - Axis travel distance up to velocity change position: 40000
- FOR-TIMES K2: Designation of repeat count (2 times)
- VINC ... VINC: Setting of velocity change position and velocity

Axis No.	Axis 2	
Velocity change point	30000	50000
Velocity up to the velocity change point	20000	40000

- NEXT: End of repeat area
- VEND: End of velocity change control

(5) Sequence program

The sequence program used to execute the servo programs is indicated below.



PC ready (M2000) ON

Detection of X80 leading edge, Servo program No. 501 start instruction flag (M511) is set at the leading edge of X80.

Servo program No. 501 execution request

M511 is reset after the the completion of the execution request for servo program No. 501.

7.14 Constant Velocity Control

(1) With a single start instruction, positioning control through the designated pass-points is executed at a constant velocity.

(2) Pass-points and the positioning method to the pass-point are designated in the servo program.

(3) By using the repeat command, it is possible to repeat the control between the required pass-points.

(4) The M code and torque limit can be changed at each pass-point.

7.14.1 Designation of constant velocity control start, pass-points, and end of control

<table>
<tr><th colspan="3" rowspan="2">Servo Command</th><th colspan="2">Start</th><th>End</th><th colspan="9">Pass-points</th></tr>
<tr><th>[CP START 2]</th><th>[CP START 3]</th><th>[CP END]</th><th>[ABS-2]</th><th>[ABS-3]</th><th>[ABS]</th><th>[ABS]</th><th>[ABS]</th><th>[ABS]</th><th>[ABS]</th><th>[ABS]</th><th>[ABS]</th></tr>
<tr><td colspan="3">Positioning Control Mode</td><td colspan="3">—</td><td colspan="9">Absolute</td></tr>
<tr><td colspan="3">Number of axes to be controlled</td><td colspan="3"></td><td colspan="9"></td></tr>
<tr><td rowspan="22">Process-ing Con-tents</td><td rowspan="7">Common</td><td>Parameter block No.</td><td>Δ</td><td>Δ</td><td></td><td></td><td></td><td></td><td colspan="4"></td><td colspan="2"></td></tr>
<tr><td>Axis</td><td>O</td><td>O</td><td></td><td>O</td><td>O</td><td>O</td><td colspan="4">O</td><td colspan="2">O</td></tr>
<tr><td>Address/travel distance</td><td></td><td></td><td></td><td>O</td><td>O</td><td>O</td><td colspan="4">O</td><td colspan="2">O</td></tr>
<tr><td>Command velocity</td><td>O</td><td>O</td><td></td><td></td><td></td><td></td><td colspan="4"></td><td colspan="2"></td></tr>
<tr><td>Dwell time</td><td></td><td></td><td>Δ</td><td></td><td></td><td></td><td colspan="4"></td><td colspan="2"></td></tr>
<tr><td>M code</td><td></td><td></td><td></td><td>Δ</td><td>Δ</td><td>Δ</td><td colspan="4">Δ</td><td colspan="2">Δ</td></tr>
<tr><td>Torque limit</td><td></td><td></td><td></td><td>Δ</td><td>Δ</td><td>Δ</td><td colspan="4">Δ</td><td colspan="2">Δ</td></tr>
<tr><td rowspan="3">Circular interpola-tion</td><td>Assist-point</td><td></td><td></td><td></td><td></td><td></td><td>O</td><td colspan="4"></td><td colspan="2"></td></tr>
<tr><td>Radius</td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="4">O</td><td colspan="2"></td></tr>
<tr><td>Center</td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="4"></td><td colspan="2">O</td></tr>
<tr><td rowspan="9">Para-meter block</td><td>Control units</td><td>Δ</td><td>Δ</td><td></td><td></td><td></td><td></td><td colspan="4"></td><td colspan="2"></td></tr>
<tr><td>Velocity limit</td><td>Δ</td><td>Δ</td><td></td><td></td><td></td><td></td><td colspan="4"></td><td colspan="2"></td></tr>
<tr><td>Acceleration time</td><td>Δ</td><td>Δ</td><td></td><td></td><td></td><td></td><td colspan="4"></td><td colspan="2"></td></tr>
<tr><td>Deceleration time</td><td>Δ</td><td>Δ</td><td></td><td></td><td></td><td></td><td colspan="4"></td><td colspan="2"></td></tr>
<tr><td>Immediate stop deceleration time</td><td>Δ</td><td>Δ</td><td></td><td></td><td></td><td></td><td colspan="4"></td><td colspan="2"></td></tr>
<tr><td>Torque limit</td><td>Δ</td><td>Δ</td><td></td><td></td><td></td><td></td><td colspan="4"></td><td colspan="2"></td></tr>
<tr><td>STOP input deceleration processing</td><td>Δ</td><td>Δ</td><td></td><td></td><td></td><td></td><td colspan="4"></td><td colspan="2"></td></tr>
<tr><td>Circular interpola-tion allowable error range</td><td>Δ</td><td></td><td></td><td></td><td></td><td></td><td colspan="4"></td><td colspan="2"></td></tr>
<tr><td rowspan="3">Others</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="4"></td><td colspan="2"></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="4"></td><td colspan="2"></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="4"></td><td colspan="2"></td></tr>
<tr><td colspan="3">Velocity change</td><td colspan="12">Possible</td></tr>
</table>

O: Items to be set always

Δ: Items to be set as needed

Servo Command			[INC-2]	[INC-3]	[INC]	[INC]	[INC]	[INC]	[INC]	[INC]	[INC]
			Pass-points								
Positioning Control Mode			Incremental								
Number of axes to be controlled											
Processing Contents	Common	Parameter block No.									
		Axis	O	O	O	O				O	
		Address/travel distance	O	O	O	O				O	
		Command velocity									
		Dwell time									
		M code	Δ	Δ	Δ	Δ				Δ	
		Torque limit	Δ	Δ	Δ	Δ				Δ	
	Circular interpolation	Assist-point			O						
		Radius				O					
		Center								O	
	Parameter block	Control units									
		Velocity limit									
		Acceleration time									
		Deceleration time									
		Immediate stop deceleration time									
		Torque limit									
		STOP input deceleration processing									
		Circular interpolation allowable error range									
	Others										
Velocity change			Possible								

O: Items to be set always

Δ: Items to be set as needed

[Control details]

Start and end of constant velocity control

The following commands are used to start and end the velocity change control.

(1) CPSTART2

Starts the constant velocity control with two axes.

Set the number of axes to be controlled and the required velocity.

(2) CPSTART3

Starts the constant velocity control with three axes.

Set the number of axes to be controlled and the required velocity.

(3) CPEND

Ends the constant velocity control called by the CPSTART2 or CPSTART3 commands.

Setting positioning control mode up to the pass-point

The following commands are used to designate the positioning mode up to the position where the control is changed.

(1) ABS-2/INC-2

Designates two-axis linear interpolation control.

For details, refer to Section 7.3.

(2) ABS/INC

Designates circular interpolation control with assist point designation.

For details, refer to Section 7.5.

(3) ABS/INC , ABS/INC , ABS/INC , ABS/INC

Designates circular interpolation control with radius designation.

For details, refer to Section 7.6.

(4) ABS/INC , ABS/INC

Designates circular interpolation control with center point designation.

For details, refer to Section 7.7.

Programming servo program and operation timing

The servo program that calls the constant velocity control and the operation timing are indicated in Fig. 7.26.

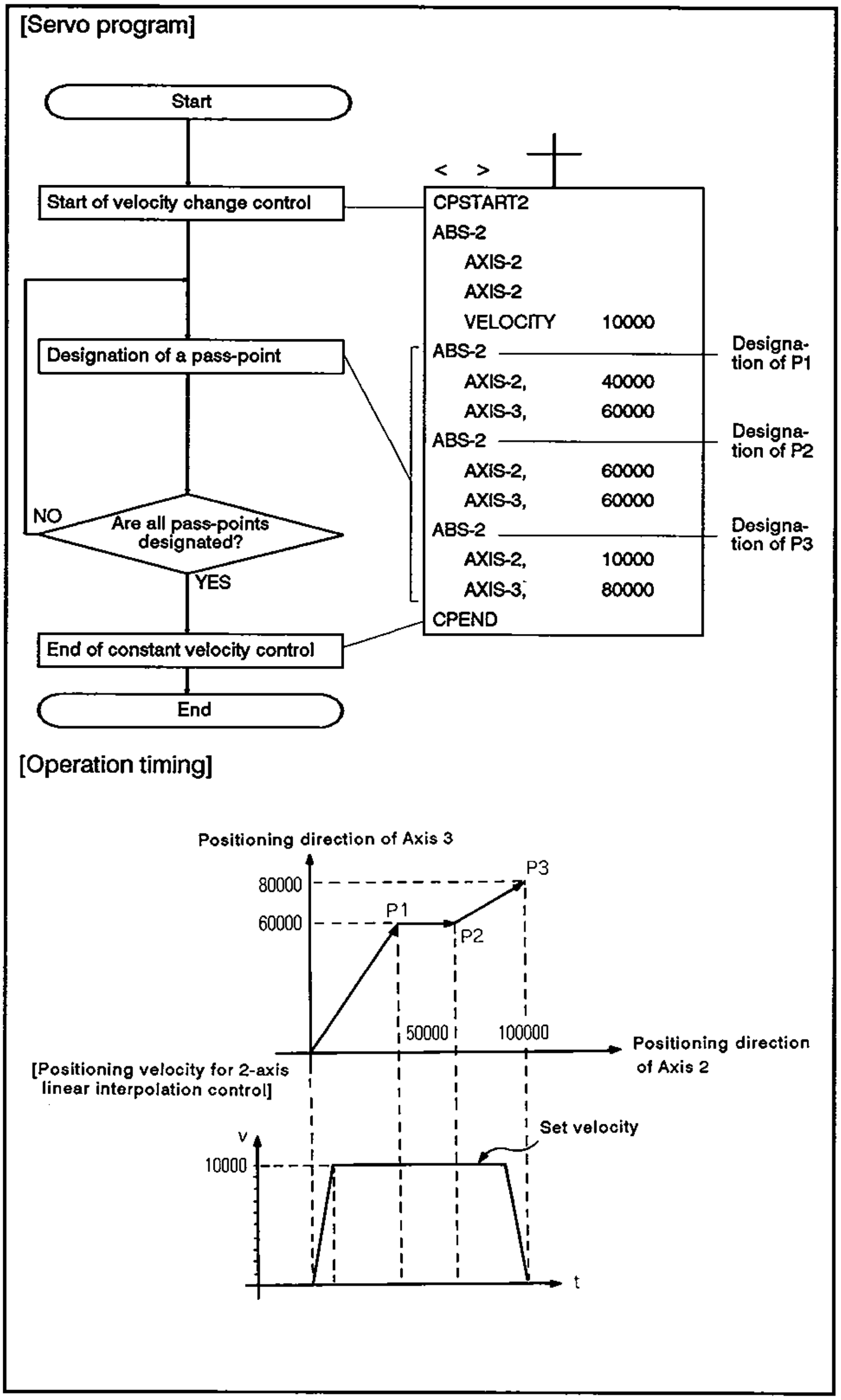


Fig. 7.26 Servo Program for Constant Velocity Control and Operation

IB (NA) 66233-A

[Caution]

(1) The number of controlled axes cannot be changed before completion.

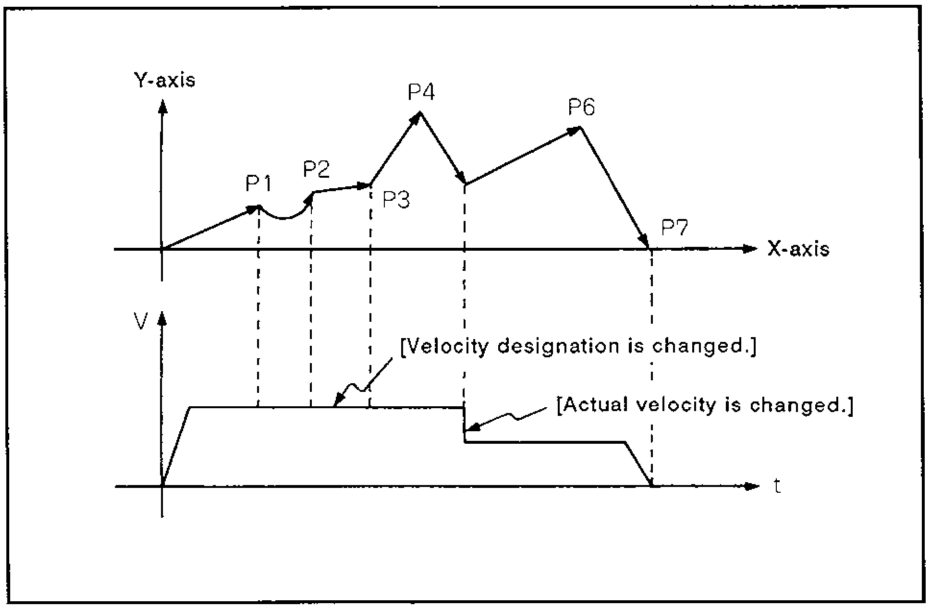
(2) For the positioning control mode up to the pass-point, absolute (ABS[ ]) and incremental (INC[ ]) control can be used in combination.

(3) For the pass-points, designation is allowed for an address which causes the axis travel direction to change.

However, because the acceleration/deceleration control is not executed at the pass-points, an error, such as a servo error, might occur.

(4) It is possible to change the velocity after the start of positioning control.

When the designated velocity is changed, the new velocity becomes effective from the second positioning mode counted from the positioning mode in which the velocity is changed.



(5) After the start of positioning and at the final positioning point is detected, the distance up to the final positioning point is checked; if the distance is shorter than the distance required for deceleration, an overrun error occurs.

In this case, an error code (211) is registered in the minor error storage register for the axis in question.

(6) The maximum number of steps which can be designated in a single constant velocity control program is 768 steps, which is approximately 100 points.

(7) After the start of positioning control, if positioning is to be executed beyond the stroke limit, an error code (106) is registered in the minor error storage register for the axis in question; the axis decelerates and stops.

(8) The minimum axis travel distance between the two pass-points allowable for the constant velocity control is indicated below.

(a) When the A6MD is not used, or when the A6MD screen is used in a mode other than the test or monitor mode, there are no limits on travel distance.

However, positioning velocity will be lowered if the distance between the two pass-points is extremely short.

Example:
If pass-points are set in intervals of one pulse, positioning is executed at approximately 280 PPS regardless of the velocity setting.

(b) When the A6MD is used in the monitor mode or for monitoring the program in the test mode, the distance between the two pass-points must be greater than the distance indicated below.

Designated velocity x 0.02 < Travel distance (PLS)

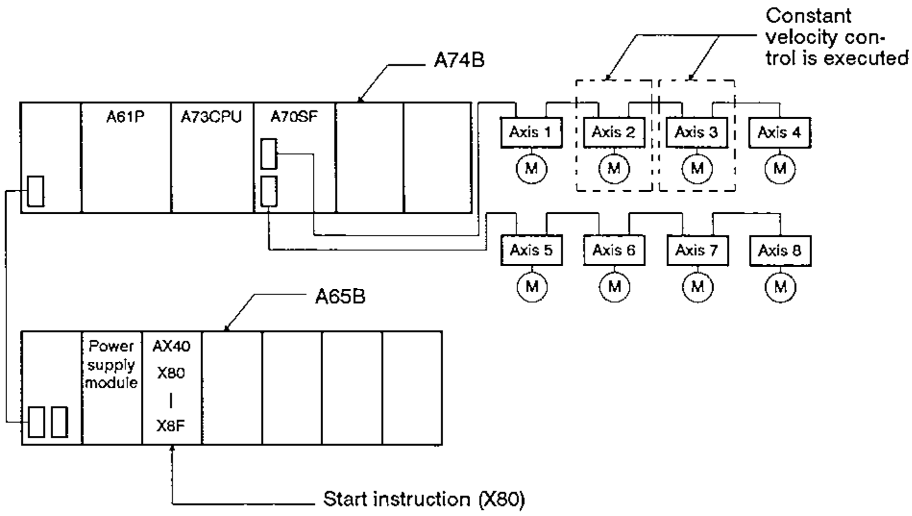
POINT

While the constant velocity control program is being monitored, never change the screen display. The motor will stop at each pass-point when the travel distance is small or screen is changed.

[Program example]

The program which calls constant velocity control is explained below.

(1) System configuration



(2) Positioning conditions

(a) Constant velocity control conditions are indicated below.

Item		Setting		
Servo program No.		505		
Positioning velocity		10000		
Positioning mode		2-axis linear interpolation	Circular interpolation with the radius designated	2-axis linear interpolation
Pass-point	Axis 2	30000	50000	90000
	Axis 3	30000	50000	100000

(b) Start of constant velocity control

Leading edge (OFF → ON) of X80

(3) Servo program

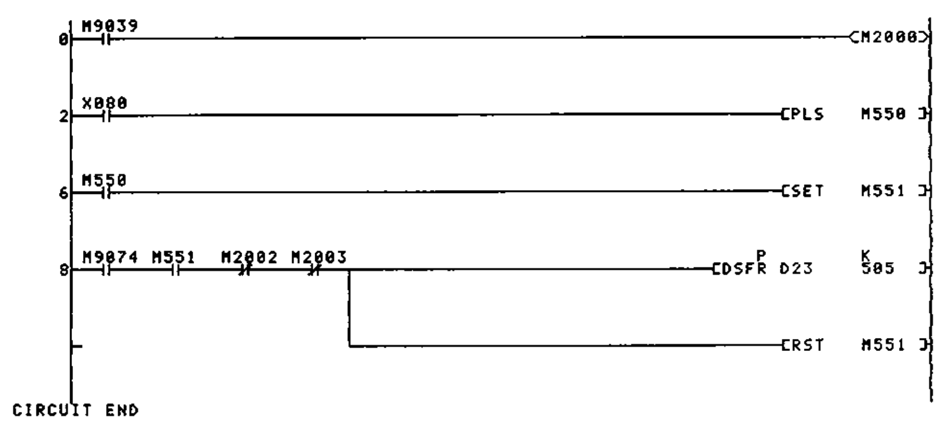
Servo program No. 505, which calls the constant velocity control, is explained below.

<K505>

CPSTART 2		Start of constant velocity control
AXIS-2		Control axes: Axis 2 and Axis 3
AXIS-3		
VELOCITY	10000	Positioning velocity: 10000
ABS-2		2-axis linear interpolation control
AXIS-2,	30000	Positioning addresses: Axis 2 30000
AXIS-3,	30000	Axis 3 30000
ABS		Circular interpolation
AXIS-2,	50000	Positioning addresses: Axis 2 50000
AXIS-3,	50000	Axis 3 50000
RADIUS	20000	Radius 20000
ABS-2		2-axis linear interpolation control
AXIS-2,	90000	Positioning addresses: Axis 2 90000
AXIS-3,	100000	Axis 3 100000
CPEND		End of constant velocity control

(5) Sequence program

The sequence program used to execute the servo programs is indicated below.



PC ready (M2000) ON

Servo program No. 505 start instruction flag (M551) is set at the leading edge of X80.

Servo program No. 505 execution request

M551 is reset after the the completion of the execution request for servo program No. 505.

7.14.2 Designating pass-points with repeat command

Servo Command			[FOR-TIMES]	[FOR-ON]	[FOR-OFF]	[NEXT]
Positioning Control Mode				—		—
No of axes to be controlled				—		—
Processing Contents	Common	Parameter block No.				
		Axis				
		Address/travel distance				
		Command velocity				
		Dwell time				
		M code				
		Torque limit				
	Circular interpolation	Assist-point				
		Radius				
		Center				
	Parameter block	Control units				
		Velocity limit				
		Acceleration time				
		Deceleration time				
		Immediate stop deceleration time				
		Torque limit				
		STOP input deceleration processing				
		Circular interpolation allowable error range				
	Others	Repeat condition		O		
Velocity Change					—	

O: Items to be set always

Δ: Items to be set as needed

[Control details]

Setting the repeat range head address

The following commands are used to designate the repeat range head address.

(1) FOR-TIMES (setting loop-out count)

(a) The set repeat range is repeatedly executed for the set number of times.

(b) The setting range is 1 to 32767.

If the set data is outside the allowable setting range (–32768 to 0), it is regarded as "1".

(c) The devices that can be used to set the repeat count are indicated below:

1) Data register (D)

2) Link register (W) For indirect setting

3) Decimal constant (K)

4) Hexadecimal constant (H)

(2) FOR-ON (setting loop-out trigger conditions)

(a) The designated range is executed repeatedly until the designated bit device is turned on.

(b) The devices that can be used for the loop-out trigger condition are indicated below:

1) Input (X)

2) Output (Y)

3) Internal relay (M) / special relay (SP.M)

4) Latch relay (L)

5) Link relay (B)

6) Annunciator (F)

(3) FOR-OFF (setting loop-out trigger conditions)

(a) The designated range is executed repeatedly until the designated bit device is turned off.

(b) The devices that can be used as the loop-out trigger condition are indicated below:

1) Input (X)

2) Output (Y)

3) Internal relay (M) / special relay (SP.M)

4) Latch relay (L)

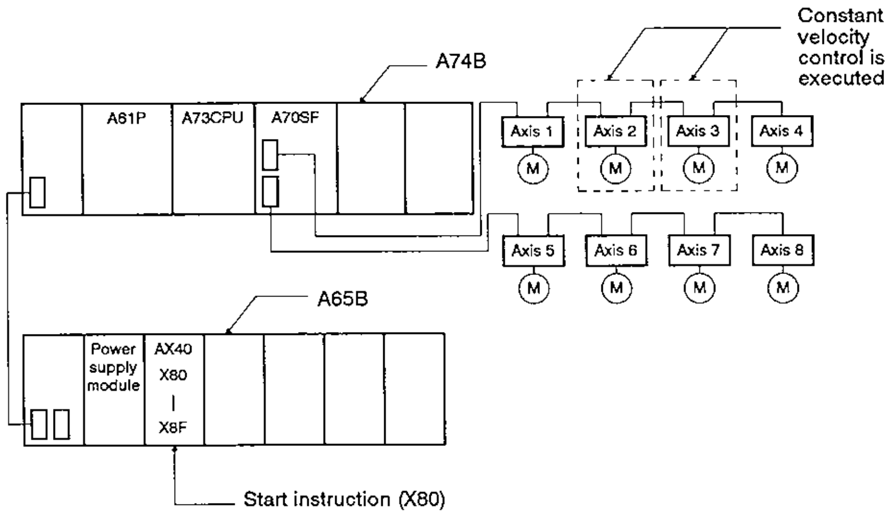
5) Link relay (B)

6) Annunciator (F)

[Program example]

The program repeatedly executes the same processing in the constant velocity control mode is explained below.

(1) System configuration



(2) Positioning conditions

(a) The constant velocity control conditions are indicated below.

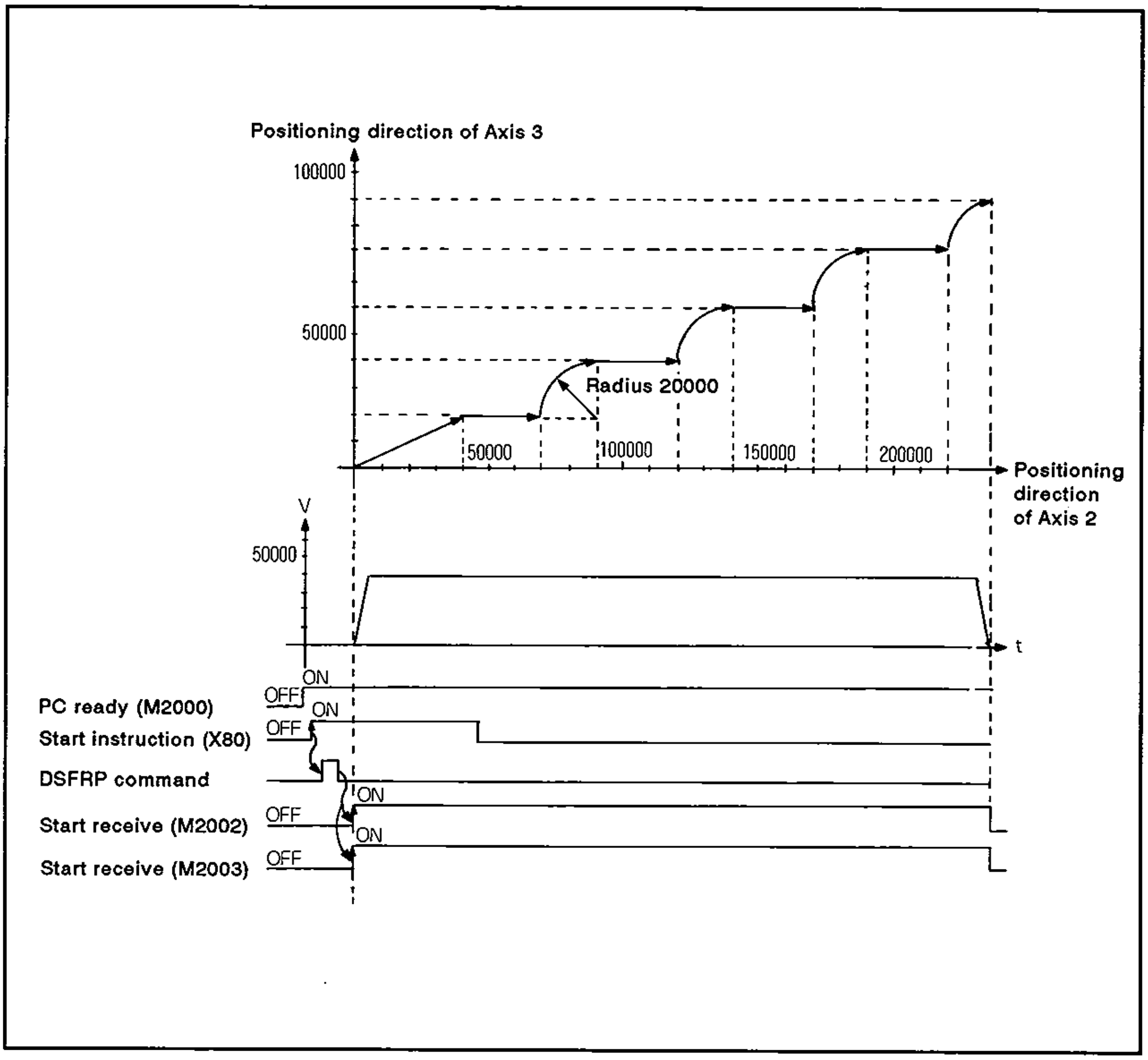
Item	Setting
Servo program No.	510
Control axes	Axis 2, Axis 3
Positioning velocity	10000

(b) Start of constant velocity control

Leading edge (OFF → ON) of X80

(3) Operation timing and axis travel in the constant velocity control mode

Timing chart and axis motion in the constant velocity control mode are illustrated below.



(4) Servo program

Servo program No. 510, which calls the constant velocity control, is explained below.

```
<K510>
CPSTART2                        ——— Start of constant velocity control
    AXIS-2                      ——— Control axis:     Axis 2, Axis 3
    AXIS-3
    VELOCITY      10000         ——— Positioning velocity: 10000
ABS-2                           ——— Setting pass-points
    AXIS-2,       40000
    AXIS-3,       20000
FOR-TIMES                       ——— Designation of repeat count (4 times)
    K4
INC-2                           ——— Setting of pass-points
    AXIS-2,       30000
    AXIS-3,           0
INC ↶
    AXIS-2,       20000
    AXIS-3,       20000
    RADIUS        20000
NEXT                            ——— End of repeat area
CPEND                           ——— End of velocity change control
```

Positioning mode		2-axis linear interpolation	Circular interpolation with radius designation
Axis travel distance	Axis 2	30000	20000
	Axis 3	0	20000

(5) Sequence program

The sequence program used to execute servo programs is indicated below.

```
   M9039
0 ─┤├──────────────────────────────────────────────(M2000)   PC ready (M2000) ON

   X080
2 ─┤├─────────────────────────────────────[PLS    M560 ]    Servo program No. 510
                                                             start instruction flag
   M560                                                      (M561) is set at the lead-
6 ─┤├─────────────────────────────────────[SET    M561 ]    ing edge of X80.

   M9074 M561  M2002 M2003                    P        K
8 ─┤├────┤├────┤/├───┤/├──┬──────────────[DSFR  D23   510 ]   Servo program No. 510
                          │                                     execution request
                          └──────────────[RST         M561 ]   M561 is reset after the
                                                                the completion of the ex-
                                                                ecution request for
CIRCUIT END                                                     servo program No. 510.
```

7.15 Simultaneous Start of Servo Programs

A single start instruction can start multiple servo programs simultaneously.

The START command in the servo program is used for this operation.

Servo Command			[START]
Positioning Control Mode			*
Number of axes to be controlled			*
Processing Contents	Common	Parameter block No.	
		Axis	
		Address/travel distance	
		Command velocity	
		Dwell time	
		M code	
		Torque limit	
	Circular interpolation	Assist-point	
		Radius	
		Center	
	Parameter block	Control units	
		Velocity limit	
		Acceleration time	
		Deceleration time	
		Immediate stop deceleration time	
		Torque limit	
		STOP input deceleration processing	
		Circular interpolation allowable error range	
	Others	Repeat condition	
		Program No.	O
Velocity Change			*

O: Items to be set always　　*: Varies with servo program to execute simultaneous start.

Δ: Items to be set as needed

[Control details]

Control by START command

(1) Multiple servo programs are started simultaneously.

(2) Any servo program, excluding the one containing the START command, can be designated for simultaneous start.

(3) Simultaneous start of up to three servo programs is possible.

When each of the servo programs is for one axis control, simultaneous start of up to three axes is possible. When the servo program is for two or three axis control, simultaneous start of up to eight axes is possible.

(4) After the start, control of each axis is carried out according to the setting in the designated servo program.

[Caution]

(1) At the start of control, the program is checked and if any of the following case are detected, an error will occur and positioning will not start.

Error Contents	Error Processing	Code Storing Registers	
		D9189	D9190
The designated servo program is not found.	Servo program setting error flag (M9079): ON Start receive flag (M200n): OFF	The number of servo program causing an error	19
The START command is set in the designated servo program.			
A servo program which cannot be started is designated.		The number of servo program causing an error	Wrong servo program number

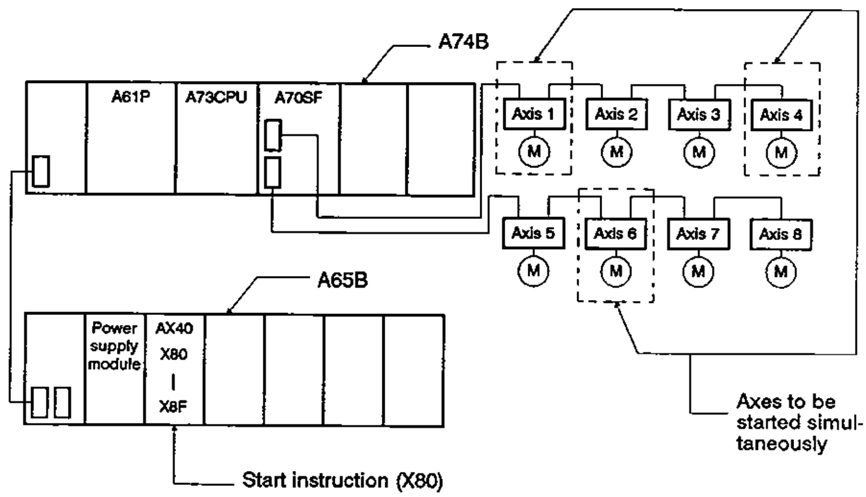
(2) For the designation of the servo program to be started simultaneously, indirect designation is not possible.

(3) If a servo program designated with the START command calls for "incremental feed control" or the "V/P control", execution of such a program might be delayed up to one second compared with other position control or velocity control programs.

[Program example]

The servo program which calls for the simultaneous start of other servo programs is explained below.

(1) System configuration



(2) Number of servo programs to be designated and servo program numbers

(a) Number of servo programs which can be designated:

3 servo programs

(b) Designated program number

Servo Program No.	Control Axis	Control Contents
No. 1	Axis 1 and Axis 3	Circular interpolation
No. 14	Axis 4	Velocity control
No. 45	Axis 6	Zero return control

(3) Start conditions

(a) Servo program calling the simultaneous start:

No. 121

(b) Execution of simultaneous start:

Leading edge (OFF → ON) of X80

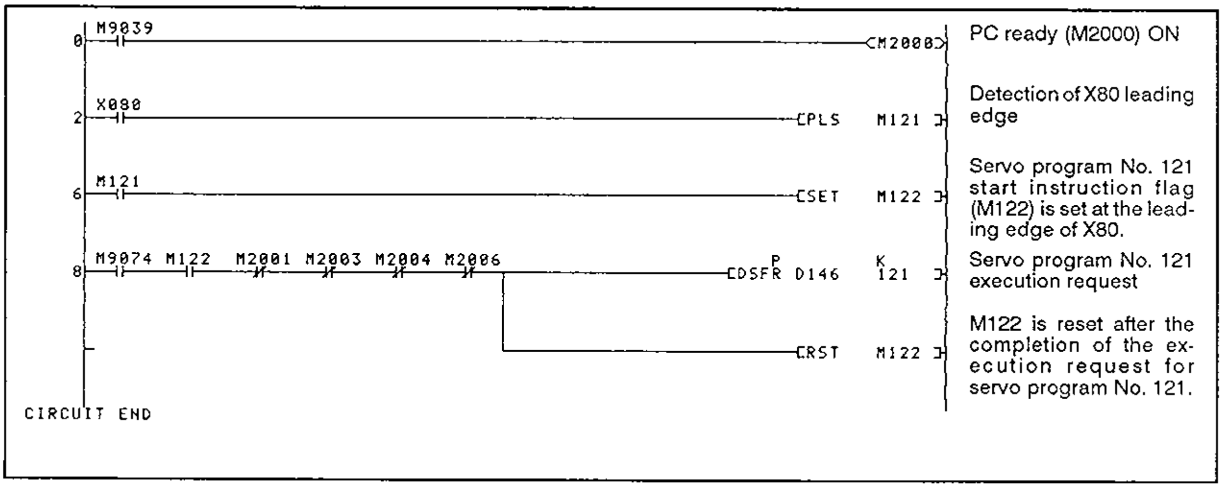
(4) Servo program

Servo program No. 121, which calls for the simultaneous start of servo programs, is indicated below.

```
<K121>
START                 ---- Simultaneous start control
  K         1         ---- Servo program No. 1
  K         14        ---- Servo program No. 14
  K         45        ---- Servo program No. 45
```

(5) Sequence program

The servo program used to execute other servo programs is illustrated below.



7.16 Jog Operation

The set jog operation is executed.

Jog operation can be executed independently or with other servo programs.

Jog operation is called by the sequence program or carried out in a peripheral device test mode. For the procedure to execute jog operation in a peripheral device test mode, refer to the peripheral device operation manual.

To execute jog operation, a setting is required for each individual axis.

7.16.1 Jog operation data

Jog operation data is the data required to execute jog operation.

The defaults are set before shipment; to change defaults, set the required data with a peripheral device.

Table 7.15 Jog Operation Data List

No.	Item	Setting range								Default		Remarks	Refer to
		mm		inch		degree		PULSE					
		Setting range	Unit	Setting range	Unit	Setting range	Unit	Setting range	Unit	Initial value	Unit		
1	Jog operation velocity limit	0.01 to 6000000.00	mm / min	0.01 to 6000000.000	inch / min	0.01 to 6000000.000	degree / min	1 to 1000000	PLS / sec	20000	PLS / sec	Set the maximum velocity allowed for jog operation. If the designated jog velocity exceeds this limit, the designated jog velocity is replaced with the maximum velocity.	—
2	Parameter block designation	1 to 16								1	—	Set the parameter block number to be used for jog operation.	4.3

(1) Checking the jog operation data

The set jog operation data is cross checked at the following times.

(a) When the power is turned on.

(b) At the leading edge of the PC ready (M2000) signal

(c) When the mode is changed to the test mode.

(2) Error processing

(a) If an error is detected in the cross check, the error data is replaced with the default and jog operation is executed.

(b) The error code corresponding to the error data is registered in the data register.

7.16.2 Independent start of jog operation

Jog operation of the designated axis starts.

Jog operation is started with the following jog operation signal.

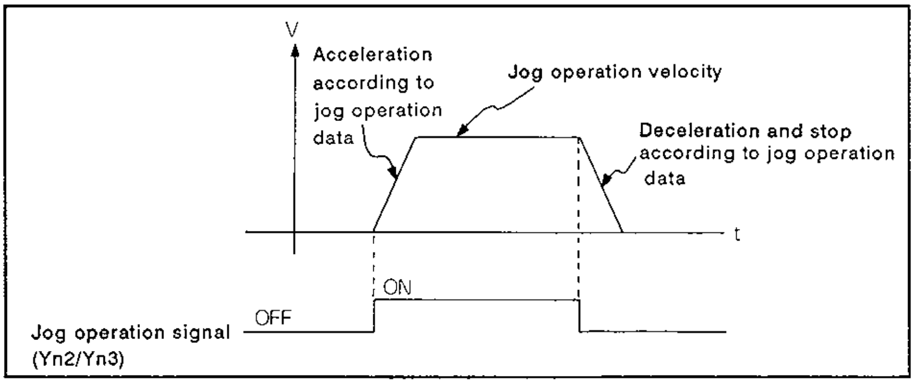
(a) Forward jog operation Yn2

(b) Reverse jog operation Yn3

[Control details]

(1) Jog operation is executed at the velocity set in the jog operation velocity setting register while the jog operation signal is on; when the jog operation signal is turned off, the axis decelerates and stops.

Acceleration and deceleration are controlled according to the data set as jog operation data.



Jog operation is executed on the axis for which the jog operation signal is on.

(2) The jog operation signal and jog operation data setting register and setting range are indicated below.

<table>
<tr><th rowspan="3">Axis No.</th><th colspan="2">Jog Operation Signal</th><th colspan="2">Jog Operation Velocity Setting Register</th><th colspan="8">Velocity Setting Range</th></tr>
<tr><th colspan="2"></th><th colspan="2"></th><th colspan="2">mm</th><th colspan="2">inch</th><th colspan="2">degree</th><th colspan="2">PULSE</th></tr>
<tr><th>Forward Jog</th><th>Reverse Jog</th><th>Upper digit</th><th>Lower digit</th><th>Setting range</th><th>Unit</th><th>Setting range</th><th>Unit</th><th>Setting range</th><th>Unit</th><th>Setting range</th><th>Unit</th></tr>
<tr><td>1</td><td>Y2</td><td>Y3</td><td>965</td><td>964</td><td rowspan="8">1 to 600000000</td><td rowspan="8">X10^{-2} mm / min</td><td rowspan="8">1 to 600000000</td><td rowspan="8">X10^{-3} inch / min</td><td rowspan="8">1 to 600000000</td><td rowspan="8">X10^{-3} degree / min</td><td rowspan="8">1 to 1000000</td><td rowspan="8">PLS / sec</td></tr>
<tr><td>2</td><td>Y12</td><td>Y13</td><td>971</td><td>970</td></tr>
<tr><td>3</td><td>Y22</td><td>Y23</td><td>977</td><td>976</td></tr>
<tr><td>4</td><td>Y32</td><td>Y33</td><td>983</td><td>982</td></tr>
<tr><td>5</td><td>Y42</td><td>Y43</td><td>989</td><td>988</td></tr>
<tr><td>6</td><td>Y52</td><td>Y33</td><td>995</td><td>994</td></tr>
<tr><td>7</td><td>Y62</td><td>Y63</td><td>1001</td><td>1000</td></tr>
<tr><td>8</td><td>Y72</td><td>Y73</td><td>1007</td><td>1006</td></tr>
</table>

POINT

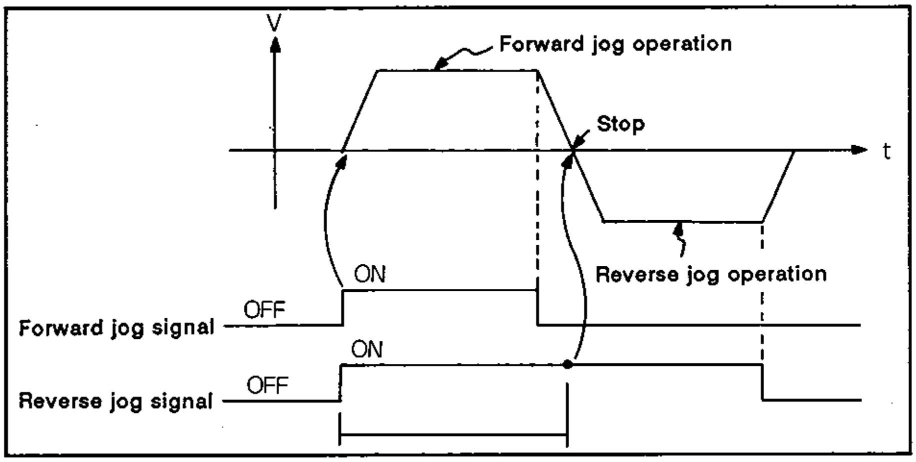
To set the jog operation velocity in the sequence program, set it in the jog operation velocity setting register after multiplying it by 100 (for mm) or 1000 (for inches or degrees).

