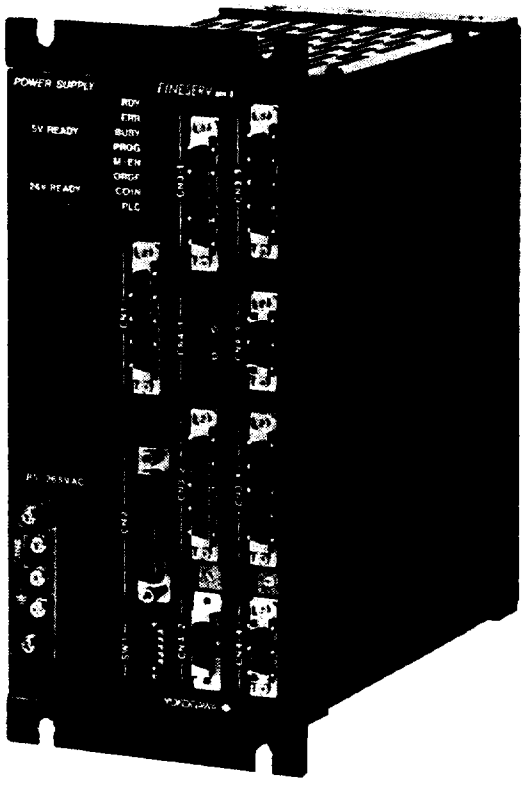
Instruction Manual

FINESERV MK II

Dedicated Controller
PC1 Series



YOKOGAWA ◆
Yokogawa Precision Corporation

IM B201-E
Sep. 19, 1994

PREFACE

Thank you for purchasing of our PC1 Series FINESERV MKII Dedicated Controller.

The FINESERV MKII Dedicated Controller is provided with high performance and sophisticated functions which are required for automatic controllers used for factory automated machines. A variety of functions are built into a compact body of the FINESERV MKII Dedicated Controller so that many types of curves can be drawn with ease.

To use fully all the available functions of the FINESERV MKII Dedicated Controller, we recommend you, besides reading this manual thoroughly, to read the instruction manuals of the other devices you are going to use with the FINESERV MKII Dedicated Controller, such as the DYNASERV, AC/DC servo motors, stepping motors, and inverter/encoder for synchronization.

NOTICES

- Copying or reproduction by any means of all or any part of the contents of this manual without permission is strictly prohibited.
- The contents of this manual are subject to change without prior notice.
- Every effort to ensure accuracy has been made in the preparation of this manual. However, if you should notice any errors or omissions, please contact your dealer or service personnel of Yokogawa Precision representative.
- Yokogawa Precision shall have no responsibility for indirect or consequential damages such as, but not so as to limit the foregoing, loss of profit, or loss of production, caused by the use of our products in accordance with this manual.

Caution on Request of Reparation

The contents of your file created with the FINESERV MKII will be deleted at the factory for reparation, and the file will be set in the same conditions as when it was shipped first time. Be sure to make a copy of your file before shipment if you need the same file.

FD No. IM PCB-01E

ABOUT THIS MANUAL

This manual describes functions and operations of the PC1 Series FINESERV MKII Dedicated Controller. For operations of display operation equipment, please refer to the instruction manuals of the equipment or utility programs listed below.

- IM-PA1A001-E for Operation Display Panel PA100AP
- IM-PC000AT-E for Operation Display Pendant PC000AT
- IM PCB-03A-E for Maintenance Utility in the Tool Box Utilities
- IM PCB-03B-E for Cam Curve Generation Utility in the Tool Box Utilities

This manual is provided with the following five chapters and an appendix. The readers who operate the FINESERV MKII Dedicated Controller for the first time shall read all the chapters in order. Note that Chapter "4. OPERATIONS" is provided with full details of operations required for the FINESERV MKII Dedicated Controller.

Following explains briefly the contents of Chapter 1 through Chapter 5 and the APPENDIX.

Chapter 1. OUTLINE
describes features, functions of programs, system configuration, specifications and related products in the FINESERV MKII Dedicated Controller system.

Chapter 2. APPEARANCE AND NAME OF PARTS
shows appearance and dimensions. It also introduces names of each part.

Chapter 3. INSTALLATION AND CONNECTIONS
covers required conditions for installation and connections, and is provided with details of connecting diagrams, connection cables and connectors. It also explains functions of the DYNASERVs (DD motors), AC/DC servo motors, stepping motors, and inverter to be connected with the FINESERV MKII Dedicated Controller.

Chapter 4. OPERATIONS
describes fully operations of the PC1 Series FINESERV MKII Dedicated Controller.

Chapter 5. MALFUNCTION AND REMEDIES / MAINTENANCE AND CHECKS
covers remedies when the FINESERV MKII Dedicated Controller malfunctions, and maintenance and checks.

APPENDIX
lists commands, errors / alarms and connector specifications.

CAUTION

Before connecting the FINESERV MKII Dedicated Controller to the DYNASERV driver (DD driver), AC/DC servo driver, and/or stepping motor driver, adjustments and operation checks in corresponding combination of only the driver and the motor shall be completed successfully in accordance with the instruction manuals supplied with them. When making connections between the FINESERV MKII Dedicated Controller and the AC / DC servo driver, stepping motor driver, and/or inverter, be sure to refer to the connection specifications of this instruction manual and of the instruction manuals of the devices to be connected at the same time.

CONTENTS

1. OUTLINE

1.1 Features

■ Versatile Control Functions built into a Compact Body!

- ◆ Besides connecting a DYNASERV (DD motor), an AC / DC servo motor, a stepping motor and an inverter / encoder for synchronization can be connected.
- ◆ Synchronizing time or position is possible between external units or internal axes.
- ◆ 5 input unit systems are available.
- ◆ Smaller than a letter size with 4 axes control functions

■ Flexible Cam Curve is realized on Site!

- ◆ 8 cam curves are available in addition to the 3rd order spline and trapezoid for acceleration / deceleration curve.
- ◆ A desirable cam curve can be obtained on site using the tool box utilities (optional). The curve can be downloaded to the FINESERV by drawing the curve with a mouse.

■ Fit to High Speed Motion Control Systems for Short Cycle Time!

- ◆ Feed forward control
- ◆ Response time is just a quarter of our current models.

■ High Accuracy and Reliability!

- ◆ Highly accurate backlashless motion control is possible using with DYNASERV (DD motor).
- ◆ 9-digits data can be set for any motors.
- ◆ 4 error histories can be displayed using 48 error messages.