Example: To designate jog operation velocity of 600.00 mm/min, set "6000000" in the jog operation velocity setting register.

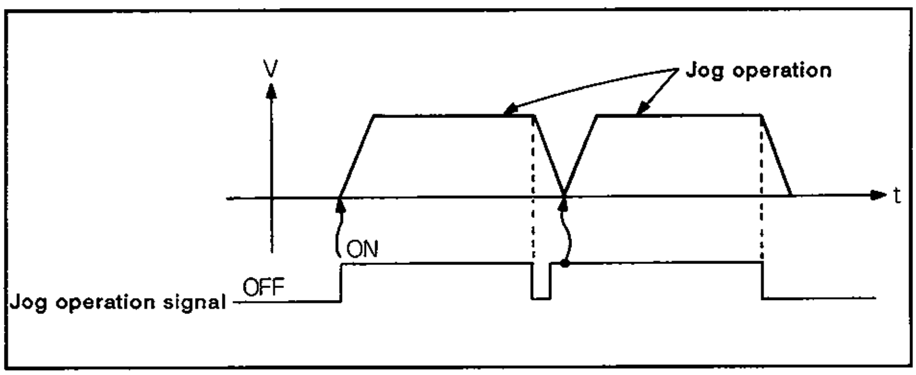
[Caution]

(1) If both the forward jog signal (Yn2) and reverse jog signal (Yn3) are turned on at the same time, the forward jog operation will be executed.

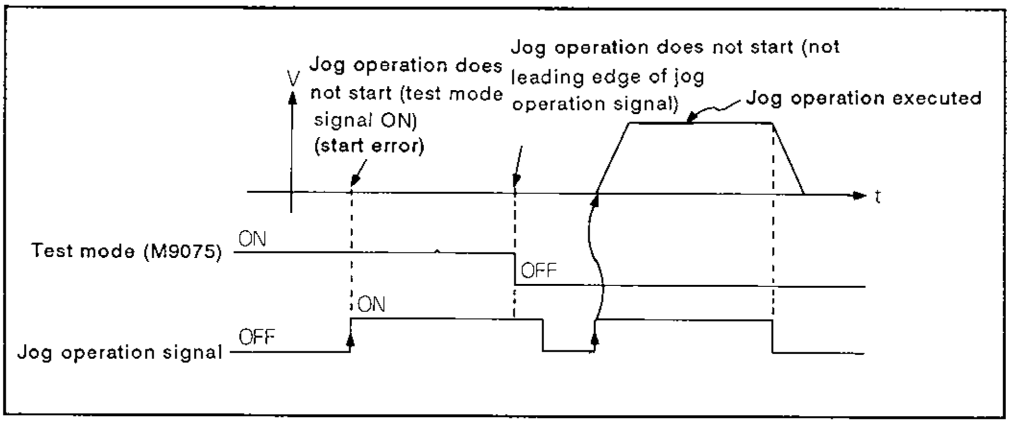
If the reverse jog signal is on when the axis stops after deceleration with the jog reverse signal turned off, the reverse rotation jog operation is executed.



(2) If the jog operation signal is turned on during deceleration after the jog operation signal has been turned off, the axis is decelerated and stopped. After that, jog operation will be executed.

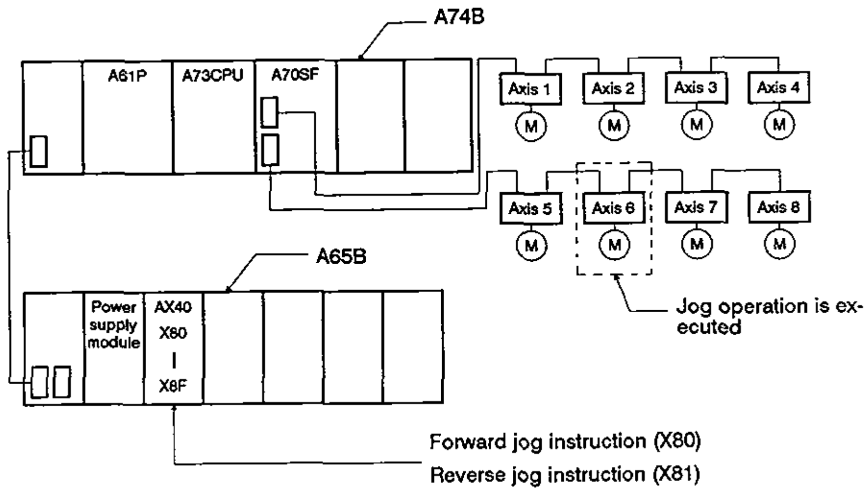


(3) In a peripheral device test mode, jog operation does not start when the jog operation signal is turned on. Jog operation will start at the leading edge of the jog operation signal after the test mode is cleared.



[Program example]

(1) System configuration



(2) Jog operation conditions

(a) Axis No. .... Axis 6

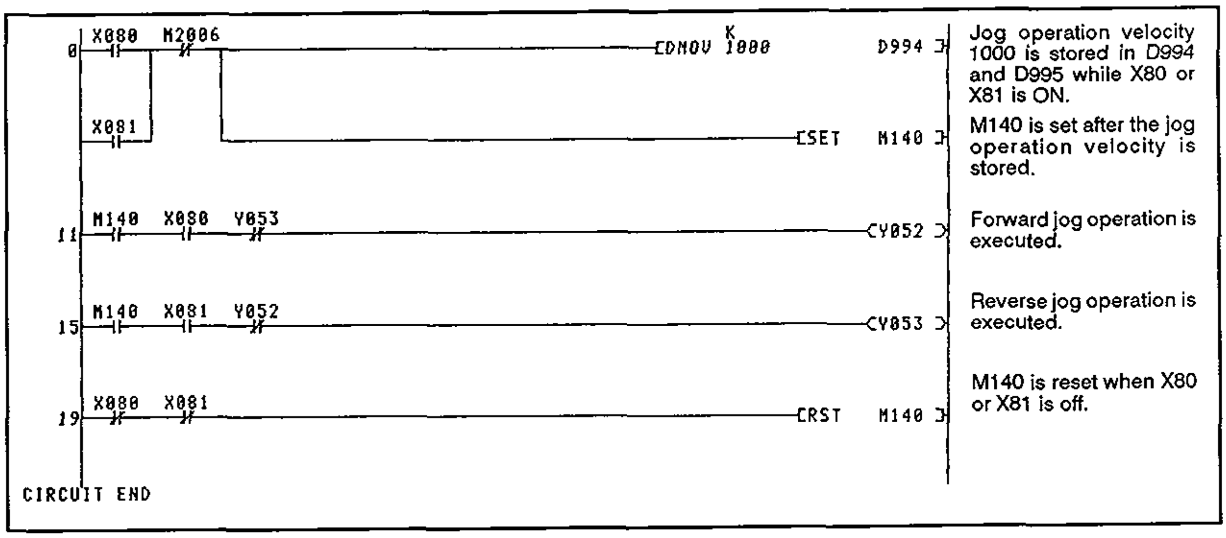
(b) Jog operation velocity 1000

(c) Jog operation instruction

1) Forward jog operation......... During X80 ON

2) Reverse jog operation .... During X81 ON

(3) Sequence program

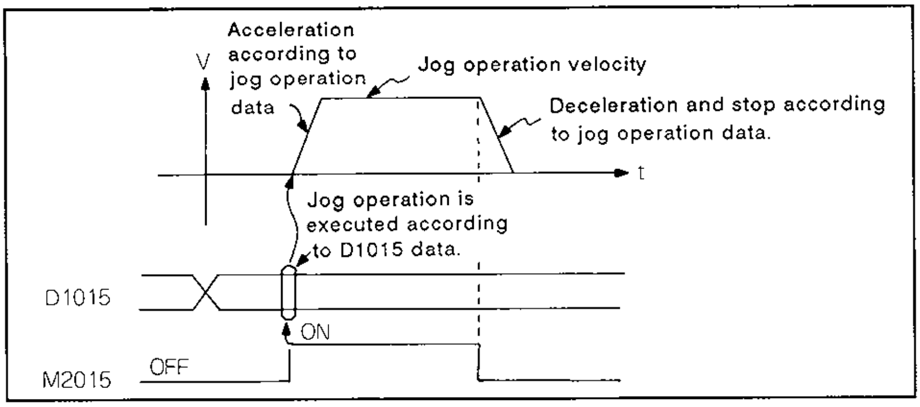


7.16.3 Simultaneous start of jog operation for multiple axes

[Control details]

(1) While the simultaneous jog operation flag (M2015) is on, jog operation of each axis is executed at the value set in the jog operation velocity setting registers. The axes decelerate and stop when M2015 is turned off.

Acceleration and deceleration are controlled according to the jog operation data.



(2) The axes to be operated in the jog operation mode are set in D1015.

	Reverse jog								Forward jog							
	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

Axis numbers corresponding to b0 to b15 in D1015

1: Jog operation executed
0: Jog operation not executed

The program to execute forward jog operation with axis 1 and axis 5 and reverse jog operation with axis 8 using the MOV instruction is indicated below.

a) Hexadecimal (H) designation: [MOV | H8011 | D1015]

b) Decimal (K) designation: [MOV | K32785 | D1015]

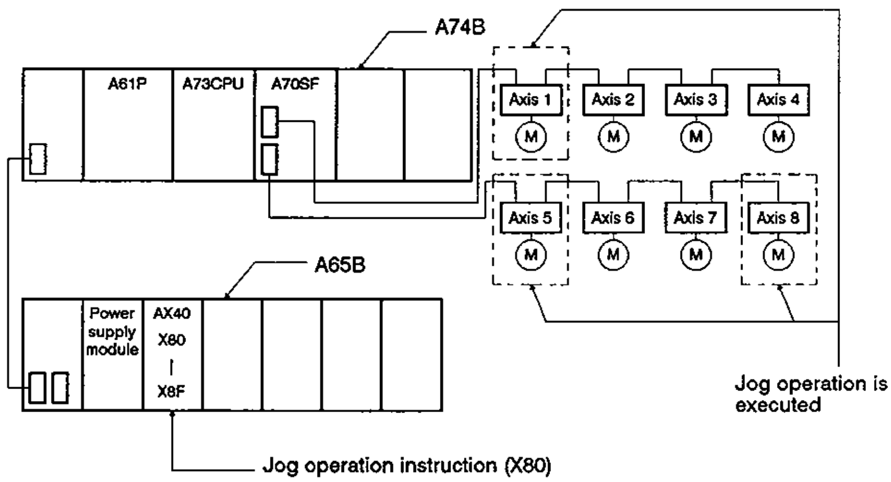
(3) Jog operation velocity setting registers are indicated below.

Axis No.	Jog Operation Velocity Setting Register		Velocity Setting Range							
			mm		inch		degree		PULSE	
	Upper digit	Lower digit	Setting range	Unit	Setting range	Unit	Setting range	Unit	Setting range	Unit
1	965	964	1 to 600000000	$X10^{-2}$ mm /min	1 to 600000000	$X10^{-3}$ mm /min	1 to 600000000	$X10^{-3}$ mm /min	1 to 1000000	PLS / sec
2	971	970								
3	977	976								
4	983	982								
5	989	988								
6	995	994								
7	1001	1000								
8	1007	1006								

[Program example]

The program which calls for the simultaneous start of multiple axes jog operation is explained below.

(1) System configuration



(2) Jog operation conditions

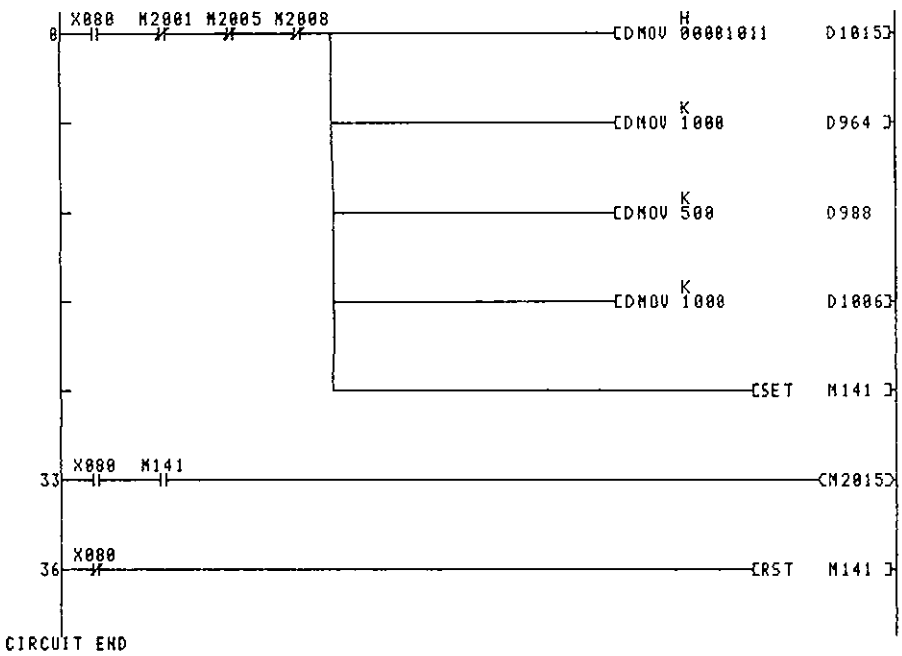
(a) Jog operation conditions are indicated below.

Item	Jog Operation Conditions		
Controlled axis	Axis 1	Axis 5	Axis 8
Jog operation velocity	1000	500	1000
Jog operation direction	Forward	Reverse	Forward

(b) Jog operation instruction:

While X80 is ON.

(3) Sequence program



Axes to be started simultaneously are stored in D1015 while X80 is ON.

Jog operation velocity of each axis is stored to each jog operation velocity setting register.

M141 is set after the completion of axes setting and jog operation velocity.

Jog operation starts.

M141 is reset when X80 is off.

7.17 Manual Pulse Generator (MPG) Operation

Positioning control is executed in response to the number of pulses input by the MPG.

Up to three MPGs can be connected to the A70SF and each MPG can simultaneously operate one to three axes.

[Control details]

(1) In response to the pulses input from the MPG, positioning is executed for the axis set in the MPG axis setting register.

MPG operation is only possible when the MPG operation enable flag is set.

MPG Connection Position *	MPG Operation Axis Setting Register	MPG Operation Enable Flag
P1	D1012	M2012
P2	D1013	M2013
P3	D1014	M2014

(2) Axis travel distance and output velocity for MPG operation are calculated as indicated below.

(a) Axis travel distance

Axis travel distance is calculated with the following formula.

Axis travel distance =
[Travel distance per pulse] x [Number of pulses] x [Multiplication ratio]

(b) Output velocity

In the MPG operation mode, positioning is executed at the velocity in response to the number of pulses input in unit time.

Output velocity =
[Input pulses input in 1 msec] x [Multiplication ratio]

(3) Setting the axes for MPG operation

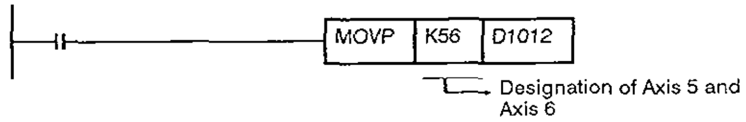
(a) Set the axes to be operated in the MPG operation mode to the MPG axis setting registers (D1012 to D1014).

Set the axes with 3-digit decimal data; an axis number (1 to 8) is set by each digit.

The number of digits indicates the number of axes to be simultaneously operated.

Example:

Set as indicated below to control Axis 5 and Axis 6 with the MPG connected to P1.



REMARKS

*: The MPG connection position indicates the name of the A70SF terminal where the MPG is connected.

For details, refer to the A73PCU User's Manual.

(4) Pulse multiplication ratio

(a) The multiplication ratio of one MPG pulse input is set using registers D1016 to D1023.

Multiplication Ratio Setting Register	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	

(5) The setting of the MPG multiplication ratio setting register for a corresponding axis is checked at the leading edge of the MPG operation enable flag. If the setting is outside the allowable setting range, "1" is set for the MPG axis setting error register (D9187) and the MPG axis setting error flag (M9077) is set; MPG operation is executed with a pulse multiplication ratio setting of "1".

(6) Error during MPG operation

Error Contents	Error Processing
The axis setting is not between "1" and "8".	The setting causing the error is ignored. The axes set correctly are controlled in the MPG operation mode.
An axis already designated for MPG operation is designated again.	The setting is ignored. MPG operation is executed according to the first setting.
4 or more digits are set.	All settings are ignored.

[Caution]

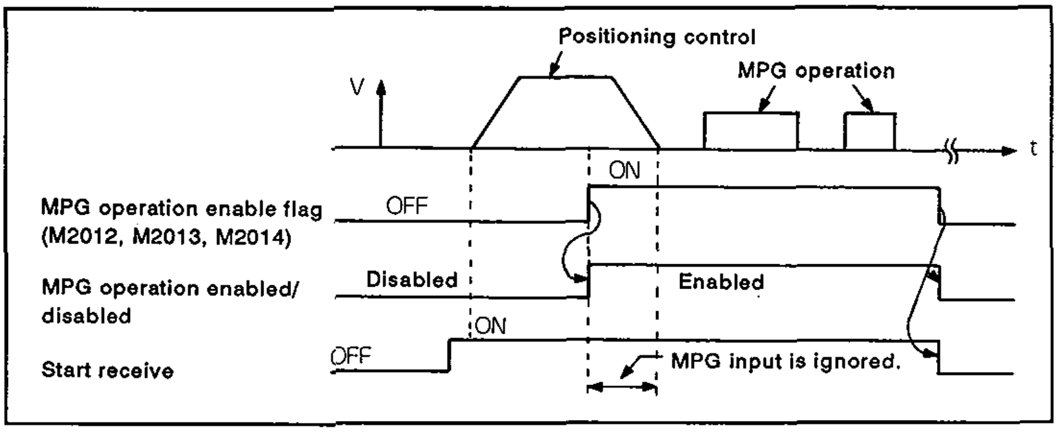
(1) For the axis set for MPG operation, the start receive flag is set.

Therefore, positioning control, zero return, and other operations are not allowed.

After the completion of MPG operation, reset the MPG operation enable flag.

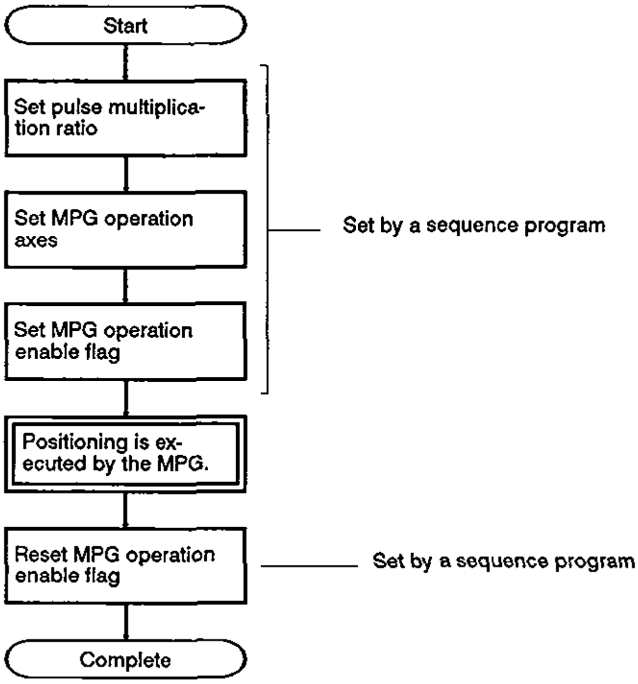
(2) During MPG operation, the torque limit is fixed at 300%.

(3) If the MPG operation enable flag is set for axes already started for positioning control, jog operation, etc., MPG operation is enabled; however, the MPG inputs are ignored.



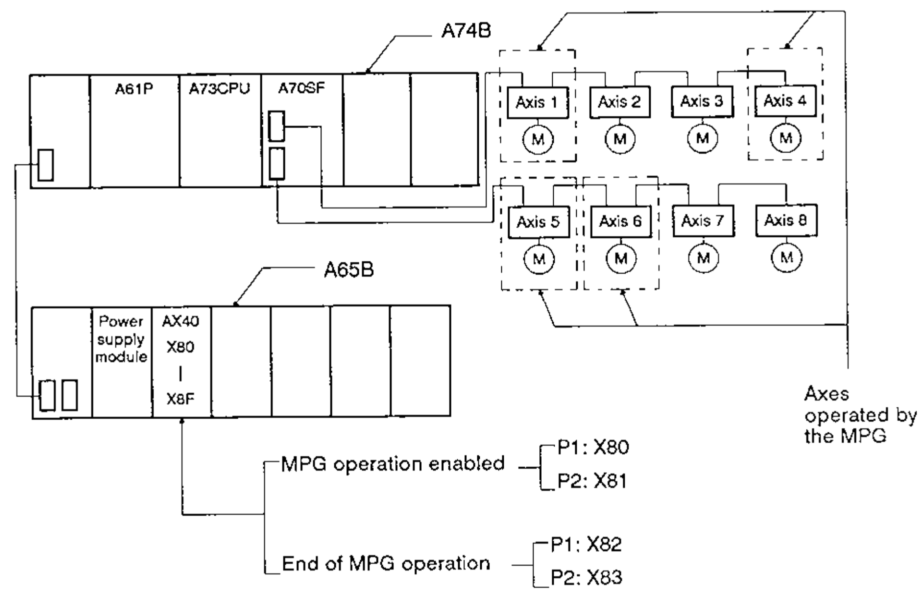
[MPG Operation procedure]

The procedure to execute MPG operation is indicated below.



[Program example]

(1) System configuration



(2) MPG operation conditions

(a) MPG connection positions: P1, P2

(b) MPG operation axes

1) P1 MPG: Axis 1

2) P2 MPG: Axis 4, Axis 5, Axis 6

(c) Pulse multiplication ratio

1) Axis 1: 100

2) Axis 4: 2

3) Axis 5: 2

4) Axis 6: 2

(d) MPG operation enabled

1) P1: Leading edge of X80

2) P2: Leading edge of X81

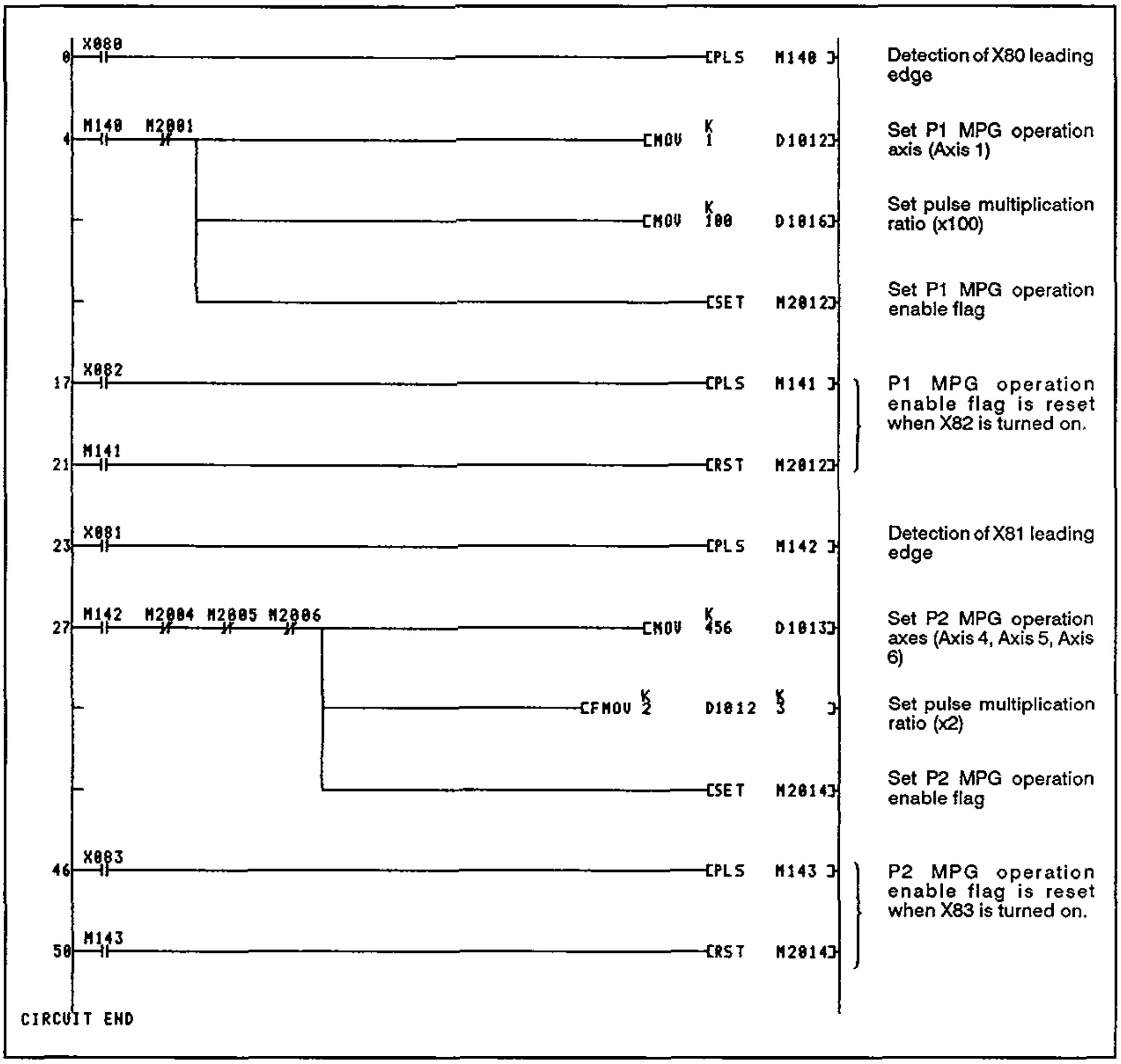
(e) End of MPG operation

1) P1: Leading edge of X82

2) P2: Leading edge of X83

(2) Sequence program

The sequence program used to execute MPG operation is indicated below.



7.18 Zero Return

(1) Execute zero return when a confirmation of the machine zero point is required, such as when power is turned on.

(2) For zero return operation, the following three methods are available.

(a) Near-zero point dog ⎤ Used for systems other than

(b) Count ⎦ the absolute system

(c) Data set:

Recommended for the absolute system

(3) To execute zero return operation, zero return data must be set for each axis.

7.18.1 Zero return data

Zero return data is used to execute zero return.

Table 7.17 Zero Return Data List

No.	Item	Setting Range								Default	Remarks	Refer to
		mm		inch		degree		PULSE		Initial value		
		Setting range	Unit	Setting range	Unit	Setting range	Unit	Setting range	Unit			
1	Zero return direction	1: Reverse direction (addresses decreasing) 0: Forward direction (addresses increasing)								0	Set the direction in which zero return is executed. When zero return is started, the axis moves in the designated direction.	—
2	Zero return method	0: Near-zero point dog 1: Count 2: Data set								0	Set the required zero return method Servo amplifier not absolute: Near-zero point dog and count Absolute servo amplifier: Data set	—
3	Zero point address	-2^{31} to $2^{31}-1$	X10^{-1} μm	-2^{31} to $2^{31}-1$	X10^{-5} inch	0 to 35999999	X10^{-5} degree	-2^{31} to 2^{31}	PLS	0	Set the present position data of the zero point after the completion of the zero return. It is recommended to set the zero point at either upper or lower limit of stroke.	—
4	Zero return velocity	0.01 to 6000000.00	mm / min	0.01 to 6000000.000	inch / min	0.01 to 6000000.000	degree / min	1 to 1000000	PLS / sec	1	Set the velocity for zero return operation.	—
5	Creep velocity	0.01 to 6000000.00	mm / min	0.01 to 6000000.000	inch / min	0.01 to 6000000.000	degree / min	1 to 1000000	PLS / sec	1	Set the creep velocity (low velocity to be used just before the stop after passing the near-zero point dog).	—
6	Axis travel distance after passing the near-zero point dog	0 to $2^{31}-1$	X10^{-1} μm	0 to $2^{31}-1$	X10^{-5}	0 to $2^{31}-1$	X10^{-5} degree	0 to $2^{31}-1$	PLS	0	For the count zero return, set the axis travel distance to reach the zero point after passing the near-zero point dog. Set a distance greater than the deceleration distance necessary for the zero return velocity.	7-114
7	Parameter block designation	1 to 16								1	Set the parameter block numbers used for zero return. See Section 5.3.5.	—

(1) Setting distance after passing near-zero point dog

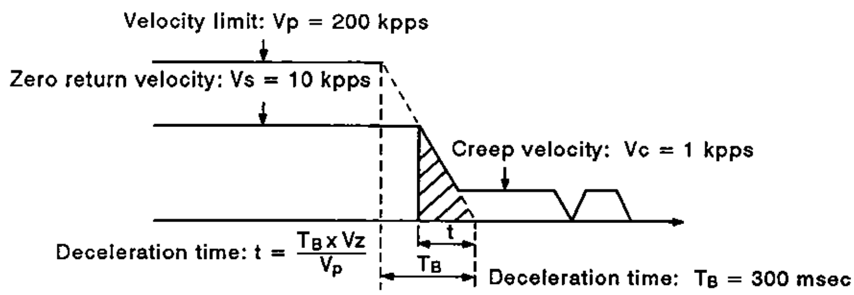
(a) The data is set in the count zero return operation; set the distance to reach the zero point after passing the near-zero point dog.

(b) The reference point appearing first after the axis has moved by the set distance after passing the near-zero point is established as the zero point.

(c) Set a distance greater than the deceleration distance required for the set zero return velocity.

Example:

Calculate the deceleration distance required for the zero return velocity from the velocity limit, zero return velocity, creep velocity, and deceleration time.



Deceleration distance (▨ area in the illustration)

$$= \frac{Vz \times t}{1000 \times 2}$$

$$= \frac{Vz}{2000} \times \frac{T_B \times Vz}{Vp}$$

$$= \frac{10k \times 300 \times 10k}{2000 \times 200k}$$

$$= 75$$

Set a value greater than "75".

7.18.2 Zero return mode when the MR-SB servo amplifier is used

When the MR-SB servo amplifier is used, zero return is possible in any of the following methods: near-zero point dog, count, and data set.

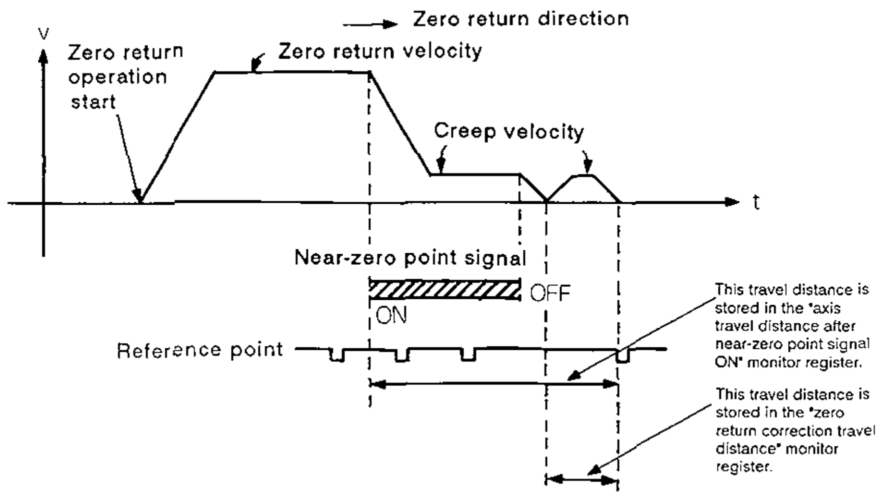
Near-zero point zero return

(1) General

With near-zero point dog zero return operation, the reference point appearing first after the signal triggered by the near-zero point dog is turned off is established as the zero point.

(2) Zero return sequence

The zero return sequence is indicated below.



* The axis decelerates and stops after the signal triggered by the near-zero point dog is turned off. From that point, positioning is executed at the reference point.
(The distance to the reference point is obtained based on the servo data.)

Fig. 7.17 Zero Return Operation Sequence - Near-zero Point Dog

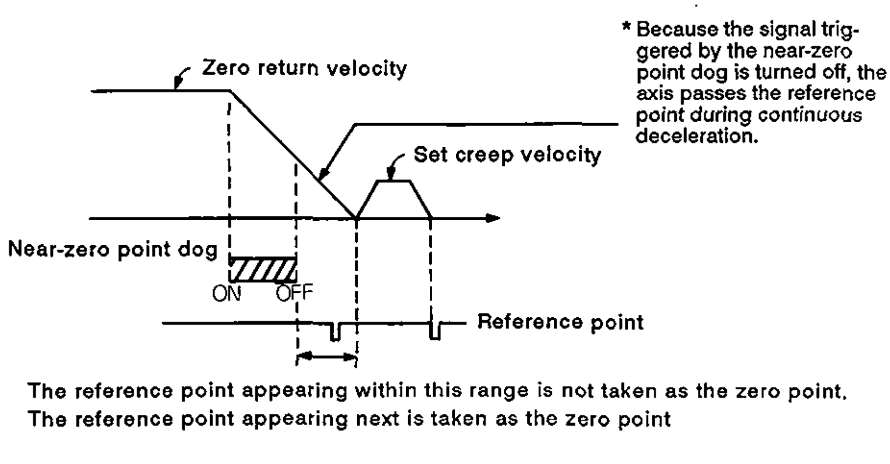
(3) Zero return execution

Zero return is executed with the servo program in Section 7.18.4.

(4) Caution

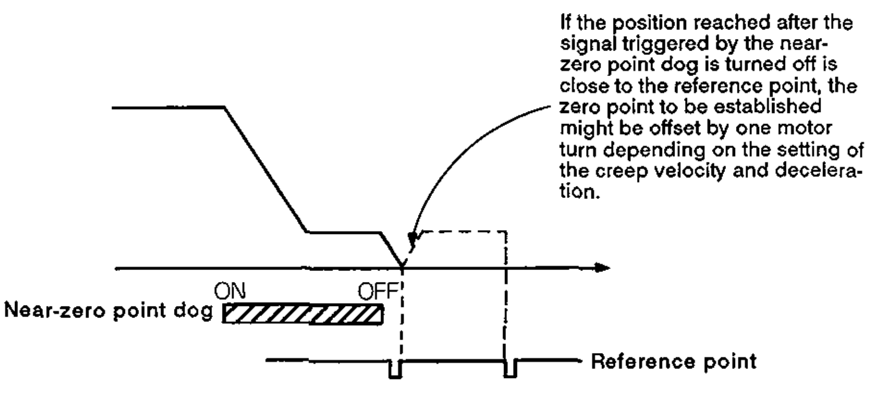
Maintain the signal triggered by the near-zero point dog until the velocity is decelerated from the zero return velocity to creep velocity.

(a) If this signal is turned off before the velocity is decelerated to creep velocity, the axis will continuously decelerate at the same rate and stop; the next reference point is established as the zero point.



(b) Adjust the near-zero point dog position so that the "zero return correction travel distance" will be equivalent to a half turn of the motor.

If this setting is smaller than the distance corresponding to a half turn of the motor, the zero position might be offset by one motor turn.



IMPORTANT

In the following cases, execute zero return operation after returning the axis to a position away from the near-zero point dog position by jog operation, etc. Zero return operation cannot be executed without returning the axis as indicated above.

(a) Zero return operation from a position where the signal triggered by the near-zero point dog is turned from on to off.

(b) Zero return operation when the power is turned on after the completion of zero return operation.

Count zero return

(1) General

The reference point appearing first after the axis has travelled the designated distance (axis travel distance after near-zero point signal ON) is taken as the zero point.

The travel distance after the signal triggered by the near-zero point dog is turned on is set as the zero return data. See Section 7.18.1.

(2) Zero return sequence

The zero return sequence is indicated below.

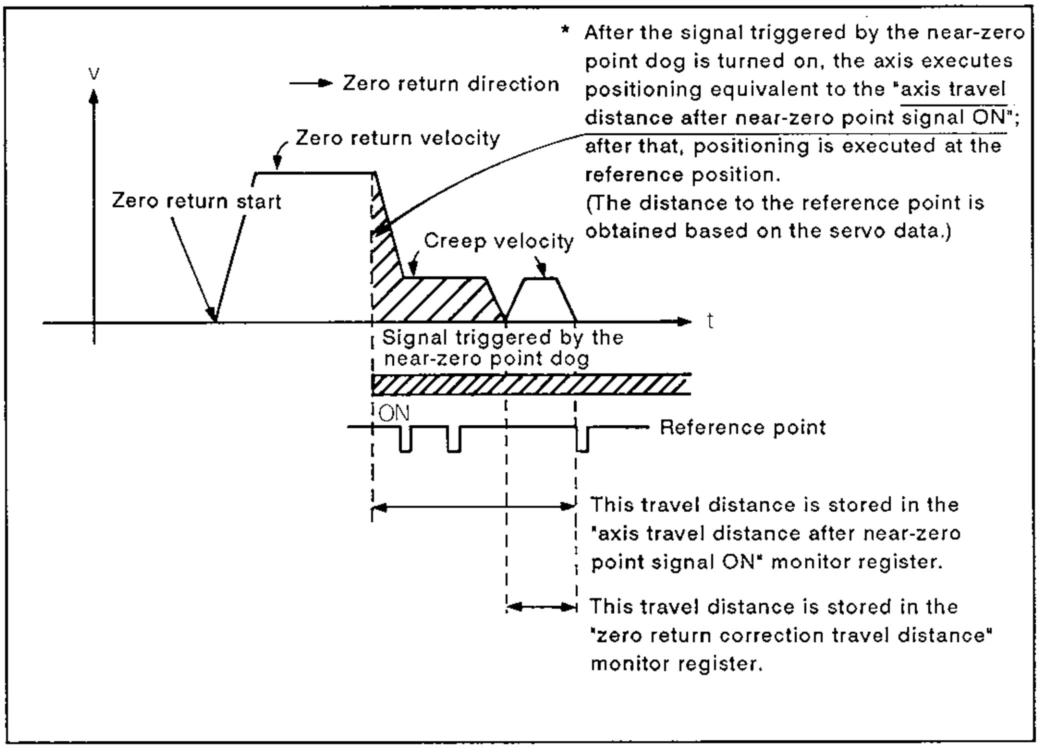


Fig. 7.18 Zero Return Operation Sequence – Count

(3) Zero return execution

Zero return is executed with the servo program in Section 7.18.4.

(4) Caution

(a) Turn off the signal triggered by the near-zero point dog to allow sufficient distance from the zero point.

(b) In this method, zero return can be started while the near-zero point dog triggered signal is on or it can be started continuously.

If zero return operation is started continuously or while the near-zero point dog triggered signal is on, the axis is returned to a position where the near-zero point signal is off. After this, zero return operation is executed.

Data set zero return

(1) General

This zero return method is only used in the absolute system; it does not use the near-zero point dog.

(2) Zero return sequence

The present position when zero return is executed with the DSFRP command is taken as the zero point address.

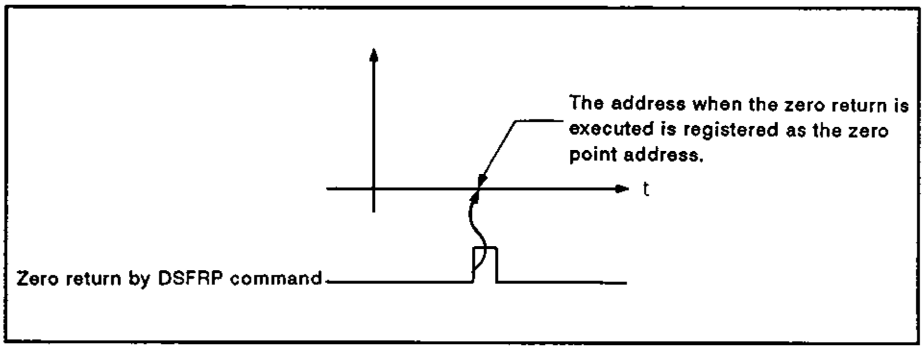


Fig. 7.19 Zero Return Operation Sequence – Data set

(3) Zero return execution

The zero return is executed with the servo program in Section 7.18.4.

(4) Caution

(a) It is necessary to pass the reference point before executing zero return operation after turning on the power.

If the zero return is executed without passing the reference point, an error (reference point not passing error) will occur. If this occurs, reset the error and execute jog operation so that the motor rotates more than one turn. After this, execute zero return again.

Check whether or not the reference point has been passed with the zero point passing signal (X0[ ]6).

(b) The data-set zero return operation, if attempted with a system other than the absolute system, has the same effect as the present position data change function.

(c) The address when the zero return is executed is registered as the zero point address.

(d) For other zero return data, set a value within the allowable setting range.

In the data-set zero return operation, present position data when the power has been turned off and that when the power is turned on show an error of approximately 1% (maximum) in motor rotation. This error is caused by the linear error of the resolver, which is used for position sensing within one motor turn. It is compensated when the position sensing is passed from the resolver to the encoder when the Z-phase is first passed after the power is turned on.

IB (NA) 66233-A

7.18.3 Zero return when a general-purpose servo amplifier is used

When a general-purpose servo amplifier is used, either near-zero point dog and count zero return is possible. Note that the data-set zero return is not allowed.

Near-zero point zero return

(1) General

With near-zero point dog zero return operation, the reference point appearing first after the signal triggered by the near-zero point dog is turned off is established as the zero point.

(2) Zero return sequence

The zero return sequence is indicated below.

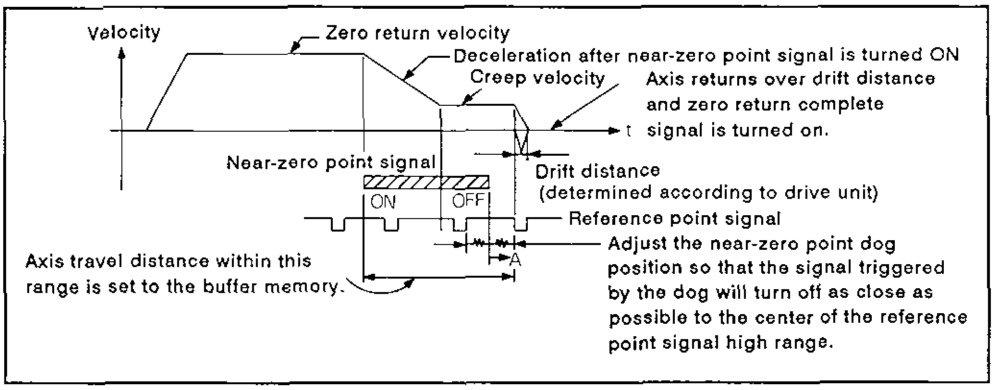


Fig. 7.20 Zero Return Operation Sequence - Near-zero Point Dog

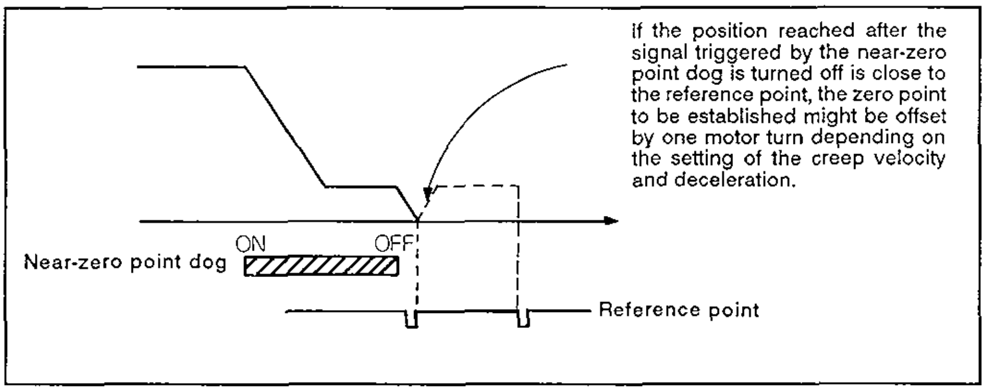
(3) Zero return execution

Zero return is executed by the servo program in Section 7.18.4.

(4) Caution

Adjust the near-zero point dog position so that the "zero return correction travel distance" will be equivalent to a half turn of the motor.

If this setting is smaller than the distance corresponding to a half turn of the motor, the zero position might be offset by one motor turn.



IMPORTANT

In the following cases, execute zero return operation after returning the axis to a position away from the near-zero point dog position by jog operation, etc. It is not possible to execute zero return operation without returning the axis as indicated above.

(a) Zero return operation from a position where the signal triggered by the near-zero point dog is turned from on to off.

(b) Zero return operation when the power is turned on after the completion of zero return operation.

7.18.4 Zero return servo program

Zero return operation is called by the ZERO servo command.

Servo Command			[ZERO]
Positioning Control Mode			—
Number of axes to be controlled			1
Processing Contents	Common	Parameter block No.	
		Axis	O
		Address/travel distance	
		Command velocity	
		Dwell time	
		M code	
		Torque limit	
	Circular interpolation	Assist-point	
		Radius	
		Center	
	Parameter block	Control units	
		Velocity limit	
		Acceleration time	
		Deceleration time	
		Immediate stop deceleration time	
		Torque limit	
		STOP input deceleration processing	
		Circular interpolation allowable error range	
	Others		
Velocity Change			

[Control details]

(1) Zero return operation is executed with the zero return method designated by the zero return data (Section 7.18.1).

For details on zero return, refer to the following sections.

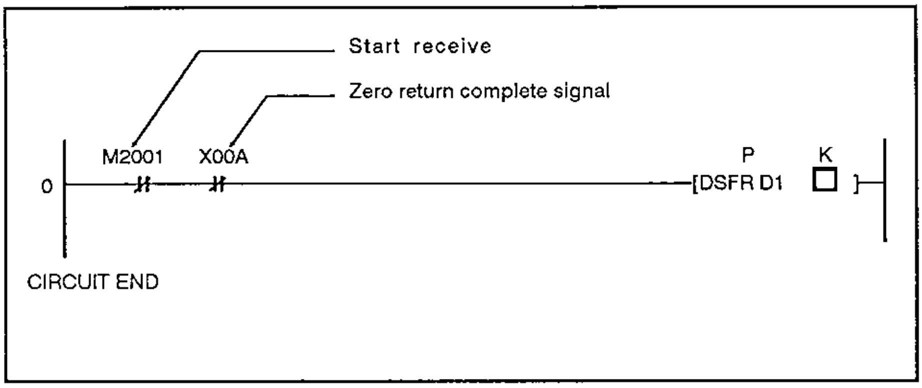
(a) With the MR-SB servo amplifier : Section 7.18.2

(b) With a general-purpose servo amplifier : Section 7.18.3

[Caution]

(1) If zero return is executed with the following circuit before the PCPU read flag (M9074) is set and after the PC ready flag (M2000) has been set, the zero return request is given after the completion of zero return operation.

To execute zero return, use M9074 for interlock. See the program example.



Count zero return

(1) General

The reference point appearing after the axis has been moved by the designated distance (axis travel distance after near-zero point signal ON) is taken as the zero point.

The travel distance after the signal triggered by the near-zero point dog is turned on is set as the zero return data. See Section 7.18.1.

(2) Zero return sequence

The sequence of zero return is indicated below.

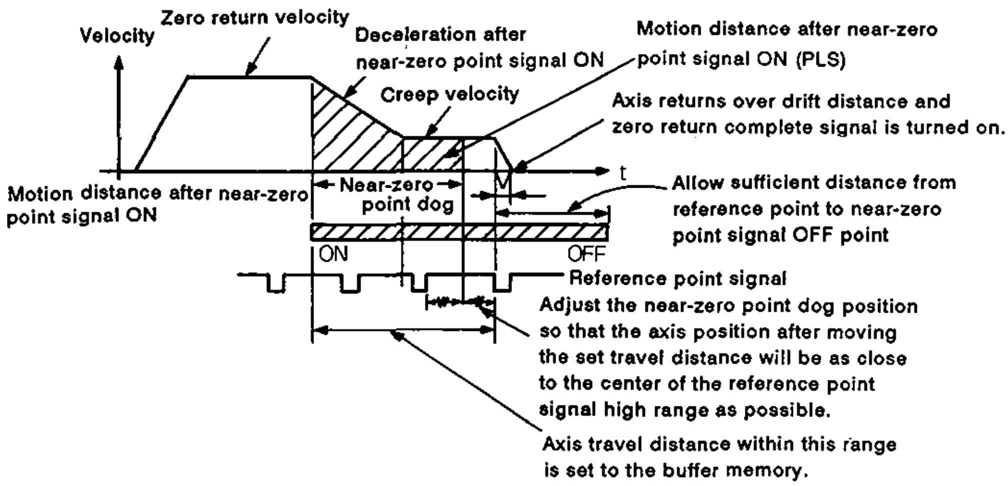


Fig. 7.21 Zero Return Operation Sequence – Count

(3) Zero return execution

Zero return is executed by the servo program in Section 7.18.4.

(4) Caution

(a) Turn off the near-zero point signal to provide sufficient distance from the zero point.

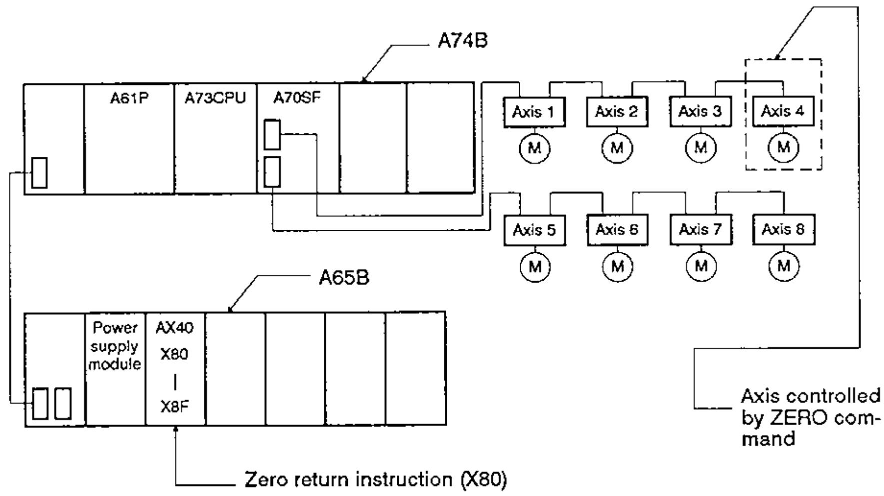
(b) In this method, zero return can be started while the near-zero point signal is on or it can be started continuously.

If the zero return operation is started while the near-zero point signal is on or continuous, the axis is returned to a position where the near-zero point signal is off. Zero return operation is then executed.

[Program example]

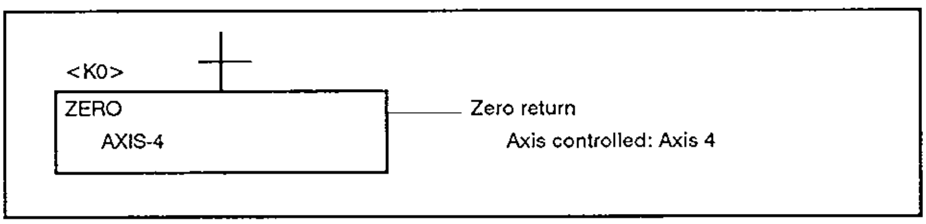
Servo program No. 0 that calls zero return operation is explained below.

(1) System configuration



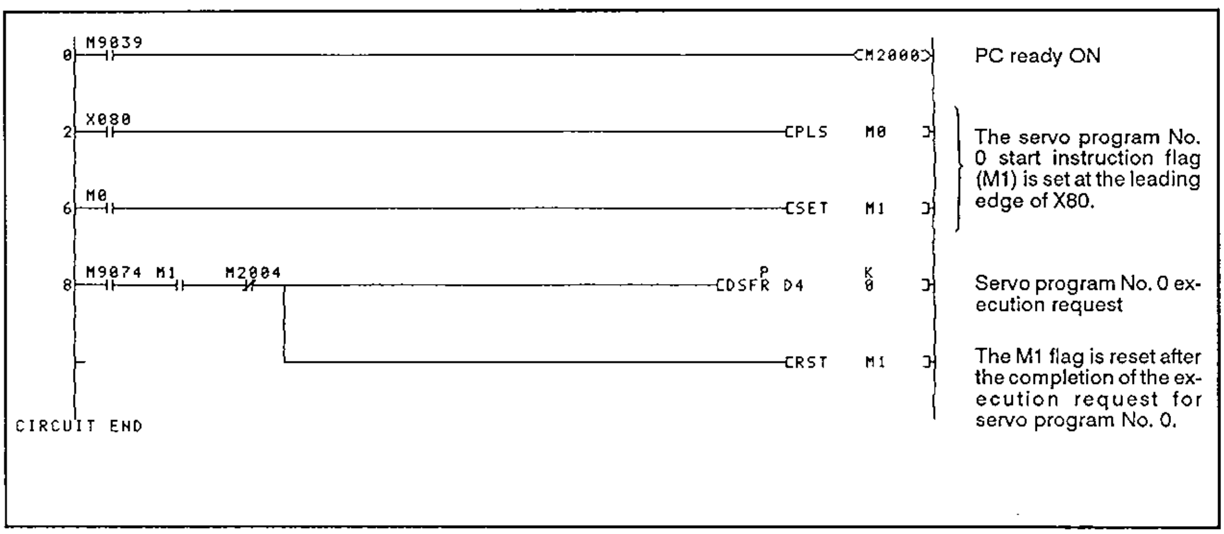
(2) Servo program example

Servo program No. 0, which calls zero return, is indicated below.



(3) Sequence program

The sequence program used to execute the servo program is indicated below.



8. AUXILIARY AND APPLICATION FUNCTIONS

The following functions used for positioning control with the A73CPU are explained in this section.

8.1 Limit Switch Output Function

The limit switch output function outputs the ON/OFF signal, corresponding to the positioning address set for each axis, from the AY42-S1 output module.

8.1.1 Limit switch output data

Item	Setting Contents		Default	Remark
ON/OFF point setting	* -2^{31} to $2^{31}-1$ ($\times 10^{-1}\mu m \times 10^{5}$ inch, PLS) * 0 to 35999999 ($\times 10^{-5}$ degree)	Unit $\times 10^{-1}\mu m$ $\times 10^{-5}$ inch $\times 10^{-5}$ degree PLS	0	Setting is possible for up to ten points for each axis.

8.1.2 Limit switch output function

[Control details]

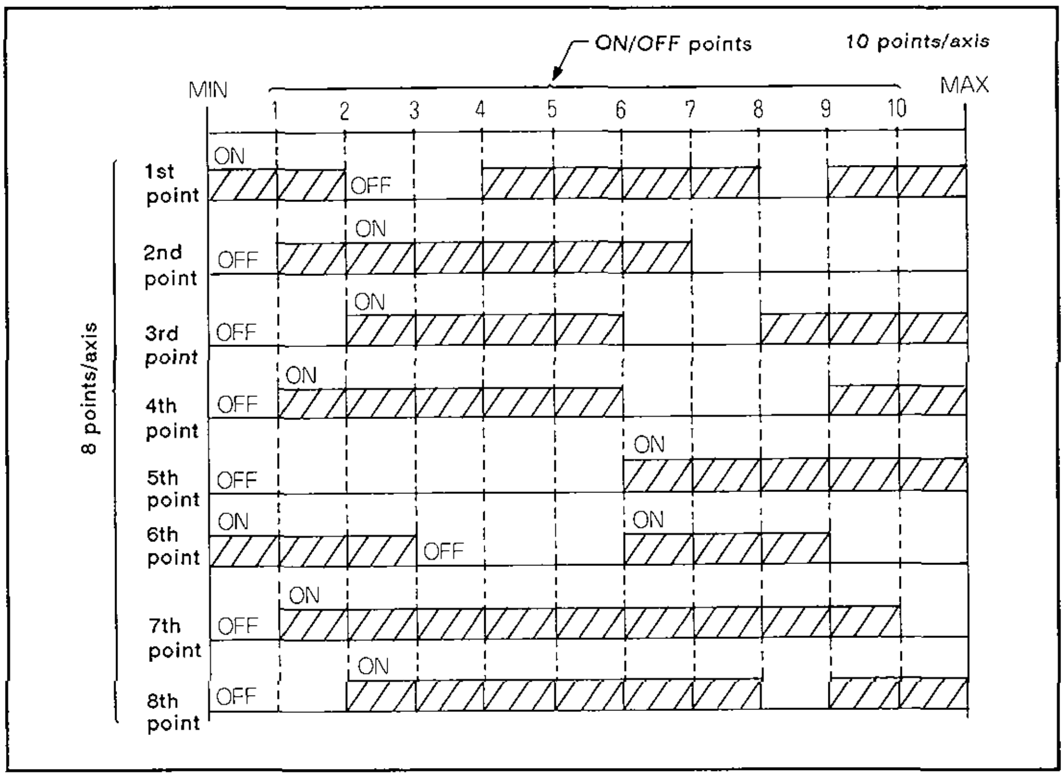
(1) The limit switch output function outputs the ON/OFF pattern at the set addresses from the AY42 module.

To execute the limit switch output function, it is necessary to set the ON/OFF pattern and the point address where the limit switches should be turned on/off with a peripheral device. Note that this setting is not possible with a sequence program.

The number of limit switch output points and the ON/OFF points are indicated below.

(a) Limit switch output points.......8 points/axis
64 points in total

(b) ON/OFF points..........................10 points/axis



(2) Limit switch output enable/disable setting

The limit switch output function is enabled or disabled in units of axes and points using the following devices.

Table 8.1 Limit Switch Output Enable/Disable Setting

Setting Data/Device	Setting Unit	Processing	Timing when Setting Data Becomes Effective
Setting of "limit switch used/not used" (fixed parameter)	Axis	Used: The corresponding axis can output the set ON/OFF pattern. Not used: All outputs of the corresponding axis are ON.	1) At the PC ready (M2000) flag leading edge 2) When the test mode is set.
Limit switch output enable signal (Yn6)	Axis	ON: The corresponding axis outputs the set ON/OFF pattern according to the limit switch output disable data storing register (D1008 to D1011). OFF: All outputs of the corresponding axis are OFF.	When the "limit switch output used/not used" setting is "not used".
Limit switch output disable data storage register (D1008 to D1011)	Point	Disable (bit: "1"): The output corresponding to the bits for which "1" is set is OFF. Enable (bit "0"): The output corresponding to the bits for which "0" is set is the set ON/OFF pattern.	When Yn6 ON

REMARKS

The information in Table 8.1 is effective during peripheral device test mode operation.

(3) Cautions

(a) The limit switch output is given according to the "command position data" of each axis at the leading edge of the PC ready (M2000) flag while the PCPU ready flag (M9094) stays on.

When the PCPU read flag (M9074) is reset, all output points are set OFF.

(b) While the PCPU ready flag is on, the limit switch output is given according to Yn6, even when the command position data is outside the stroke limit.

Therefore, take an interlock with a sequence program so that the Yn6 is turned on only while an axis is within the stroke limit.

(c) The limit switch output enable/disable setting is possible with a peripheral device while the A73CPU is in the stop state (PCPU ready flag M7094 OFF) or during the peripheral device test mode operation.

8.2 M Code Output Function

The M code is the code number (1 to 255) which can be set for each mode of positioning control.

While positioning control is being executed, the M code can be read by a sequence program to give a variety of instructions, such as confirming the servo program currently executed and calling miscellaneous operations (clamp, starting drill rotation, tool change, etc.).

(1) Setting an M code

M codes can be set when programming or editing a servo program with a peripheral device.

Each servo program can contain one M code.

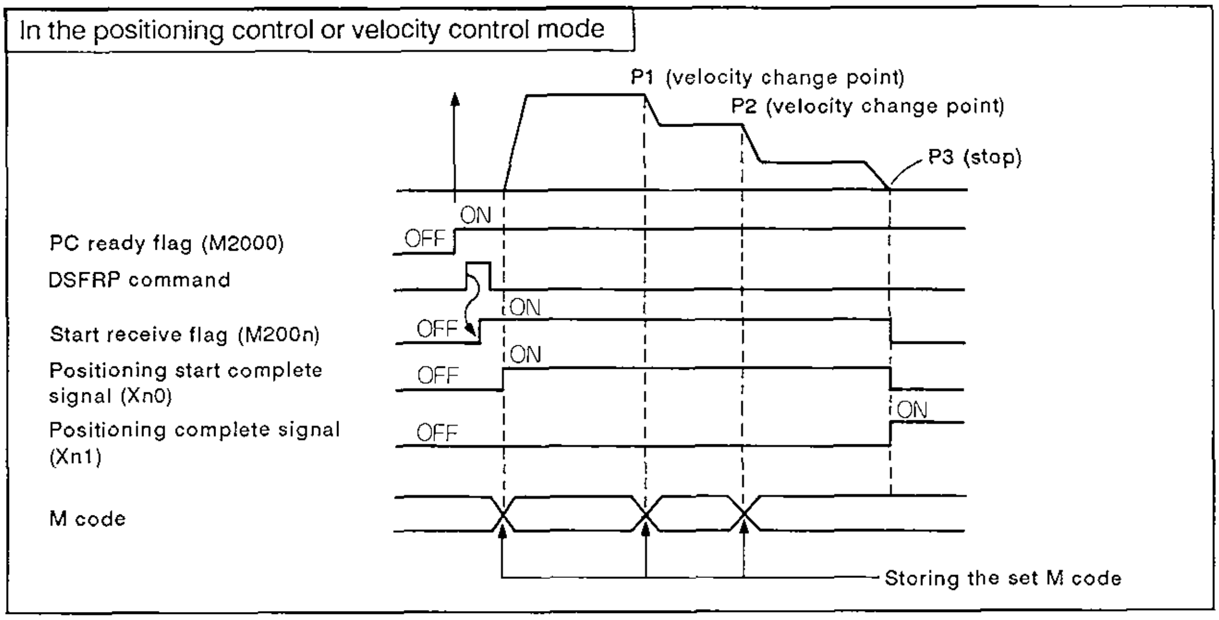
(2) Timing for storing and reading M code

(a) An M code is stored to the M code storage register of the designated axis when the positioning start is completed or at the designated point (in velocity change control or constant velocity control).

In interpolation control, the M code is stored for all axes which are used for interpolation control.

(b) To read the M code after the completion of positioning start, use the positioning start complete signal (Xn0) for the read instruction.

(c) To read the M code after the completion of positioning, use the positioning complete signal (Xn1) for the read instruction.



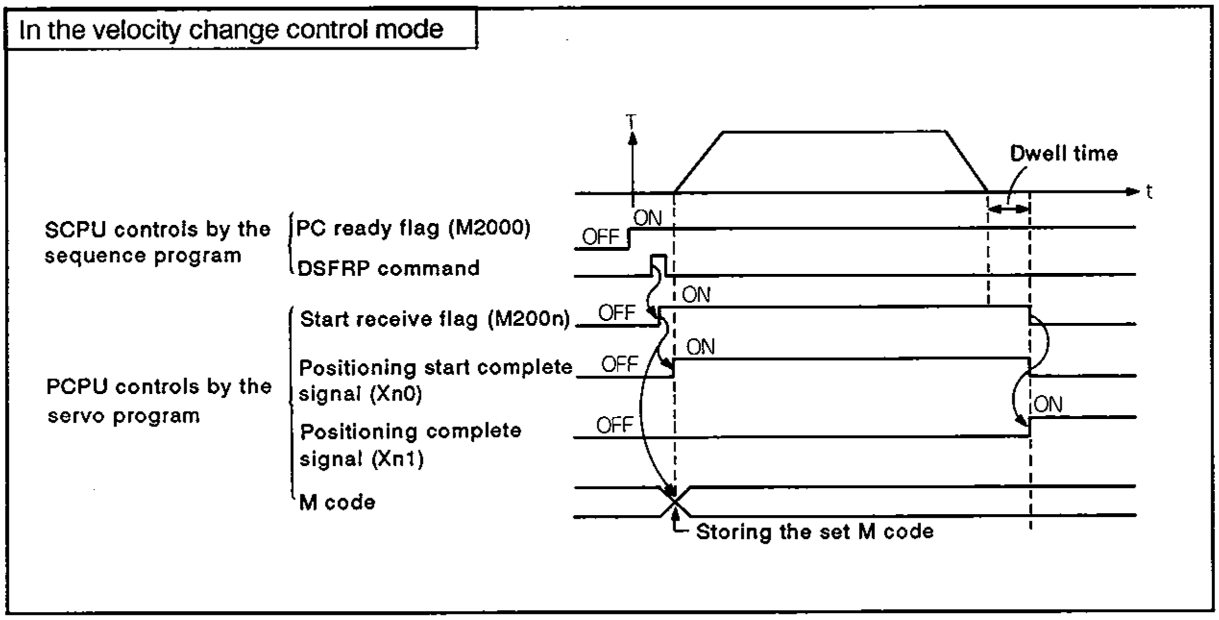


Fig. 8.1 Stroring and Reading the M Code

(3) Program example

(a) A sample sequence program is indicated below.

1) Controlled axis: Axis 3

2) Processing at the start of positioning called by an M code number is output at Y90 to Y98 in BCD code

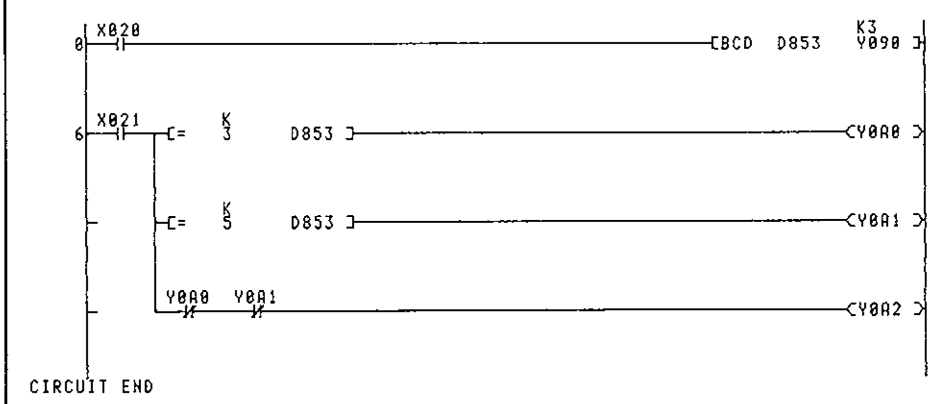
3) Processing after the completion of positioning called by the M code

i) When the M code is "3":
YA0 is turned ON.

ii) When the M code is "5":
YA1 is turned ON.

iii) When the M code is not "3" or "5":
YA2 is turned ON.

(b) The sequence program is indicated below.



The M code for the 3 axes is output to Y90 to Y98 in BCD code when the positioning start complete flag (Xn0) is set.

At the completion of positioning, YA0 to YA2 are set ON.

8.3 Backlash Compensation Function

Backlash is the total amount of play in the drive mechanism.

By setting the backlash amount, the backlash compensation function generates axis feed pulses corresponding to the set backlash amount each time the axis motion direction changes in positioning control, jog operations or MPG operation.

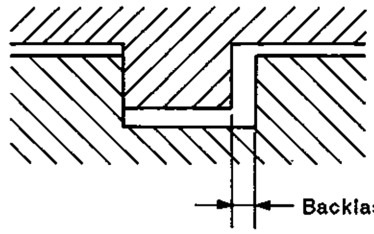


Fig. 8.2 Backlash Compensation

(1) Setting backlash compensation amount

The backlash compensation amount is fixed parameter data and can be set for each axis using a peripheral device.

Setting range varies depending on the units to be used:

Units: mm, inches, degrees

* 0 to 65535

$$* \ 0 \leqq \frac{\text{(Backlash compensation amount)}}{\text{(Travel distance per pulse)}} \leqq 255$$

(fraction is discarded)

Units: PLUSE

* 0 to 255

(2) Backlash compensation processing

Table 8.2 Backlash Compensation Processing

Conditions	Processing
At the first start after power is turned on	Motion direction = Zero return direction: Backlash is not compensated Motion direction ≠ Zero return direction: Backlash is compensated
At the start of jog operation	When axis motion direction is changed, backlash compensation amount is fed in a single jog operation.
At the start of positioning operation	Backlash is compensated when motion direction is changed.
During MPG operation	Backlash is compensated when motion direction is changed.
At the start of zero return operation	Backlash compensation amount becomes effective after zero return has started.
Absolute value system	The state when the power is turned off is stored.

POINT

(1) The number of pulses generated for backlash compensation is not added to the command position data.

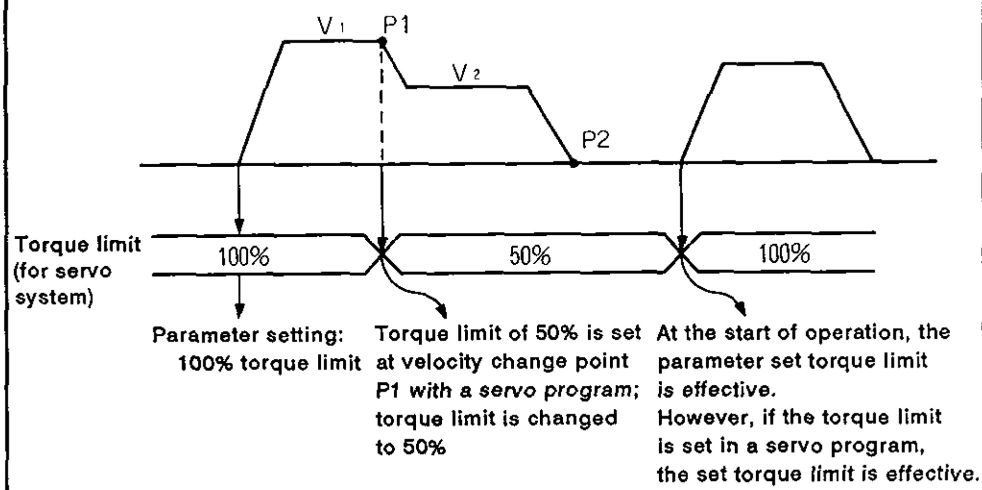
(2) When the backlash amount is changed, zero return operation must be executed.

If zero return is not executed, the previous backlash compensation amount remains unchanged.

8.4 Torque Limit Function

The torque limit function limits the torque generated by the servo motor within the set range.

If the torque required to execute the designated positioning control exceeds the limit, control is executed at a torque within the set range.



(1) Setting range

1 to 500% of the rated torque

(2) Setting method

(a) Set the torque limit following the procedure indicated below.

1) Set the torque limit with the parameter block (see Section 4.3).

By designating the parameter block number to be used by the servo program, the set torque limit is applied to the positioning control, limiting the servo motor generation torque.

2) Set the torque limit in a servo program. The servo motor torque is limited at this setting when the servo program is executed.

POINT

The torque limit function is only effective only when an MR-SB servo amplifier is used.

When a general-purpose servo amplifier is used, the torque limit function cannot be used.

8.5 Electronic Gear Function

The electronic gear function allows the mechanical axis travel distance per pulse to be changed as required by setting parameter data (travel distance per pulse). See Section 4.1.1.

Due to the electronic gear function, it is not necessary to select an encoder to match the mechanical system, permitting flexibility in designing the servo system.

The electronic gear function is effective for positioning control, zero return control, jog operation, and MPG operation.

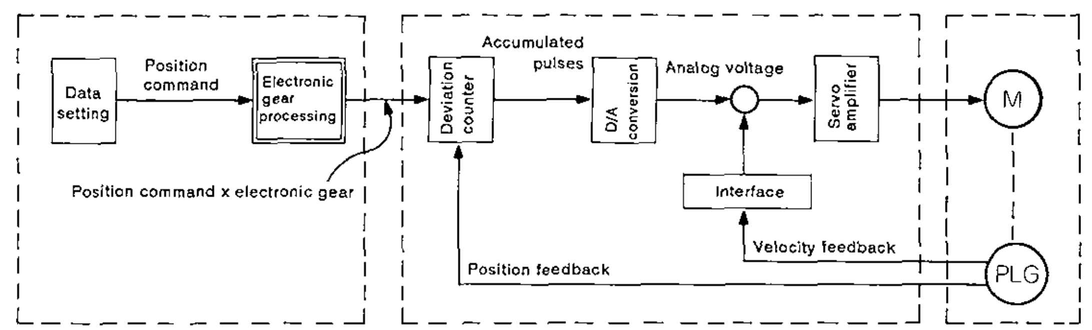


Fig. 8.3 Electronic Gear Processing

By setting the electronic gear, the relationship between the command velocity (positioning velocity set by the servo program) and the actual velocity (actual positioning velocity) is indicated below.

(a) Electronic gear setting = 1

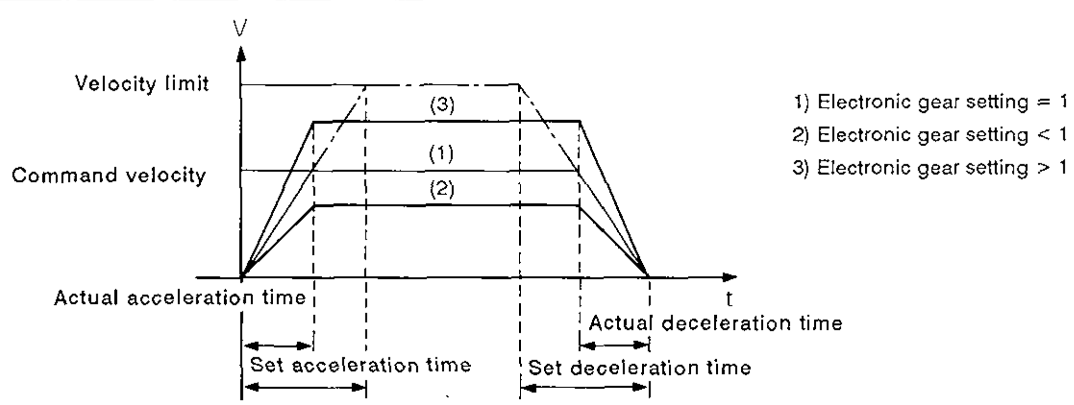
Command velocity = Actual velocity

(b) Electronic gear setting < 1

Command velocity < Actual velocity

(c) Electronic gear setting > 1

Command velocity > Actual velocity



The data in the designated parameter block is used for the velocity limit, acceleration time, and deceleration time.

Fig. 8.4 The Relationship between Command Velocity and Actual Velocity

8.6 Absolute System

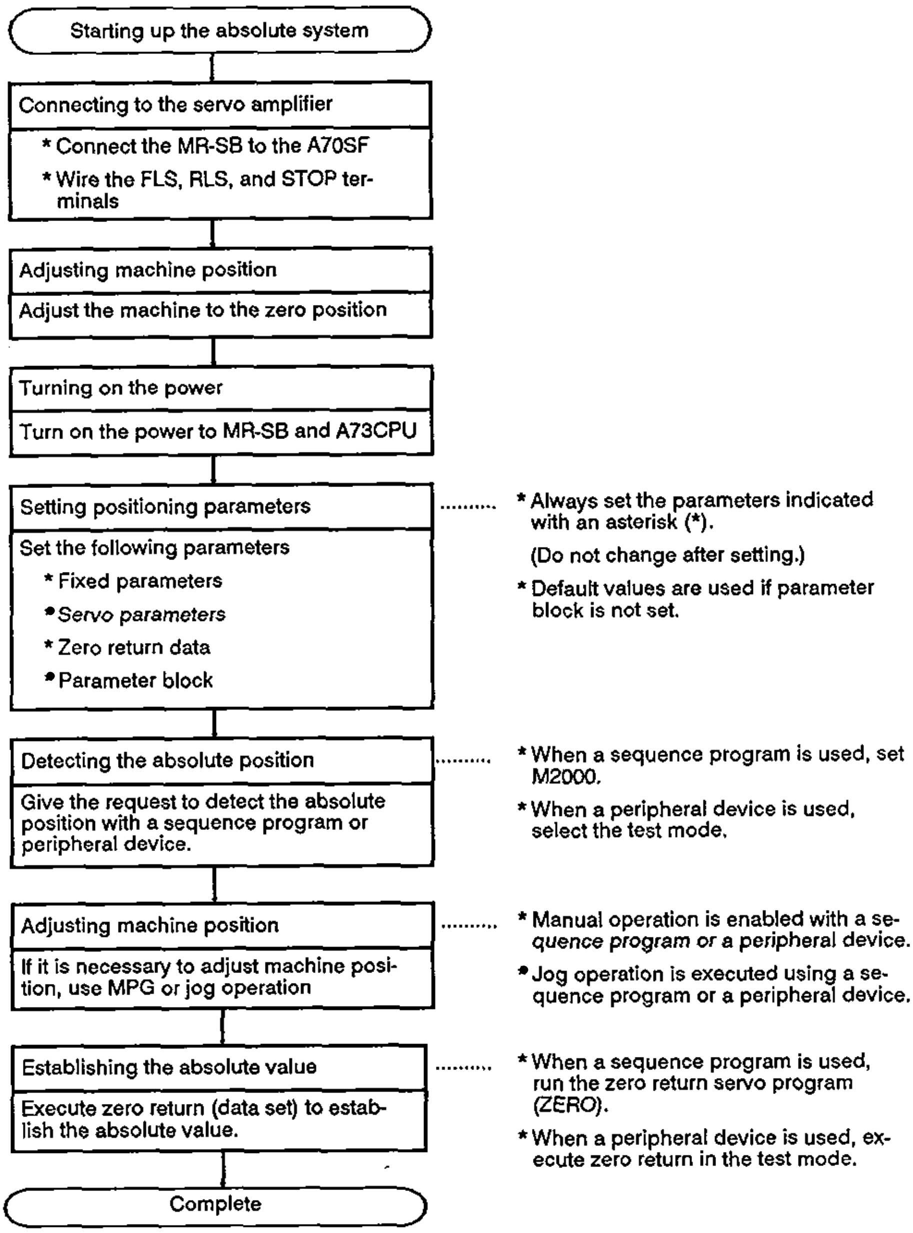
By using an MR-SB servo amplifier, positioning control with absolute position data is possible.

When a general-purpose servo amplifier is used, positioning control with absolute position data is not possible.

With the absolute system, after the machine position is set when the system is initially set up, it is not necessary to execute zero return because the absolute position of an axis is detected when the power is turned on.

Machine position is established by zero return called by the sequence program or using a peripheral device.

(1) Starting up the absolute system



(2) The absolute system might lose the absolute position in the following cases.

In such a case, establish the absolute position by executing zero return or changing the present position data after setting the machine to the zero position.

(a) When an absolute position encoder is removed or changed.

(b) When the cable connecting the absolute position encoder to the servo amplifier is removed or changed.

(c) When the servo amplifier is changed.

(d) When a servo battery alarm occurs (detected when the servo amplifier power is turned on).

(e) When mechanical system is misaligned due to a collision.

8.7 Velocity Change Function

The function changes velocity during positioning control (excluding circular interpolation) and jog operation.

Velocity is changed with the DSFLP command in a sequence program or in the test mode of a peripheral device.

For the procedure for changing velocity with the test mode of a peripheral device, refer to the Operating Manual for the peripheral device used.

[Control details]

(1) The velocity of the axis started is forcibly changed to the velocity set in the velocity change register.

(2) The velocity is changed with the DSFLP command. For details, see Section 5.3.

[Data setting]

(1) The velocity change registers for each axis are indicated below.

Axis No.	Velocity Change Register	
	Upper Digit	Lower Digit
1	D963	D962
2	D969	D968
3	D975	D974
4	D979	D980
5	D987	D986
6	D993	D992
7	D999	D998
8	D1005	D1004

(2) The setting range of the velocity is indicated below.

Unit / Item	mm		inch		degree		PULSE	
	Setting range	Unit	Setting range	Unit	Setting range	Unit	Setting range	Unit
Velocity change value	1 to 600000000	$\times 10^{-2}$ mm/min	1 to 600000000	$\times 10^{-3}$ inch/min	1 to 600000000	$\times 10^{-3}$ degree/min	1 to 1000000	PLS/sec

POINT

To set the jog operation velocity in the sequence program, set it in the jog operation velocity setting register after multiplying it by 100 (for mm) or 1000 (for inches or degrees).

Example: To designate jog operation velocity of 10000.00 mm/min, set "1000000" in the jog operation velocity setting register.

[Caution]

(1) When changing the velocity during linear interpolation control of two or three axes, change the velocity for one of the two or three axes.

(2) The unit of velocity is the control unit set in the parameter block.

(3) In the test mode of a peripheral device, values in the velocity change registers are ignored.

(4) The velocity is not changed if any of the following errors occur. The conditions necessary to change the velocity are checked when the DSFLP command is executed.