1.2 System Configuration

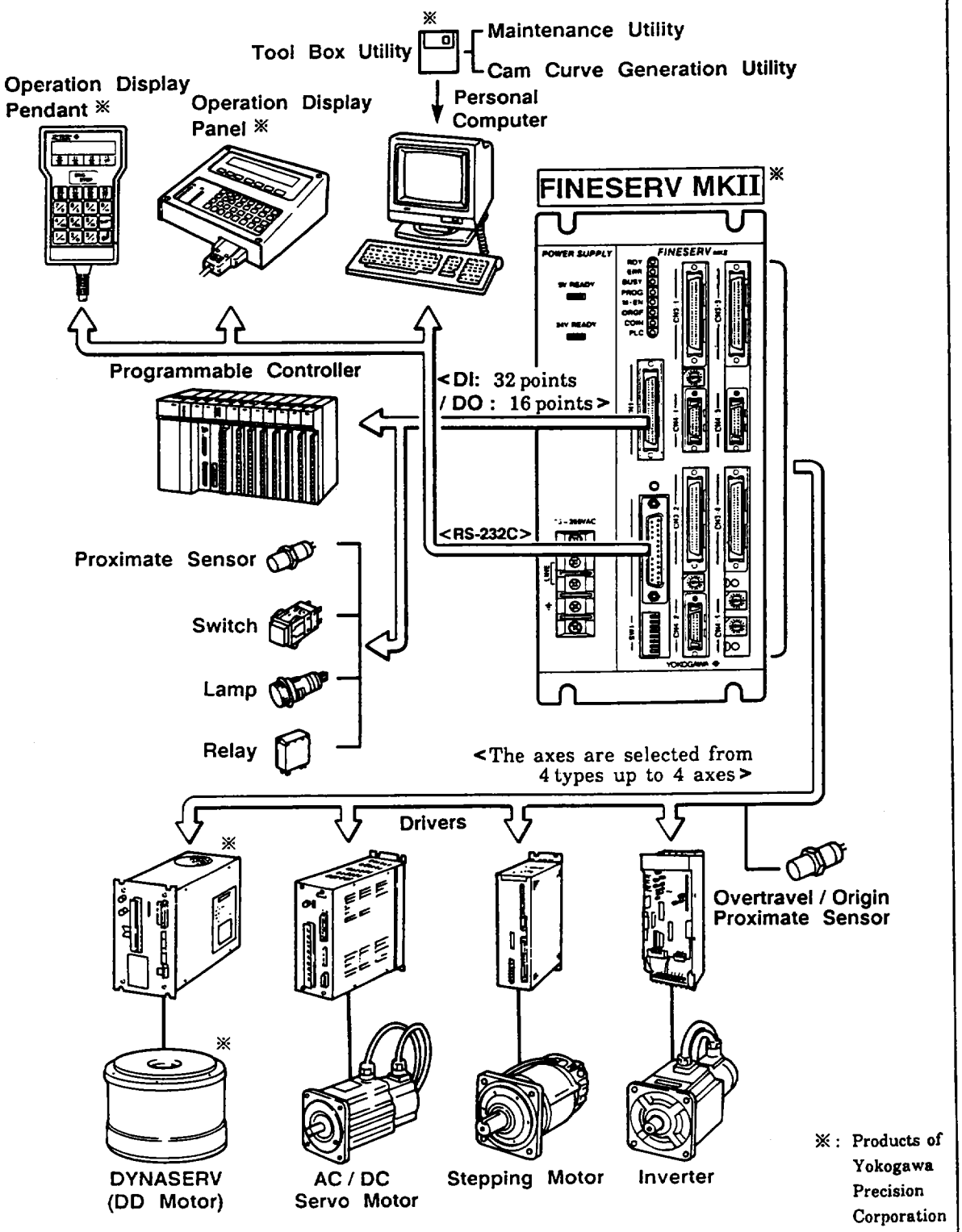
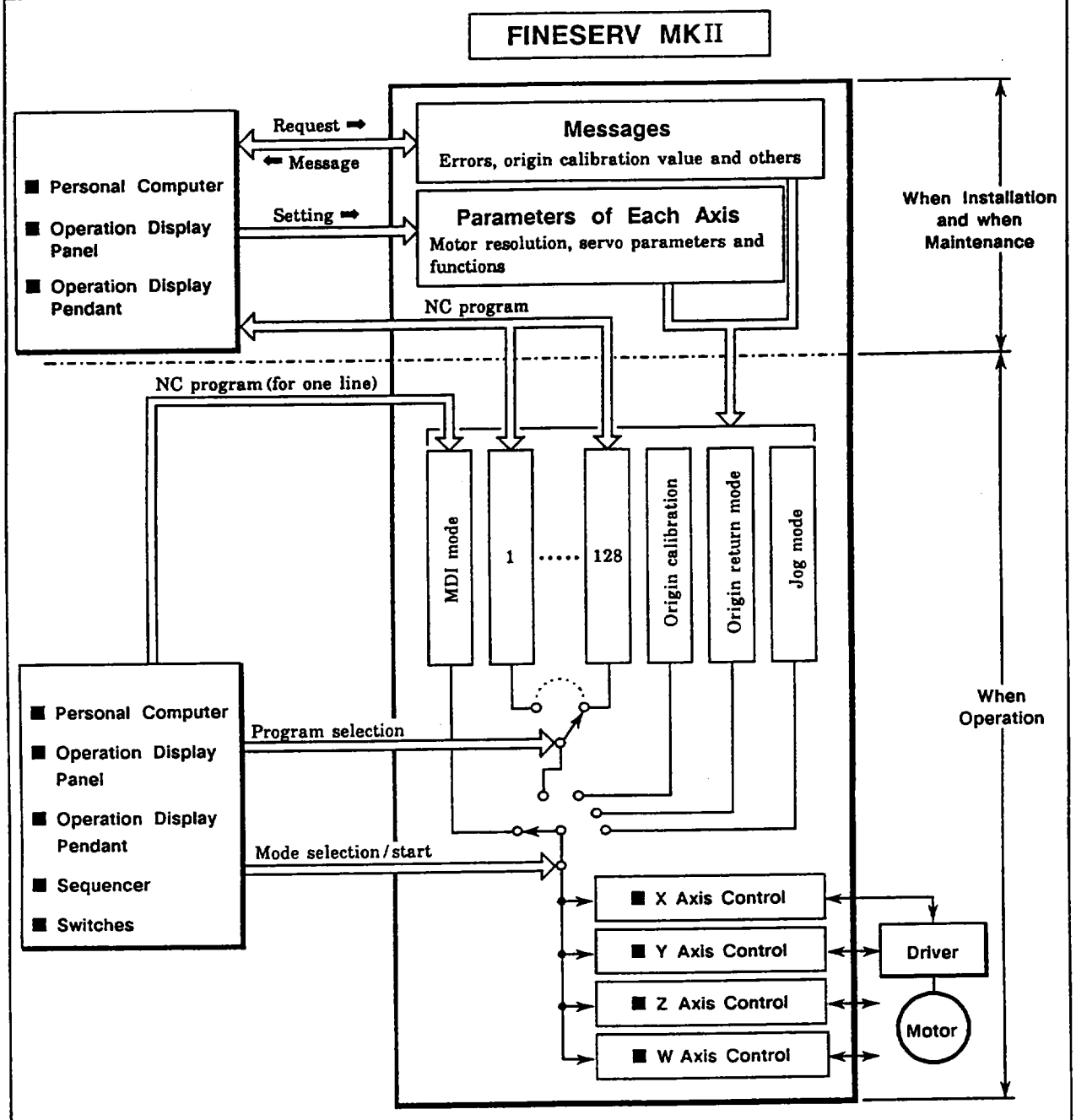


Figure 1.1 System Configuration of FINESERV MKII Dedicated Controller PC1 Series

1.3 Program Function



Note : The parameters of each axis mean that resolution of the motor encoder, servo parameters and functions for X, Y, Z and W axes. NC program is written in NC language. Reading and setting of each parameter is done using @ commands.

Figure 1.2 Outline of Program Function

1.4 Specifications

(1) General Specifications

(a) Main Unit

Table 1.1 Main Unit Specifications

Item		PC100 □0 / without Power Supply	PC100 □1 / with Power Supply
Power Supply	Voltage	5V DC±3%, 24V DC±5%	85V AC to 265V AC
	Current	5V DC : 3A / 1 axis, 2 axes 5A / 3 axes, 4 axes 24V DC : 0.7A / 1 axis, 2 axes 0.7A / 3 axes, 4 axes	30W / 1 axis, 2 axes 40W / 3 axes, 4 axes
Power Supply for Memory Backup		Lithium battery (Life: 20,000 hours)	
Weight		2.3kg (5lb 1oz) (For 4 axes)	3.2kg (7lb 1oz) (For 4 axes)
Structure		Mountable in a rack or on the wall	
Withstand Voltage		1.5kV AC for one minute between the input and the case	
Insulation Resistance		5MΩ or more at 500V AC between the output and the case (25°C (77°F) 70%RH)	

(b) Display Operation Unit

Table 1.2 Display Operation Unit Specifications

Item	Operation Display Panel / PA100AP	Operation Display Pendant / PC000AT
Power Supply	5V DC / 0.2A supplied from the main unit	
Weight	480g (1lb 1oz)	300g (11oz)
Structure	Separation board type (operation and display parts) Casing	Handy type
Life of the Switches	500,000 times	500,000 times

* : For details, refer to the instruction manuals of the display operation equipment.

(c) Tool Box Utilities

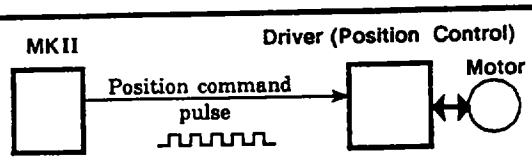
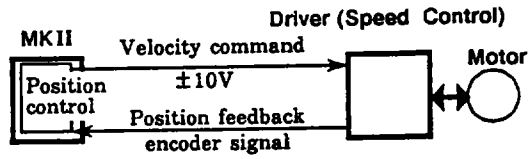
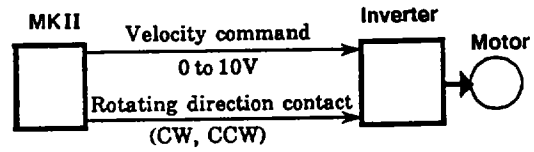
Table 1.3 Tool Box Utilities Specifications

Item	Description
Floppy Disc	3.5-inch / 2DD type
Operation Environment	IBM PC AT or compatibles with 640KB or more, MS-DOS
Message	English

*: For details, refer to the instruction manuals of the Tool Box utilities (utility for maintenance and utility for cam curve generation).

(2) Control Function Specifications

Table 1.4 Control Function Specifications

Control Mode	Control Block Diagram	Connectable Device	Specification
Position		DYNASERV (DD motor) driver Stepping motor driver	Output frequency • DD driver: 1.36MHz at max. • Stepping driver: 86kHz at max.
Velocity (for Position Control)		Speed control driver of AC/DC servo motors (use with an incremental encoder)	Position control system: PID and speed feed forward Position loop gain: 0.03125 to 64 Speed feed forward gain: 0 to 120% Position settling check width: 0 to 32767 pulses
Velocity (without Position Control)		Inverter	See item "(3) Connection Specifications"

The number of axes to be controlled: 4 axes for all motors except for the inverter (only 1 axis for inverter)
The number of axes simultaneously controllable : 4 axes

(3) Connection Specifications

Table 1.5 Connection Specifications

Item			AC / DC Servo Driver	Stepping Driver	Inverter	
Position Command	System		—	UP/DOWN pulse	—	
	Electrical Specifications of Output		—	Open-collector	—	
Velocity Command (for Position Control)	Electrical Specifications of Output		±10V DC	—	—	
Velocity Command (without Position Control)	Electrical Specifications of Output		—	—	0 to 10V DC at max. Max value variable between 6 and 10V DC Rotating direction : CW/CCW with contact output	
Encoder Input	Multiplication		4	—	4	
	Electrical Specifications of Input		Use with a differential signal	Open-collector	—	Open-collector
	Maximum Frequency	UP / DOWN	3MHz	—	—	300kHz
		A / B	750kHz	—	—	75kHz
		Z	750kHz	75kHz	—	75kHz

Note 1. Connectable with the following DYNASERV (DD motor)
DM1□□□A, DM1□□□B, DM8□□□A, DM8□□□B, DR1□□□A,
DR1□□□B, DR1□□□E, DR5□□□A, DR5□□□B.