<table>
<tr><th colspan="2">Error Contents</th><th>Error Processing</th><th>Error Code</th></tr>
<tr><td rowspan="5">Data setting error</td><td>Axis number setting is not between 1 and 8.</td><td rowspan="4">The error step is stored in D9010 and D9011.
M9010 and M9011 are set.</td><td rowspan="4">—</td></tr>
<tr><td>Axis number is set indirectly with index modifier.</td></tr>
<tr><td>The data set for present position data change is not 0 or 1.</td></tr>
<tr><td>The present position data change is set indirectly with the index modifier.</td></tr>
<tr><td>The set velocity is outside the allowable setting range.</td><td>The error detection flag (Xn7) set.
The error code indicated in the right column is stored in the minor error code register for the axis in question.</td><td>305</td></tr>
<tr><td rowspan="4">Error when changing velocity</td><td>The designated axis is executing zero return.</td><td rowspan="4">The error detection flag (Xn7) set.
The error code indicated in the right column is stored in the minor error code register for the axis in question.</td><td>301</td></tr>
<tr><td>The designated axis is executing circular interpolation.</td><td>302</td></tr>
<tr><td>The designated axis is decelerated automatically during positioning control.</td><td>303</td></tr>
<tr><td>The designated axis is decelerated with the jog operation signal turned off.</td><td>304</td></tr>
</table>

(5) Set values for velocity changing are ignored in the following cases. (No error occurs.)

(a) During deceleration controlled by a STOP instruction.

(b) When axis movement is being stopped.

(c) During the MPG operation.

IB (NA) 66233-A

[Timing chart]

The operation timing for executing velocity change is indicated below.

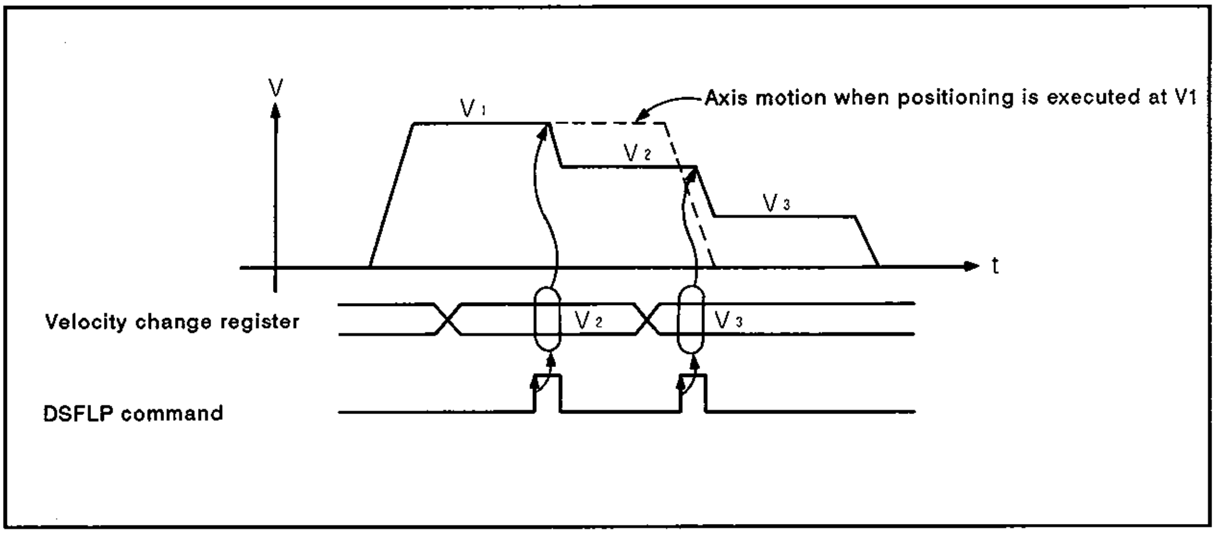


Fig. 8.5 Timing Chart of Velocity Change Operation

[Program example]

The program which calls velocity change control is indicated below.

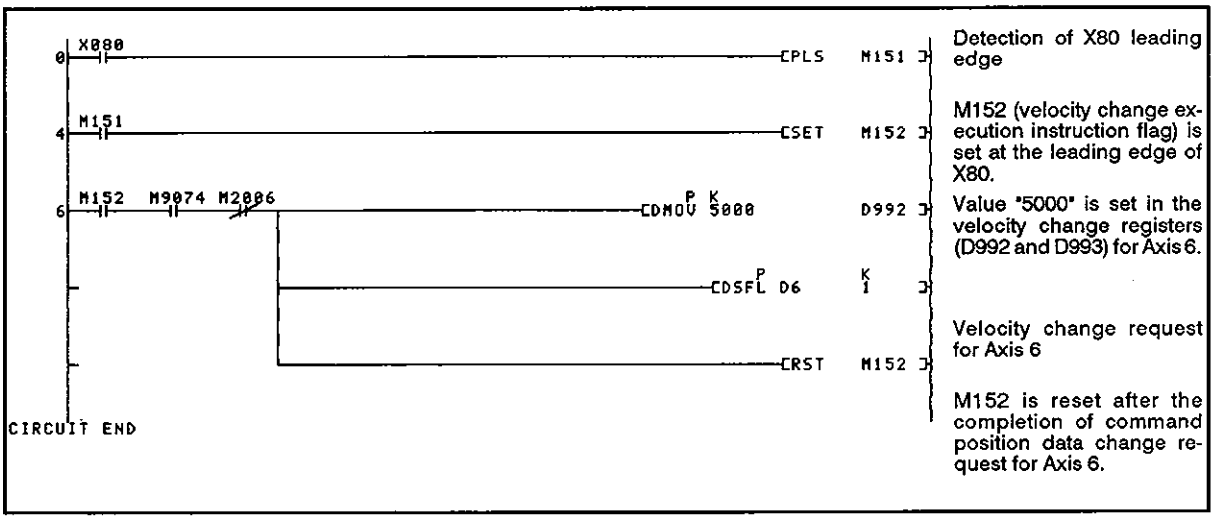
(1) Velocity change conditions

(a) Axis .... Axis 6

(b) New velocity 5000

(c) Velocity change instruction....... X80

(2) Sequence program



8.8 Command Position Data Change Function

This function is used to change the command position data.

To change the command position data of a stopped axis, use the DSFLP command in a sequence program. It is also possible to change it in the test mode of a peripheral device. For details, see the Operating Manual for the peripheral device used.

[Control details]

(1) The command position data is changed to the position data set in the command position data change registers.

Note that the actual position data is changed at the same time when the command position data is changed.

(2) Use the DEFLP command. For details on the DSFLP command, see Section 5.3.

(3) If the command position data is changed in the absolute system, the machine position address is changed.

[Data setting]

(1) The registers used for command position data are indicated below.

Axis No.	Register	
	Upper Digit	Lower Digit
1	D961	D960
2	D967	D966
3	D973	D972
4	D979	D978
5	D985	D984
6	D991	D990
7	D997	D996
8	D1003	D1002

(2) Setting range of the position data in the registers is indicated below.

Item \ Unit	mm		inch		degree		PULSE		Remark
	Setting Range	Unit	Setting Range	Unit	Setting Range	Unit	Setting Range	Unit	
Position data change	-2^{31} to $2^{31}-1$	x 10^{-1} mm	-2^{31} to $2^{31}-1$	x 10^{-5} inch	-2^{31} to $2^{31}-1$	x 10^{-5} degree	-2^{31} to $2^{31}-1$	PLS	If the setting is outside the travel range, no error will occur.

[Caution]

(1) The command position data of the axis being moved cannot be changed.

Any attempt to change it will cause a minor error; the error detection flag (X[ ]7) is set. The error code is stored in the minor error code storage register of the corresponding axis.

(2) The data set in the command position data change register is ignored in the test mode of a peripheral device.

(3) If the absolute system is used, establish the reference position corresponding to the machine position address using the command position data change function when the system is booted.

(4) Command position data is not changed if any of the following errors occur. (Checked when the DSFLP command is executed.)

Error Contents		Error Processing
Data setting error	Axis number is not 1 to 8.	The number of steps causing errors is stored in D9010 and D9011. M9010 and M9011 are turned on.
	Axis number is set indirectly with an index modifier.	
	Data set for the command position change is not 0 or 1.	
	Command position change data is set indirectly with index modifier.	
Command position data change execution error	The axis designated by command position data changing has already started.	The error detection flag (Xn7) set. The error code is stored in the minor error code storage register of the corresponding axis.
	The servo amplifier of the designated axis is not yet turned on.	

(5) While the command position data is changed, the start receive flag is set.

(6) Command position data can be changed regardless of the PC ready flag (M2000) state.

[Timing chart]

The timing chart for the command position data change processing is illustrated below.

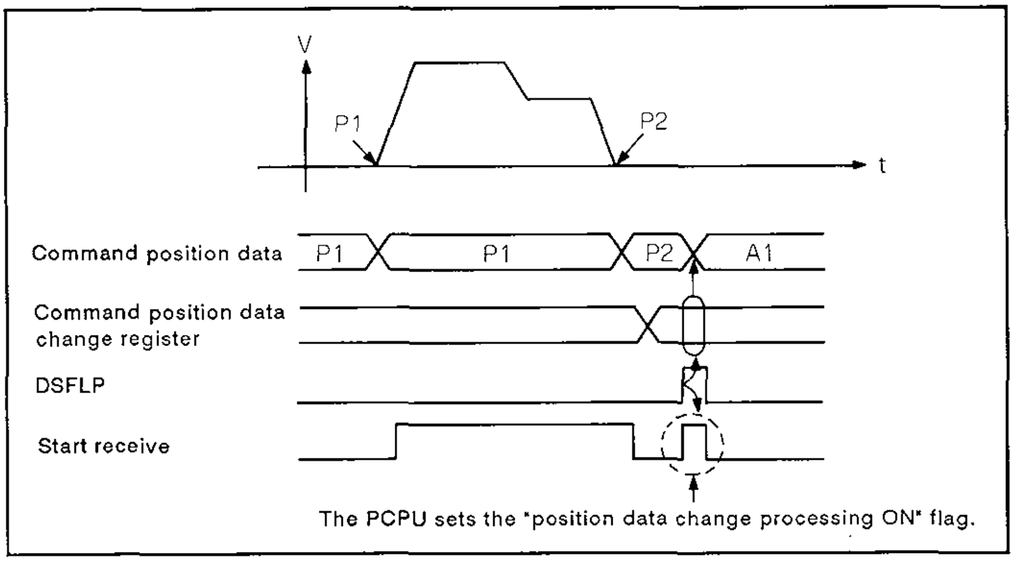


Fig. 8.6 Timing Chart for the Command Position Data Change Processing

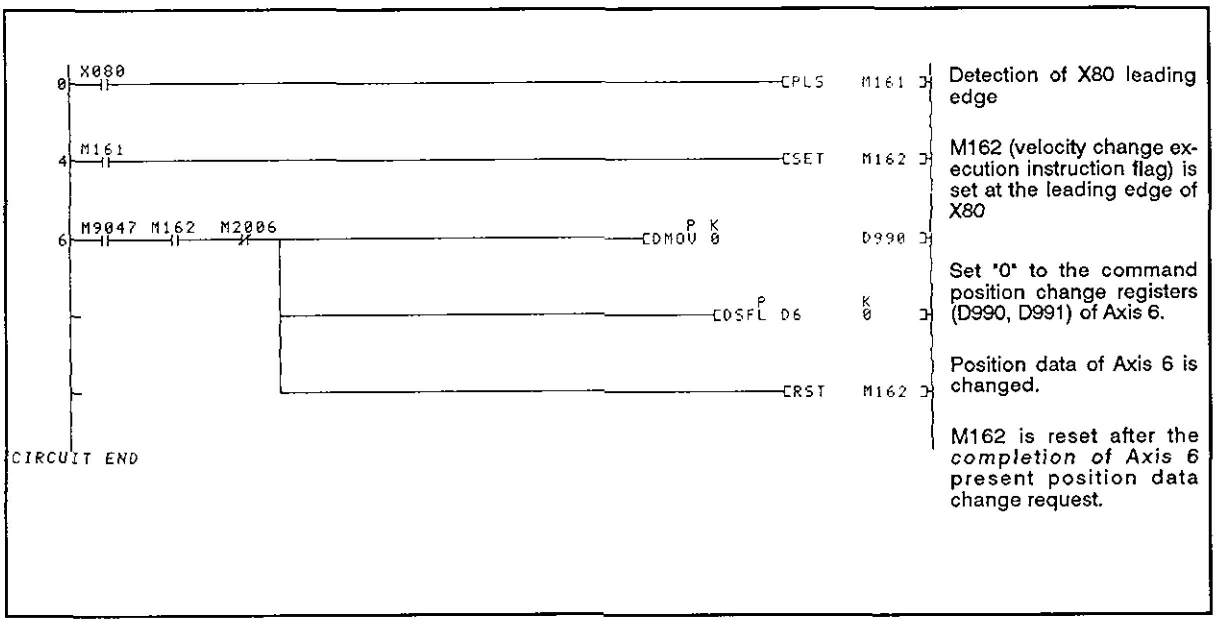
[Program example]

(1) Position data change conditions

(a) AxisAxis 6

(b) Position data change instruction ...X80

(2) *Sequence program*

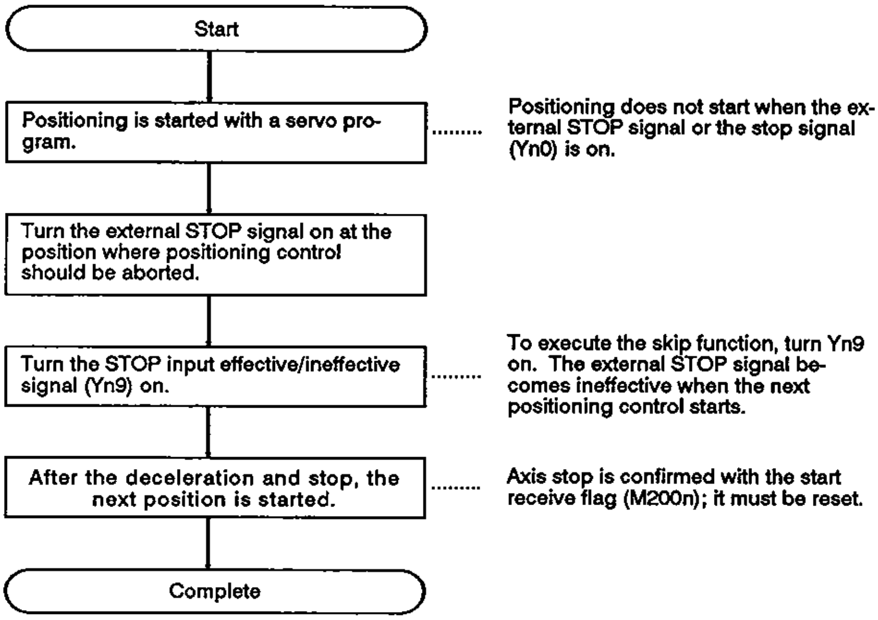


8.9 Skip Function

The skip function aborts the currently executed positioning control in response to the external input and executes the next positioning control.

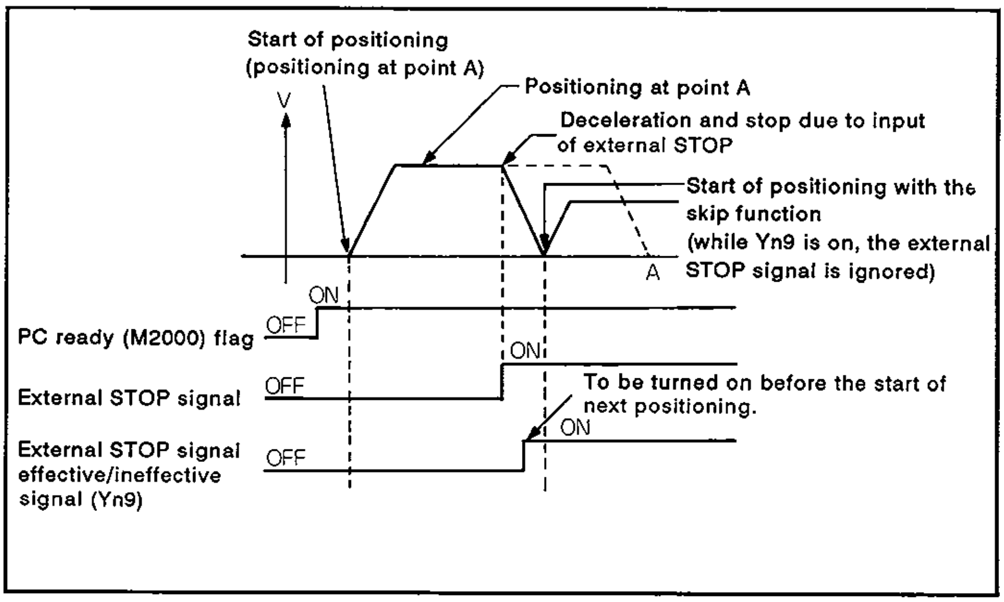
The A73CPU executes the skip function using an external STOP signal and a sequence program.

(1) The skip function execution sequence is explained below.



(2) Timing chart

The skip function operation timing is indicated below.



SEQUENCE CONTROL

1. INTRODUCTION

This section describes the sequence processing and functions of the SCPU which are necessary to execute sequence control with the A73CPU multi-axis positioning module (to be referred to as the A73CPU in this manual).

The A73CPU SCPU can perform the same sequence control as the A3NCPU with the exception of the items indicated in Table 1.1.

For the programming procedure of the sequence program used to execute sequence control, see the ACPU Programming Manual.

Table 1.1 Differences between A73CPU (SCPU) and A3NCPU

Item			A73CPU	A3NCPU
Usable extension base			A65B, A68B (A55B, A58B: Not usable)	A65B, A68B, A55B, A58B
Base unit			A74B	A32B, A35B, A38B
Power supply module loaded in the base unit			A61P *1	A61P, A62P, A63P, A65P
Number of instructions		Sequence instruction	22	22
		Basic instruction	132	132
		Application instruction	107 (DSFL, DSFR: Not usable DSFLP, DSFRP: Function changed)	109
I/O points		For sequence control	1920 points (X/Y80 to X/Y7FF)	2048 points (X/Y0 to X/Y7FF)
		For positioning control	128 points (X/Y0 to X/Y7F)	—
Devices	Internal relay/ Latch relay	For sequence control	2000 points (M/L0 to M/L1999)	2048 points (M/L0 to M/L2047)
		For positioning control	48 points (M2000 to M2047)	—
	Data register	For sequence control	800 points (D0 to D799)	1024 points (D0 to D1023)
		For positioning control	224 points (D800 to D1023)	—
	Special relay		Functions of M9073 to M9079 added	—
	Special register		Functions of D9180 to D9199 added	—
Latch range *2			D0 to D799 (D800 to D1023 cannot be set for latch range)	D0 to D1023

POINT

*1: A62P and A65P cannot be used because their 5 VDC current capacity is insufficient.

*2: This indicates the range that is latched when the latch range is set with the parameter.

2. SCPU SPECIFICATIONS

2.1 Specifications

Table 2.1 SCPU Performance List

Item			Type: SCPU
Control system			Repeated operation (with stored program)
I/O control method			Refresh/direct mode (selectable)
Programming language			Special sequence control language (relay symbolic language, logic symbolic language, MELSAP language)
Instruction number	Sequence instruction		22
	Basic instruction		132
	Application instruction		107
Processing speed (sequence instruction) (μsec/step)			Direct mode : 1.0 to 2.3 Refresh mode : 1.0
I/O points			1920 (X/Y80 to X/Y7FF)
Watchdog timer (WDT) (msec)			10 to 2000
Memory capacity			Equivalent to the capacity of a loaded memory cassette
Program capacity			Main sequence program + Main microcomputer program : Max. 30K steps Main microcomputer program: Max. 58K bytes (29K steps)
			Sub-sequence program + Sub-microcomputer program : Max. 30K steps Sub-microcomputer program: Max. 58K bytes (29K steps)
Devices	Internal relay (M) (points)		1000 (M0 to M999) — M + L + S = 2000 points (set with parameters)
	Latch relay (L) (points)		1000 (L1000 to L1999) — M + L + S = 2000 points (set with parameters)
	Step relay (S) (points)		0 (default) — M + L + S = 2000 points (set with parameters)
	Link relay (B) (points)		1024 (B0 to B3FF)
	Timer (T)	Number of points	256
		Specifica-tions	100 msec timer : 0.1 to 3276.7 sec (T0 to T199) 10 msec timer : 0.01 to 327.67 sec (T200 to T255) 100 msec retentive timer : 0.1 to 3276.7 sec Set with parameters
	Counter (C)	Number of points	256
		Specifica-tions	Normal counter : 1 to 32767 (C0 to C255) Interrupt program counter: 1 to 32767 Counters used in interrupt programs Set with parameters

Item / Type		SCPU
Devices	Data register (D) (points)	800 (D0 to D799)
	Link register (W) (points)	1024 (W0 to W3FF)
	Annunciator (F) (points)	256 (F0 to F255)
	File register (R) (points)	Max. 8192 (R0 to R8191)
	Accumulator (A) (points)	2 (A0, A1)
	Index register (V, Z) (points)	2 (V, Z)
	Pointer (P) (points)	256 (P0 to P255)
	Pointer for interruption (I) (points)	32 (I0 to I31)
	Special relay (M) (points)	256 (M9000 to M9255)
	Special register (D) (points)	256 (D9000 to D9255)
Comment (points)		Max. 4032 (set in units of 64 points)
Self-diagnosis		Watchdog error monitor, memory error, CPU error, I/O error, battery error, etc.
Operation mode at the occurrence of error		Stop or continue
STOP → RUN output mode		Output data when STOP is restored, or data after operation is output.

2.1.1 I/O control modes

I/O modules are controlled in either the direct or refresh mode.

The control mode of the I/O modules can be set in one of the following three modes. For details on mode settings, see the A73CPU User's Manual.

* Direct mode for both input and output
* Refresh mode for input and direct mode for output
* Refresh mode for both input and output

(1) Direct mode

I/O modules are accessed each time the I/O signals are used for processing in sequence program operation. See Fig. 2.1.

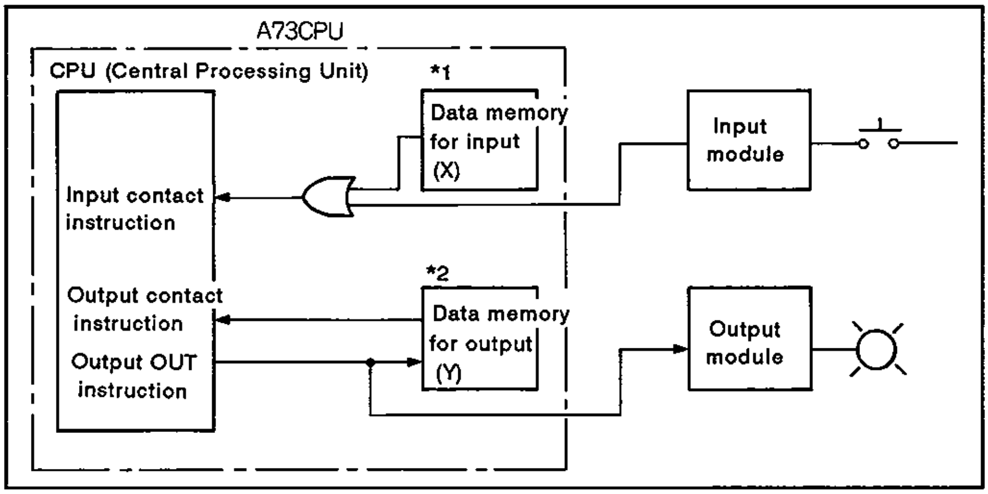


Fig. 2.1 Direct Mode

(a) When an input contact instruction is executed:

The input module data and data memory input data are ORed. The result is used as the input data to execute the sequence program.

(b) When an output contact instruction is executed:

Output data is read from the data memory to execute the sequence program.

(c) When the output OUT instruction is executed:

The operation result of the sequence program is output to the output module and stored in the output data memory (Y).

(d) Operation delay time

The maximum time lag between the change in the input signal and the change in the output signal is one scan.

REMARKS

*1: Operations which turn ON/OFF the data in the data memory for input (X) are:

1) Test operation with a peripheral device
2) Link refresh in the MELSECNET
3) Write from the computer link module (AJ71C24(S3), AD51(S3), etc.)

*2: Operations which turn ON/OFF the data in the data memory for output (Y) are:

1) Execution of OUT instruction in a sequence program
2) Test operation with a peripheral device
3) Write from the computer link module (AJ71C24(S3), AD51(S3), etc.)

(2) Refresh mode

I/O modules are accessed in a batch before step 0 is executed. See Fig. 2.2.

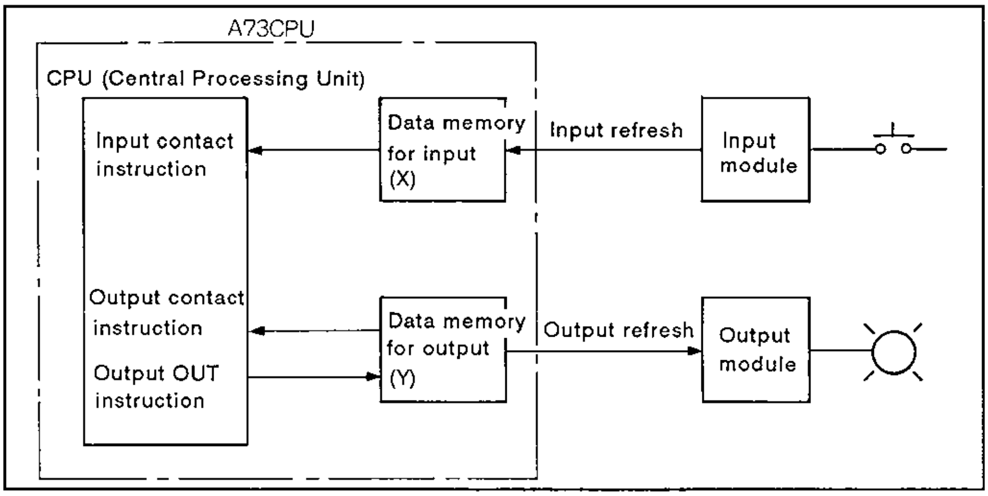


Fig. 2.2 Refresh Mode

(a) Input refresh

Before step 0 is executed, input data is read in a batch from the input module and stored in the input data memory (X).

(b) Output refresh

Before step 0 is executed, the data in the data memory for output (Y) is output to the output module in a batch.

(c) When an input contact instruction is executed:

Input data is read from the data memory for input (X) to execute a sequence program.

(d) When an output contact instruction is executed:

Output data is read from the data memory for output (Y) to execute a sequence program.

(e) When the output OUT instruction is executed:

The operation result of a sequence program is output to the output module and stored in the output data memory (Y).

(f) Operation delay time

The maximum time lag between the change in the input signal and the change in the output signal is two scans.

POINT

To access a part of I/O module directly as in the direct mode, use the SEG instruction (partial refresh instruction).

For details, see the ACPU Programming Manual.

(3) Cautions for the refresh mode

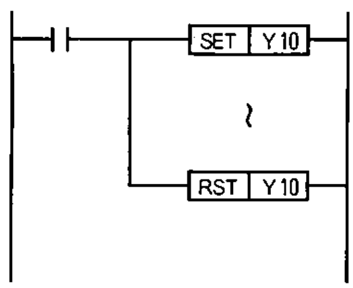
(a) When output (Y) is the refresh mode:

As illustrated in Fig. 2.3, it is not possible to output pulse signals externally by turning ON/OFF the output (Y) of the same device number within one scan of operation.

If such a circuit is programmed, the instruction set in a larger step number becomes effective and the instruction set in a smaller step number is ignored.

To output a pulse signal externally, use the SEG instruction (partial refresh instruction) as shown in the program in Fig. 2.4.

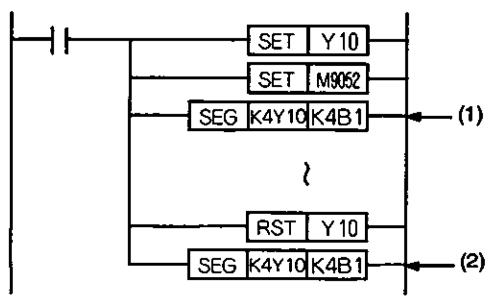
For details on the SEG instruction, see the ACPU Programming Manual.



If Y10 is turned on and off within the same scan, the state just before the output refresh is executed, i.e., the off state after the execution of the RST instruction, is always output to the output module.

However, in the sequence program, Y10 is on after the execution of the SET instruction and before the execution of the RST instruction.

Fig. 2.3 When the SEG Instruction Is Not Used



The "SET M9052" is used to change the SEG instruction to the partial refresh instruction.

The partial refresh instruction in (1) outputs the Y10 ON data to the output module.

The partial refresh instruction in (2) outputs the Y10 OFF data to the output module.

Fig. 2.4 When the SEG Instruction Is Used

(b) When input (X) is the refresh mode:

To execute the fetch processing of the external input signals in a batch at the head of the scan, maintain the external signal ON/OFF state for more than one scan.

If the ON/OFF state of the external input signal is changed during a scan, the state of the input (X) in the sequence program does not change.

To fetch the ON/OFF state of the external input signal during a scan, execute the SEG instruction.

A program example is shown in Fig. 2.5.

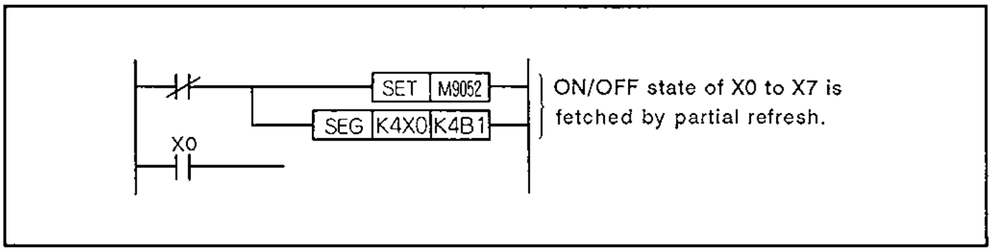


Fig. 2.5 Partial Refresh of Input Signals

2.1.2 Initial processing

Initial processing is a series of processing which is required to execute sequence program operation processing.

Initial processing is executed when the SCPU is powered up or reset with the reset switch of the SCPU.

Initial processing usually takes 2 to 4 seconds, varying depending on the system configuration.

(1) I/O module initialization

The I/O module is reset to be initialized.

(2) Clearing data memory

(a) Unlatched data memory is cleared.

The latch range is set with a parameter using a peripheral device.

(b) Y data contents are cleared if Y, which corresponds to the areas occupied by the input modules and the areas not occupied by any module, is used as internal relay M.

(3) Setting link parameter

When the A73CPU is used as the master station in the MELSECNET, the link parameter data is set to the data link module and the data link starts.

(4) I/O address assignment

I/O addresses are automatically assigned to the I/O modules loaded in the base unit.

(5) Entering I/O module data

The types of I/O modules loaded in the base unit are entered.

The I/O module data is used to verify the I/O module.

(6) Self-diagnosis

Among the check items conducted by the SCPU for self-diagnosis, those conducted when the power is turned on or the SCPU is reset are conducted.

For details, see Section 2.1.9.

2.1.3 END processing

The END processing returns the operation step to step 0 in repeated operation processing.

After the execution of the END (FEND) instruction, the following series of processing is executed as the end processing.

(1) Self-diagnosis

The SCPU checks for a blown fuse, I/O module verify error, low battery power, etc. For details, see Section 2.1.9.

(2) Timer/counter processing

Present values of the timers and counters are updated and their contacts are turned on or off.

For details, see Sections 2.1.4 and 2.1.5.

(3) Constant scan processing

When the constant scan function is used, the SCPU waits until the constant scan time (set for the special data register D9020) is reached to execute the repeated operation processing.

(4) Data communication processing with computer link module

The SCPU communicates with the computer link module when a data read/write request is given by the computer link module (AJ71C24(S3), AD51(S3)).

(5) Link refresh processing

Link refresh processing is executed when a link refresh request is given by the MELSECNET data link module.

For link refresh timing, see the MELSECNET Data Link System User's Manual.

(6) Sampling trace processing

The status of the set devices is stored in the sampling trace area as the sampling trace is executed every scan (after END instruction execution).

(7) RUN/STOP keyswitch position check

The SCPU operation status is changed according to the RUN/STOP switch position.

For details of transition processing to RUN, STOP, PAUSE, and STEP-RUN, see Section 2.1.8.

2.1.4 Timer processing

The SCPU uses timers which increase the present values according to the measured duration.

Three types of timers are used:

• 100 msec timers	Range :	0.1 to 3276.7 sec
	Units :	100 msec
* 10 msec timers	Range :	0.01 to 327.67 sec
	Units :	10 msec
* 100 msec retentive timers	Range :	0.01 to 3276.7 sec
	Units :	100 msec

Present value is retained after the power is turned off.

(1) Updating the timer present value

When the timer coil is turned on with the OUT T[ ] instruction, the timer present value is updated after the execution of the END (FEND) instruction. After the timer counts up, the timer contacts close.

(a) 100 msec and 10 msec timers

When the timer coil is turned off with the input conditions not satisfied, the present value of the timer is reset to 0 and the timer contacts open after the execution of the END (FEND) instruction.

(b) 100 msec retentive timers

When the timer coil is turned off with the input conditions not satisfied, the present value update stops but the present value is retained.

(2) When the RST T[ ] instruction is executed:

When the timer is reset with the RST T[ ] instruction, the present value is reset to 0 and the timer contacts open.

The 100 msec retentive timer retains the present value and the contact status when the coil is turned off.

To reset the 100 msec retentive timer, use the RST T[ ] instruction.

(3) When the OUT T[ ] instruction is skipped:

Once the timer is started with the OUT T[ ] instruction, the timer present value is updated even when the OUT T[ ] instruction is skipped with the CJ instruction. The contacts close when the timer counts up.

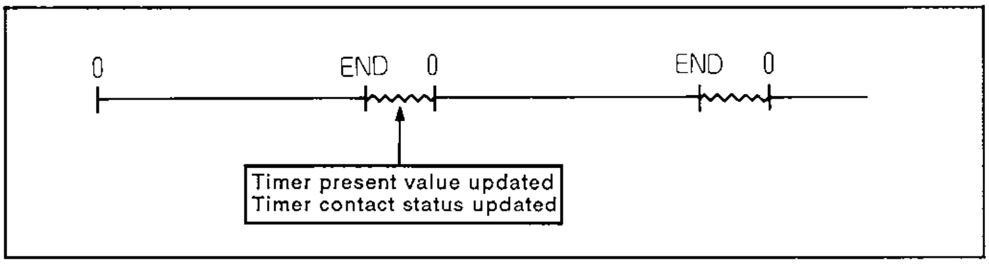


Fig. 2.6 Timer Processing

REMARKS

Timer accuracies are indicated below. For further details, see the ACPU Programming Manual.

Timer	Scan Time T	Direct Input	Refresh Input
10 msec	T < 10 msec	+2 scan time to −10 msec	+2 scan time to 0
10 msec	T ≥ 10 msec	+2 scan time to −1 scan time	
100 msec 100 msec retentive	T < 100 msec	+2 scan time to −100 msec	
100 msec 100 msec retentive	T ≥ 100 msec	+2 scan time to −1 scan time	

2.1.5 Counter processing

The SCPU uses counters which increase the present value at the leading edge of an input signal. Two types of counters, normal and interrupt counters, are used. In this manual, the word "counter" indicates a normal counter unless it is specifically expressed as an "interrupt counter".

* The counter is used in a main routine or subroutine.
* The interrupt counter is used in an interrupt program.

(1) Updating the counter present value

The counter coil is turned on and off by the OUT C[ ] instruction. The present value is updated at the leading edge of the coil signal and the coil contacts close when the counter counts out.

(a) The counter used in a main or subroutine

The present value is updated and the coil status is changed after the execution of the END (FEND) instruction.

(b) The interrupt counter used in an interrupt program

The present value is updated and the coil status is changed after the execution of the IRET instruction.

(2) When the RST C[ ] instruction is executed:

When the counter is reset by the RST C[ ] instruction, the present value is reset to 0 and the counter contacts open.

The present value and the contact status are retained when the counter coil is turned off.

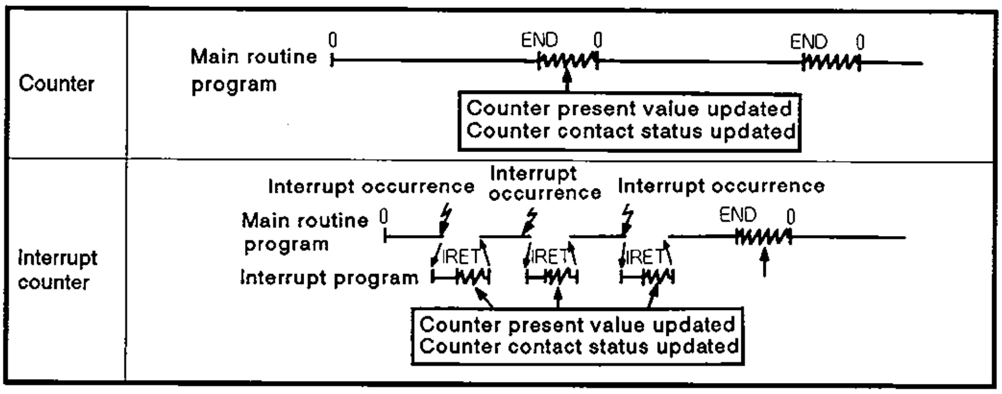


Fig. 2.7 Counter Processing

REMARKS

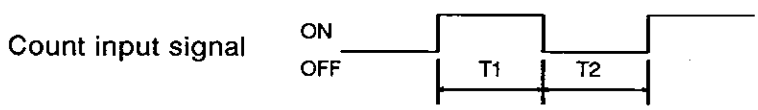
The maximum speed of the counter depends on the scan time. Counting is only possible when the on/off time of the input conditions is longer than one scan. For details, see the ACPU Programming Manual.

$$\text{Maximum counting speed Cmax} = \frac{n}{100} \times \frac{1}{ts} \text{ [time/sec]}$$

where,

n: duty (%)

Duty is the ratio of the input signal on or off time to one cycle



of input signal on/off time, expressed in percent.

If $T1 \leq T2$ $\quad n = \frac{T1}{T1 + T2} \times 100\ (\%)$

If $T1 > T2$ $\quad n = \frac{T2}{T1 + T2} \times 100\ (\%)$

ts: program scan time (sec)

2.1.6 Watchdog timer (WDT)

(1) Watchdog timer

The watchdog timer is an internal timer used to detect errors of the PC's repeated operation function.

Setting is possible with a parameter in the range of 10 msec to 2000 msec in units of 10 msec. Default is 200 msec.

(2) Operation

The watchdog timer monitors program execution scan time over one scan.

If any PC hardware fault occurs or processing is not completed within the preset period, it stops operation with warning signal output.

(3) Reset timing

The watchdog timer is reset after the execution of the END instruction if the PC operation is completed normally.

(4) Error

The watchdog timer error is indicated by code "22" or "25".

Error code "22": The END instruction is executed after the set period has elapsed.

Error code "25": The END instruction is not executed due to endless loop.

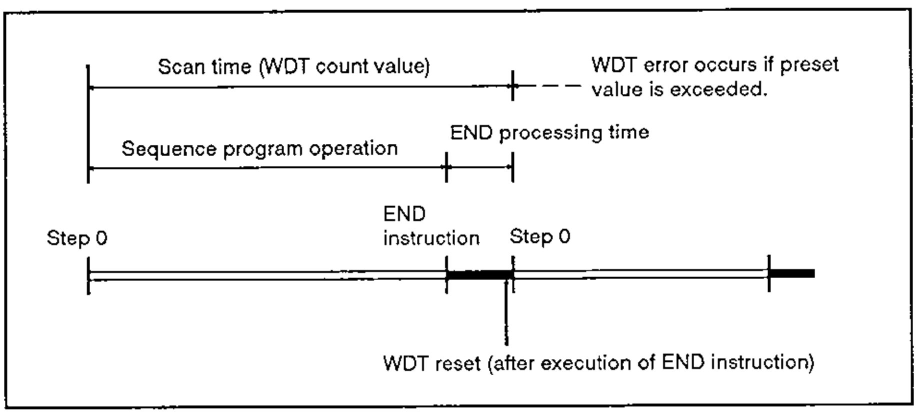
For details, see the A73CPU User's Manual.

(5) Operation after an occurrence of watchdog timer error

(a) The PC stops operation and turns off all outputs.

(b) The RUN LED in the CPU module front panel flashes.

(c) The "WDT ERROR" message is displayed on the LED display unit on the CPU module front panel.

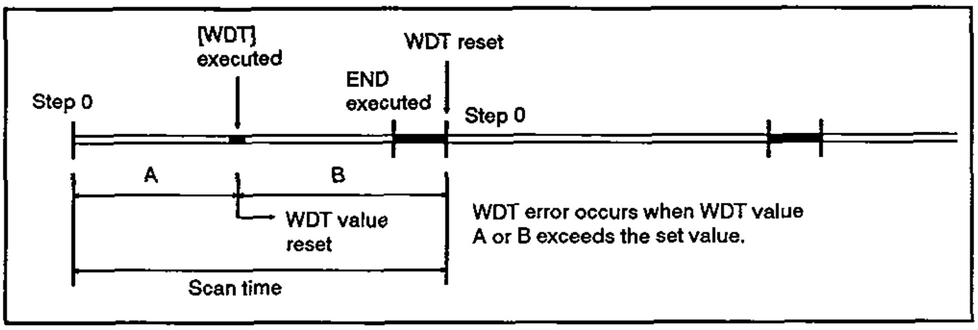


(6) Resetting

The watchdog timer can be reset in a sequence program.

The watchdog timer preset value is reset when the WDT instruction is executed; counting begins again from "0".

However, the scan time stored in D9017 to D9019 is not reset when the WDT instruction is executed.



(7) If the WDT error occurs, check the nature of the error by referring to the troubleshooting information in the A73CPU User's Manual. After placing the reset switch in the RESET position, remove the cause of the error.

2.1.7 Operation processing at the occurrence of momentary power failure

The A73CPU detects a momentary power failure if the input line voltage to the power supply module falls below the specified value.

If the momentary power failure within the allowable limit (20 msec) occurs, the SCPU executes the momentary power failure processing as indicated below.

(1) Momentary power failure under 20 msec

(a) Operation processing stops with the output status retained.

(b) Operation processing is resumed when normal state is restored.

(c) The watchdog timer keeps counting while the operation is stopped. Therefore, if the watchdog timer setting is 200 msec and scan time is 190 msec, the WDT error occurs if a momentary power failure occurs for 20 msec.

(2) Momentary power failure longer than 20 msec

The SCPU is reset.

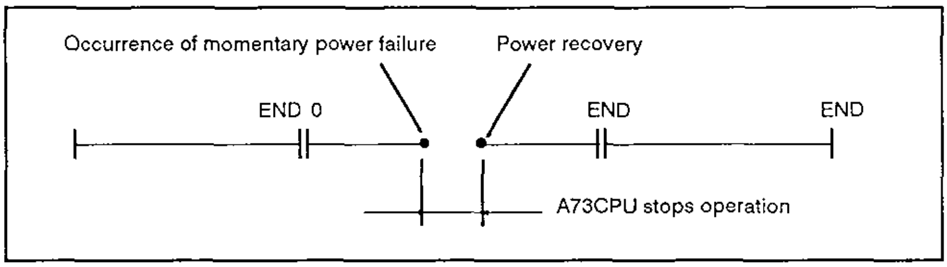


Fig. 2.8 Operation Processing at Occurrence of Momentary Power Failure

2.1.8 RUN, STOP, PAUSE, STEP-RUN operation processing

The SCPU operation state are RUN, STOP, PAUSE, or STEP-RUN.

Each operation state is explained below.

(1) RUN

The RUN state indicates that the SCPU is executing a sequence program repeatedly in the order of step 0 to the END (FEND) instruction.

When the SCPU is set in the RUN state, the output status saved when it had been set to the STOP state is output according to the set output mode (STOP → RUN).

The processing required to start sequence program operation after the operation state has been changed from the STOP to RUN is usually one to three seconds. This time varies according to the system configuration.

The processing shown in Fig. 2.9 is repeated in the RUN state until it is changed to another operation state.

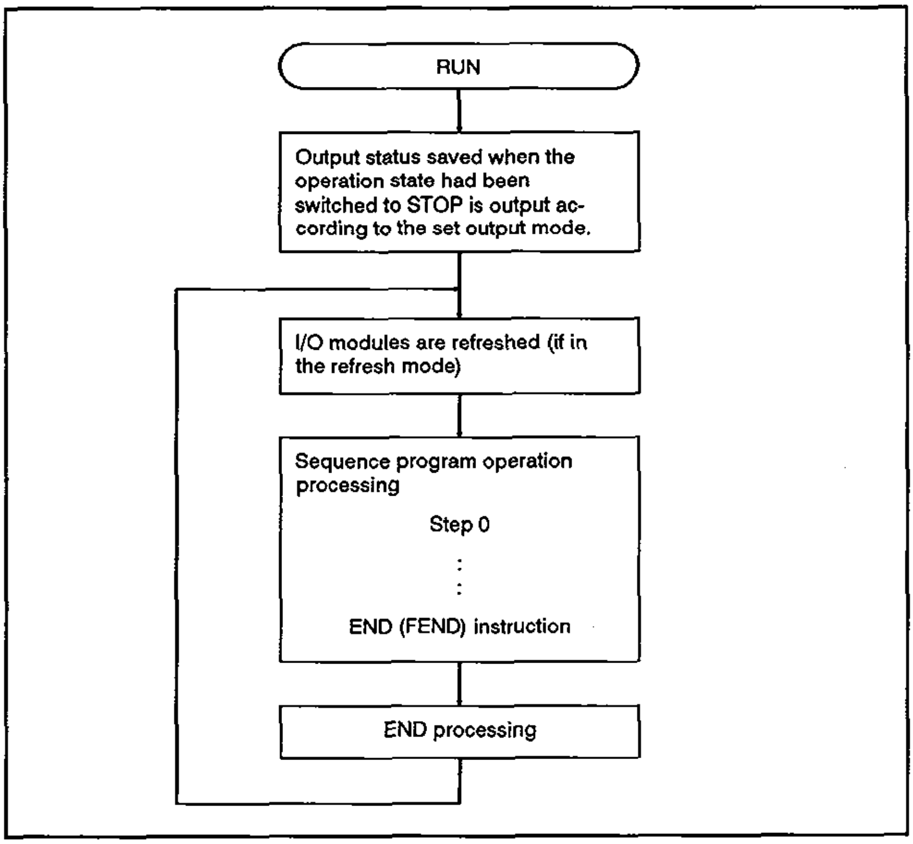


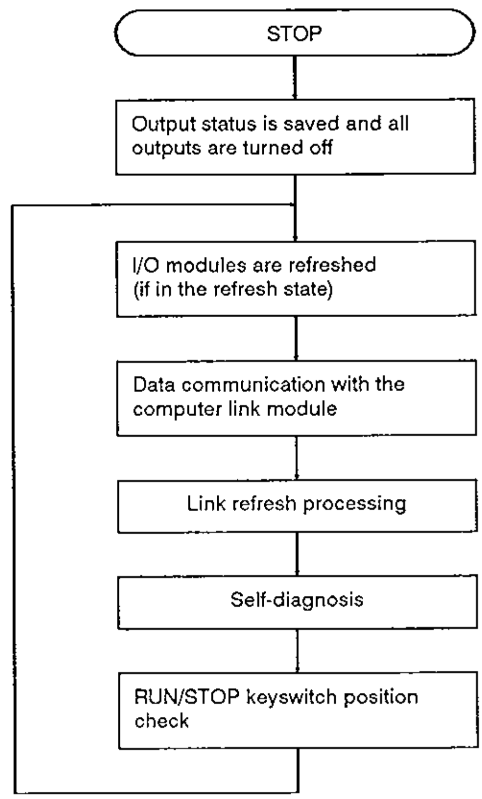
Fig. 2.9 RUN Operation Processing

(2) STOP

The STOP state indicates that sequence program operation is stopped using the RUN/STOP keyswitch or by the remote STOP signal (Section 2.2.3).

When the SCPU is set in the STOP state, the output status is saved and all outputs are turned off. Data other than the outputs (Y) is retained.

The processing shown in Fig. 2.10 is repeated in the STOP state until it is changed to another operation state.



* For details on processing, see Section 2.1.3.

Fig. 2.10 STOP Operation Processing

(3) PAUSE

The PAUSE state indicates that sequence program operation is stopped with the output and data memory statuses retained.

The processing shown in Fig. 2.11 is repeated in the STOP state until it is changed to another operation state.

For the procedure to set the SCPU in the PAUSE state, see Section 2.2.4.

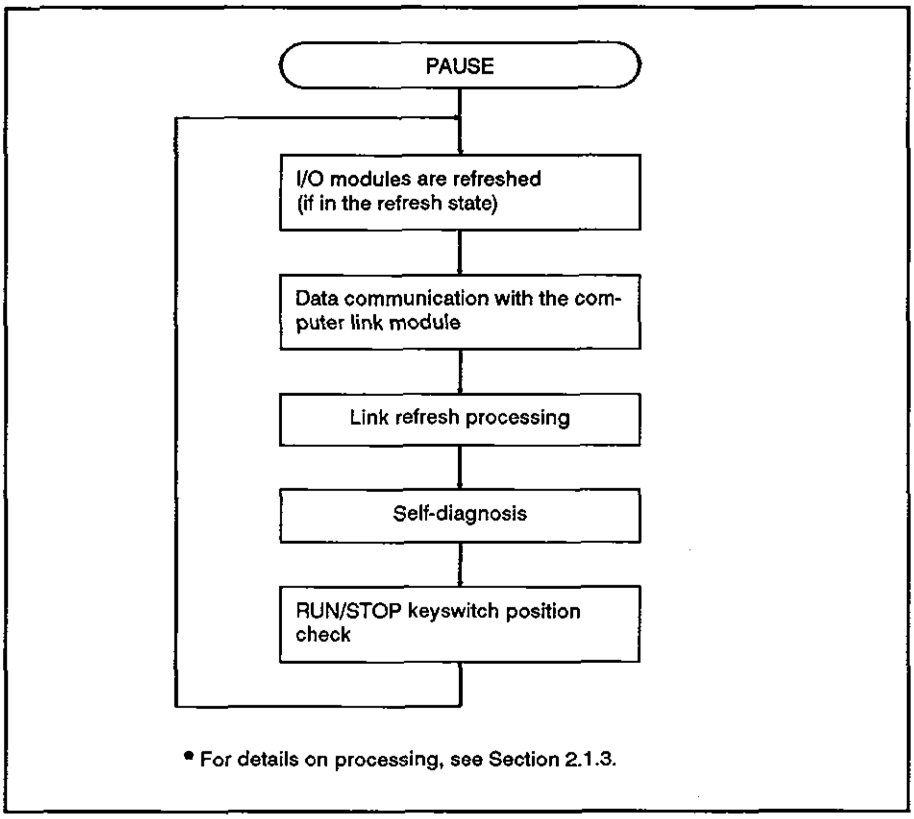


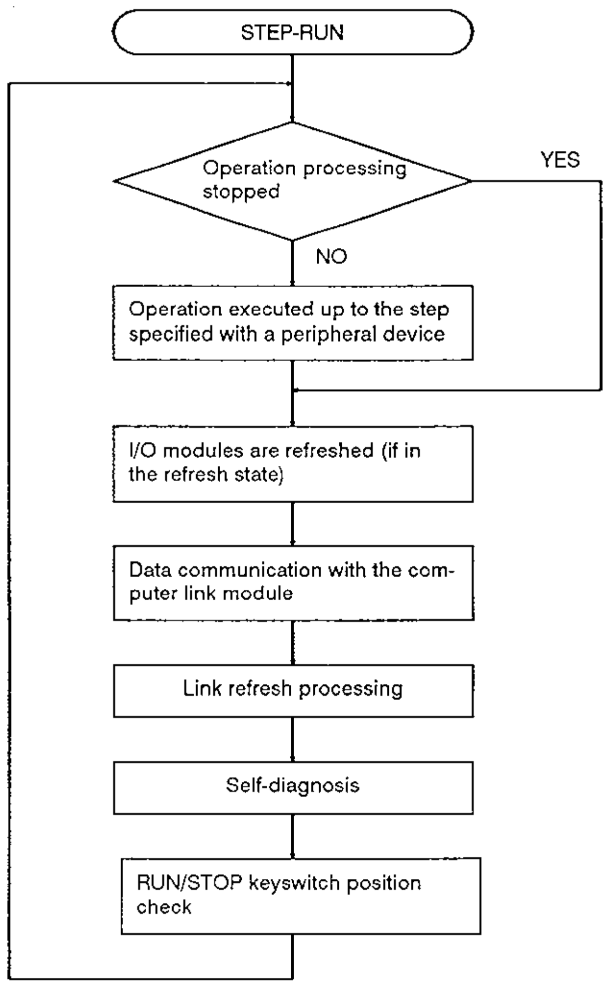
Fig. 2.11 PAUSE Operation Processing

(4) STEP-RUN

The STEP-RUN state indicates the run state in which sequence program operation processing can be stopped or continued in units of instructions; run/stop control is possible using a peripheral device.

Because operation processing is stopped with the output and data memory statuses retained, it is possible to check the execution state of the sequence program.

The processing shown in Fig. 2.12 is repeated in the STEP-RUN state until it is changed to another operation state.



* For details on processing, see Section 2.1.3.

Fig. 2.12 STEP-RUN Operation Processing

(5) Relationship between RUN/STOP switch operation and SCPU operation processing

Table 2.2 RUN/STOP Switch Operation and SCPU Operation Processing

<table>
<tr><th colspan="2">SCPU
RUN/STOP</th><th>Sequence Program Operation Processing</th><th>External Output</th><th>Data Memory (Y, M, L, S, T, C, D)</th><th>Remarks</th></tr>
<tr><td colspan="2">RUN → STOP
STEP-RUN → STOP</td><td>Stopped</td><td>Output status is saved by OS and all outputs are turned off.</td><td>Status at the entry to STOP state is retained.</td><td></td></tr>
<tr><td colspan="2">STOP → RUN</td><td>Started</td><td>Depends on the parameter setting for "STOP → RUN output state".</td><td>Operation is resumed in the same status as that at the entry to the STOP state.</td><td></td></tr>
<tr><td colspan="2">RUN → PAUSE
(with M9040.ON)</td><td>Stopped</td><td rowspan="3">Output status is retained.</td><td>The same status as that at the entry to the PAUSE state is retained.</td><td>If M9040 is off, the same operation processing will occur as that executed when the RUN/STOP switch is in the RUN position. (PAUSE state is not set.)</td></tr>
<tr><td rowspan="2">STOP → STEP-RUN

PAUSE → STEP-RUN</td><td>Operation stopped from a peripheral device.</td><td>Operation stops at the step specified by a peripheral device.</td><td>Status at the entry to operation stop is retained.</td><td></td></tr>
<tr><td>Operation resumed from a peripheral device.</td><td>Operation resumes from the step next to the operation stopped step.</td><td>Operation resumes from the status at the entry to operation stop.</td><td></td></tr>
<tr><td colspan="2">PAUSE → RUN</td><td>Started</td><td>Operation resumes from the PAUSE output status.</td><td>Operation resumes from the status at the entry to the PAUSE state.</td><td></td></tr>
</table>

(6) Processing during sequence program stop operation processing

Table 2.3 Processing when Operation Processing Stopped

RUN/STOP \ Processing	RUN (END processing)	STOP	PAUSE	STEP-RUN
I/O Refresh (for refresh state)	Executed	Executed	Executed	Executed
Self-diagnosis	Executed	Executed	Executed	Executed
Timer/counter (Updating of present value and opening/closing contacts)	Executed	—	—	—
Constant Scan Processing	Executed	—	—	—
Communication with Computer Link Module	Allowed	Allowed	Allowed	Allowed
Link Refresh Processing	Allowed	Allowed	Allowed	Allowed
Sampling Trace Processing	Executed	—	—	—
RUN/STOP Switch Position Check	Executed	Executed	Executed	Executed
Remarks				If the END (FEND) instruction is executed during STEP state operation, the END processing begins. Timer present value update: 10 msec timer .... "1" is added after each scan. 100 msec timer .... "1" is added after each 10 scans.

2.1.9 Self-diagnosis

(1) Self-diagnosis

The SCPU checks itself for errors when the power is turned on and while the SCPU is running. If an error is detected, the SCPU provides an error indication or stops operation, depending on the level of error, thereby preventing SCPU malfunctioning and ensuring reliable operation.

The SCPU's operation states at the detection of an error are operation stop state and operation continue state. For some errors included in the operation continue state, it is possible to stop operation when such errors are detected, by setting the proper parameter.

If an error is detected, the error occurrence event and the contents of the error are stored in the special relays (M) and special registers (D). Read the stored error data in the program, especially for the operation continue state, to prevent malfunctioning of the SPCU and mechanical systems.

If an operation stop error is detected by the self-diagnosis function, the operation stops at the point the error is detected and all outputs (Y) are turned off.

If an operation continue error is detected, the program portion involved with the error is skipped and other portions of the program are executed.

If an I/O module verify error is detected, operation is continued with the I/O addresses before the occurrence of the error.

For errors detected by the self-diagnosis function, refer to Table 2.4.

REMARKS

1. If two states are indicated in the "CPU Status" and "RUN LED Status" columns in Table 2.4, either of them can be selected by setting a peripheral device.
2. For an operation check error marked with (*), the error message displayed on the LED display unit is as indicated below only for the error caused by the CHK instruction.

<CHK> ERROR [ ][ ][ ]

3-digit fault code number

Table 2.4 Self-diagnosis Error List

Diagnosis		Diagnosing Timing	CPU Status	RUN LED Status	Error Message
Memory error	Instruction code check	When each instruction is executed.	Stop	Off	INSTRCT.CODE ERR
	Parameter setting check	When the power is turned on or the SCPU is reset. When the operation state is changed: From [STOP/PAUSE] to [RUN/STEP RUN]			PARAMETER ERROR
	No END instruction	When M9056 or M9057 is set. When the operation state is changed: From [STOP/PAUSE] to [RUN/STEP RUN]			MISSING END INS.
	Instruction execution impossible	When the following instruction is executed: CJ, SCJ, JMP, CALL(P), FOR-NEXT When the operation state is changed: From [STOP/PAUSE] to [RUN/STEP RUN]			CAN'T EXECUTE (P)
	Format (CHK instruction) check	When the operation state is changed: From [STOP/PAUSE] to [RUN/STEP RUN]			CHK FORMAT ERR.
	Instruction execution impossible	When an interrupt occurs. When the operation state is changed: From [STOP/PAUSE] to [RUN/STEP RUN]			CAN'T EXECUTE (I)
	No memory cassette	When the power is turned on or the SCPU is reset.			CASSETTE ERROR
CPU error	RAM check	When the power is turned on or the SCPU is reset. When the M9084 is set during STOP state.	Stop	Off	RAM ERROR
	Operation circuit check	When the power is turned on or the SCPU is reset.			OPE. CIRCUIT ERR.
	Watchdog error check	When the END instruction is executed.			WDT ERROR
	END instruction not executed	When the END instruction is executed.			END NOT EXECUTE
	Sub-CPU check	Always			SUB-CPU ERROR
	Interrupt error	Always			MAIN CPU DOWN
I/O error	I/O module verification	When the END instruction is executed. (Not checked if M9084 or M9094 is ON.)	Stop .... Run	Off .... On	UNIT VERIFY ERR.
	Fuse blown	When the END instruction is executed. (Not checked if M9084 or M9094 is ON.)			FUSE BREAK OFF.

Table 2.4 Self-diagnosis Error List (Continued)

Diagnosis		Diagnosing Timing	CPU Status	RUN LED Status	Error Message
Special function module error	Control bus check	When the FROM or TO instruction is executed.	Stop	Off	CONTROL-BUS ERR.
	Special function module error	When the FROM or TO instruction is executed.			SP. UNIT DOWN
	Link module error	When the power is turned on or the SCPU is reset. When the operation state is changed: From [STOP/PAUSE] to [RUN/STEP RUN]			LINK UNIT ERROR
	I/O interruption error	When an interrupt occurs.			I/O INT. ERROR
	Special function module assignment error	When the power is turned on or the SCPU is reset. When the operation state is changed: From [STOP/PAUSE] to [RUN/STEP RUN]			SP. UNIT LAY. ERR.
	Special function module error	When the FROM or TO instruction is executed.	Stop ---------- Run	Off ---------- On	SP. UNIT ERROR
	Link parameter error	When the power is turned on or the SCPU is reset. When the operation state is changed: From [STOP/PAUSE] to [RUN/STEP RUN]	Run	On	LINK PARA. ERROR
Battery	Battery low	Always (Not checked when M9084 is ON.)	Run	On	BATTERY ERROR
* Operation check error		When each instruction is executed.	Stop ---------- Run	Off ---------- On	OPERATION ERROR

2.1.10 Parameters

(1) Parameters assign the user memory area in the CPU module to specific applications and functions.

(2) Parameters default values are determined as indicated below. If no parameter setting is made, the default values will be used.

(3) The allowable setting range for each parameter is indicated in Table 2.5. Setting is possible with a peripheral device.

For details on parameter setting, refer to the Operating Manual for the peripheral device.

Table 2.5 Parameter Setting Range

Item \ Setting		Default Value	Setting Range	Peripheral Device PU	GPP	HGP	PHP
Main sequence program capacity		6K steps	1K to 30K steps (in units of 1K steps)	o	o	o	o
Sub-sequence program capacity		None	1K to 30K steps (in units of 1K steps)	o	o	o	o
File register capacity		None	1K to 8K points (in units of 1K points)	o	o	o	o
Comment capacity		None	0 to 4032 points (in units of 64 points)	–	o	o	o
Status latch	Memory capacity	None	0 or 8K bytes to 24K bytes	–	o	o	o
	Data memory		NA/A (0/8K bytes)				
	File register		NA/A (2K to 16K bytes)				
Sampling trace		None	0 or 8K bytes	–	o	o	o
Microcomputer program capacity		None	0 to 58K bytes (in units of 2K bytes)	–	o	o	o
I/O control setting		Direct for inputs and outputs	Direct or refresh setting is possible for inputs and outputs individually	o	o	o	o
Latch range setting	Link relay (B)	L1000 to L2047 None for others	B0 to 3FF (in units of points)	o	o	o	o
	Timer (T)		T0 to 255 (in units of points)				
	Counter (C)		C0 to 255 (in units of points)				
	Data register (D)		D0 to 1023 (in units of points)				
	Link register (W)		W0 to 3FF (in units of points)				

Table 2.5 Parameter Setting Range (Continued)

Item		Default Value	Setting Range	Peripheral Device: PU	GPP	HGP	PHP
Link range setting	Number of link stations	None	1 to 64	–	o	o	o
	Input (X)		X0 to 7FF (in units of 16 points)				
	Output (Y)		Y0 to 7FF (in units of 16 points)				
	Link relay (B)		B0 to 3FF (in units of 16 points)				
	Link register (W)		W0 to 3FF (in units of points)				
Setting of Internal relay (M), Latch relay (L), and Step relay (S)		M0 to M999 L1000 to L2047 None for S	M/L/S: 0 to 2047 (must be serial numbers)	o	o	o	o
Timer setting		100 msec: T0 to T199 10 msec: T200 to T255	256 points (in units of 8 points) in total for 10 msec, 100 msec, and 100 msec retentive timers (must be serial numbers for each timer)	o	o	o	o
Interrupt counter setting		No interrupt counter	Used status of interrupt counter (C244 to C255) assigned to each interrupt pointer (I), is set in units of points.	–	o	o	o
I/O number assignment	Input (X) module	None	0 to 64 points (in units of 16 points)	–	o	o	o
	Output (Y) module						
	Special function module						
	Empty slot						
Remote RUN/PAUSE contact setting		None	X0 to 7FF (1 point each for RUN/PAUSE contact. Setting of PAUSE contact only is not allowed.)	–	o	o	o
Operation state at the occurrence of an error	Fuse blown	Continue	Stop or Continue	–	o	o	o
	I/O verify error	Stop					
	Operation error	Continue					
	Special function module check error	Stop					
Annunciator display state		F number display	F number of alternate display of F number and comment (comment: alphanumeric only)	–	o	o	o
STOP → RUN display state		Operation status before STOP state entry is output	Output of operation status: Before entry to STOP state After operation execution	–	o	o	o
Print title entry		None	128 characters (all keys on MELSAP are accepted)	–	o	o	o
Keyword entry		None	Max. 6 hexadecimal digits (0 to 9, A to F)	o	o	o	o

REMARKS

The memory capacity changes according to the parameter settings.

Select a memory cassette by calculating the required memory capacity using Table 2.6.

Table 2.6 Parameter Setting and Memory Capacity

Item		Setting Units	Memory Capacity	ROM	Remarks
Main program	Parameter and T/C setting value	—	4K bytes (fixed)	Possible	4K byte memory area is used for parameter settings and T/C settings.
	Sequence program	1K steps	Main sequence program capacity x 2K bytes		
	Microcomputer program	2K bytes	(Main microcomputer program capacity)K bytes		
Sub-program	T/C settings	—	6K bytes (fixed)	Impossible	6K byte memory area is used for setting the T/C settings and PI addresses.
	Sequence program	1K steps	Sub-sequence program capacity x 2K bytes		
	Microcomputer program	2K bytes	(Sub-microcomputer program capacity)K bytes		
Sampling trace		None/Used	0/8K bytes		
Status latch	Data memory	None/Used	0/8K bytes		The memory capacity for file register status latch is determined by the number of file register points set by the parameter.
	File register	None/Used	(File register memory capacity)K bytes		
File register		1K points	(File register points) x 2K bytes		
Comment		64 points	$\frac{\text{(Comment points)}}{64}$ + 1K bytes		The system occupies 1K bytes when a comment capacity is set.

2.2 Function List

Table 2.7 Function List

Function	Contents	Refer to
Constant scan	Executes a sequence program at the predetermined intervals independent of the scan time. Setting range: 10 msec to 190 msec	Section 2.2.1
Latch	Retains the device data when the power is turned off, the PC is reset, or a momentary power failure longer than 20 msec occurs. The devices that can be latched are: L, B, T, C, D, and W.	Section 2.2.2
Remote RUN/STOP	Allows remote run/stop control from the external device (peripheral device, external input, computer, etc.) when the RUN keyswitch is in the RUN position.	Section 2.2.3
PAUSE	Stop operation while retaining the status of outputs (Y). The PAUSE state can be set by any of the following methods: • RUN keyswitch on the front panel of the CPU module • Remote PAUSE contact * Peripheral device	Section 2.2.4
Status latch	Stores all device data to the status latch area of the memory cassette when the status latch conditions are satisfied. The device data stored in the status latch area can be monitored by a peripheral device.	Section 2.2.5
Sampling trace	Executes sampling of the specified device operation status at predetermined intervals and stores the sampling result in the sampling trace area of the memory cassette. The device data stored in the sampling trace area can be monitored by a peripheral device.	Section 2.2.6
STEP-RUN	Executes a sequence program in units of instructions. Step run can be executed by any of the following methods: * Specifying the loop count * In units of instructions	Section 2.2.7
I/O module replacement in online state	Allows an I/O module to be replaced while the CPU is running (power on).	Section 2.2.8
BASIC program interrupt	Allows the BASIC program interrupt processing routine to start from the sequence program. Starts up to 16 interrupt subroutines using the SUBP instruction.	Section 2.2.9

2.2.1 Constant scan

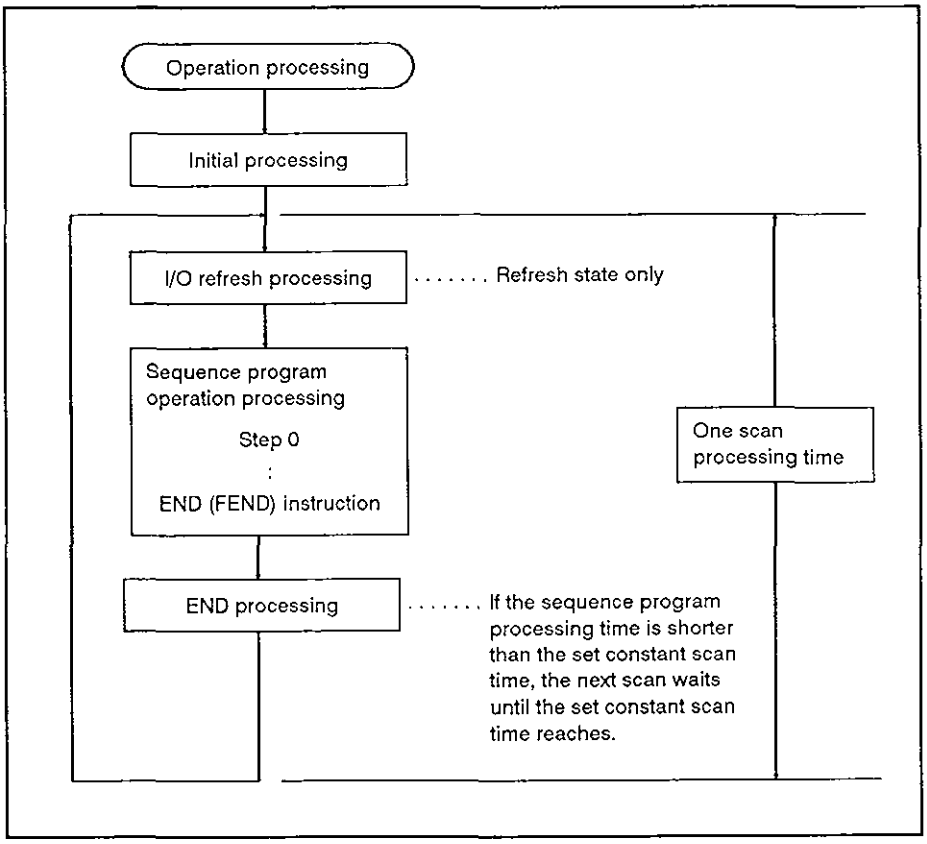
APPLICATION

The scan time of a sequence program varies depending on whether or not the instructions are executed, resulting in an error in positioning. The constant scan function minimizes such errors by permitting the sequence program to be executed at specified intervals.

FUNCTION

(1) Definition

The constant scan function makes one scan processing time of a sequence program constant.



(2) Setting range

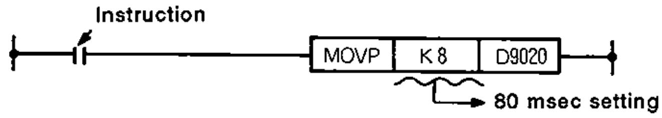
The constant scan time setting is written to D9020 using a value between 1 and 19 in units of 10 msec.

If the setting is outside the allowable setting range, it is processed as indicated below:

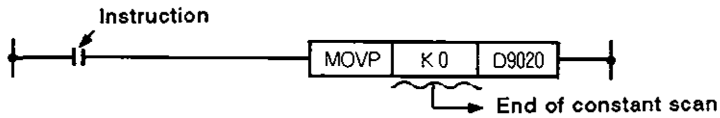
-32768 to 0....... No constant scan setting
1 to 19................ Constant scan setting, 10 msec to 190 msec
20 to 32767....... Constant scan setting, 190 msec

(3) Program example

(a) Constant scan time = 80 msec



(b) To terminate constant scan



(4) Operation

(a) Constant scan is executed from the scan during which a setting is written to D9020.

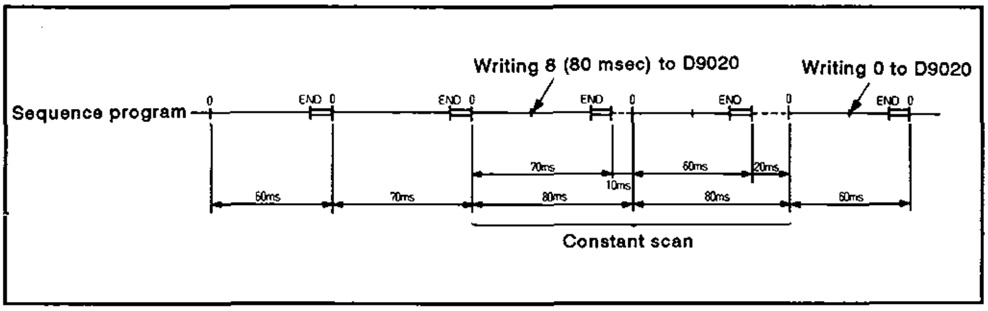


Fig. 2.13 Constant Scan Execution

(b) The constant scan time must be greater than the maximum scan time in the sequence program.

The constant scan is not executed correctly if the setting is shorter than the sequence program scan time.

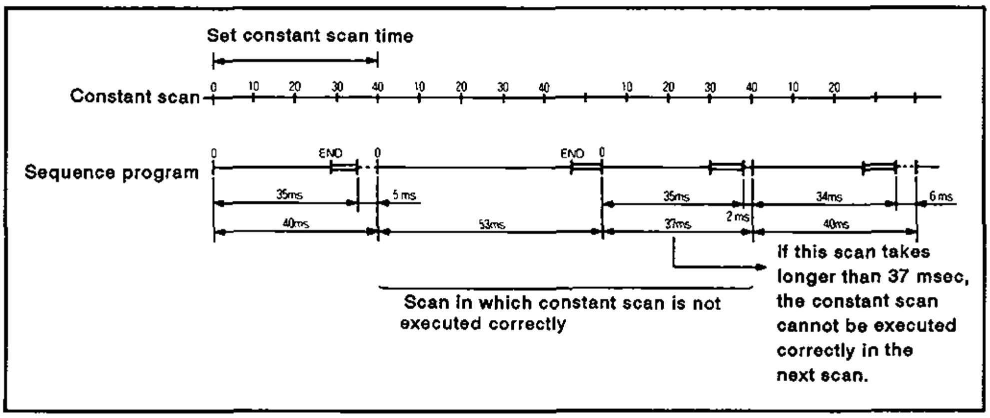


Fig. 2.14 Program Execution if Scan Time is Longer than the Set Constant Scan Time

(5) Accuracy

(a) The following processing is possible during waiting time in END processing.

Therefore, constant scan accuracy will be deteriorated by the corresponding interrupt processing time.

Interrupt	Processing Time
I/O interrupt	General data processing of AD51(S3): 0.2 msec to 0.5 msec Interrupt from AI61 or AD51(S3): 0.2 msec + (I0 to I17 interrupt program execution period)
10 msec interrupt	1.0 msec + I29 to I31 interrupt program execution time
Interrupt from a peripheral device	0.2 msec

If more than one interrupt indicated above occurs, the sum of these interrupt processing times is required.

OPERATION

(a) To execute a constant scan

1) Write the constant scan setting to D9020 with a sequence program.

2) Write the constant scan setting to D9020 with a peripheral device set in the test state.

(b) To terminate constant scan

1) Write "0" to D9020 with a sequence program.

2) Write "0" to D9020 with a peripheral device set in the test state.

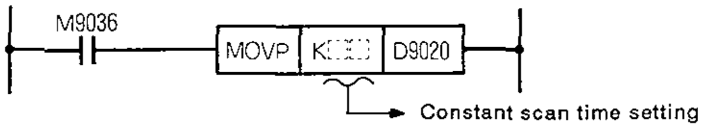
(c) To change the constant scan setting during run

1) Modify the program which writes the constant scan setting to D2020 using a peripheral device; rewrite the setting during run and set the constant setting instruction on.

2) Write the required setting to D9020 with a peripheral device in the test state.

CAUTION

(a) D9020 is cleared when the power supply to the SCPU is turned or when it is reset. The following program is required to initiate the constant scan function from the time the power is turned on after the SCPU is reset.



(b) The constant scan function is not executed correctly if a momentary power failure shorter than 20 msec occurs because the constant scan time is extended for the duration of momentary power failure.

2.2.2 Latch

APPLICATION

The latch function retains the data if a momentary power failure longer than 20 msec occurs to allow to continuous execution of the control.

FUNCTION

(1) Definition

The latch function retains the device data in the CPU module when the power supply to the SCPU is turned off and then on or it is reset.

(2) Latchable device

(a) The devices which can be latched are indicated below.

1) Latch relay (L)

2) Link relay (B)

3) Timer (T)

4) Counter (C)

5) Data register (D)

6) Link register (W)

(3) Clearing the latched data

(a) The latched data can be cleared by either of the following methods.

1) Reset switch

After placing the RUN switch in the STOP position, set the reset switch in the LATCH CLEAR position.

2) GPP/HGP/PHP

Execute the device clear operation.

(b) Clearing the latched data also clears the data in the unlatched area at the same time.

OPERATION

The latch range is set for each device by parameters.

CAUTION

(a) The device data in the latch range is retained by the battery (A6BAT) in the memory cassette. The battery is, therefore, required when operation is executed with a ROM which stores the sequence program.

(b) The latched and unlatched device data is stored in the CPU module. The data in the latch range is therefore destroyed if following operation is attempted while the power is off.

1) Disconnect the battery connector from the memory cassette.

2) Remove the memory cassette from the CPU module.

2.2.3 Remote RUN/STOP

APPLICATION

(a) In the following cases, remote run/stop control of the SCPU is possible without using the RUN keyswitch on the front panel of the CPU module.

1) The CPU module is out of reach.

2) The CPU module is stored in the control box.

3) The CPU is under the control of a computer, etc.

FUNCTION

(1) Definition

The remote RUN/STOP function controls run/stop of the SCPU using an external device (peripheral device, external input, computer, etc.) while the RUN keyswitch is in the RUN position.

(2) Operation

(a) Operation processing is controlled as indicated below:

1) Remote STOP

The SPCU is set in the STOP state after the sequence program is executed to the END (FEND) instruction.

2) Remote RUN

Remote run operation after the remote stop sets the SCPU in the RUN state in which the sequence program is executed from step 0.

OPERATION

(a) Remote run/stop operation is possible in the following three methods.

1) Remote RUN contact (external input)

The remote RUN contact is set with a peripheral device.

2) Peripheral device

3) Computer or AD51 (S3)

Remote RUN contact

The SCPU is set to the run or stop state according to the status of the remote run contact set with a parameter.

How the SCPU is controlled according to the status of the remote run contact is indicated below.

Remote run contact: OFF..........RUN state
ON ....STOP state

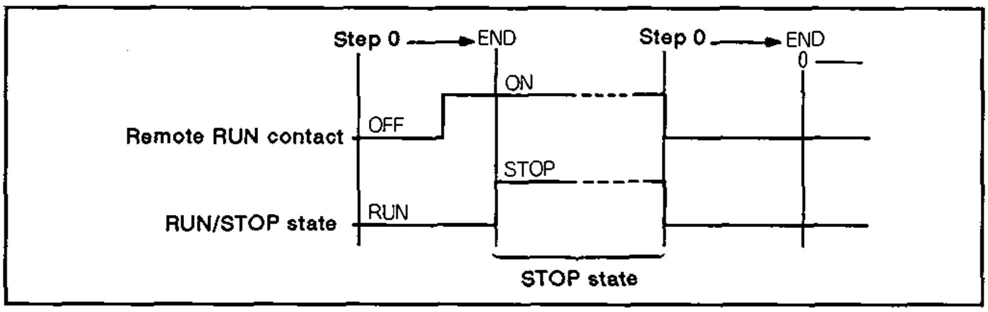


Fig. 2.15 Remote RUN/STOP Timing Using Remote Run Contacts

Peripheral device, computer, or AD51(S3)

The SCPU is set to the run or stop state according to the remote run/stop instruction input from a peripheral device, computer, or AD51 (S3).

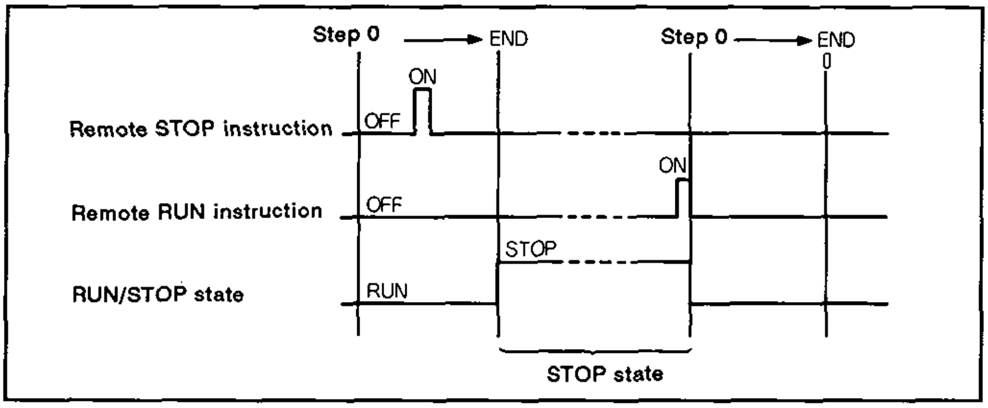


Fig. 2.16 Remote RUN/STOP Timing Using a Peripheral Device, Computer, or AD51(S3)

CAUTION

(a) Pay attention to the following points because the STOP state is given priority for remote RUN/STOP control.