(4) Operation Function Specifications

Table 1.6 Operation Function Specifications

Item				Description
Transmission Mode (Instruction)				■ RS-232C ■ Parallel transmission
Equipment to be Connected	RS-232C			■ Personal computer ■ Operation display panel ■ Operation display pendant
	Parallel Transmission			■ Sequencer ■ Switches
Operation Mode	Jog Mode			■ H/L speed switching ■ Data setting of H/L ■ Rotating direction switching
	Origin Return Mode	System		■ Mechanical coordinates origin return ■ Program coordinates origin return
		Operation Type		■ 4 types ● Reverse (with/without overtravel sensor) ● Non-reverse (with/without overtravel sensor)
		Variable Functions		■ Return rotating direction ■ Origin search speed
	Origin Calibration Mode			■ Distance measurement of origin sensor and origin mark
	Program Operation Mode (AUTO/STEP/CONT)	Position Setting Function	System	■ Point to point
			Acceleration/Deceleration Type	■ 3rd order spline/trapezoid ■ Standard 8 cam curves ■ Two user defined cam curves (optional)
			Acceleration/Deceleration Data Setting	■ Acceleration/deceleration duration (3rd order spline/trapezoid) ..... 10ms to 60s ■ Feeding duration (cam curves) ..... 10ms to 6000s
			Position Data Type	■ ABS/INC
			Position Data Input Unit System	■ Pulse ■ Degree (one rotation) ■ Degree (multi-rotation) ■ Detection number (1 to 1024) ■ Feeding length (μm)
			Position Data Digits	■ Sign and 9-digits (depending on the input unit system)
		Synchronization Function	Start Delay Synchronization	■ Start delay with respect to the start signal (X/Y/Z/W)
			Time Synchronization	■ Assigned time synchronization start with respect to start of X(Y/Z/W)
			Position Synchronization	■ Start of X after assigned position operation (Y/Z/W)
		Temporary Stop Function		■ Stops deceleration by switching on the external contact. Remaining amount is operated by switching off the external contact.
		Origin Return Function		■ The same as the origin return mode
		Supplementary Functions (M/O)		■ BCD 2-digits output
		Servo Parameter Remote Setting Function		■ fc, COIN width, integral/proportional control switching and DC gain for the DYNASERV driver ■ Integral/proportional control switching for the AC/DC driver
		Dwell Function		■ Starts the operation of the next block delayed by the specified times
	MDI Mode (only for RS-232C)			■ The same as the programmed operation mode (STEP)
Supplementary Function*	Overtravel Check			■ Stops by detecting the stroke limit ■ 4 values of the stroke limit available for each axis • Soft limit value (+, −) • Overtravel sensor (+, −)
	Override			■ 0 to 100%

*: Applicable to all operation modes

(5) Program Function Specifications

Table 1.7 Program Function Specifications

Item		Description
NC Language	The Number of Programs	128 (99 for parallel transmission) at maximum
	Capacity of Program	32KB
	The Number of Blocks	Not limited (Limited by 32KB)
	Details of Codes	Separately described
	Special Codes	Multiple input unit systems, synchronization function type and remote setting of servo parameters
Command Function	Parameter Setting for Each Axis	Servo parameter
		Function parameter
	Commands with Actions	Mode selection start
		NC program file management
		Message request and others

1.5 Product Configuration and Product Names

The controller has been inspected at the factory, and then shipped. However, please check the following items before installing the controller.

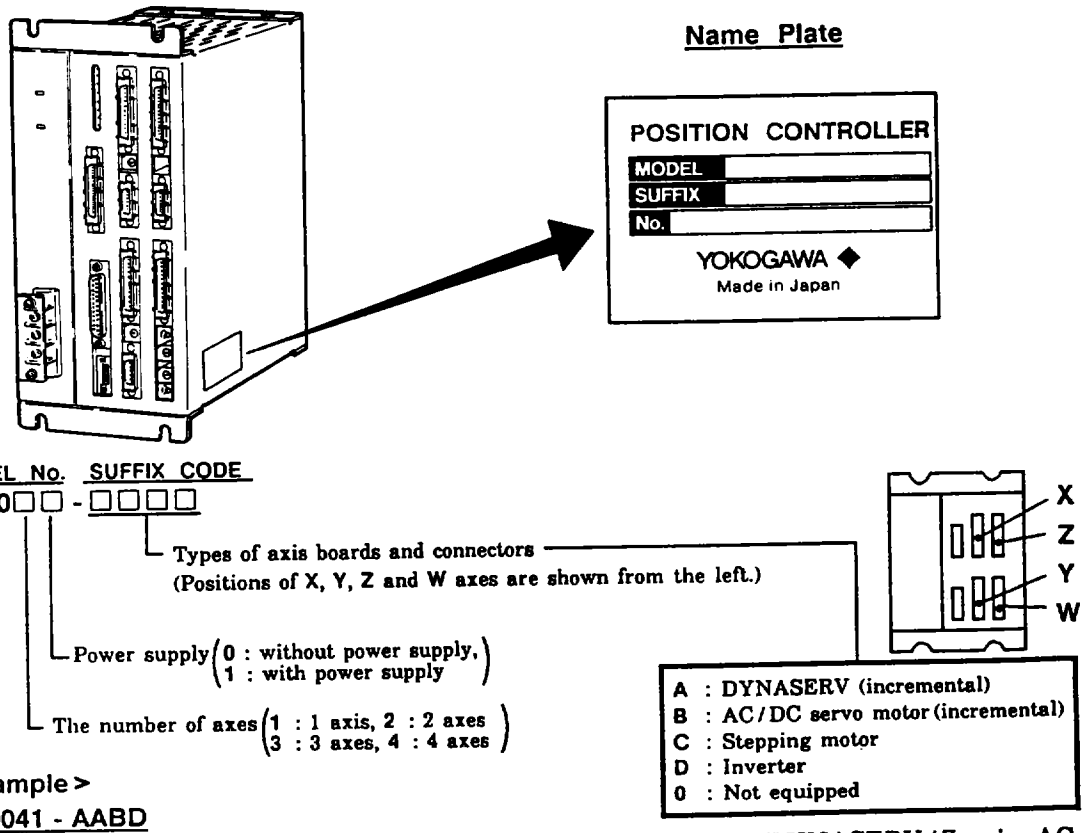
(1) Main Unit and Standard Parts

The list below shows composition of the PC1 Series FINESERV MKII.

Product Type	Name / Model	Standard Parts	Remarks
Main Unit (with Power Supply)	FINESERV MKII PC1 Series / PC100□1	■ Instruction Manual	Checking of the number of axes, types of axis board
Main Unit (without Power Supply)	FINESERV MKII PC1 Series / PC100□0	■ Instruction Manual ■ Molex connectors (Connector 5199-04 : 1 piece, pin 5194TL : 6 pieces)	Checking of the number of axes, types of axis board

(2) Checking Model Name

Model name and other information are provided on the name plate on the right side of the FINESERV MKII as shown in Figure 1.3. Check whether the model is the same one as you have specified. When you have a problem with your FINESERV MKII and you are going to contact your dealer or service personnel of Yokogawa Precision, be sure of the model name and the serial number.



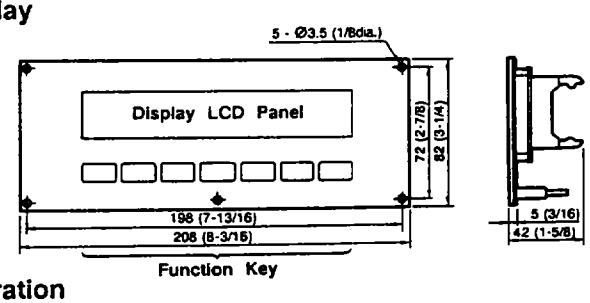
<Example>
PC10041 - AABD
(4 axes control / with power supply / X axis: DYNASERV / Y axis: DYNASERV / Z axis: AC / DC servo motor / W axis: inverter)

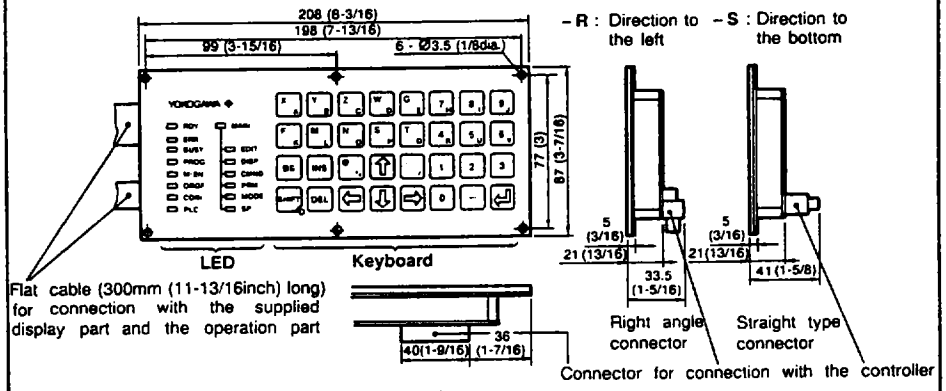
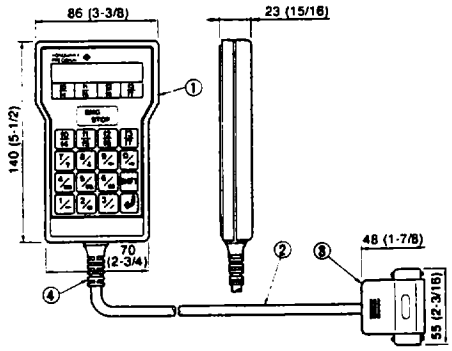
Figure 1.3 Checking Model Name

(3) Optional Parts

Following shows the list of optional parts. For separately sold cables, connectors and covers, see "3.3 (1) Connection Cables and (2) Connectors and Covers".