1) If a remote STOP signal is given from the remote RUN contact, peripheral device, computer, or AD51 (S3), the SCPU is set in the STOP state.

2) To set the SCPU in the RUN state after setting it in the STOP state with a remote STOP signal, set all the factors (remote RUN contact, peripheral device, computer, and AD51 (53)) that called the stop state of the SCPU to the run state.

2.2.4 PAUSE

APPLICATION

The PAUSE function allows the process control, etc. to be resumed after the SCPU is set in the STOP state.

FUNCTION

(1) Definition

The PAUSE function stops SCPU operation while retaining the status of all outputs (Y).

(2) Operation

(a) The M9041 is turned on after the END of the scan during which the pause conditions are satisfied. Operation stops when the END (FEND) instruction in the next scan is executed.

(b) In the PAUSE state, the status of all outputs, after the one scan executed after the M9041 has been turned on, is retained.

Therefore, any output that should be off in the PAUSE state must be interlocked using the M9041.

OPERATION

(a) The SCPU can be set in the PAUSE state by any of the following three methods.

1) RUN keyswitch

2) Remote PAUSE contact

The remote PAUSE contact is set with a peripheral device.

3) Peripheral device

RUN keyswitch

When both of the following conditions are satisfied at the same time, operation stops after the END (FEND) instruction in the next scan is executed.

Conditions: RUN keyswitch set in the PAUSE position
M9040 is on

Operation can be resumed by either setting the RUN keyswitch in the RUN position or turning the M9040 off with a peripheral device.

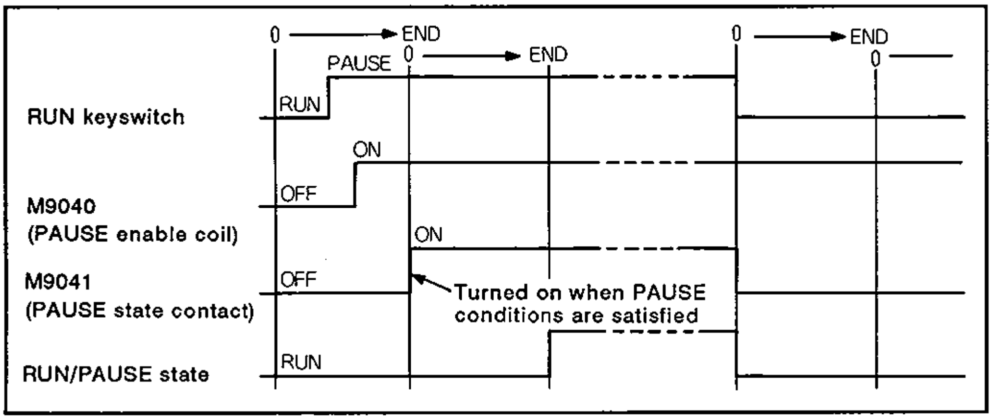


Fig. 2.17 PAUSE Timing with RUN Keyswitch

Remote PAUSE contact

1) When both of the following conditions are satisfied at the same time, operation stops after the END (FEND) instruction in the next scan is executed.

   Conditions: Remote PAUSE contact on
   M9040 is on

2) Operation can be resumed by either turning the remote PAUSE contact off or turning the M9040 off with a peripheral device.

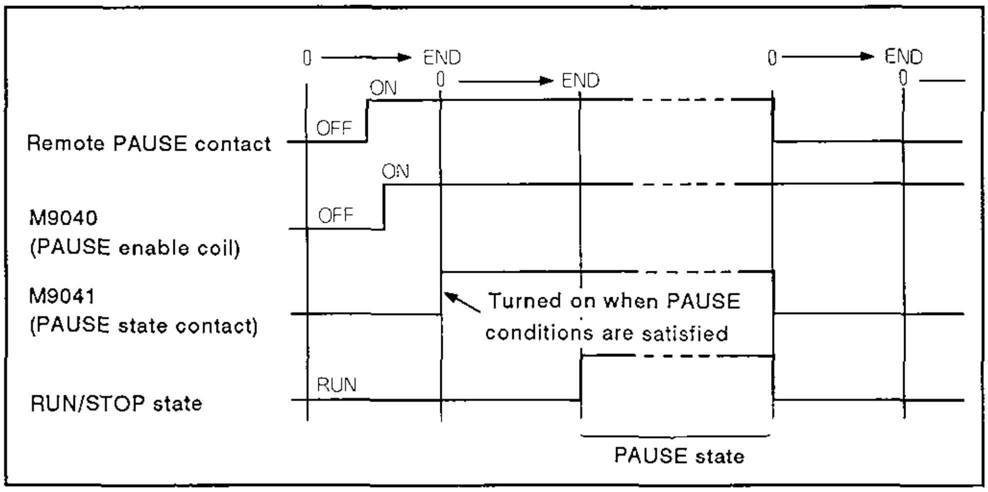


Fig. 2.18 PAUSE Timing with Remote PAUSE Contact

Peripheral device

1) Operation stops after the END (FEND) instruction, in the scan that follows the scan during which the remote PAUSE instruction is input from a peripheral device, is executed.

2) Operation can be resumed when the remote RUN instruction is input from a peripheral device.

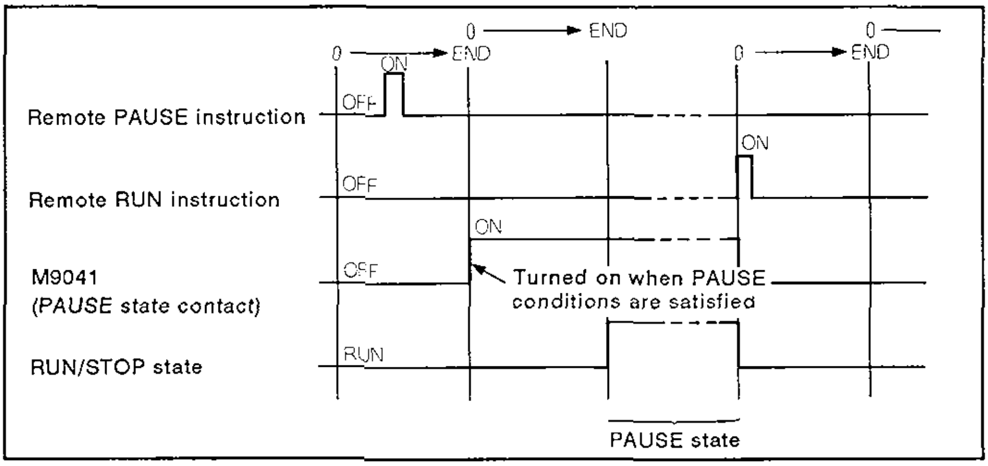


Fig. 2.19 PAUSE Timing with a Remote PAUSE Instruction

2.2.5 Status latch

APPLICATION

The status latch function is used to check device data during debugging under certain conditions.

FUNCTION

(1) Definition

The status latch function stores all the device data to the status latch area of the memory cassette by executing the SLT instruction.

The latched data can be read and monitored by the GPP/HGP/PHP.

(2) Stored data

The following data of each individual device is written to the status latch area.

1) X, Y, M, L, S, F, B ....ON/OFF data

2) T, C..................................ON/OFF data and present value

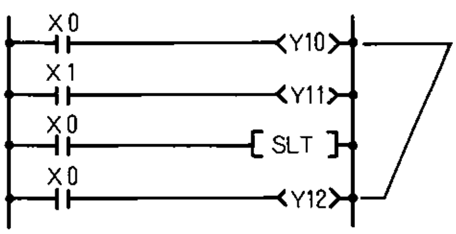
3) D, W, A, Z, V, R..............Currently stored data

(3) Data storage timing

(a) Data is stored in the status latch area when the SLT instruction is executed. Therefore, the device data, which changed after the execution of the SLT instruction, is not stored in the status latch area even if this change occurs within the same scan.

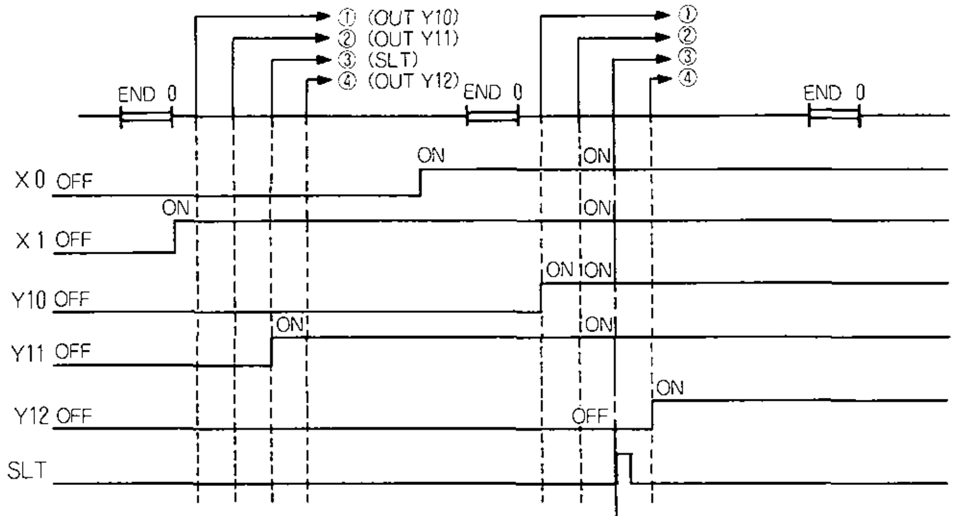
(b) Example

OUT instructions which are turned on/off by the same input (X0) are written before and after the SLT instruction.



OUT instructions which are turned on/off by the same input (X0) are written before and after the SLT instruction.

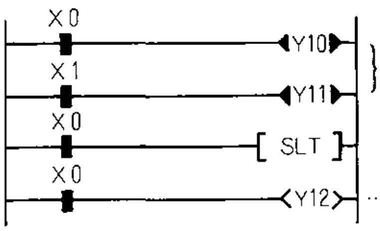
Timing chart



The following data is stored in the status latch area when the SLT instruction is executed.

X0	X1	Y10	Y11	Y12
ON	ON	ON	ON	OFF

Monitoring the status latch data



Y10 and Y11 are on because they were on before the execution of the SLT instruction.

..... Y0 is off even if X0 is on; this is because the Y12 was off when the SLT instruction is executed.

OPERATION

(a) Setting the status latch area

Specify the status latch area by setting the parameter with the GPP/HGP/PHP and write it to the SCPU.

(b) Executing status latch

Data is written to the status latch area when the SLT instruction is executed.

(c) Resuming status latch

To execute the SLT instruction again, reset the SLT instruction by executing the SLTR instruction.

CAUTION

(1) The scan time is extended as indicated below when the SLT instruction is executed.

Take these values into consideration when setting the SCPU watchdog timer.

	Device Memory Only	Device Memory and File Register
Processing time (msec)	8.5 ms	24.6 ms

2.2.6 Sampling trace

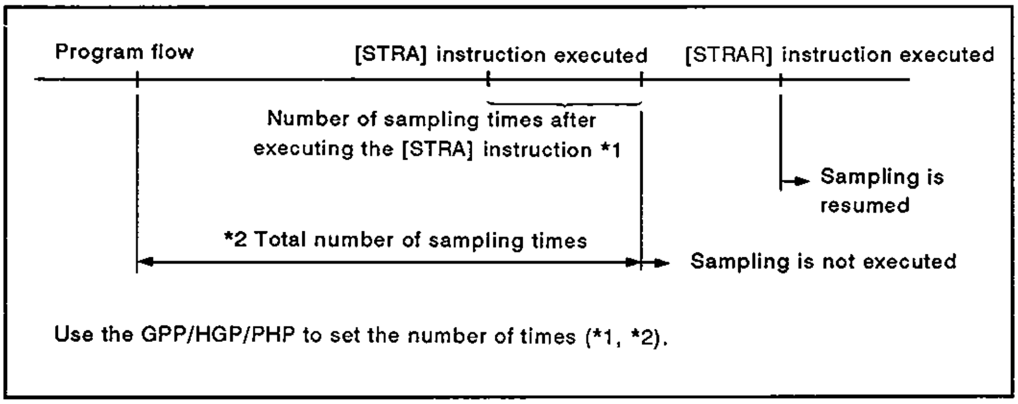
APPLICATION

The sampling trace function is used to shorten debugging time.

FUNCTION

(1) Definition

The sampling trace function executes sampling of the contents of the specified device in predetermined intervals (sampling periods) to store it in the sampling trace area of the memory cassette.



(2) Devices used

(a) The devices and allowable number of sampling points are indicated below.

1) Bit device (X, Y, M, L, S, F, B, T/C coil, T/C contact)

Maximum 8 points

2) Word device (T/C present value, D, W, R, A, Z, V)

Maximum 3 points

OPERATION

(a) Setting the sampling trace area

Set the sampling trace area using the GPP/HGP/PHP and write it to the SCPU.

(b) Setting the sampling trace data

Set the following sampling trace data using the GPP/HGP/PHP and write it to the SCPU.

1) Number of sampling trace times

2) Devices to be traced

3) Sampling period

(c) Starting the sampling trace

Sampling trace can be started by either of the following methods.

1) GPP/GHP/PHP

2) Turning on the M9047

(d) Terminating and stopping sampling trace

To terminate:

Execute the STRA instruction in a sequence program; after sampling is executed for the specified number of times and the data is latched, the sampling trace is terminated.

To stop:

Sampling trace can be stopped by either of the following methods:

* GPP/GHP/PHP
* Turning off the M9047

(e) Checking the sampling trace data

Read and monitor the sampling trace area data using the GPP/GHP/PHP.

(f) Resuming the sampling trace

Execute the STRAR instruction in a sequence program.

2.2.7 STEP-RUN operation

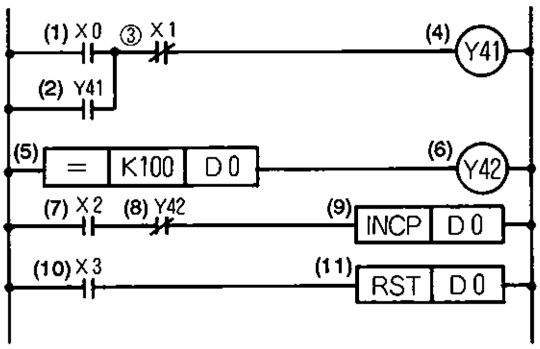
APPLICATION

The step-run function allows a sequence program to be run while checking the sequence program execution status and device data in response to on/off setting of the input signals.

FUNCTION

(1) Definition

The step-run function executes the sequence program operation in units of instructions.



In the step-run state, operation is executed step-by-step in the order of (1), (2), ... and (11).

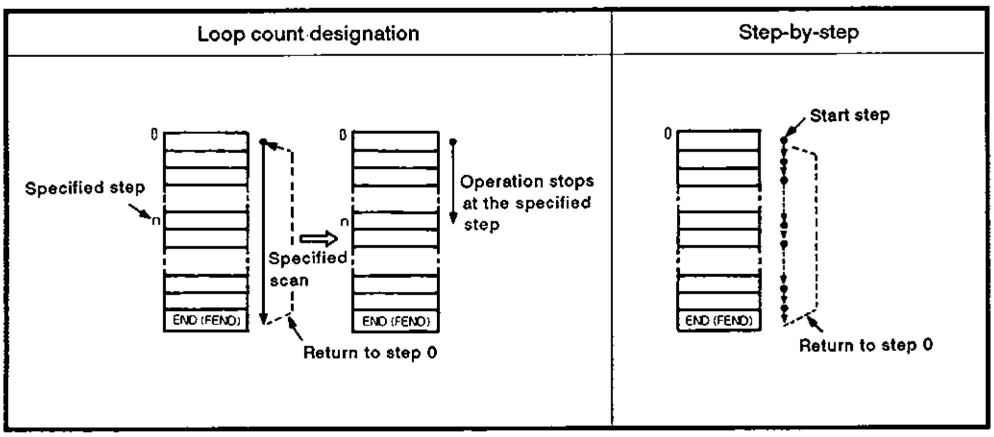
(2) Types

(a) Step-run by designating the loop count

Operation stops at the specified step after the sequence program is executed the specified number of times.

(b) Step-run in units of instructions

Operation is executed step-by-step in units of instructions starting at step 0 or the current step.



(3) Output (Y) status with the RUN switch set to STEP-RUN

(a) The RUN keyswitch can be placed in the STEP-RUN position by either of the following methods.

1) RUN → PAUSE → STEP-RUN

When the RUN keyswitch is placed in the STEP-RUN position, operation stops with all output statuses retained.

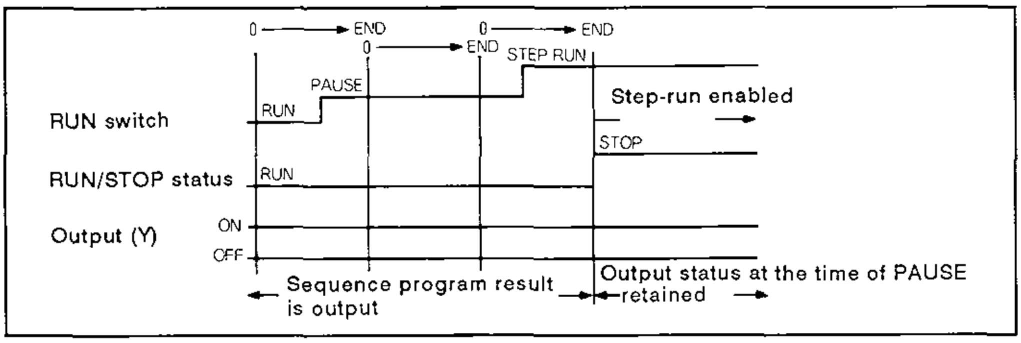


Fig. 2.20 Timing Chart When the Switch Is Turned From RUN → PAUSE → STEP-RUN

2) RUN → STOP → STEP-RUN

When the RUN keyswitch is operated in the above indicated sequence, the output status varies depending on the setting for the "STOP → RUN display state".

i) To output operation status prior to stop

When the RUN keyswitch is placed in the STOP position, operation stops with all outputs set off.

When the RUN keyswitch is then placed in the STEP-RUN position, output status at the time of STOP setting is output with the operation stopped.

ii) To output the status after operation

When the RUN keyswitch is placed in the STOP position, operation stops with all outputs set off.

When the RUN keyswitch is then placed in the STEP-RUN position, operation remains stopped with all the outputs off.

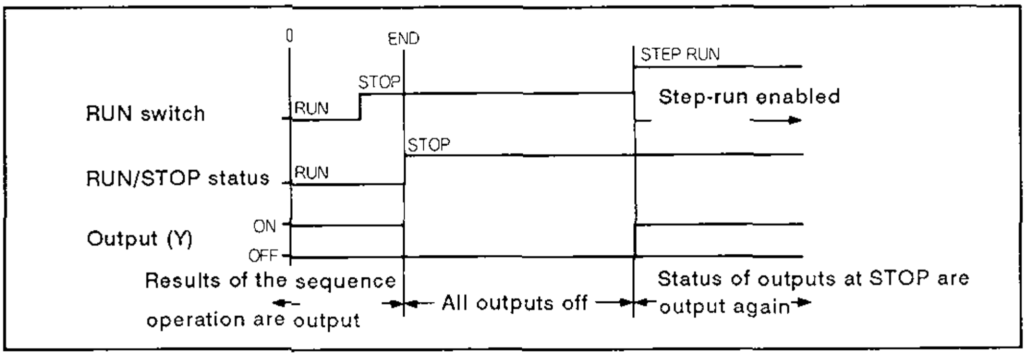


Fig. 2.21 Timing Chart When the Switch Is Turned From RUN → STOP → STEP-RUN

(4) Timer and special timing clock processing during step run

(a) Timers

1) 10 msec timer.......... 10 msec increased every scan

2) 100 msec timer........ 100 msec increased every 10 scans

(b) Special timing clocks

1) M9030 (0.1 sec clock) ...... Turned on/off every 5 scans

2) M9031 (0.2 sec clock) ...... Turned on/off every 10 scans

3) M9032 (1 sec clock).......... Turned on/off every 50 scans

4) M9033 (2 sec clock).......... Turned on/off every 100 scans

5) M9034 (1 min clock).......... Turned on/off every 3000 scans

OPERATION

(a) Turn the RUN keyswitch to the STEP-RUN position.

(b) Execute step-run using the GPP/GHP/PHP.

For details, see the GPP/GHP/PHP Operation Manual.

CAUTION

(a) When executing step-run by specifying a loop count, the number of loop execution times is counted when the specified step is executed.

Therefore, the loop count is not increased if the specified step is skipped by the CJ instruction, for example.

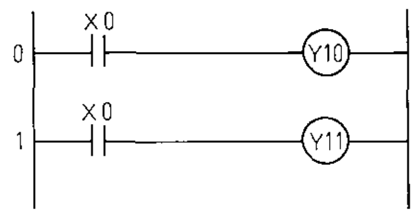
(b) When the RUN keyswitch is turned from STEP-RUN or RUN to the STOP position, the output status just before the switch is placed in the STOP position is stored in the SCPU memory. Therefore, the output stored in the memory is output when the RUN keyswitch is turned to the RUN or STEP-RUN position from the STOP position.

If output of the stored output status is not required, reset the SCPU first before turning the RUN keyswitch to RUN or STEP-RUN.

(c) I/O refresh is executed while operation stops in step-run operation. Refer to Table 2.3 in Section 2.1.8.

Note that the CPU module using the refresh state fetches the input signals or outputs the signals externally while in operation.

If step-run is executed using the following circuit in units of instructions, outputs Y10 and Y11 are off and on respectively after the execution of step 1 assuming that X0, originally off when step 0 is executed, is turned on when step 1 is executed.



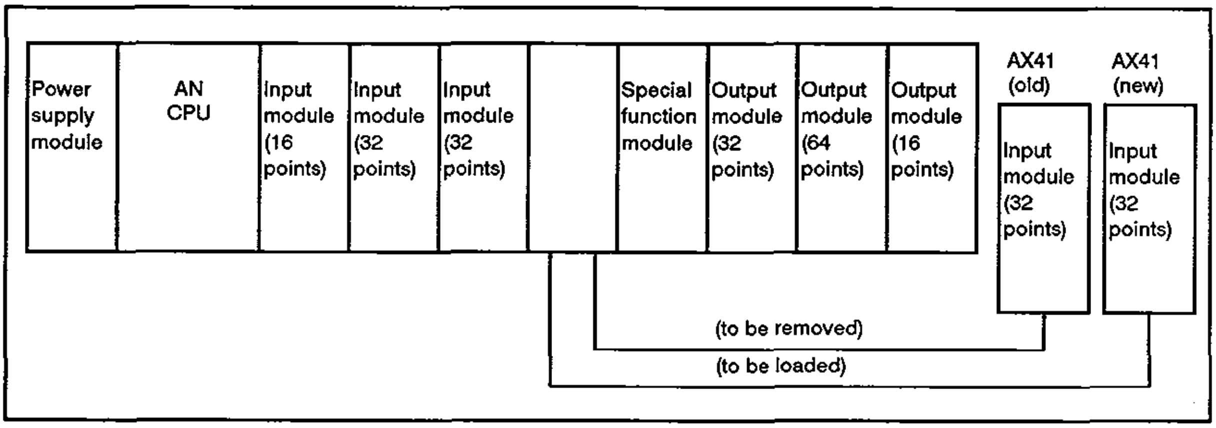
2.2.8 Online I/O module replacement

APPLICATION

(a) An input or output module can be replaced while the SCPU is running.

FUNCTION

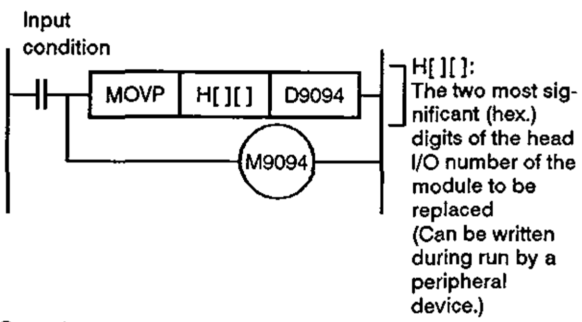
(a) The output or input module can be replaced in units of slots as indicated below while the system is in the online state.



(b) With this function, it is possible to replace the module without stopping sequence operation.

OPERATION

(a) This replacement can be executed in either of the following methods.

<table>
<tr><th>a) Using a sequence program</th><th>b) Using a peripheral device</th></tr>
<tr><td>Write the following program in a sequence program and turn the required input condition on while the CPU is running.


Procedure:
1) Turn on the input condition.
Set the module number to D9094. M9094 is set on.
2) Replace the I/O module.
3) Turn off the input condition.
The M9094 is set off. Operation of the new module starts.</td>
<td>Connect the peripheral device to the CPU.
Set the CPU to RUN or STOP state.
Procedure:
1) In the PC test state, set the two most significant digits of the head I/O number of the module to be replaced after converting it to a hexadecimal number (3 digits).
Examples:
Head I/O number 70 → Set H07
Head I/O number 170 → Set H17
2) Turn the M9094 on in the test state.
3) Replace the module.
4) Turn off (reset) the M9094 in the test state.
Operation of the new module starts.</td></tr>
</table>

(b) Processing during replacement

The modules execute the following processing depending on the on/off status of the M9094.

M9094	Module	Processing during CPU RUN
ON	CPU	1) Stores the operating status to the X or Y image memory area when the first END processing is executed after M9094 has been turned on. After this, the CPU does not access the corresponding input or output module.
		2) After M9094 is turned on, the corresponding input module operation is processed using the data (ON or OFF) in the input (X) image memory area.
		3) After M9094 is turned on for the corresponding output module operation, operation results of the CPU are stored to the output (Y) image memory area.
		4) After M9094 is turned on, the CPU does not execute module verification and blown fuse check for all modules.
	Module to be replaced	1) Stops after retaining the operating status at the first END processing after M9094 is turned off.
	Other modules	1) Execute normal operation. The module giving an output according to the data of the module to be changed operates with the input data fixed.
OFF	CPU	1) The CPU accesses the corresponding input module when the first END processing is completed after M9094 is turned off.
		2) The output module outputs the output (Y) image area data (ON or OFF) in the first END processing after M9094 is turned off. The CPU accesses the corresponding output module thereafter.
		3) After M9094 is turned off, the CPU resumes module verification and blown fuse check.
	Module to be replaced	1) Resumes access using the CPU image memory area data after the END processing after M9094 is turned off.
	Other modules	1) Execute normal operation.

* Processings during the replacement using a peripheral device with the CPU "STOP" are as described above. However, because all outputs are off, all modules are in the stop state. Outputs are restored by setting the SCPU to the RUN state.

CAUTION

No.	Item	Caution
1	Special module replacement	Only change a special module after turning off the power.
2	Replaceable I/O module	1) The new module must have an equal number of I/O points as that of the module to be replaced. An error occurs if the number of I/O points of the new module is different.
		2) Only the module specified in D9094 can be replaced. If a module which is not specified is replaced, it will cause an I/O error of the device connected to that module.
3	CPU RUN and STOP key switch operation	Set the key switch in the required position before replacing the module and leave it in the set position until the module replacement is completed. If its position is changed during replacement, CPU operation processing varies causing I/O module malfunctioning.
4	D9094	1) A hexadecimal (HFF) is set to D9094 when the CPU power is turned on; the data memory is cleared when the power is turned off. The data is overwritten with the two most significant digits of the head I/O number, expressed in three-digit hexadecimal.
		2) A correct I/O number must be specified because the D9094 set range and set value are not checked by the CPU.
5	Disabling test operation using a peripheral device	After M9094 is turned on, do not execute test operation with a peripheral device. If test operation is executed, the results will be set in the image memory area to be output when M9094 is turned off, causing malfunctions.
6	Safety measures for module replacement during CPU RUN	1) The module to be replaced stops operation when M9094 is turned on and the new module starts operation using the data in the image memory area when it is turned off. Therefore, check the operating status of the system before starting operation by turning off M9094; take safety measures as necessary.
		2) Specials safety measures are ot required when the output module is used only for display, such as A6FD. However, take proper safety measures such as writing interlock step in the sequence program or turning off the macinery power if I/Os call mechanical motion.
7	If a momentary power failure occurs for a period longer than 20 msec with the module removed (CPU initial start)	1) Turn off the power and load the new module. If the module is loaded with the power on, a "UNIT VERIFY ERROR" occurs.
		2) I/O numbers are not changed if I/O assignment is executed. This will prevent input and output errors. If the CPU module starts operation in the initial start state, it assigns I/Os automatically assuming that the empty slot are the vacant 16 points. See Section 2.1.2. Therefore, if the removed I/O module has 32 or more points, I/O numbers are changed, causing input and output errors. Turn off the power before loading the module.

2.3 I/O Address Assignment

2.3.1 Basic concept

I/O addresses are assigned in either of the following methods.

(a) I/O ASSIGNMENT not executed with a peripheral device

(b) I/O ASSIGNMENT executed with a peripheral device

(1) I/O ASSIGNMENT not executed with a peripheral device

(a) Assign I/O numbers in the order of the set stage numbers of the extension base independent of the connection order of the extension cables.

(b) Assign the I/O numbers to the main base and extension base(s) assuming that each base has eight slots.

Although the A74B provides four slots, it occupies eight slots (0H to 7FH).

If five slots are used in all bases except the final stage, I/O numbers are assigned to the next base by adding the I/O number for three slots (48 points).

(c) Assign 16 points to an empty slot.

(d) If I/O addresses are assigned skipping some extension stage numbers, "the number of skipped stages x 8 slots" are assumed to occupy 16 points per slot.

(2) I/O ASSIGNMENT executed with a peripheral device

(a) Using the I/O ASSIGNMENT function, the following settings are possible in units of slots.

1) Empty slot

This is the setting for not-used slots; assignment is possible for 0, 16, 32, 48, or 64 points.

2) Input (X) module

This is the setting for input modules; assignment is possible for 16, 32, 48, or 64 points.

3) Output (X) module

This is the setting for output modules; assignment is possible for 16, 32, 48, or 64 points.

4) Function module (F)

This is the setting for special function modules; assignment is possible for 16, 32, 48, or 64 points.

Each slot assigns the I/O numbers at the set number of points.

(b) The A74B occupies 0H to 7FH. For I/O assignment, set slots 0 to 7 for the empty 16 points.

(c) For the slot where the special function module is loaded, use the setting for special function module (F) with the same number of points as the actually loaded module.

(d) If the setting is not made for the function module or the setting of the number of points is incorrect, an error occurs.

(e) If the number of points assigned is smaller than that of the module actually loaded, the available number of I/O points is reduced.

Example:

If "16 input points" are set on the I/O assignment screen while an input module with "32 points" is loaded, the latter 16 points cannot be used.

(f) If the set number of I/O points is greater than the number of I/O points of the actually loaded module, the excess number of points are handled as dummy points.

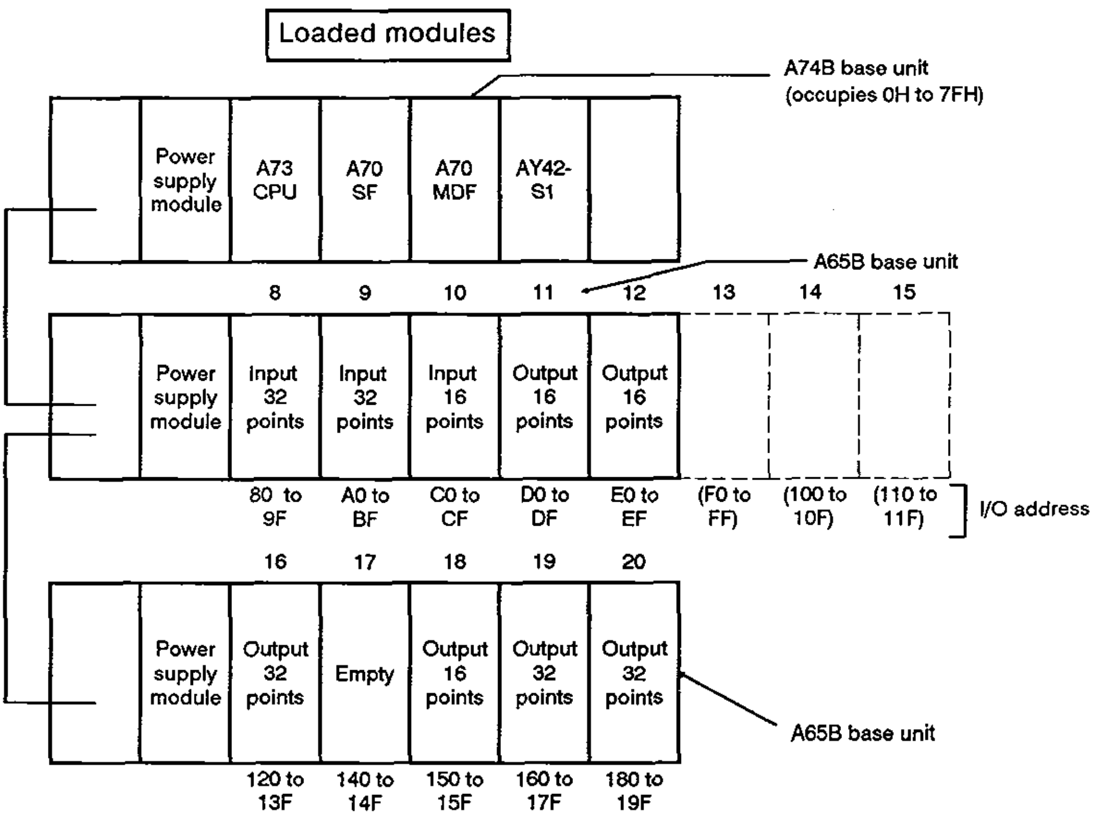
(g) If "S" (empty) is set for the slot where an I/O module is loaded, the loaded I/O module cannot be used.

(h) If a remote I/O station is used in the data link, I/O assignment for the remote station is necessary.

(i) A combined I/O composite module, such as the A42XY, is handled as an output module when executing an I/O assignment.

2.3.2 I/O address assignment

(1) I/O ASSIGNMENT of a peripheral device is not used



(2) I/O ASSIGNMENT of a peripheral device is used

(a) I/O ASSIGNMENT example

* I/O ASSIGNMENT *

SLT NO.	I/O UNT	SLT NO.	I/O UNT	SLT NO.	I/O UNT	SLT NO.	I/O UNT	SLT NO.	I/O UNT	SLT NO.	I/O UNT	SLT NO.	I/O UNT	SLT NO.	I/O UNT
0	S16	16	Y32	32		48		64		80		96		112	
1	S16	17	S32	33		49		65		81		97		113	
2	S16	18	Y16	34		50		66		82		98		114	
3	S16	19	Y48	35		51		67		83		99		115	
4	S16	20	Y32	36		52		68		84		100		116	
5	S16	21		37		53		69		85		101		117	
6	S16	22		38		54		70		86		102		118	
7	S16	23		39		55		71		87		103		119	
8	X32	24		40		56		72		88		104		120	
9	X16	25		41		57		73		89		105		121	
10	X32	26		42		58		74		90		106		122	
11	Y16	27		43		59		75		91		107		123	
12	S16	28		44		60		76		92		108		124	
13	S 0	29		45		61		77		93		109		125	
14	S 0	30		46		62		78		94		110		126	
15	S 0	31		47		63		79		95		111		127	

EMP (S) SLOT: 1:0 PTS, 2:16 PTS, 3:32 PTS, 4:48 PTS, 5:64 PTS
X (X) MODULE: 6:16 PTS, 7:32 PTS, 8:48 PTS, 9:64 PTS
Y (Y) MODULE: A:16 PTS, B:32 PTS, C:48 PTS, D:64 PTS
FCT (F) MODULE: E:16 PTS, F:32 PTS, G:48 PTS, H:64 PTS

MEMORY TOTAL 16K BYTES

PRESS <END> TO END SETTING

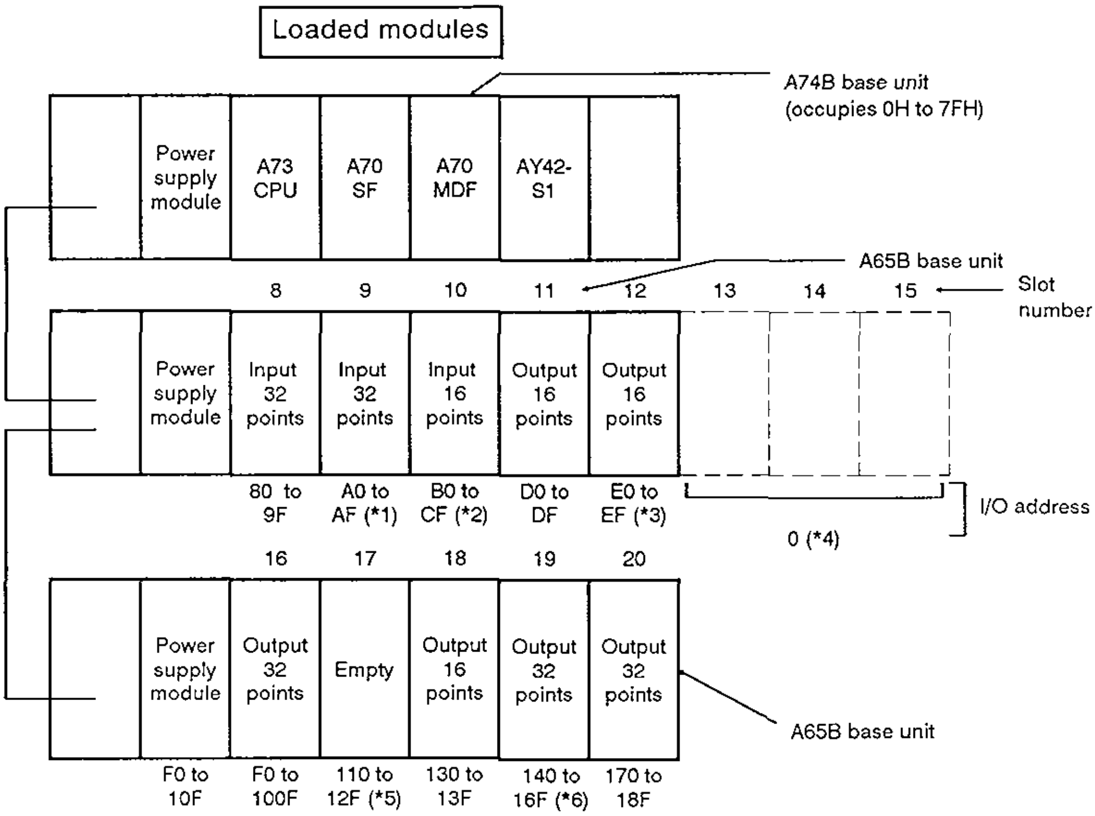
MODE PARAMETER FCT

KEY IN DATA

MESSAGE

A3

(b) I/O numbers after execution of I/O ASSIGNMENT



REMARKS

*1: Because "X16" is set, the 16 latter points cannot be used.

*2: Because "X32" is set, C0 to CF are assigned as dummy points.

*3: Because "S16" is set, the module cannot be used as an output module.

*4: Because "S0" is set for slots 13, 14 and 15, no I/O points are set.

*5: Because "S32" is set, 32 points are reserved.

*6: Because "Y48" is set, 160 to 16F are assigned as dummy points.

APPENDICES

APPENDIX 1 SCPU ERROR CODE LIST

If an error occurs while the PC starts or is running, the self-diagnosis function provides an error indication or stores the error code and the step number involved to a special register. Table 1.1 indicates the list of errors, possible causes, and necessary countermeasures.

Error codes can be read using a peripheral device. For the operation to read the error codes, refer to the Operation Manual for the peripheral device.

1.1 SCPU Error Code List

Table 1.1 Error Code List

Error Message	Content of Special Register D9008 (BIN value)	Resulting CPU Status	Error and Causes	Corrective Action
INSTRCT CODE ERR. [checked during execution of an instruction]	10	Stop	An instruction code which is undecodable by the SCPU is contained in the program. (1) Undecodable instruction contained in installed ROM. (2) Instruction code contained in memory made undecodable due to adverse effects of unknown source on stored data.	(1) Read the program step with a peripheral device and correct the program in that step. (2) When the ROM is used, correct the ROM data or change the ROM that contains the correct data.
PARAMETER ERROR Checked when : * Power ON or reset * STOP/PAUSE →RUN/STEP-RUN	11	Stop	Parameter data in the SCPU memory is changed due to noise or incorrect memory installation.	(1) Check SCPU memory installed condition; install the memory correctly. (2) Read the SCPU memory parameter data with the peripheral device; check and correct the parameter data and write the correct data to the memory.
MISSING END INS. Checked when : * M9056 or M9057 is turned on STOP/PAUSE →RUN/STEP-RUN	12	Stop	(1) Absence of END (FEND) instruction in the program. (2) Absence of END instruction in a subprogram when a subprogram is set with the parameter.	(1) Add END instruction to end of program.