Table 1.4 List of Optional Parts

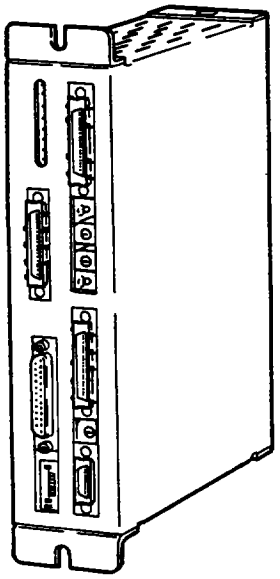
<table>
<tr><th>Item</th><th colspan="4">Parts Name</th><th>Model Name</th></tr>
<tr><td>1</td><td colspan="4">Operation Display Panel
• Display
 

• Operation
 
</td><td>PA100AP-{R S</td></tr>
<tr><td>2</td><td colspan="4">Operation Display Pendant
 

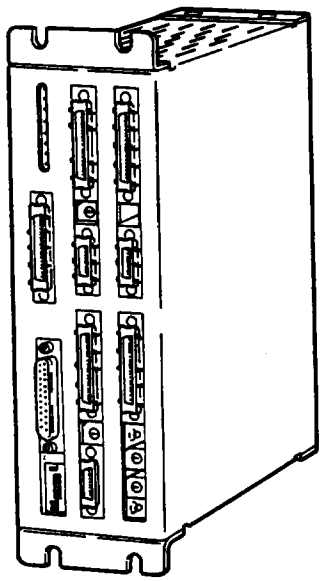
<table>
<tr><th>Number</th><th>Parts Name</th><th>Quantity</th></tr>
<tr><td>①</td><td>Unit</td><td>1</td></tr>
<tr><td>②</td><td>Cable</td><td>1.5m (59-1/16inch)</td></tr>
<tr><td>③</td><td>Connector</td><td>1</td></tr>
<tr><td>④</td><td>Cable clamp</td><td>1</td></tr>
</table></td><td>PC000AT</td></tr>
<tr><td>3</td><td colspan="4">Lithium Battery for Backup and Memory of the Main Unit (with Connector)
 
Note : This part is already equipped with the main unit at the factory. Purchase it when it is to be replaced.</td><td>R7304LT</td></tr>
<tr><td rowspan="3">4</td><td colspan="2">Parts Name</td><td>Model Name</td><td>Supplied Parts</td></tr>
<tr><td rowspan="2">Tool Box Utilities</td><td>Maintenance UTY</td><td>KC101A</td><td>Instruction manual</td></tr>
<tr><td>Cam curve Generation UTY</td><td>KC102A</td><td>Instruction manual</td></tr>
</table>

2. APPEARANCE AND NAME OF PARTS

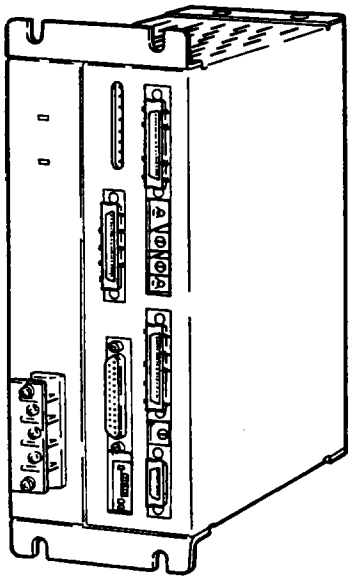
2.1 Appearance



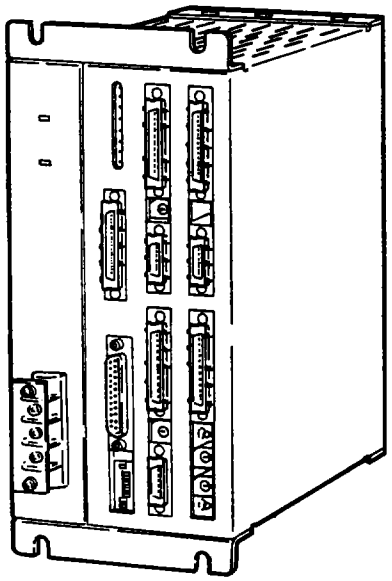
(1) PC10010 (1 Axis Control / without Power Supply)
PC10020 (2 Axes Control / without Power Supply)



(3) PC10030 (3 Axes Control / without Power Supply)
PC10040 (4 Axes Control / without Power Supply)



(2) PC10011 (1 Axis Control / with Power Supply)
PC10021 (2 Axes Control / with Power Supply)



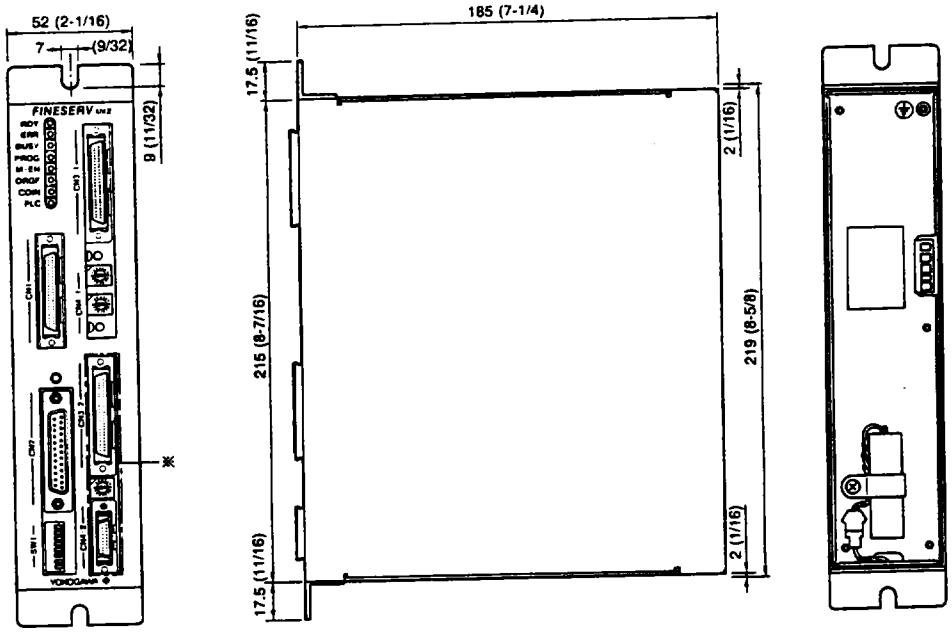
(4) PC10031 (3 Axes Control / with Power Supply)
PC10041 (4 Axes Control / with Power Supply)

Figure 2.1 Appearance of FINESERV MKII

2.2 Dimensions

(1) PC10010 (1 Axis Control / without Power Supply) / PC10020 (2 Axes Control / without Power Supply)

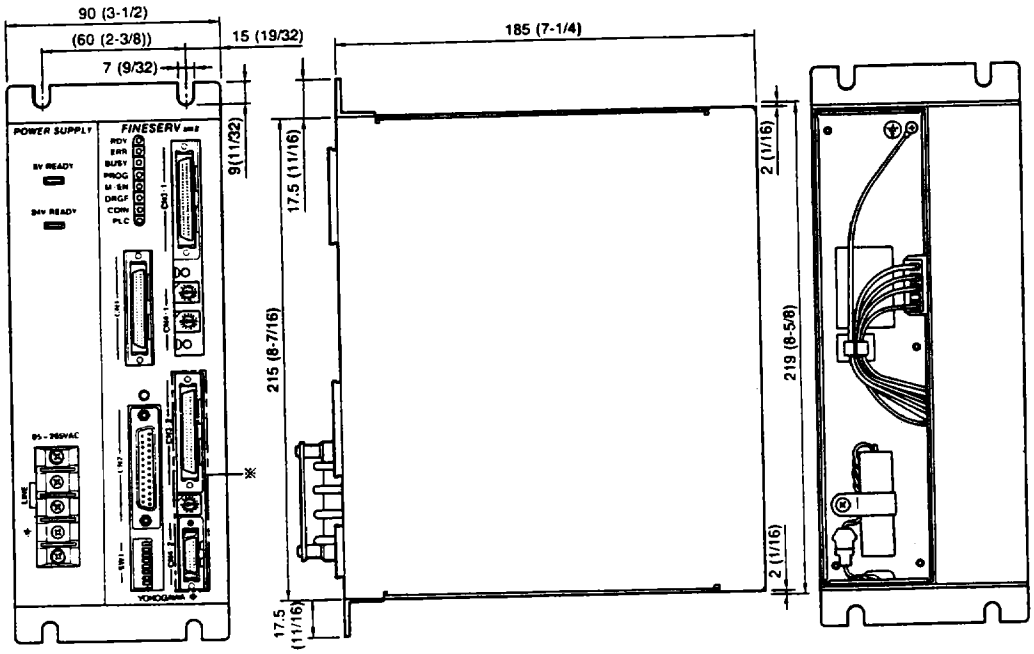
Unit : mm
(Approx. inch)



• Panel Cut Size : $52^{+1}_{+0.5}$(2-1/16)×$219^{+1}_{+0.5}$(8-5/8)

Figure 2.2 Dimensions of PC10010 / PC10020

(2) PC10011 (1 Axis Control / with Power Supply) / PC10021 (2 Axes Control / with Power Supply)



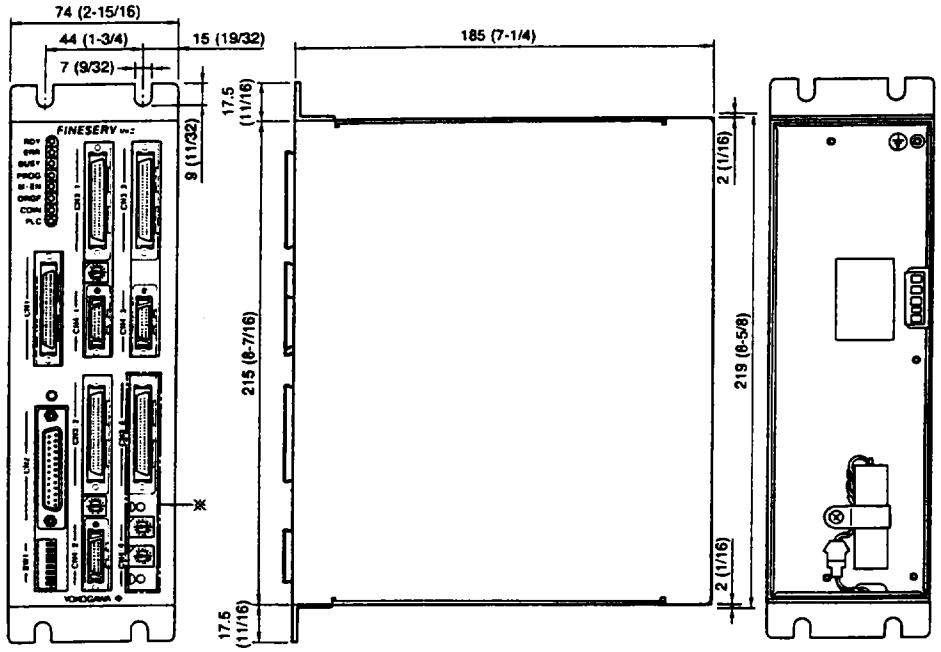
• Panel Cut Size : $90^{+1}_{+0.5}$(3-1/2)×$219^{+1}_{+0.5}$(8-5/8)

Figure 2.3 Dimensions of PC10011 / PC10021

※ : For 1 axis type, this part is covered with a blind panel.