Table 1.1 Error Code List (Continued)

Error Message	Content of Special Register D9008 (BIN value)	Resulting CPU Status	Error and Causes	Corrective Action
CAN'T EXECUTE (P) Checked when : * The CJ, SCJ, JMP, CALL(P), or FOR - NEXT instruction is executed * STOP/PAUSE →RUN/STEP-RUN	13	Stop	(1) Absence of, or multiple destinations for, a jump instruction designated by the CJ, SCJ, CALL, CALLP, or JMP instructions. (2) A subprogram is not set although the CHG instruction is specified. (3) The RET instruction is executed although coresponding CALL instruction is absent from the program. (4) The CJ, SCJ, CALL, CALLP, or JMP instruction having the jump destination after the END instruction is executed. (5) The number of FOR instructions and that of the NEXT instructions do not agree with each other. (6) A JMP instruction is writen in a step within the FOR - NEXT loop, causing the program to exit the FOR - NEXT loop. (7) The program exits the subroutine by executing the JMP instruction before executing the RET instruction. (8) The destination of the JMP instruction causes the program to jump to a step into the FOR - NEXT loop or the subroutine.	(1) Read the error step with a peripheral device and correct the program. Insert the jump destination or eliminate wrong jump destinations.

Table 1.1 Error Code List (Continued)

Error Message	Content of Special Register D9008 (BIN value)	Resulting CPU Status	Error and Causes	Corrective Action
CHK FORMAT ERR. Checked when : * STOP/PAUSE →RUN/STEP-RUN	14	Stop	(1) The circuit block of the CHK instruction contains an instruction other than LDX, LDIX, ANDX, and ANIX including the NOP instruction. (2) More than one CHK instruction is written. (3) The circuit block of the CHK instruction contains more than 150 contacts. (4) The X device number in the CHK instruction circuit block is greater than X7FE. (5) The following circuit block is not written preceding the CHK instruction circuit block. CJ[ ] (6) The D1 device (number) in the following instruction and the device (number) of the contact which is written preceding the "CJ [ ]" instruction do not agree with each other. CHK | D1 | D2 (7) There is no P254 pointer located on the head of the CHK instruction circuit block. P254 CHK | D1 | D2	(1) Check the program of the CHK instruction circuit block for (1) to (6) in the left column. Correct the error in the program with a peripheral device and execute the program again. (2) This error code is valid only when the direct input/output control method is used.
CAN'T EXECUTE (I) Checked when : * An interruption occurs * STOP/PAUSE →RUN/STEP-RUN	15	Stop	(1) Absence of, or multiple numbers for, interrupt pointer (I) in the program corresponding to the used interrupt module. (2) The IRET instruction is not written in the interrupt program. (3) The IRET instruction is written in a program other than an interrupt program.	(1) Check if a interrupt program corresponding to the interrupt module is present; create the interrupt program or eliminate the identical I numbers. (2) Check if an IRET instruction is written in the interrupt program. Write it in the program if not present. (3) Check if the IRET instruction is written in a program other than an interrupt program; delete the IRET instruction written in a program other than the interrupt program.
CASSETTE ERROR Checked when : * Power ON or reset	16	Stop	The SCPU memory cassette is not installed.	Install the SCPU memory cassette and reset it.

IB (NA) 66233-A

Table 1.1 Error Code List (Continued)

Error Message	Content of Special Register D9008 (BIN value)	Resulting CPU Status	Error and Causes	Corrective Action
RAM ERROR Checked when : * Power ON or reset * M9084 is turned on during stop	20	Stop	(1) The SCPU checks if the data can be correctly written to or read from the PC CPU data memory area. This error indicates that either read or write, or both read and write are impossible.	The SCPU hardware is faulty. Contact the system supplier.
OPE. CIRCUIT ERR. Checked when : * Power ON or reset	21	Stop	(1) The operation circuit in the SCPU does not operate correctly.	
WDT ERROR [checked continuously]	22	Stop	The scan time exceeds the watchdog error monitor time. (1) User program scan time is longer than the watchdog error monitor time. (2) Temporary power failure during program scan caused the scan time to be extended.	(1) Calculate user program scan time and change parameter setting for scan time to a larger value. (2) Monitor contents of special register D9005. If it is not "0", the supply voltage is not stable: check the power supply and minimize voltage fluctuation.
END NOT EXECUTE Checked when : * END processing is executed	24	Stop	(1) The END instruction is read in another instruction code due to noise or other causes. (2) The END instruction is changed to another instruction code.	(1) Reset the SCPU using the board reset operation or the reset key switch located on the front panel and run the SCPU again. If the same error occurs again, the CPU hardware is faulty. Contact the system supplier.
UNIT VERIFY ERR. Checked when : * END instruction is executed * Not checked when the M9048 or M9094 is on	31	Stop (Operation)	The I/O information differs with that existent when power was turned on. (1) An I/O module (including a special function module) failure occurred during operation. Or, different module is loaded.	(1) If the module position is correct and loaded correctly, reset the SCPU by resetting the board or using the reset key switch located on the front panel. (2) The bit corresponding to the module with which the verify error for special registers D9116 to D9123 has occurred is "1". Check the bits with a peripheral device; check the module corresponding to bit "1" indication. If the module is loaded correctly, the module is faulty.
FUSE BREAK OFF Checked when : * END instruction is executed * Not checked when the M9048 or M9094 is on	32	Operation (Stop)	Fuse blown in an output module.	(1) Check the fuse blown indicating LED on the output modules. Replace the fuse if blown. (2) It is possible to check the fuse blown output module with a peripheral device. Bit "1" registered in special registers D9100 to D9107 indicates the fuse blown output module.

IB (NA) 66233-A

Table 1.1 Error Code List (Continued)

Error Message	Content of Special Register D9008 (BIN value)	Resulting CPU Status	Error and Causes	Corrective Action
CONTROL-BUS ERR. Checked when : * FROM/TO instruction is executed * Power ON or reset * STOP/PAUSE →RUN/STEP-RUN	40	Stop	The FROM/TO instruction is not executed. (1) Special function module control bus error	(1) Hardware fault of special function module, SCPU board, or base unit. Change and check the hardware. Contact the system supplier.
SP. UNIT DOWN Checked when : * FROM/TO instruction is executed * Power ON or reset * STOP/PAUSE →RUN/STEP-RUN	41	Stop	There is no response from a special function module although the special function module is accessed when the FROM/TO instruction is executed. (1) The accessed special funcion module is faulty.	Hardware faulty of the accessed special function module. Contact the system supplier.
LINK UNIT ERROR Checked when : * Power ON or reset * STOP/PAUSE →RUN/STEP-RUN	42	Stop	(1) The AJ71R22 module or the AJ71P22 module is loaded in the master station.	(1) Remove the AJ71R22 module or the AJ71P22 module from the master station. Next, reset the board or reset the SCPU with the reset key switch on the front panel.
I/O INT. ERROR Checked when : * An interruption occurs	43	Stop	An interruption occurred although an interruption module is not loaded.	(1) Hardware fault of any of the modules on the expansion base unit. Check the module loaded conditions.
SP.UNIT LAY. ERR. Checked when : * Power ON or reset * STOP/PAUSE →RUN/STEP-RUN	44	Stop	(1) More than 3 computer link modules are loaded for one CPU module. (2) More than one AJ71P22 or AJ71R22 board is loaded. (3) More than one interruption module is loaded. (4) Using the parameter setting with a peripheral device, I/O assignment is executed for a special function module while an I/O unit is actually loaded. Or, I/O assignments executed for an I/O module while a special function module is actually loaded.	(1) Reduce the number of CPU modules to two or one. (2) Reduce the number of AJ71P22 or AJ71R22 board to one none. (3) Reduce the number of the interrupt module to one. (4) Re-set the I/O assignment meeting the actual board loaded status.
SP. UNIT ERROR Checked when : * FROM/TO instruction is executed	46	Stop (Operation)	(1) A FROM/TO instruction accessed a slot in which a special function module is not loaded.	(1) Read the error step; check and correct the FROM/TO instruction.
LINK PARA. ERROR Checked when : ** Power ON or reset * STOP/PAUSE →RUN/STEP-RUN	47	Operation	(1) The contents concerning link range setting that were written to the parameter area and the link parameter settings that are read by the SCPU do not agree with each other. (2) The setting for the total number of slave stations is "0".	(1) Write the parameters and check them again. (2) If the same error is displayed again, the hardware is faulty. Contact the system supplier.

Table 1.1 Error Code List (Continued)

Error Message	Content of Special Register D9008 (BIN value)	Resulting CPU Status	Error and Causes	Corrective Action
OPERATION ERROR Checked when : * Each instruction is executed	50	Operation (Stop)	(1) The result of BCD conversion exceeds the specified range (9999 or 99999999). (2) The setting exceeds the specified device range making operation impossible. (3) The file register is used in the program without setting the file register capacity.	(1) Read the error step with a peripheral device and check and correct the program (device setting range, BCD conversion value, etc.).
BATTERY ERROR Checked continuously * Not checked when M9048 is on	70	Operation	(1) Low battery voltage (2) Battery is not connected	(1) Replace battery. (2) When RAM is used or power failure data backup function is used, connect the battery leads.

APPENDIX 2 ERROR CODES STORED BY PCPU

The errors detected by the PCPU are servo program setting errors and positioning errors.

(1) Servo program setting error

A servo program setting error is an error in the positioning data set by a servo program. This type of error is checked when the servo program is started.

An error will occur when the positioning data is designated indirectly.

When an error occurs,

1) The servo program setting error flag (M9079) is set,

2) The program number is stored in the error program number register (D9189), and

3) The error code is stored in the error item information storage register (D9190).

(2) Positioning error

(a) A positioning error is an error which occurs at the start of or during positioning control. Positioning errors are classified into three levels: minor errors, major errors, and servo errors.

1) Minor errors

An error caused by a sequence program or the servo program. Codes "1" to "999" are used to indicate this type of error.

After checking the error code, correct the sequence program or the servo program to remove the cause of the error.

2) Major errors

This is an error caused by the externally input signals or the control instructions input from the SCPU. Codes "1000" to "1999" are used to indicate this type of error.

After checking the error code, remove the cause of the error; check the status of the external input signal or the sequence program.

3) Servo errors

This is an error detected by the servo amplifier. Codes "2000" to "3999" are used to indicate this type of error.

After checking the error code, remove the cause of the error in the servo system.

(b) If an error occurs, the error detection signal of the error occurrence axis is turned on and the corresponding error code is stored in the minor error code, major error code, or servo error code storage register.

Table 2.1 Error Code Storing Registers and Error Detection Flags

Error Level \ Device	Error Code Storing Register								Error Detection Signal
	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5	Axis 6	Axis 7	Axis 8	
Minor error	D806	D826	D846	D866	D886	D906	D926	D946	Xn7
Major error	D807	D827	D847	D867	D887	D907	D927	D947	
Servo error	D809	D828	D848	D868	D888	D908	D928	D948	Xn8

(c) The error code is overwritten. If an error occurs while the error code of the previous error is stored in the register, the error code of current error overwrites the previous error code.

A peripheral device (A6MD or A6GPP/A6PHP booted by SW0GP-A73P) is used to check the error log.

(d) The error detection flag and the error code are retained until the error reset signal (Yn7) or the servo error reset signal (Yn8) is turned on.

POINT

(1) With some servo errors, the same servo error code is stored again when the servo error reset (Xn8: ON) is executed.

(2) If a servo error occurs, remove the error cause in the servo amplifier and execute servo reset operation.

2.1 Servo Program Setting Error

Table 2.2 indicates the list of error codes, error details, and necessary measures to be taken for a servo program setting error.

An "n" in the error code, indicated by an asterisk (*), indicates an axis number (1 to 8).

Table 2.2 Servo Program Setting Error List

Error Code in D9190	Error Name	Error Details	Error Processing	Corrective Action
1	Parameter block number error	Designated parameter block number is not between "1" and "16".	The servo program is executed assuming that the parameter block number is "1" (default).	Correct the parameter block number.
n03 *	Address/axis travel distance setting error (excluding velocity control, and V/P control)	1) In absolute positioning control, an address outside the setting range is designated. Units: degree — Address Setting Range: 0 to 35999999, $\times 10^{-5}$ degree/min 2) In incremental positioning control, axis travel distance setting is "−2147483648" (H80000000).	1) Positioning control does not start. (In interpolation control, any axis involved with the designated interpolation control does not start.) 2) An axis decelerates and stops if an error is detected in the velocity change control or constant velocity control. 3) If the control is to start more than one servo program simultaneously, no servo program is executed if one of these servo programs contains an error.	1) Correct the set address for degree units. Setting range: 0 to 35999999 2) Correct the set axis travel distance. Setting range: 0 to ± $(2^{31} - 1)$
4	Command velocity error	1) Designated velocity is outside the allowable setting range. Setting range: 1 to velocity limit 2) Designated velocity is outside the allowable address setting range. Units / Address Setting Range: mm: 1 to 600000000, $\times 10^{-2}$ mm/min inch: 1 to 600000000, $\times 10^{-3}$ inch/min degree: 1 to 600000000, $\times 10^{-3}$ degree/min PULSE: 1 to 1000000, PLS/sec	1) If the designated velocity is 0 or smaller, the axis will not start. 2) If the designated velocity exceeds the limit, the control is made at the velocity limit.	1) Correct the designated velocity. Setting range: 1 to Velocity limit
5	Dwell time setting error	The set dwell time is outside the allowable setting range: Setting range: 0 to 5000	Positioning control is executed assuming a dwell time setting of "0" (default).	Correct the dwell time setting. Setting range: 0 to 5000
6	M code setting error	The set M code is outside the allowable setting range. Setting range: 0 to 255	Positioning control is executed assuming an M code setting of "0" (default).	Correct the M code setting. Setting range: 0 to 255
7	Torque limit setting error	The set torque limit is outside the allowable setting range. Setting range: 1 to 500	Positioning control is executing assuming the torque limit of the designated parameter block.	Correct the torque limit setting. Setting range: 1 to 500

Table 2.2 Servo Program Setting Error List (Continued)

<table>
<tr><th>Error Code in D9190</th><th>Error Name</th><th>Error Details</th><th>Error Processing</th><th>Corrective Action</th></tr>
<tr><td rowspan="4">n08*</td><td rowspan="4">Assist point setting error (circular interpolation with assist-point designation)</td><td>1) In absolute mode positioning control, an address outside the setting range is designated.
<table><tr><th>Units</th><th colspan="2">Address Setting Range</th></tr><tr><td>degree</td><td>0 to 35999999</td><td>x10^{-5} degree/min</td></tr></table></td><td rowspan="4">Positioning control does not start.</td><td>1) Correct the set address for degree units.
Setting range:
0 to 35999999</td></tr>
<tr><td>2) In incremental mode positioning control, axis travel distance setting is "-2147483648" (H80000000).</td><td>2) Correct the set axis travel distance.
Setting range:
0 to ± $(2^{31} - 1)$</td></tr>
<tr><td>3) Designated assist point is either the start point or end point.</td><td>3) Designate the assist point so that it does not lie on the start point or end point.</td></tr>
<tr><td>4) Designated assist point is on the line connecting the start and end points.</td><td>4) Designate the assist point so that it does not lie on the line connecting the start and end points.</td></tr>
<tr><td rowspan="4">n09*</td><td rowspan="4">Radius setting error (circular interpolation with radius designation)</td><td>1) In absolute positioning control, an address outside the setting range is designated.
<table><tr><th>Units</th><th colspan="2">Address Setting Range</th></tr><tr><td>degree</td><td>0 to 35999999</td><td>x10^{-5} degree/min</td></tr></table></td><td rowspan="4">Positioning control does not start.</td><td>1) Correct the set address for degree units.
Setting range:
0 to 35999999</td></tr>
<tr><td>2) In incremental positioning control, axis travel distance setting is "-2147483648" (H80000000).</td><td>2) Correct the set axis travel distance.
Setting range:
0 to ± $(2^{31} - 1)$</td></tr>
<tr><td>3) Start point = end point</td><td>3) Designate the start point and end point so that start point ≠ end point.</td></tr>
<tr><td>4) The distance between the start point and end point is greater than the diameter.</td><td>4) Correct the designated data so that the following is satisfied.
$L/2R \leq 1$
L: Distance between start point and end point
R: Radius</td></tr>
</table>

Table 2.2 Servo Program Setting Error List (Continued)

<table>
<tr><th>Error Code in D9190</th><th>Error Name</th><th>Error Details</th><th>Error Processing</th><th>Corrective Action</th></tr>
<tr><td rowspan="4">n010*</td><td rowspan="4">Center point setting error</td><td>1) In absolute mode positioning control, an address outside the setting range is designated.
<table><tr><th>Units</th><th colspan="2">Address Setting Range</th></tr><tr><td>degree</td><td>0 to 35999999</td><td>x10^{-5} degree/min</td></tr></table></td><td rowspan="4">Positioning control does not start.</td><td>1) Correct the set address for degree units.
Setting range:
0 to 35999999</td></tr>
<tr><td>2) In incremental mode positioning control, axis travel distance setting is "-2147483648" (H80000000).</td><td>2) Correct the set axis travel distance.
Setting range:
0 to $\pm(2^{31}-1)$</td></tr>
<tr><td>3) Designated assist point is either the start point or end point.</td><td>3) Designate the assist point so that it does not lie on the start point or end point.</td></tr>
<tr><td>4) Designated assist point is on the line connecting the start and end points.</td><td>4) Designate the assist point so that it does not lie on the line connecting the start and end points.</td></tr>
<tr><td>11</td><td>Interpolation control unit setting error</td><td>Designated interpolation setting error is not between 0 and 3.</td><td>Positioning control is executed assuming a setting of "3" (default).</td><td>Correct the interpolation control unit setting.
Setting range:
0 to 3</td></tr>
<tr><td>12</td><td>Velocity limit setting error</td><td>Designated velocity limit is outside the allowable range.</td><td>Positioning control is executed assuming a setting of "200000 PLS/sec" (default).</td><td>Correct the velocity limit setting.</td></tr>
<tr><td>13</td><td>Acceleration time setting error</td><td>Designated acceleration time is "0".</td><td rowspan="3">Positioning control is executed assuming a setting of "1000" (default).</td><td>Correct the acceleration time setting.
Setting range:
1 to 65535</td></tr>
<tr><td>14</td><td>Deceleration time setting error</td><td>Designated deceleration time is "0".</td><td>Correct the deceleration time setting.
Setting range:
1 to 65535</td></tr>
<tr><td>15</td><td>Immediate stop deceleration time setting error</td><td>Designated immediate stop deceleration time is "0".</td><td>Correct the immediate stop deceleration time.
Setting range:
1 to 65535</td></tr>
<tr><td>16</td><td>Torque limit setting error</td><td>Designated torque limit is outside the allowable setting range.
Setting range: 1 to 500</td><td>Positioning control is executed assuming a setting of "300%" (default).</td><td>Correct the torque limit setting.
Setting range:
1 to 500</td></tr>
</table>

Table 2.2 Servo Program Setting Error List (Continued)

<table>
<tr><th>Error Code in D9190</th><th>Error Name</th><th>Error Details</th><th>Error Processing</th><th>Corrective Action</th></tr>
<tr><td>17</td><td>Allowable circular interpolation error setting error</td><td>Designated allowable circular interpolation setting error is outside the allowable setting range.
Setting range:
<table><tr><th>Units</th><th colspan="2">Address Setting Range</th></tr><tr><td>mm</td><td rowspan="4">1 to 100000</td><td>$\times 10^{-1}$ μm</td></tr><tr><td>inch</td><td>$\times 10^{-5}$ inch</td></tr><tr><td>degree</td><td>$\times 10^{-5}$ degree</td></tr><tr><td>PULSE</td><td>PLS</td></tr></table></td><td>Positioning control is executed assuming a setting of "100 PLS" (default).</td><td>Correct the allowable circular interpolation setting error.</td></tr>
<tr><td>18</td><td>Repeat count setting error</td><td>Designated repeat count is outside the allowable setting range:
Setting range: 1 to 32767</td><td>Positioning control is executed assuming a repeat count of "1".</td><td>Correct the repeat count setting.
Setting range:
1 to 32767</td></tr>
<tr><td rowspan="3">19</td><td rowspan="3">START instruction setting error</td><td>1) The servo program designated with the START instruction is not found.</td><td rowspan="3">Positioning control does not start.</td><td>1) Change the servo program number designated with the START instruction with a peripheral device.</td></tr>
<tr><td rowspan="2">2) The START instruction is present in the designated servo program.</td><td>2) Create a servo program designated with the START instruction.</td></tr>
<tr><td>3) Eliminate the servo program containing the START instruction.</td></tr>
</table>

2.2 Minor Errors

A minor error is an error that occurs in the sequence or servo program and is represented by error codes of 1 to 999.

A minor error occurs in the following cases:

* Setting data error
* Error at the start of positioning control
* Error during positioning control
* Error occurring when changing the data

(1) Setting data error (1 to 99)

Table 2.3 shows the error code, error cause, error processing, and corrective action.

Table 2.3 Setting Data Error List (1 to 99)

Error Code	Error Data	Checked When	Error Causes	Error Processing	Corrective Action
10	Servo parameter	A73CPU is powered up	For the axis to which the general-purpose servo is connected, either MR-SB (0) or absolute MR-SB (1) is selected.	The general-purpose servo does not start.	Correct the setting for the servo system setting parameters with a peripheral device.
22	Zero return data	Zero return operation starts	Designated zero return velocity is outside the allowable setting range. Setting range: 1 to the velocity limit	Zero return does not start.	Correct the zero return velocity setting with a peripheral device. Setting range: 1 to the velocity limit
23			Designated creep feed velocity is outside the allowable setting range. Setting range: 1 to the zero return velocity		Correct the creep feed velocity setting with a peripheral device. Setting range: 1 to the zero return velocity
40 *	Parameter block	Positioning control starts Zero return starts Jog operation starts	Interpolation control units designated in the parameter block and those set with the fixed parameters do not agree with each other.		Change either of the set control units.

POINT

If the interpolation control units in the parameter block and those set using the fixed parameters do not agree with each other, there are cases when the error code is not stored for some combinations of the units.

For details, see Section 7.1.4.

(2) Error at the start of positioning control (100 to 199)

Table 2.4 shows the error code, error cause, error processing, and corrective action.

*: For interpolation control, the error code is stored in all of the error code storing areas of the axis to be controlled in the designated interpolation mode.

Table 2.4 Positioning Control Start-up Error List (100 to 199)

Error Code	Control Mode									Error Causes	Error Processing	Corrective Action
	*1	*2	*3	*4	*5	*6	*7	*8	*9			
100	o	o	o	o	o	o	o	o	o	The PC ready flag (M2000) or PCPCU ready flag (M9074) is off.	Positioning control does not start.	Set the A73CPU to the RUN mode. Set the PC ready flag (M2000).
101	o	o	o	o	o	o	o	o	o	The start receive flag (M2001 to M2008) of the corresponding axis is already on.		Take an interlock so that the axis already controlled is not started again. Use the "start receive flag off" as an interlock condition
103	o	o	o	o	o	o	o	o	o	The stop instruction (Yn0) of the corresponding axis is on.		Turn off the stop instruction (Yn0) of the corresponding axis before starting positioning control.
104	o	o	o	o	o	o	o	o	o	The immediate stop instruction (Yn1) of the corresponding axis is on.		Turn off the immediate stop instruction (Yn1) of the corresponding axis before starting positioning control.
106*	o	o			o	o				Positioning beyond the stroke limit is called.		Positioning end (target) point must be within the set stroke limit.
107	o					o				An address, which does not create an arc, is designated in circular interpolation with assist point designation. (Relationship among the start, assist, and end point addresses)		Correct the servo program address.
108*	o					o				An address, which does not create an arc, is designated in circular interpolation with radius designation. (Relationship among the start, radius, and end point addresses)		

*1: Positioning
*2: Incremental Feed
*3: Velocity
*4: Velocity Position
*5: Velocity Change
*6: Constant Velocity
*7: Jog
*8: Manual Pulse Generator
*9: Zero Return

Table 2.4 Positioning Control Start-up Error List (100 to 199) (Continued)

Error Code	Control Mode									Error Causes	Error Processing	Corrective Action
	*1	*2	*3	*4	*5	*6	*7	*8	*9			
109	o					o				An address, which does not create an arc, is designated in circular interpolation with center point designation. (Relationship among the start, center, and end point addresses)	Positioning control does not start.	Correct the servo program address.
110*	o					o				In circular interpolation, the difference between the end point address and the theoretical end point address is greater than the allowable circular interpolation error range.		
111				o						The V/P control is started again while the current V/P control is not stopped before completion.		Do not restart the V/P control unless the control has been stopped before completion.
115									o	When the near-zero point dog zero return is started, the zero point return signal (XnA) is already on.		Zero return operation cannot be started continuously. Return the axis in the jog or positioning mode control to a point before the near-zero point dog triggered signal turning on point. After that, execute zero return.
116							o			The set jog velocity is "0".		Correct the jog velocity setting.
										The set jog velocity is greater than the jog velocity limit.	Jog velocity is clamped at the limit velocity.	
117							o			Restart is designated in both forward and reverse rotation for the same axis.	Positioning control starts in the forward direction.	Correct the setting.
118					o					The velocity change point exceeds the end point address.	Positioning control does not start.	Set the velocity change point within a range of the preceding velocity change point or start point and the end point.
										An address causing positioning control to be executed in the opposite direction is set.		
120									o	Count zero return ZCT not set		

*1: Positioning
*2: Incremental Feed
*3: Velocity
*4: Velocity Position
*5: Velocity Change
*6: Constant Velocity
*7: Jog
*8: Manual Pulse Generator
*9: Zero Return

IB (NA) 66233-A

Table 2.4 Positioning Control Start-up Error List (100 to 199) (Continued)

Error Code	Control Mode *1	*2	*3	*4	*5	*6	*7	*8	*9	Error Causes	Error Processing	Corrective Action
121									o	Zero return absolute value storage error EEROM error		The service life of the A73CPU internal memory (EEPROM) has expired. Replace the memory.

*1: Positioning
*2: Incremental Feed
*3: Velocity
*4: Velocity Position
*5: Velocity Change
*6: Constant Velocity
*7: Jog
*8: Manual Pulse Generator
*9: Zero Return

(3) Position control errors (200 to 299)

Table 2.5 shows the error code, error cause, error processing, and corrective action.

Table 2.5 Positioning Control Error List (200 to 299)

Error Code	Control Mode *1	*2	*3	*4	*5	*6	*7	*8	*9	Error Causes	Error Processing	Corrective Action
200	o	o	o	o	o	o	o	o		The PC ready flag (M2000) is reset while positioning is started in response to the start request given from a sequence program.	The axis decelerates and stops.	Reset the PC ready flag (M2000) after all axes have stopped.
201									o	The PC ready flag (M2000) is reset during zero return operation.		Execute zero return again after setting the PC ready flag (M2000) or turning off the stop instruction (Yn0) or the immediate stop instruction (Yn1). (In the case of near-zero point dog zero return, return the axis to a point before the near-zero point dog-triggered signal turning on point in either the positioning or jog operation.)
202									o	The stop instruction (Yn0) is turned on during zero return		
203									o	The immediate stop instruction (Yn1) is turned on during zero return	Immediate stop	
204	o	o	o	o	o	o	o	o	o	The PC ready flag (M2000) is set again while the axis is decelerating in response to resetting of the PC ready flag.	No processing	Set the PC ready flag (M2000) again after all the axes have stopped. The PC ready flag, which is set again during deceleration, is ignored and no processing occurs.

*1: Positioning
*2: Incremental Feed
*3: Velocity
*4: Velocity Position
*5: Velocity Change
*6: Constant Velocity
*7: Jog
*8: Manual Pulse Generator
*9: Zero Return

Table 2.5 Positioning Control Error List (200 to 299) (Continued)

Error Code	Control Mode *1	*2	*3	*4	*5	*6	*7	*8	*9	Error Causes	Error Processing	Corrective Action
206									o	During zero return operation, emergency stop is attempted by pressing the [BREAK] or [STOP] key of a peripheral device.	Immediate stop	In the case of near-zero point dog zero return, return the axis to a point before the near-zero point dog-triggered signal turning on point in the positioning or jog operation. With the count zero return mode, if the near-zero point dog signal is off, return the axis to a point before the near-zero point dog-triggered signal turning on point in either the positioning or jog operation. With the count zero return mode, if the near-zero point dog-triggered signal is on, execute zero return again.
207	o				o	o				During positioning, the present value exceeds the stroke limit. In the circular interpolation mode, only the axis which moves beyond the limit is stored. In the linear interpolation mode, all the axes involved with the interpolation control are stored.	The axes decelerate and stop	Correct the stroke limit settings or the set axis travel distance so that positioning control is executed within the stroke limit range.
208	o				o	o				During circular interpolation control, the present value of the mating axis exceeds the stroke limit. (For detecting an error of other axis)		
209				o					o	During V/P control, overrun occurs because the set axis travel distance is smaller than the required deceleration distance when the CHANGE signal is input.		Correct the velocity setting so that overrun does not occur. Correct the set axis travel distance so that overrun does not occur.

*1: Positioning
*2: Incremental Feed
*3: Velocity
*4: Velocity Position
*5: Velocity Change
*6: Constant Velocity
*7: Jog
*8: Manual Pulse Generator
*9: Zero Return

Table 2.5 Positioning Control Error List (200 to 299) (Continued)

Error Code	Control Mode									Error Causes	Error Processing	Corrective Action
	*1	*2	*3	*4	*5	*6	*7	*8	*9			
210				o						During V/P control, the set axis travel distance exceeds the stroke limit when the CHANGE signal is input.	The axis decelerates and stops	Correct the stroke limit setting or axis travel distance setting so that positioning control is executed within the stroke limit.
211						o				During positioning control, the distance from the point where the final positioning address is detected to the final positioning address is not sufficient for deceleration at the present output velocity, resulting in overrun.	The axis decelerates and stops	1) Correct the velocity so that overrun does not occur. 2) Correct the set axis travel distance so that overrun does not occur.
213								o		Output velocity during MPG operation exceeds 1 MPPS.	Velocity is clamped at 1 MPPS.	Operate the MPG so that output velocity is less than 1 MPPS. Output velocity = (No. of input pulses/msec) x (Pulse multiplication ratio)
214								o		An attempt is made to control an axis that has already been moved with the MPG by setting the MPG operation enable flag for that axis.	Pulse input from the manual pulse is ignored until the current axis travel stops.	Try MPG operation after the travel of the required axis has stopped.
215					o					The velocity change point address exceeds the end point address. Change the velocity change point address so that the designated velocity change address is within the previous velocity change address and the end point address.	The axis decelerates and stops.	An address is set causing positioning control to be executed in the opposite direction.

*1: Positioning
*2: Incremental Feed
*3: Velocity
*4: Velocity Position
*5: Velocity Change
*6: Constant Velocity
*7: Jog
*8: Manual Pulse Generator
*9: Zero Return

(4) Present V/P change error (300 to 399)

Table 2.6 shows the error code, error cause, error processing, and corrective action.

For interpolation control, the error code is stored in all of the error code storage areas of the axis to be controlled in the designated interpolation mode.

Table 2.6 Present V/P Change Error List (300 to 399)

Error Code	Control Mode *1	*2	*3	*4	*5	*6	*7	*8	*9	Error Causes	Error Processing	Corrective Action
300	o	o	o	o	o	o	o	o	o	An attempt is made to change the command position data for an axis which is moving. An attempt is made to change the command position data for an axis which has not been booted.	Command position data is not changed.	Use the following devices as an interlock so that the command position data of the axis currently moving is not changed. 1) Off state of the start receive flag (M2001 to M2008) of the corresponding axis 2) On state of the servo start signal XnF
301									o	An attempt is made to change the velocity of the axis being returned to the zero point.	Velocity is not changed.	Do not change the velocity during zero return.
302	o									An attempt is made to change the velocity of the axis controlled in the circular interpolation mode.		Do not change the velocity during circular interpolation.
303	o	o		o	o	o				An attempt is made to change the velocity after automatic deceleration has started.		Do not change the velocity after automatic deceleration has started.
304							o			An attempt is made to change the velocity while the axis is decelerating with the jog operation start signal (Yn2/Yn3) off.		Do not change the velocity during deceleration called by the jog operation start signal (Yn2/Yn3) being turned off.
305	o	o	o	o	o	o	o			The velocity after velocity change is outside the allowable range. Allowable range: 1 to the velocity limit	0: Velocity change processing is ignored. Not 0: Velocity is clamped at the velocity limit.	Correct the velocity after the change so that it will be within the range of 1 to velocity limit.

*1: Positioning
*2: Incremental Feed
*3: Velocity
*4: Velocity Position
*5: Velocity Change
*6: Constant Velocity
*7: Jog
*8: Manual Pulse Generator
*9: Zero Return

IB (NA) 66233-A

2.3 Major Error

A major error is an error caused by an external signal or a control instruction given by the SCPU. It is represented by error codes 1000 to 1999.

A major error occurs in the following cases:

* Error at the start of positioning control
* Error during positioning control
* Absolute system error

(1) Start-up error (1000 to 1099)

Table 2.7 shows the error code, error cause, error processing, and corrective action.

Table 2.7 Start-up Error List (1000 to 1099)

Error Code	Control Mode									Error Causes	Error Processing	Corrective Action
	*1	*2	*3	*4	*5	*6	*7	*8	*9			
1000	o	o	o	o	o	o	o	o	o	The external STOP signal of the corresponding axis is turned on.	Positioning control does not start.	Turn off the STOP signal.
1001	o	o	o	o	o	o	o	o	o	When positioning is started in the forward direction (addresses increasing), the external FLS (upper limit LS) signal is off.		Move the axis in the reverse direction in the jog mode until the axis enters the axis movable range set with the limit switch.
1002	o	o	o	o	o	o	o	o	o	When positioning is started in the reverse direction (addresses decreasing), the external RLS (lower limit LS) signal is off.		Move the axis in the forward direction in the jog mode until the axis enters the axis movable range set with the limit switch.
1003									o	When near-zero point dog zero return is started, the external DOG (near-zero point dog) signal is turned on.		Return the axis to a point before the near-zero point dog-triggered signal turning on point in the jog mode and then execute zero return.
1004	o	o	o	o	o	o	o	o	o	The servo of the corresponding axis is not in the ready state (XnF: OFF). 1) Servo amplifier power is off. 2) Initial processing with the servo amplifier power turned on 3) Servo amplifier not installed 4) Servo error existing 5) Cable		Wait until the servo gets ready (XnF: OFF).

*1: Positioning
*2: Incremental Feed
*3: Velocity
*4: Velocity Position
*5: Velocity Change
*6: Constant Velocity
*7: Jog
*8: Manual Pulse Generator
*9: Zero Return

Table 2.7 Start-up Error List (1000 to 1099) (Continued)

Error Code	Control Mode *1	*2	*3	*4	*5	*6	*7	*8	*9	Error Causes	Error Processing	Corrective Action
1005	o	o	o	o	o	o	o	o	o	The servo error detection signal (Xn8) of the corresponding axis is turned on.	Positioning control does not start.	After removing the cause of servo error, reset the Xn8 with the servo error reset signal (Yn8). After that, start positioning control.

*1: Positioning
*2: Incremental Feed
*3: Velocity
*4: Velocity Position
*5: Velocity Change
*6: Constant Velocity
*7: Jog
*8: Manual Pulse Generator
*9: Zero Return

(2) Position control errors (1100 to 1199)

Table 2.8 shows the error code, error cause, error processing, and corrective action.

Table 2.8 Positioning Control Error List (1100 to 1199)

Error Code	Control Mode *1	*2	*3	*4	*5	*6	*7	*8	*9	Error Causes	Error Processing	Corrective Action
1101	o	o	o	o	o	o	o	o	o	The external FLS (upper limit LS) signal is turned off while the axis is moving in the forward direction (addresses increasing).	The axis decelerates and stops according to the setting for "stop processing when STOP signal is input" of the parameter block.	Move the axis in the reverse direction in the jog mode until the axis enters the axis movable range set with the limit switch.
1102	o	o	o	o	o	o	o	o	o	The external RLS (lower limit LS) signal is turned off while the axis is moving in the reverse direction (addresses decreasing).		Move the axis in the forward direction in the jog mode until the axis enters the axis movable range set with the limit switch.
1103									o	The external STOP signal is turned on during zero return operation.		Return the axis to a point before the near-zero point dog-triggered signal turning on point in the jog mode and then execute zero return.
1104	o	o	o	o	o	o	o	o	o	The servo error detection signal (Xn8) is turned on while an axis is moving.	The axis stops at the spot without decelerating.	Take corrective action for servo error. The axis can be restarted.
1105	o	o	o	o	o	o	o	o	o	The servo amplifier power is turned off while an axis is moving (servo not installed, cable failure, etc.).	Turn X00F OFF.	Turn on the servo power. Check the cable connecting to the servo amplifier.

*1: Positioning
*2: Incremental Feed
*3: Velocity
*4: Velocity Position
*5: Velocity Change
*6: Constant Velocity
*7: Jog
*8: Manual Pulse Generator
*9: Zero Return

(3) Absolute system error (1200 to 1299)

Table 2.9 shows the error code, error cause, error processing, and corrective action.

Table 2.8 Absolute System Error List (1200 to 1299)

Error Code	Control Mode									Error Causes	Error Processing	Corrective Action
	*1	*2	*3	*4	*5	*6	*7	*8	*9			
1201										Sum check error occurs with the backup data when the power is turned on.	Positioning control does not start.	The service life of the A73CPU internal memory (EEPROM) has expired. Replace the memory.

*1: Positioning
*2: Incremental Feed
*3: Velocity
*4: Velocity Position
*5: Velocity Change
*6: Constant Velocity
*7: Jog
*8: Manual Pulse Generator
*9: Zero Return

2.4 Servo Errors

An error detected by the servo system is a servo error. It is assigned an error number in the range of 2000 to 3999.

Servo errors are classified into two types depending on the servo amplifier: MR-SB or general-purpose servo amplifier.

(1) When an MR-SB is used (2000 to 2999):

Table 2.9 MR-SB Servo Error List (2000 to 2999)

Error Code	Error Causes		Error Is Checked when	Processing	Corrective Action
	Name	Contents			
2010	Low voltage 1	AC power supply voltage (200/220 VAC) is lowered below 160 ± 5 VDC.	Always	Immediate stop	
2012	Internal memory error	EPROM check sum error or SRAM or 2-port RAM check error EPROM, SRAM, or 2-port RAM error	When servo power is turned on Leading edge of the PC ready flag (M2000)		After resetting the servo error (Yn8), set the PC ready flag (M2000). Reset the A73CPU. Reset the servo amplifier.
2013	External clock error	Data processing for the position command given by the A73CPU is not completed within the specified period. Timing signal error (BCLK, SCLK) from the A73CPU.	Always	Immediate stop	Hardware fault of the A73CPU or servo. Reset the A73CPU and check if the same error occurs.
2014	WD	Timing error of 2-port RAM signals of the A73CPU.			
2015	2-port memory error	2-port RAM check error or parity error in initial parameters. Faulty cable connecting to A70SF or 2-port RAM error	When servo power is turned on Leading edge of the PC ready flag (M2000)		
2016	Magnetic pole error	In the initial magnetic pole detection, the magnetic pole position cannot be detected correctly (U, V, W phase error). Faulty cable or encoder.			
2017	PC board error	AD converter output during initialization is incorrect; higher than ±0.5 V. Faulty AD converter on the PCB			
2020	No-signal ME	Signal error of the encoder connected to the RF01 card (U, V, W, A, B, Z error). Faulty cable or encoder	Always	Immediate stop	Encoder hardware fault (connected to the servo). Replace the servo amplifier.
2021	No-signal AE	Signal error of the encoder connected to the RF31 or RF33 card (A, B, Z error) Faulty cable or encoder			
2022	No-signal IX	Signal error of the resolver connected to the RF32 or RF33 card. Faulty cable or resolver or an excitation signal error.			Resolver hardware fault (connected to the servo). Replace the servo.