(3) PC10030 (3 Axes Control / without Power Supply) / PC10040 (4 Axes Control / without Power Supply)

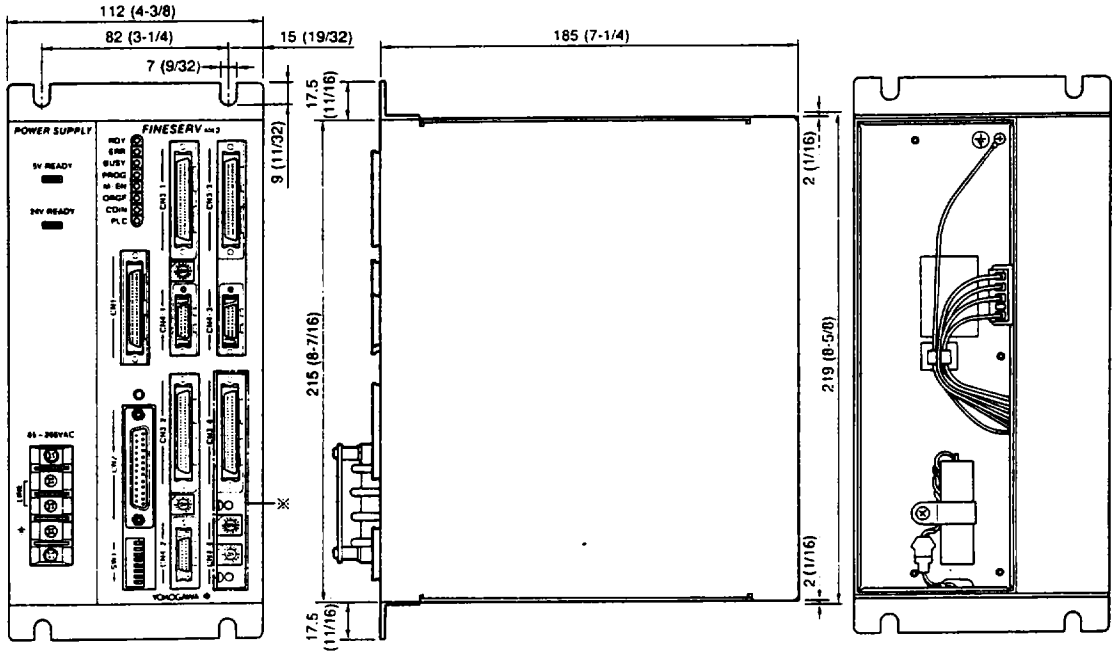
Unit : mm
(Approx. inch)



• Panel Cut Size : $74^{+1}_{+0.5}$ (2-15/16) × $219^{+1}_{+0.5}$ (8-5/8)

Figure 2.4 Dimensions of PC10030 / PC10040

(4) PC10031 (3 Axes Control / with Power Supply) / PC10041 (4 Axes Control / with Power Supply)

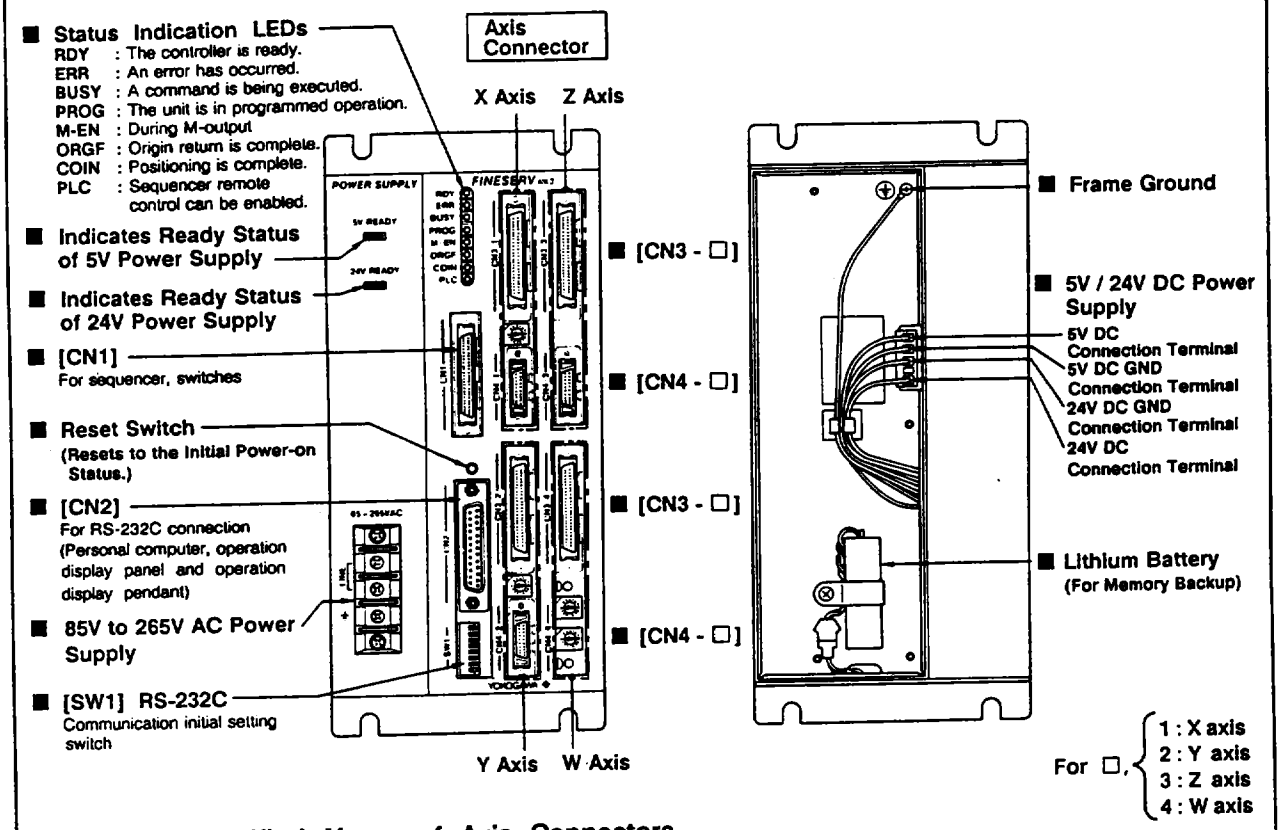


• Panel Cut Size : $112^{+1}_{+0.5}$ (4-3/8) × $219^{+1}_{+0.5}$ (8-5/8)

Figure 2.5 Dimensions of PC10031 / PC10041

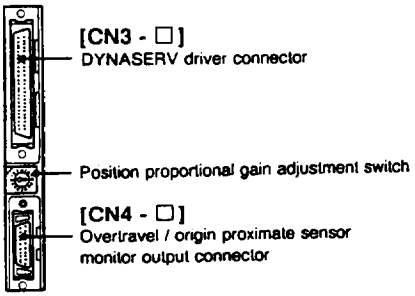
※ : For 3 axes type, this part is covered with a blind panel.

2.3 Name of Parts on the Main Unit

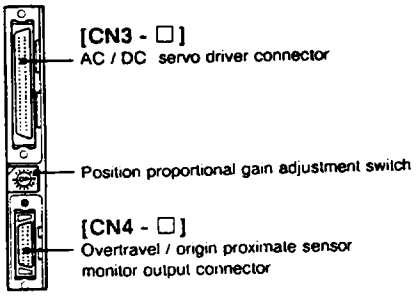


• Device Specified Name of Axis Connectors

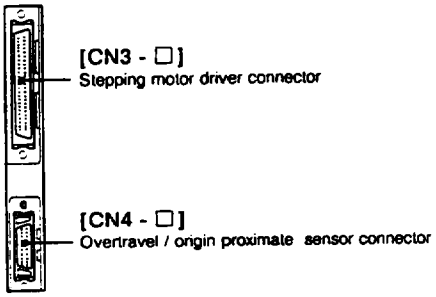
(1) DYNASERV (DD Motor)



(2) AC / DC Servo Motor



(3) Stepping Motor



(4) Inverter / Synchronization Encoder

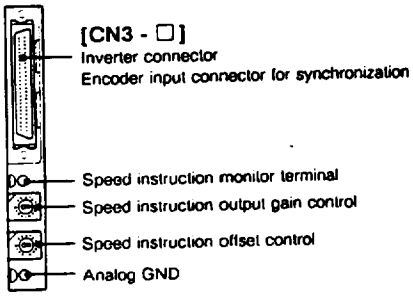


Figure 2.6 Name of Parts on the MKII Main Unit

3. INSTALLATION AND CONNECTIONS

3.1 Conditions of Installation

(1) Environmental Conditions

Table 3.1 Environmental Conditions

Item	Description and Installation Places
Range of Operating Temperature / Humidity	0°C to 50°C (32°F to 122°F) (Humidity : 20 to 90% RH, no condensation)
Range of Storage Temperature / Humidity	−20°C to 80°C (−4°F to 176°F) (Humidity : 20 to 90% RH, no condensation)
Vibration	Avoid installing the unit in the places subject to vibration. (Conform to JIS C-0911 IIB [0.5G / 9 to 150Hz])
Shock	Avoid installing the unit in the places subject to physical shock. (Conform to JIS C-0912 [10G / 5ms])
Atmospheric Conditions	Avoid installing the unit in the places: outdoors, in direct sunlight, and where subject to corrosive gas, explosive gas, steam, dust, polishing liquid, and metal powders. (JEM 1103 grade C)

(2) Power Supply and Grounding

Table 3.2 Power Supply and Grounding

Item		PC100 □0 / without Power Supply	PC100 □1 / with Power Supply
Power Supply	Voltage	5V DC±3%, 24V DC±5%	85V AC to 265V AC
	Current	5V DC : 3A / 1 axis, 2 axes 5A / 3 axes, 4 axes 24V DC : 0.7A / 1 axis, 2 axes 0.7A / 3 axes, 4 axes	30W / 1 axis, 2 axes 40W / 3 axes, 4 axes
Grounding		Ground	

(3) Mounting in the Rack

As shown in Figure 3.1, by changing the position of the metal fixture, either the back or the front of the controller can be attached to the rack. However, do not attach the controller by side. Have a space around the top and bottom of the controller in order to allow adequate air circulation. When installing, never drop screws or similar materials into the controller through the air vents.