IB (NA) 66233-A

Table 2.9 MR-SB Servo Error List (2000 to 2999) (Continued)

Error Code	Error Causes: Name	Error Causes: Contents	Error Is Checked when	Processing	Corrective Action
2025	Battery alarm	Backup battery voltage for the RF32 or RF33 card absolute position detection circuit is low. Absolute values might have been lost; execute zero return again.	When servo power is turned on Leading edge of the PC ready flag (M2000)	Zero return request (Xn9) is turned on.	This check is only available in the absolute system. Replace the servo battery. After resetting the servo error (Yn8), establish the absolute coordinate system by executing zero return or command position data change operation.
2030	Excessive regeneration	Regeneration control power transistor is turned on and off too frequently. (Regenerative resistor is overheaded.) Lower acceleration/deceleration frequency or positioning velocity.	Always	Immediate stop	After resetting the servo error (Yn8), set the PC ready flag (M2000). Reset the A73CPU.
2031	Over-speed	Excessive motor rpm: Max. 2400 rpm ..... HA40, 80, 100, 200, and 300 Max. 3600 rpm ..... HA43, 83 Command velocity is too high or overshoot occurs during acceleration.			
2032	Over-current	Excessive current flows from the DC bus line (+ side). Motor wiring grounded or shorted.			
2033	Over-voltage	Voltage applied to the DC bus line is higher than 400 V. Acceleration/deceleration is repeated too frequently, exceeding regeneration performance. Connection error at the regeneration circuit terminals.			
2034	Communication error	Parity error in the data sent from the A73CPU. Faulty cable or noise exists.			
2035	Data error	Excessive variation rate of position data given by the A73CPU. Too high velocity, faulty cable, or noise exists.			
2036	Transmission error	Communication with the A73CPU is disabled. Faulty cable, noise exists, or an MCP processing error.			
2037	Parameter error	The parameters transmitted during initialization processing were incorrect. Check the parameters.	When servo power is turned on Leading edge of the PC ready flag (M2000)		

Table 2.9 MR-SB Servo Error List (2000 to 2999) (Continued)

Error Code	Error Causes: Name	Error Causes: Contents	Error Is Checked when	Processing	Corrective Action
2045	Fin overheat	The thermal protector in the amplifier power circuit has tripped. The servo is operated exceeding the allowable continuous output current.	Always	Immediate stop	Reset the servo error (Yn8). Lower the load.
2046	Motor overheat	The thermal protector in the motor has tripped. The motor is operated exceeding the allowable continuous output current.			
2050	Overload 1	Motor current, converted into the stall rating, which exceeds the value set for parameter OLL (overload detection level) for a period longer than the value set for parameter OLT (overload time constant). Load inertia or friction is too high or hunting due to parameter setting error.			Reset the servo error (Yn8).
2051	Overload 2	Current command exceeding 95% of the current limit is present for more than 0.5 sec. Collision of the machine or load inertia is too high.			
2052	Excessive error	An error between the actual position data and the command position data exceeds parameter setting for OD1 (error limit during servo on) or OD2 (error limit during servo off). Due to excessively high inertia, acceleration is not as designated and overshoot or hunting occurs.			
2055	External emergency stop	Terminals B and R, usually shorted, are opened.			
2100	Initial communication warning	Serial signal error when the A73CPU power supply is turned on or the absolute counter data is not the same when the present position data is read several times.	Always	Continued	Replace the cable or card. Replace the position encoder/resolver.
2101	Serial communication warning	Serial signal communications from the absolute position encoder are abnormal.			
2102	Battery warning	The voltage of the battery mounted on the RF371 card is lower than 3.2 ± 0.2 V.			Replace the battery.
2103	Battery cable open warning	The voltage supplied to the absolute position encoder is lower than 2.8 ± 0.2 V.			
2104	Position offset warning	The relationship between the feedback position data and Z-phase of the encoder is incorrect.			

Table 2.9 MR-SB Servo Error List (2000 to 2999) (Continued)

Error Code	Error Causes		Error Is Checked when	Processing	Corrective Action
	Name	Contents			
2144	Parameter error	Illegal parameters are set. The illegal parameter and the following parameters are ignored. Check the parameters. (Servo is not turned off.)	When servo power is turned on Leading edge of the PC ready flag (M2000)		After resetting the servo error (Yn8), set the PC ready flag (M2000). Reset the A73CPU.
2145	Absolute position error	The value "1X" of absolute position detection system is incorrect. Absolute position detection is not executed correctly. (Servo is not turned off.)	When servo power is turned on Leading edge of the PC ready flag (M2000)		This check is available only in the absolute system. After resetting the servo error (Yn8), establish the absolute coordinate system by executing the zero return operation.
2147	PC emergency stop	The emergency stop signal is output from the A73CPU.			

(2) When a general-purpose servo is used:

Table 2.10 General-purpose Servo Error List (3000 to 3999)

Error Code	Error Causes		Error Is Checked when	Processing	Corrective Action
	Name	Contents			
3000	A70AF setting error	The same axis number is set for multiple axes.	When the power is turned on.	The corresponding axis does not start.	Check the A70AF axis setting.
3002	Excessive error	Excessive error of the general-purpose servo.	Always	Immediate stop	Check the command velocity, set rotation direction, and feedback loop.
3003	WDT error	A70AF WDT error	Always	Immediate stop	A70AF hardware error
3004	Zero return error	The servo ready input signal is turned off during zero return.	During zero return	Immediate stop	Check the servo ready signal.

IB (NA) 66233-A

APPENDIX 3 SPECIAL RELAYS AND SPECIAL REGISTERS

3.1 Special Relays (SP-M)

Table 3.1 Special Relays List

Number	Name	Description	Details
M9000*1	Fuse blown	OFF: Normal ON: Presence of fuse blow module	Turned on when there is one or more output modules of which fuse has been blown. Remains on if normal status is restored.
M9002*1	I/O module verify error	OFF: Normal ON: Presence of error	Turned on if the status of I/O module is different from entered status when power is turned on. Remains on if normal status is restored.
M9005*1	AC DOWN detection	OFF: AC is good ON: AC is down	Turned on if power failure of within 20 ms occurs. Reset when POWER switch is moved from OFF to ON position.
M9006	Battery low	OFF: Normal ON: Battery low	Turned on when battery voltage reduces to less than specified. Turned off when battery voltage becomes normal.
M9007*1	Battery low latch	OFF: Normal ON: Battery low	Turned on when battery voltage reduces to less than specified. Remains on if battery voltage becomes normal.
M9008*1	Self-diagnostic error	OFF: Absence of error ON: Presence of error	Turned on when error is found as a result of self-diagnosis.
M9009	Annunciator detection	OFF: Absence of detection ON: Presence of detection	Turned on when OUT F or SET F instruction is executed. Turned off when the data in D9124 is cleared to "0".
M9011*1	Operation error flag	OFF: Absence of error ON: Presence of error	Turned on when operation error occurs during execution of application instruction. Remains on if normal status is restored.
M9012	Carry flag	OFF: Carry off ON: Carry on	Carry flag used in application instruction.
M9016	Data memory clear flag	OFF: No processing ON: Output clear	Clears all data memory (except special relays and special registers) including latched range in remote mode from computer, etc. when M9016 is ON.
M9017	Data memory clear flag	OFF: No processing ON: Output clear	Clears all unlatched data memory (except special relays and special registers) in remote run mode from computer, etc. when M9017 is ON.
M9020	User timing clock No.0	n2 scan n2 scan n1 scan	Relay which repeats on/off at intervals of predetermined scan. When power is turned on or reset is performed, the clock starts with off. Set the intervals of on/off [DUTY] instruction. [DUTY n1 n2 M9020]
M9021	User timing clock No.1		
M9022	User timing clock No.2		
M9023	User timing clock No.3		
M9024	User timing clock No.4		
M9030	0.1 second clock	0.05 second 0.05 second	0.1 second, 0.2 second, 1 second, 2 seconds, and 1 minute clocks are generated. Not turned on and off per scan but turned on and off even during scan if corresponding time has elapsed. Starts when power is turned on or reset is performed.
M9031	0.2 second clock	0.1 second 0.1 second	
M9032	1 second clock	0.5 second 0.5 second	
M9033	2 second clock	1 second 1 second	
M9034	1 minute clock	30 seconds 30 seconds	

Table 3.1 Special Relays List (Continued)

Number	Name	Description	Details
M9036	Normally ON	ON OFF	Used as dummy contacts of initialization and application instruction in sequence program. M9036 and M9037 are switched on/off independently of the CPU RUN/STOP switch position. M9038 and M9039 are switched on/off in accordance with the RUN/STOP switch position, i.e. switched off when the switch is set to STOP. When the switch is set to other than STOP, M9038 is only switched on during 1 scan and M9039 is only switched off during 1 scan.
M9037	Normally OFF	ON OFF	
M9038	On only for 1 scan after run	ON OFF 1 scan	
M9039	RUN flag (Off only for 1 scan after run)	ON OFF 1 scan	
M9040	PAUSE enable coil	OFF: PAUSE disabled ON: PAUSE enabled	When RUN key switch is at PAUSE position or remote pause contact has turned on and if M9040 is on, PAUSE status is set and M9041 is turned on.
M9041	PAUSE status contact	OFF: During pause ON: Not during pause	
M9042	Stop status contact	OFF: During stop ON: Not during pause	Set ON when the RUN/STOP switch is set in the STOP position.
M9043	Sampling trace completion	OFF: During sampling trace ON: Sampling trace completion	Turned on upon completion of sampling trace performed the number of times preset by parameter after [STRA] instruction is executed. Reset when [STRAR] instruction is executed.
M9046	Sampling trace	OFF: Except during trace ON: During trace	Set ON while sampling trace is executed.
M9047	Sampling trace preparation	OFF: Sampling trace stop ON: Sampling trace start	Sampling trace is not executed until M9047 is turned on. By turning off M9047, sampling trace is stopped.
M9049	Changing the number of output characters	OFF: Output to NULL character ON: 16 characters are out.	When M9049 is OFF, characters are output to the NULL (00H) code. When M9049 is ON, 16 characters are output in the ASCII code.
M9051	CHG instruction execution disable	OFF: Enable ON: Disable	Set M9051 OFF to disable the execution of the CHG instruction. Turn this flag ON to request program transmission; it is automatically turned OFF at the completion of program transmission.
M9052*2	Switching SEG instruction	OFF: 7SEG display ON: I/O partial refresh	The SEG instruction is executed as the I/O partial refresh instruction when M9052 is ON. The SEG instruction is executed as the 7SEG indication instruction when M9052 is OFF.
M9053*2	Switching EI/DI instruction	OFF: Sequence interrupt control ON: Link interrupt control	Set M9053 ON to execute the link refresh enable/disable (EI, DI) instruction.
M9054	STEP RUN flag	OFF: Not during step run ON: During step run	M9054 is set ON when the RUN/STOP switch is in the STEP-RUN position.
M9055	Status latch completion flag	OFF: Uncompleted ON: Completed	Turned on when status latch is completed. Turned off by reset instruction.
M9056	Main program P, I set request	OFF: During P, I set request ON: Except during P, I set request	Switch on upon completion of the transfer to another program (e.g. subprogram during RUN of the main program). Automatically switched off when P, I setting is complete.
M9057	Sub-program P, I set request	OFF: During P, I set request ON: Except during P, I set request	
M9084*2	Error check	OFF: Error check executed ON: Error check not executed	This sets whether the following error check is executed when the END instruction is processed (to reduce END instruction processing time). * Fuse blown check * I/O module verify check * Battery check

Table 3.1 Special Relays List (Continued)

Number	Name	Description	Details
M9094*2*3	I/O replace flag	OFF: Module not replaced ON: Module replaced	Turn the M9094 ON after setting the head I/O number of the I/O module to be replaced to D9094, and the I/O module can be replaced during online operation. Only one module can be replaced in one setting. To change the I/O module during run, set the M9094 ON using the program or by the test mode of a peripheral device, or by the test mode of a peripheral device. Do not change the mode setting of the RUN/STOP until the I/O module is replaced.

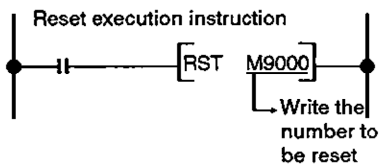
POINT

(1) Contents are all turned off when any of the following operation is executed: power switching off, latch clear, and reset. Contents are retained when the RUN key switch is set STOP.

(2) Among the special relays,, those marked with (*1) remain ON when normal state is restored. To turn OFF these relays, follow the steps below.

1) To turn off using a user program:

Insert the circuit shown right in the program; turn on the reset execution instruction contact to clear the special Ms.



2) To turn off using a peripheral device:

Reset the special relay forcibly using the test function of the peripheral device.

For operation procedure, refer to the Manual of the peripheral device to be used.

3) To turn off using the RESET switch:

Place the RESET switch in the RESET position.

(3) Those marked with (*2) are turned ON/OFF by the sequence program.

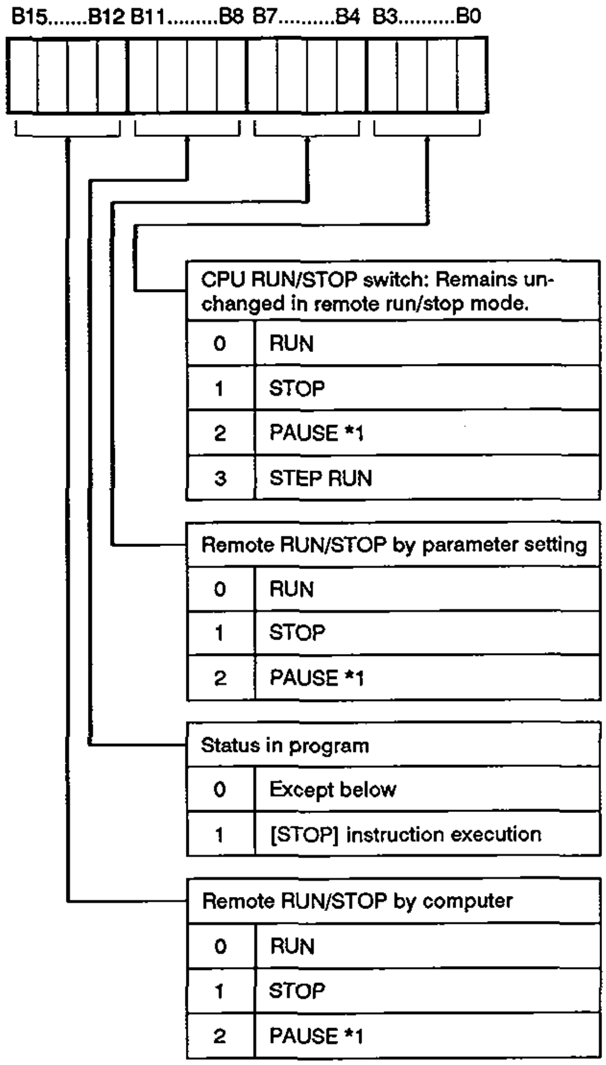
(4) Those marked with (*3) are turned ON/OFF by the test mode of the peripheral device.

3.2 Special Registers D

Special Register List

Number	Name	Stored Date	Explanation
D9000	Fuse blown	Fuse blow module number	When fuse flow modules are detected, the lowest number of detected units is stored in hexadecimal. (Example: When fuses of Y50 to 6F output modules have blown, "50" is stored in hexadecimal) The module number monitored by the peripheral is hexadecimal. (Cleared when all contents of D9100 to D9107 are reset to 0.)
D9002	I/O unit verify error	I/O module verify error module number	If I/O module data is different from data entered are detected when the power is turned on, the first I/O number of the lowest number module among the detected modules stored in hexadecimal. (Storing method is the same as that of D9000.) The module number monitored by the peripheral is hexadecimal. (Cleared when all contents of D9116 of D 9123 are reset to 0.)
D9005*1	AC DOWN counter	AC DOWN time count	1 is added each time input voltage becomes 80% or less of rating while the CPU unit is performing operation, and the value is stored in BIN code.
D9008*1	Self-diagnostic error	Self-diagnostic error number	When error is found as a result of self-diagnosis, error number 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 [OUT F[ ]] or [SET F[ ]], the F number, which has been detected earliest among the F numbers which have turned on, is stored in BIN code. D9009 can be cleared by [RST F[ ]] or [LEDR] instruction. If another F number has been detected, the clearing of D9009 causes the next number to be stored in D9009.
			When one of F0 to 255 is turned on by [OUT F[ ]] or [SET F[ ]], the F number, which has been detected earliest among the F numbers which have turned on, is stored in BIN code. D9009 can be cleared by [RST F[ ]] or [LEDR] instruction or moving INDICATOR RESET switch on CPU front to ON position. If another F number has been detected, the clearing of D9009 causes the next number to be stored in D9009.
D9011	Error step	Step number at which operation error has occurred	When operation error has occurred during execution of application instruction, the step number, at which the error has occurred, is stored in BIN code. Since storage into D9011 is made when M9011 changes from off to on, the contents of D9011 cannot be renewed unless M9011 is cleared by user program.
D9014	I/O control mode	I/O control mode number	The set I/O control mode is returned with the following code number. 0: Direct mode (I/O) 1: Refresh mode (I) and direct mode (O) 3: Refresh mode (I/O)

Special Register List (Continued)

Number	Name	Stored Date	Explanation
D9015	CPU operating states	Operating states of CPU	The operating states of CPU as shown below are stored in D9015.  *1 When the CPU is in RUN mode and M9040 is off, the CPU remains in RUN mode if changed to PAUSE mode.

Special Register List (Continued)

<table>
<tr><th>Number</th><th>Name</th><th>Stored Date</th><th>Explanation</th></tr>
<tr><td>D9016</td><td>Program number</td><td>0: Main program (ROM)
1: Main proram (RAM)
2: Subprogram (RAM)</td><td>Indicates which sequence program is run presently. One value of 0 to 2 is stored in BIN code. ("2" only for A3NCPU)</td></tr>
<tr><td>D9017</td><td>Scan time</td><td>Minimum scan time (per 10ms)</td><td>If scan time is smaller than the content of D9017, the value is newly stored at each END. Namely, the minimum value of scan time is stored into D9017 in BIN code.</td></tr>
<tr><td>D9018</td><td>Scan time</td><td>Scan time(per 10ms)</td><td>Scan time is stored in BIN code at each END and always rewritten.</td></tr>
<tr><td>D9019</td><td>Scan time</td><td>Maximum scan time (per 10ms)</td><td>If scan time is larger than the content of D9019, the value is newly stored at each END. Namely, the maximum value of scan time is stored into D9019 in BIN code.</td></tr>
<tr><td>D9020*2</td><td>Constant scan</td><td>Constant scan time (to be set by user in units of 10 msec)</td><td>Set the user program execution intervals when a user program is executed in fixed intervals.
0 : Constant scan not executed
1 - 200 : Constant scan executed
(Intervals: setting x 10 msec)</td></tr>
<tr><td>D9094*2</td><td>Replacing I/O module head I/O number</td><td>Replacing I/O module head I/O number</td><td>Stores the upper two digits, in binary, of the head I/O number of the I/O module to be replaced during online operation.
Example: Input module X2F0 → H2F</td></tr>
<tr><td>D9100*1</td><td rowspan="8">Fuse blown module</td><td rowspan="8">Bit pattern in modules of 16 points of fuse blow modules</td><td rowspan="8">Output module numbers (in units of 16 points), of which fuses have blown, are entered in bit pattern. (Preset output number when parameter setting has been performed.)
<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</td><td>0</td><td>0</td><td>0</td><td>1</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</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</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></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></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</td><td>0</td><td>0</td><td>0</td></tr>
</table>
Indicates fuse blow.
(If normal status is restored, clear is not performed. Therefore, it is required to perform clear by user program.)</td></tr>
<tr><td>D9101*1</td></tr>
<tr><td>D9102*1</td></tr>
<tr><td>D9103*1</td></tr>
<tr><td>D9104*1</td></tr>
<tr><td>D9105*1</td></tr>
<tr><td>D9106*1</td></tr>
<tr><td>D9107*1</td></tr>
<tr><td>D9116*1</td><td rowspan="8">I/O module verify error</td><td rowspan="8">Bit pattern in modules of 16 points of verify error modules</td><td rowspan="8">When I/O module data is different from those entered at power-on have been detected, the I/O module numbers (in units of 16 points) are entered in bit pattern. (Preset I/O module numbers when parameter setting has been performed.)
<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</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>0</td><td>1</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></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>D9123</td><td>0</td><td>1 *1</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>
*1...(X/Y7E0)
Indicates I/O unit verify error.
(If normal status is restored, clear is not performed. Therefore, it is required to perform clear by user program.)</td></tr>
<tr><td>D9117*1</td></tr>
<tr><td>D9118*1</td></tr>
<tr><td>D9119*1</td></tr>
<tr><td>D9120*1</td></tr>
<tr><td>D9121*1</td></tr>
<tr><td>D9122*1</td></tr>
<tr><td>D9123*1</td></tr>
</table>

Special Register List (Continued)

Number	Name	Stored Date	Explanation
D9124	Annunciator detection quantity	Annunciator detection quantity	When one of F0 to 255 is turned on by [OUT F[ ]] or [SET F[ ]], 1 is added to the contents of D9124. When [RST F[ ]] or [LED R] instruction is executed, 1 is subtracted from the contents of D9124. (For A3MNCPU, it can be performed by use of INDICATOR RESET switch on front face of CPU module.) Quantity, which has been turned on by [OUT F[ ]] or [SET F[ ]] is stored. The value of D9124 is maximum 8.
D9125 D9126 D9127 D9128 D9129 D9130 D9131 D9132	Annunciator detection quantity	Annunciator detection quantity	When one of F0 to 255 is turned on by [OUT F[ ]] or [SET F[ ]], F number, which has turned on, is entered into D9125 to D9132 in due order. F number, which has been turned off by [RST F[ ]], is erased from D9125 to D9132, and the contents of data registers succeeding the data register, where the erased F number was stored, are shifted to the preceding data registers. By executing [LED R] instruction, the contents of D9125 to D9132 are shifted upward by one. (For A3MCPU, it can be performed by use of INDICATOR RESET switch on front of CPU unit.) When there are 8 annunciator detections, the 9th one is not stored into D9125 to 9132 even if detected.

		SET F50	SET F25	SET F19	SET F25	SET F15	SET F70	SET F65	SET F38	SET F110	SET F151	SET F210	LED R
D9009	0	50	50	50	50	50	50	50	50	50	50	50	99
D9124	0	1	2	3	2	3	4	5	6	7	8	8	8
D9125	0	50	50	50	50	50	50	50	50	50	50	50	99
D9126	0	0	25	25	99	99	99	99	99	99	99	99	15
D9127	0	0	0	99	0	15	15	15	15	15	15	15	70
D9128	0	0	0	0	0	0	70	70	70	70	70	70	65
D9129	0	0	0	0	0	0	0	65	65	65	65	65	38
D9130	0	0	0	0	0	0	0	0	38	38	38	38	110
D9131	0	0	0	0	0	0	0	0		110	110	110	151
D9132	0	0	0	0	0	0	0	0	0		151	151	210

IB (NA) 66233-A

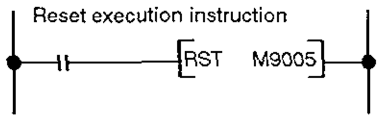
POINT

(1) Contents are cleared when any of the following operations is executed: power switching off, latch clear, and reset. Contents are retained when the RUN key switch is set STOP.

(2) Among the special registers, contents of those marked with (*1) are not cleared when normal state is restored. To clear the contents of these registers, follow the steps below.

1) To clear a user program:

Insert the circuit shown right in the program; turn on the reset execution instruction contact to clear the contents of special registers.



2) To clear using a peripheral device:

Clear the contents to "0" using the test function of the peripheral device or using the forced reset operation.

For operation procedure, refer to the Manual of the peripheral device to be used.

3) To clear using the RESET switch:

Place the RESET switch in the RESET position.

(3) For those marked with (*2), data is written using a sequence program.

APPENDIX 4 PROCESSING TIME

(1) Sequence program processing time

The processing time required for the sequence instructions, basic instructions, and application instructions used in an A73CPU sequence program is the same as that required when processing them with the A3NCPU, with the exception of the instructions indicated below.

For the A3NCPU processing time, see the ACPU Programming Manual.

(a) Servo program start request command (DSFRP)
(b) Control change command (DSFLP)
(c) Program end command (END)

Processing time of these commands is indicated below.

Table 4.1 Processing Time - DSFRP, DSFLP, and END Commands

Command	Conditions		Processing Time (μsec)	
			Direct Mode	Refresh Mode
DSFRP	Starting 1-axis positioning		180	
	Starting 2/3 axis interpolation		200	
	Error		850	
DSFLP	Present position data change	Normal	120	
		Error	770	
	Velocity change	Normal	80	
		Error	700	
END			7600	

(2) PCPU processing time

The PCPU processing time after the PC ready flag (M2000) is set in response to the start request is indicated below.

Table 4.2 PCPU Processing Time

Conditions	Processing Time (msec)
Servo program processing time *1	10 to 21
Response to velocity change	13 to 16
Time between PC ready flag (M2000) on and PCPU ready flag (M9074) on	80 to 400
Processing time for multiple servo program simultaneous start *2	23 to 27

POINT

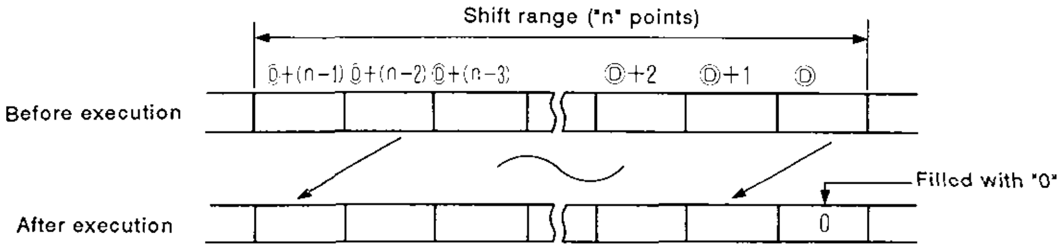
(1) *1 .... Start time varies considerably for FEED, VPF, or VPR commands (approximately 1000 msec in the worst case).

(2) *2 .... Processing time for multiple program simultaneous start varies depending on the number of axes and the combination of the commands. The indicated processing time can only be used for reference purposes.

APPENDIX 5 PROGRAM EXAMPLES

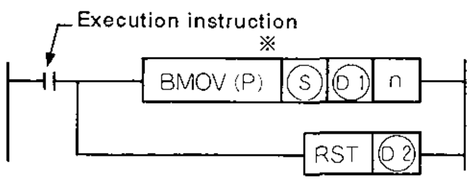
5.1 Shifting One Word to the Left (Word Data)

(1) The program which shifts "n" points of data to the left with the designated word device as the head device is indicated below.



(2) For shifting one word to the left of the word data, the BMOV(P) and RTS commands are used.

The program format is shown in Fig. 5.1.



REMARKS

1. S, D1, D2 and n

   S, D2 : Head number of the word devices to be shifted left

   D1 : The number of (S-designated word device + 1)

   n : Shift range

2. *: BMOV(P) indicates BMOV or BMOVP

Fig. 5.1 Left Shift Program Format Using BMOV(P) and RST Commands

Example:

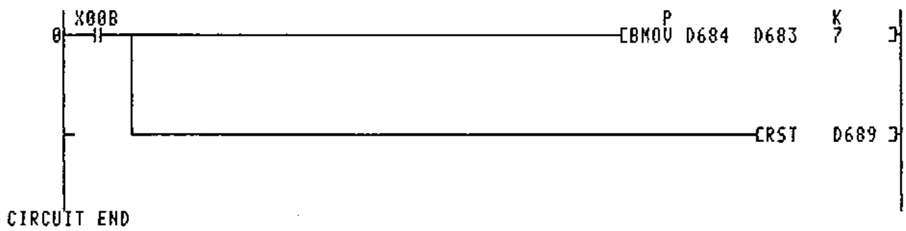
The program to shift the data in D683 to D689 one word to the left at the leading edge of XB is shown below.

[Operation]

Shift range

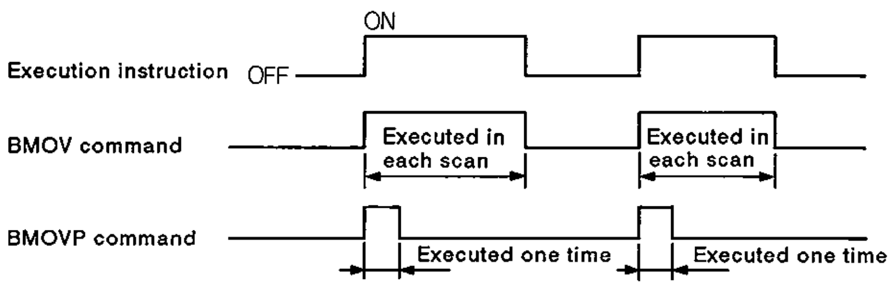
	D689	D688	D687	D686	D685	D684	D683
Before execution	−100	503	600	−336	3802	−32765	5003
After execution	503	600	−336	3802	−32765	5003	0

[Program example]



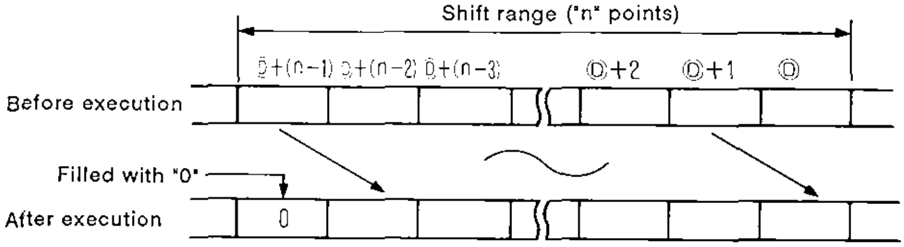
(3) Execution conditions

The BMOV or BMOVP command is executed, as indicated below.



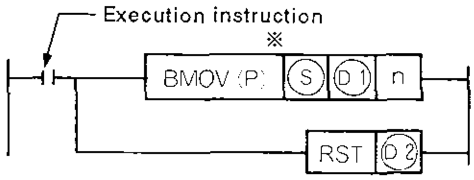
5.2 Shifting One Word to the Right (Word Data)

(1) The program which shifts "n" points of data to the right with the designated word device as the head device is indicated below.



(2) For a one-word shift to right of word data, the BMOV(P) and RTS command are used.

The program format is shown in Fig. 5.2.



REMARKS

1. S, D1, D2 and n

   S : Head number of the word devices to be shifted right

   D1 : The number of (S-designated word device + 1)

   D2 : Last number of the word devices to be shifted right.

   n : Shift range

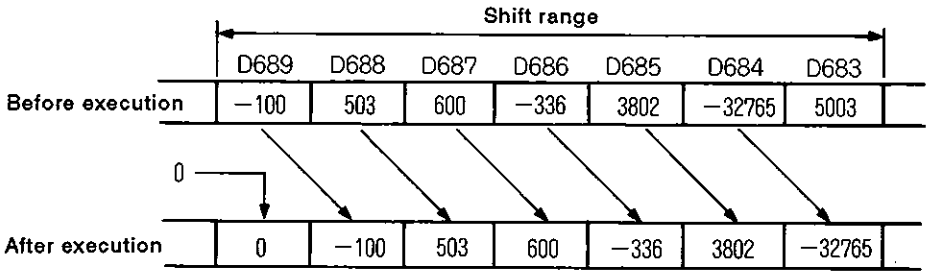
2. *: BMOV(P) indicates BMOV or BMOVP

Fig. 5.2 Right Shift Program Format Using BMOV(P) and RST Commands

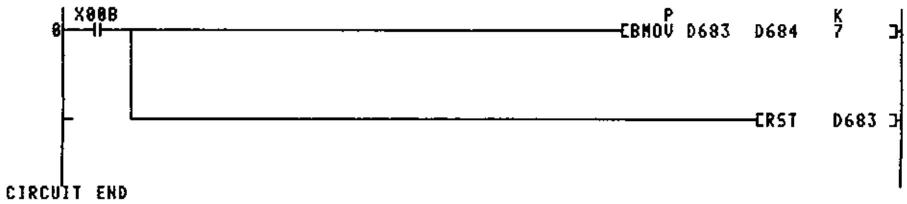
Example:

The program to shift the data in D683 to D689 to the right by one word at the leading edge of XB is shown below.

[Operation]

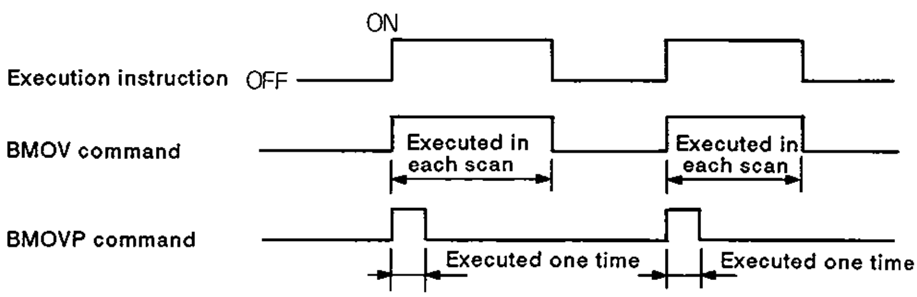


[Program example]



(3) Execution conditions

The BMOV or BMOVP command is executed, as indicated below.



5.3 Reading M Codes

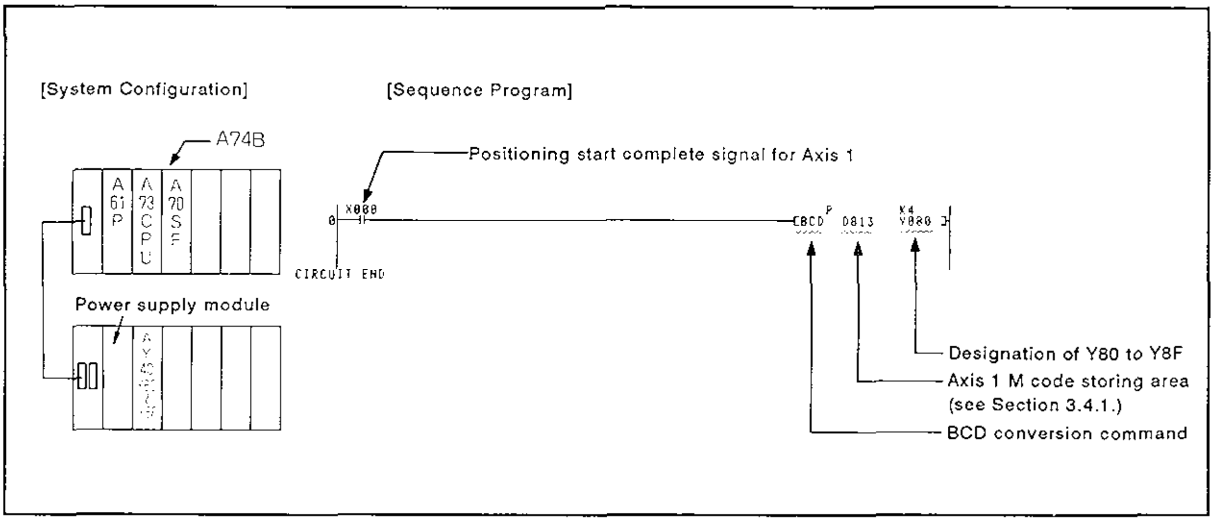
The program used to read an M code at the completion of position start or at the completion of positioning is indicated below.

The following signals are used to detect positioning start complete or positioning complete.

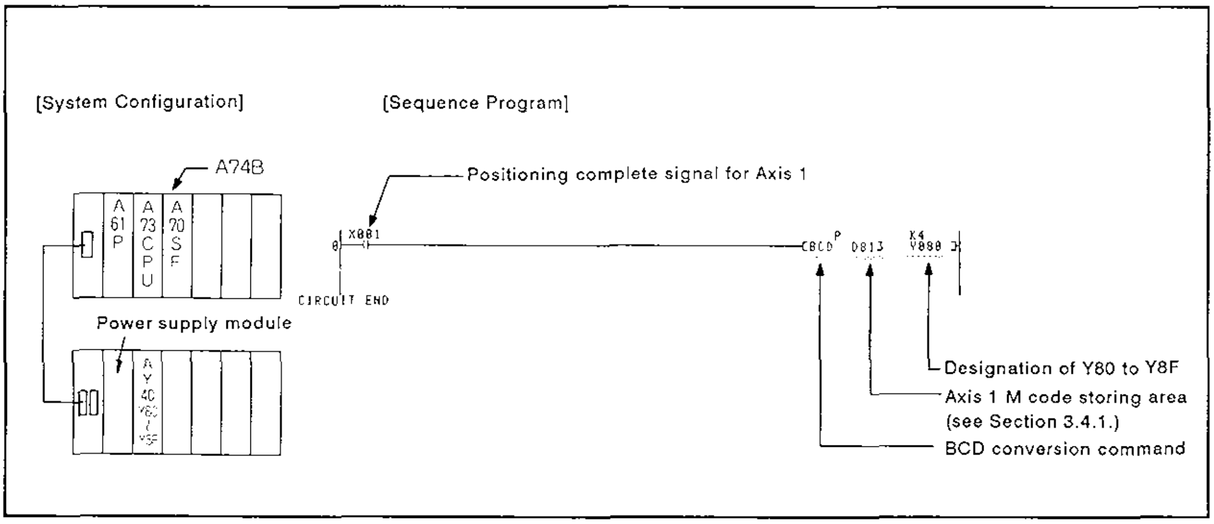
(a) Xn0 .... Positioning start complete
(b) Xn1 .... Positioning complete

[Program example]

(1) The following is the program to output an M code of Axis 1 through Y80 to Y8F after converting it into BCD code after the completion of positioning start.



(2) The following is the program to output an M code of Axis 1 through Y80 to Y8F after converting it into BCD code after the completion of positioning.



5.4 Reading Error Codes

A sample program to read an error code at the occurrence of an error is indicated below.

The following signals are used to recognize the occurrence of an error.

(a) *Error detection signal (Xn7).........Minor and Major errors*
(b) Error detection signal (Xn8).........Servo error

POINT

Storing of the error code lags behind the leading edge of Xn7/Xn8 as indicated below.

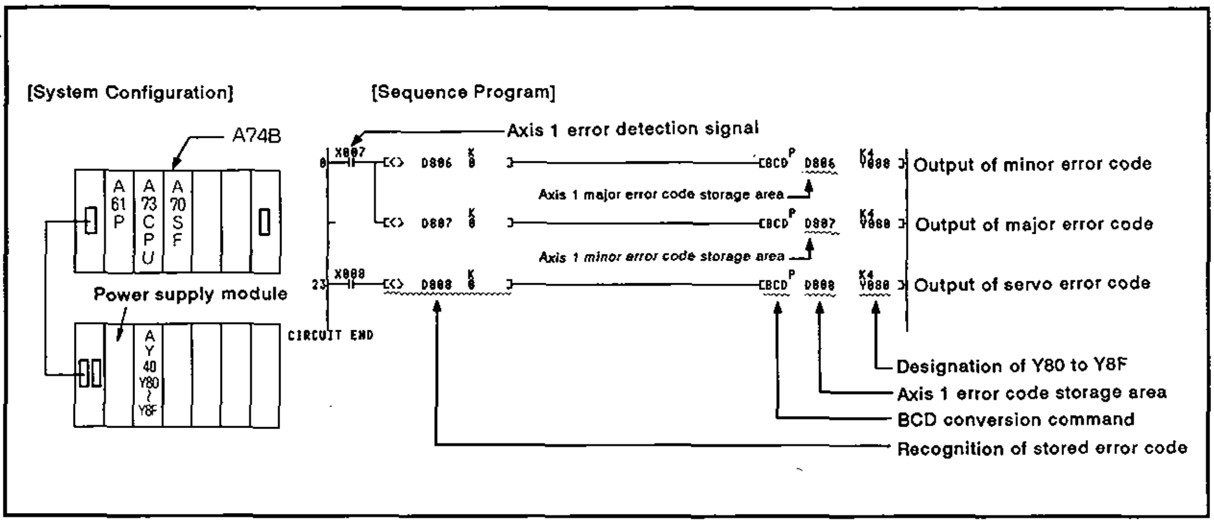
a) Maximum 80 msec when the sequence program scan time is less than 80 msec.

b) Maximum one scan time when the sequence program scan time is more than 80 msec.

Take this delay into account when reading the error code.

[Program example]

(1) The following is the program to output the error code at an occurrence of an error (minor error, major error, servo error) with Axis 1 through Y80 to Y8F after converting the error code into BDC code.



IMPORTANT

The components on the printed circuit boards will be damaged by static electricity, so avoid handling them directly. If it is necessary to handle them take the following precautions.

(1) Ground human body and work bench.

(2) Do not touch the conductive areas of the printed circuit board and its electrical parts with any non-grounded tools etc.

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) 66233-A (9004) MEE Printed in Japan Specifications subject to change without notice.