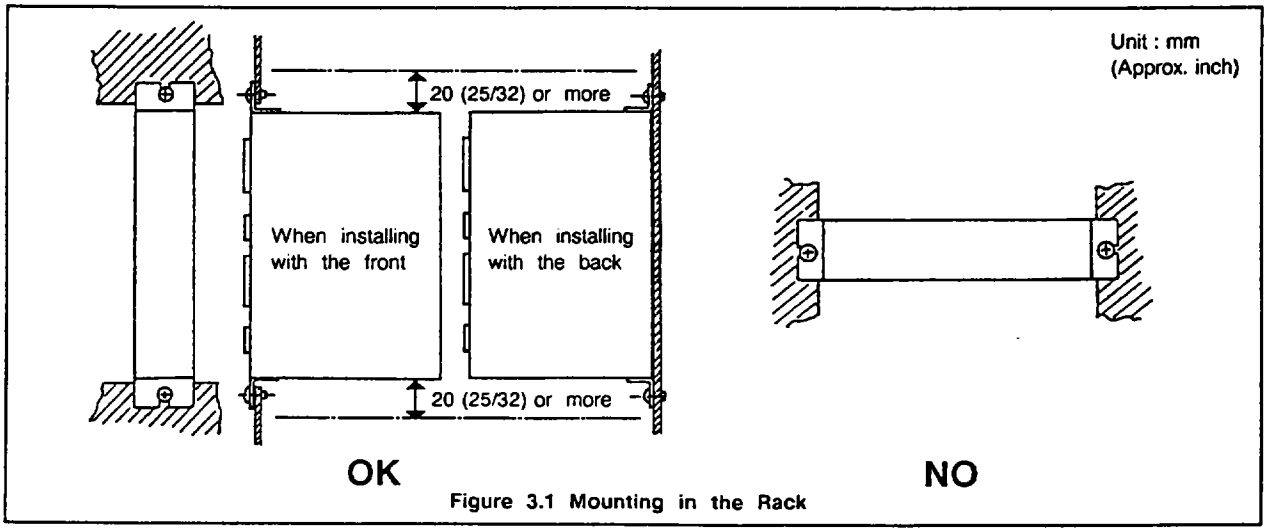


Figure 3.1 Mounting in the Rack

3.2 Connection Block Diagram

See the figure of top half of Figure 3.2 for power connection of the built-in power supply type series: PC10011 (1 axis control / with power supply); PC10021 (2 axes control / with power supply); PC10031 (3 axes control / with power supply); PC10041 (4 axes control / with power supply). See the figure of bottom half of Figure 3.2 for power connection of the separate power supply type series: PC10010 (1 axis control / without power supply); PC10020 (2 axes control / without power supply); PC10030 (3 axes control / without power supply); PC10040 (4 axes control / without power supply).

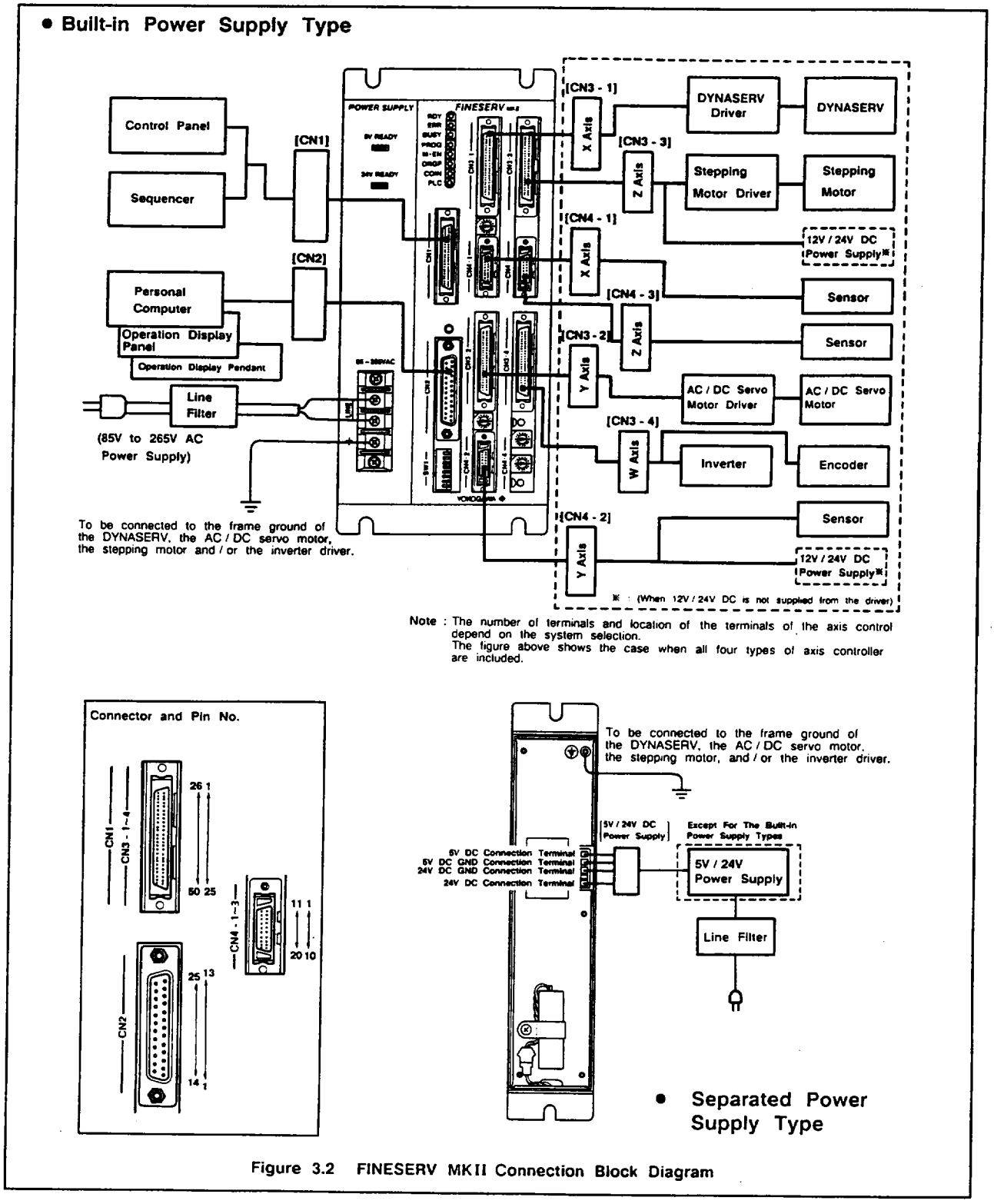


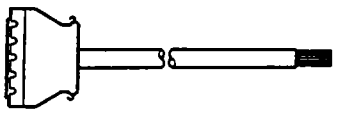
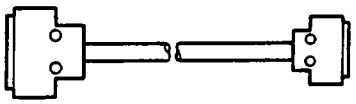
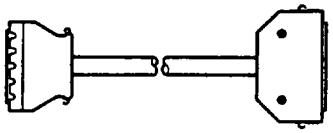
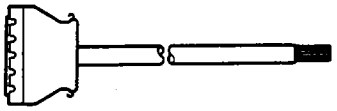
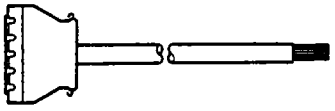
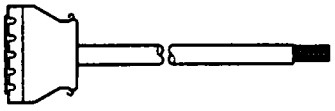
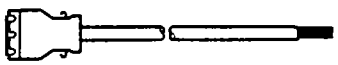
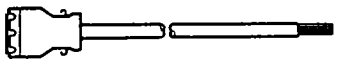
Figure 3.2 FINESERV MKII Connection Block Diagram

3.3 Connection Cables, Connectors and Covers

(1) Connection Cables

Connection cables for FINESERV MKII PC1 Series are sold separately (optional).

Table 3.3 Connection Cables

Name of Connector	Unit to be Connected	Product Type of Cable	Cable Connection Process ← To FINESERV / To the device to be connected →	Max. Length (m) (inch)	Remarks
CN1	Sequencer and Switches	CP0211S -□□□		3 (118-1/8)	A stopper (caulking) is attached to the connector to avoid reverse insertion of the connector. Caulking 5, A
CN2	Operation Display Panel	CP2221S -030 / -050		3 / 5 (118-1/8) / (196-7/8)	Two different cables in length (3m (118-1/8inch) and 5m (196-7/8inch)) are available for operation display pendant connection.
CN3-□	DYNASERV Driver	CP0307S -□□□		3 (118-1/8)	A stopper (caulking) is attached to the connector to avoid reverse insertion of the connector. Caulking 4, B
	Encoder or Inverter Synchronization	CP0212S -□□□		3 (118-1/8)	〃 Caulking 3, C
	AC / DC Servo Driver	CP0213S -□□□		3 (118-1/8)	〃 Caulking 2, D
	Stepping Motor Driver	CP0214S -□□□		3 (118-1/8)	〃 Caulking 1, E
CN4-□	Origin Sensor (DYNASERV)	CP0216S -□□□			〃 Caulking 3, A
	Origin Sensor (AC / DC Servo Moter)	CP0217S -□□□			〃 Caulking 2, B
	Origin Sensor (Stepping Motor)	CP0218S -□□□			〃 Caulking 1, C

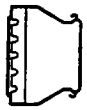
Note: 1. Length of the cable is indicated in 10cm (3-15/16 inch) unit with the last 3-digits of the product name.
<Example> - 005 : 50cm (19-11/16inch), - 030 : 300cm = 3m (118-1/8inch)

2. Use the cable supplied with the personal computer to make connections to the personal computer. (See section "3.5 RS-232C Connection through CN2".)

(2) Connectors and Covers

To assemble the desired cable, the connector and the cover listed in the Table 3.4 can be purchased from Yokogawa Precision Corporation. Each of them has a stopper (caulking) to avoid reverse insertion. (See * mark in the Table 3.4.)

Table 3.4 Connectors and Covers

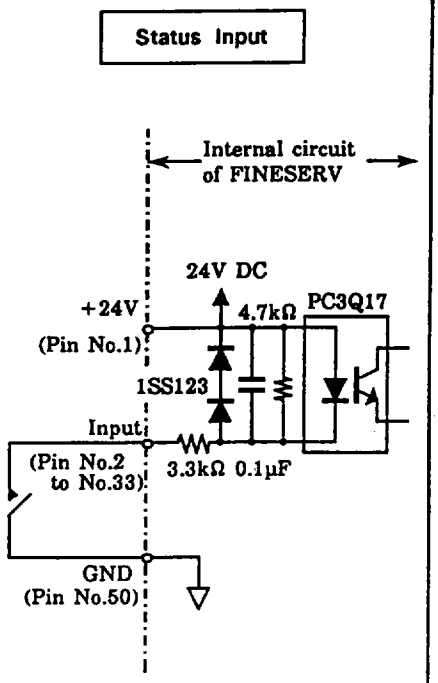
Name of Connector	Product Name	Description		Model Name by Manufacturer	Figure
CN1	R7304TB	Connector		FCN-231J050-G/E (Fujitsu Ltd.)	
	R7304VV	Cover*		FCN-230C050-D /E (Fujitsu Ltd.)	
CN3 - □	R7304TB	Connector		FCN-231J050-G/E (Fujitsu Ltd.)	
	R7304VW	Cover*	For DYNASERV	FCN-230C050-D/E (Fujitsu Ltd.)	
	R7304VY		For AC/DC servo motor		
	R7304VZ		For stepping motor		
	R7304VX		For inverter		
CN4 - □	R7304TA	Connector		FCN-230C050-G/E (Fujitsu Ltd.)	
	R7304VS	Cover*	For DYNASERV	FCN-230C020-D/E (Fujitsu Ltd.)	
	R7304VT		For AC/DC servo motor		
	R7304VU		For stepping motor		

Use a twisted pair shielded cable whose cross sectional area of each conductor is 0.2mm² or more (AWG-26). For the power supply cable, use a cable whose cross sectional area of each conductor is 0.75mm² or more.

3.4 Parallel Transmission Connection through CN1 (Input and Output of Sequencer and Switches)

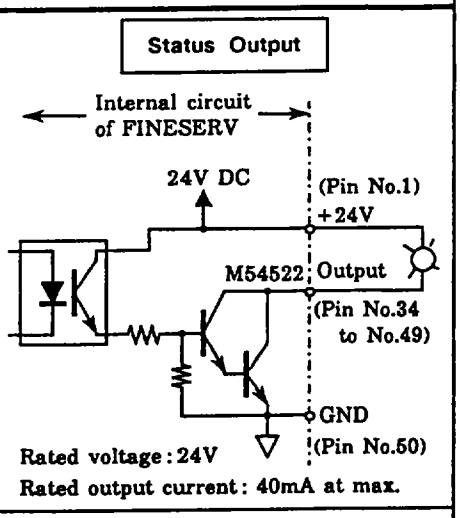
(1/2)

Pin No.	Name of Signal	Function / Meaning	Type	Electrical Connection Specification
1	+24V	Power supply +		
2	EMG ○	Emergency stop (when ON, decelerates and stops, and servo is OFF.)	Status input	
3	SERVO ON/OFF ○	Servo ON/OFF (for all axes) (servo is ON when ON.)		
4	START	Start		
5	STOP	Stop		
6	MODE 0	Mode change (4-bit binary) 0: Origin return 1: Programmed operation (AUTO) 2: Programmed operation (STEP) 3: Programmed operation (CONT) 4: Jog 8: Origin calibration 10: MDI (Manual Data Input)		
7	MODE 1			
8	MODE 2			
9	MODE 3			
10	RESET ○	Controller reset (reset after emergency stop)		
11	INTERLOCK ○	Pause		
12	M ANSWER ○	M completion		
13	JOG HIGH/LOW	Jog speed selection (ON: high speed)		
14	X AXIS SELECT	X axis selection		
15	Y AXIS SELECT	Y axis selection		
16	Z AXIS SELECT	Z axis selection		
17	W AXIS SELECT	W axis selection		
18	JOG +X	X axis jog + (ON: + direction of X axis jog)		
19	JOG −X	X axis jog − (ON: − direction of X axis jog)		
20	JOG +Y	Y axis jog +		
21	JOG −Y	Y axis jog −		
22	JOG +Z	Z axis jog +		
23	JOG −Z	Z axis jog −		
24	JOG +W	W axis jog +		
25	JOG −W	W axis jog −		
26	DATA 0	(Example) 96 → OFF		
27	DATA 1	ON		
28	DATA 2	ON		
29	DATA 3	OFF		
30	DATA 4	Program selection (BCD 2-digits) 1 to 99 → ON		
31	DATA 5	OFF		
32	DATA 6	OFF		
33	DATA 7	ON		



Input current ON : 7mA
OFF: 0.1mA at max.

(2/2)

Pin No.	Name of Signal		Function / Meaning	Type	Electrical Connection Specification
34	S RDY	○	Servo ready (for all axes)	Status output	
35	ERROR STOP	○	Error stop condition		
36	BUSY	○	Under operation		
37	UNDER PROGRAM	○	Under programmed operation		
38	PLC	○	Under parallel transmission mode		
39	BATTERY ALARM	○	Alarm for battery replacement		
40	ORG FINISH	○	Origin return completion (for all axes)		
41	M ENABLE	○	M output, effective		
42	M OUT 0	○	M output data (BCD 2-digits)		
43	M OUT 1	○			
44	M OUT 2	○			
45	M OUT 3	○			
46	M OUT 4	○			
47	M OUT 5	○			
48	M OUT 6	○			
49	M OUT 7	○			
50	GND				

Note: 1. All signals are positive logic.
2. All signals are ON status when current flows through the photocoupler.
3. The signal with a circle "○" is effective in RS-232C mode.

Caution : NEVER CONNECT 24V DC power supply to pin No.1 and No.50. Power is supplied internally.

3.5 RS-232C Connection through CN2

(Connection to the Personal Computer, Operation Display Panel and Operation Display Pendant)

The operation display panel and the operation display pendant can be connected using the cable supplied by Yokogawa Precision Corporation. Connection to a personal computer, IBM PC AT or compatibles, is shown below.

(1) Personal Computer Connection Specification

- **Transmission System** : RS-232C asynchronous
- **Baud Rate** : Selectable (9600, 4800, 2400, 1200 [BPS])
- **Stop Bit** : 1 bit (Fixed)
- **Data Length** : 8 bit (Fixed)
- **Parity** : Selectable (Odd, even, non parity, parity)
- **Termination** : CR or CR/LF
- **Communication System** : Non protocol

(2) Connections

Note: Pins 17 – 18 and pins 24 – 25 of <CN2> are shorted inside the connector. Do not connect any other pins not shown in Figure 3.3 when connecting a personal computer.

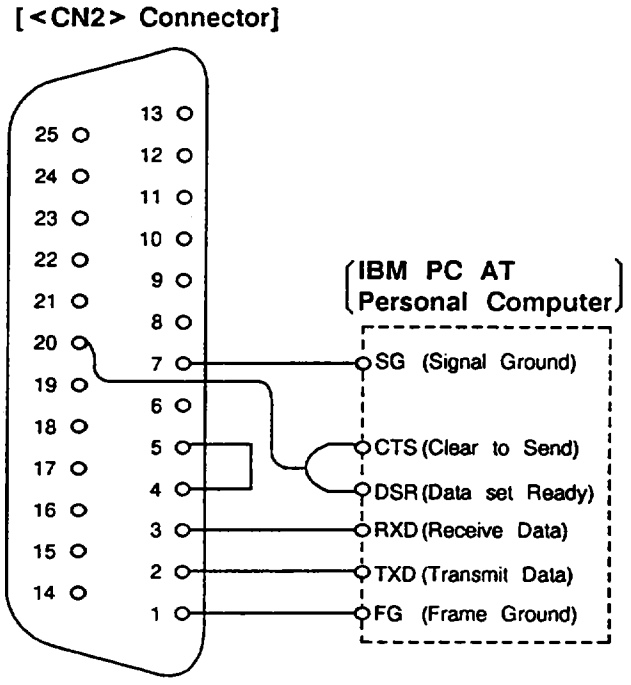


Figure 3.3 Connection Diagram

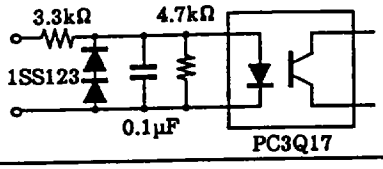
3.6 DYNASERV Connection

(1) DYNASERV Driver Connection through CN3 - □

The DYNASERV driver can be directly connected to the FINESERV using the dedicated connection cable. Refer to the table below for interface (pin assignment and signals) to the DYNASERV.

Pin No.	Name of Signal	I/O	Description	
			DM Series	DR Series
1	FN 3	Input	Compliance setting 3	Same as the left column
2	Vcc	Input	Power supply for signals	Same as the left column
3	FN 2	Input	Compliance setting 2	Same as the left column
4	Vcc	Input	Power supply for signals	Same as the left column
5	FN 1	Input	Compliance setting 1	Same as the left column
6	Vcc	Input	Power supply for signals	Same as the left column
7	FN 0	Input	Compliance setting 0	Same as the left column
8	Vcc	Input	Power supply for signals	Same as the left column
9	POSW 1	Input	Setting of positioning completion pulse width 1	Same as the left column
10	Vcc	Input	Power supply for signals	Same as the left column
11	POSW 0	Input	Setting of positioning completion pulse width 0	Same as the left column
12	Vcc	Input	Power supply for signals	Same as the left column
13	A+/U+	Output	A phase/UP pulse output signal+	Same as the left column
14	A−/U−	Output	A phase/UP pulse output signal−	Same as the left column
15	RDY	Output	Servo ready output signal	Same as the left column
16	DGND	Output	Digital ground	Same as the left column
17	VELMON	Output	Speed monitor output signal	Same as the left column
18	AGND	Output	Analog ground	Same as the left column
19	SIGN −	Input	Motor rotating direction instruction−	Same as the left column
20	SIGN +	Input	Motor rotating direction instruction+	Same as the left column
21	IRST	Input	Integration capacitor resetting	Same as the left column
22	Vcc	Input	Power supply for signals	Same as the left column
23	SRVON	Input	Servo start input signal	Same as the left column
24	Vcc	Input	Power supply for signals	Same as the left column
25	IACT/PACT	Input	Integral/proportional control switching	Same as the left column
26	Vcc	Input	Power supply for signals	Same as the left column
27	COIN	Output	Positioning completion	Same as the left column
28	DGND	Output	Digital ground	Same as the left column
29	B+/D+	Output	B phase/DOWN pulse output signal+	Same as the left column
30	B−/D−	Output	B phase/DOWN pulse output signal−	Same as the left column
31	TLIM/TFF	Input	Torque Limit, Torque feed forward	Same as the left column
32	AGND	Input	Analog ground (option)	Same as the left column
33	GAIN L	Input	Gain switching L	Same as the left column
34	Vcc	Input	Power supply for signals	Same as the left column
35	GAIN M	Input	Gain switching M	Same as the left column
36	Vcc	Input	Power supply for signals	Same as the left column
37	GAIN H	Input	Gain switching H	Same as the left column
38	Vcc	Input	Power supply for signals	Same as the left column
39	RST	Input	CPU resetting	Same as the left column
40	Vcc	Input	Power supply for signals	Same as the left column
41	OVER	Output	Deviation counter overflow	Same as the left column
42	DGND	Output	Digital ground	Same as the left column
43	Z+	Output	Origin pulse+	Same as the left column
44	Z−	Output	Origin pulse−	Same as the left column
45	PULS+	Input	Position instruction pulse+	Same as the left column
46	PULS−	Input	Position instruction pulse−	Same as the left column
47	OVL	Output	Overload	Same as the left column
48	DGND	Output	Digital ground	Same as the left column
49	VIN	Input	Speed instruction	Same as the left column
50	AGND	Input	Analog ground	Same as the left column

(2) DYNASERV Connection, and Origin Sensor and Oscilloscope Connection through CN4 - □

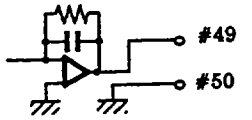
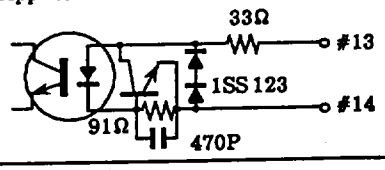
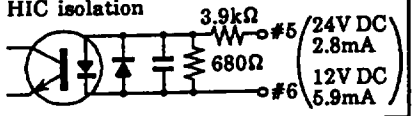
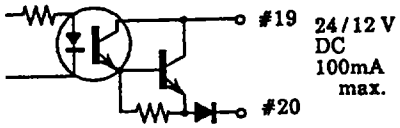
Purpose	Pin No.	Name of Signal	Function / Meaning	Type	Electrical Connection Specification
Origin Sensor Connection	3	ORGDG+ ※	Origin proximate sensor signal +	Status input	Input current: 7mA when ON and 0.1mA when OFF at max. 
	4	ORGDG− ※	Origin proximate sensor signal −		
	5	OTU+ ※	Hard limit signal + in CW direction		
	6	OTU− ※	Hard limit signal − in CW direction		
	7	OTD+ ※	Hard limit signal + in CCW direction		
	8	OTD− ※	Hard limit signal − in CCW direction		
Only for Monitoring on Oscilloscope	11	AGND	Analog ground	Analog output	±8V Never use these for any devices other than an oscilloscope.
	12	VIN	Speed instruction output monitoring		
	13	AGND	Analog ground		
	14	AOUT	Analog COIN output monitoring	Status output	
	15	GND	Digital ground		
	16	ERST ※	Axis board reset signal monitoring		
	17	GND	Digital ground		
	18	SRVRDY ※	Servo ready signal monitoring		
	19	GND	Digital ground		
	20	COIN	COIN signal monitoring		

※:Negative logic

Note : Use contact B for ORGDG, OTU and OTD signals.

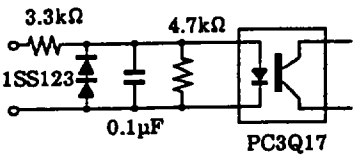
3.7 AC / DC Servo Motor Connection

(1) AC / DC Servo Driver Connection through CN3 - □

Pin No.	Name of Signal	Type	Function / Meaning	Electrical Connection Specification
49 50	VIN AGND	Analog output	Speed instruction voltage (±10V)	
13 14	A + A –	Pulse input	Encoder signal A phase pulse	Applicable to a differential driver 
29 30	B + B –		Encoder signal B phase pulse	
43 44	Z + Z –		Encoder signal Z phase pulse	
3 4	SRDY +	Status input	Servo ready	HIC isolation 
7 8	OVL + OVL –		Overload	
17 18	RST + ※ RST – ※	Status output	CPU reset	HIC isolation 
21 22	SRVON + ※ SRVON – ※		Servo ON	
23 24	PACT + PACT –		Integral/proportional control switching (ON: Proportional)	
1, 11, 15, 25, 27, 39, 41	COM +		Power supply for interface (24V/12V DC)	See "(3) Power Supply for I/F" on page 3 - 12.
2, 12, 16, 26, 28, 40, 42	COM –		Power supply GND for interface	

※ : Negative logic

(2) AC / DC Servo Moter Connection, and Origin Sensor and Oscilloscope Connection through CN4 - □

Pur-pose	Pin No.	Name of Signal	Function / Meaning	Type	Electrical Connection Specification
	1	COM +	Power supply for I/F		See "(3) Power Supply for I/F" on page 3 - 12.
	2	COM –	Power supply GND for I/F		
Origin Sensor Connection	3	ORGDG + ※	Origin proximate sensor signal +	Status input	Input current: 7mA when ON and 0.1mA when OFF at max. 
	4	ORGDG – ※	Origin proximate sensor signal –		
	5	OTU + ※	Hard limit signal + in CW direction		
	6	OTU – ※	Hard limit signal – in CW direction		
	7	OTD + ※	Hard limit signal + in CCW direction		
	8	OTD – ※	Hard limit signal – in CCW direction		
Only for Monitoring on Oscilloscope	11	AGND	Analog ground	Analog output	±10V Never use these for any devices other than an oscilloscope.
	12	VIN	Speed instruction output monitoring		
	13	AGND	Analog ground		
	14	AOUT	Analog COIN output monitoring		
	15	GND	Digital ground	Status output	
	16	ERST ※	Axis board reset signal monitoring		
	17	GND	Digital ground		
	18	SRVRDY ※	Servo ready signal monitoring		
	19	GND	Digital ground		
	20	COIN	COIN signal monitoring		

※ : Negative logic

Note : Use contact B for ORGDG, OTU and OTD signals.

(3) Power Supply for I / F

24V DC or 12V DC shall be externally applied to the status input. When the connected driver is not equipped with a power supply, supply power to pin 1 and 2 of [CN4 - □], and connect [COM+/−] of [CN3 - □] to [+/−] of the signal as shown in Figure 3.4.

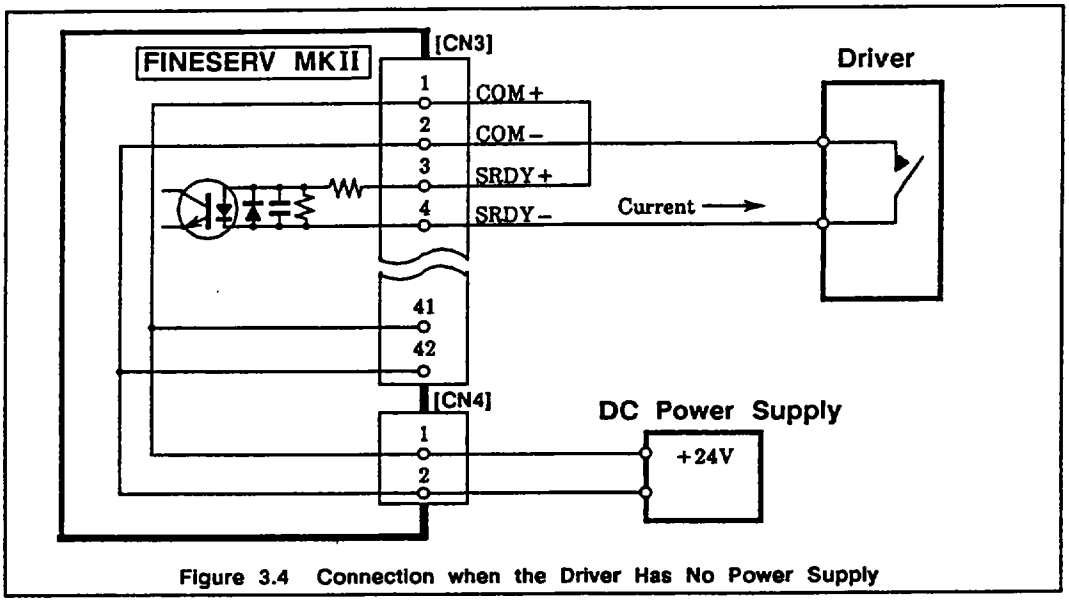


Figure 3.4 Connection when the Driver Has No Power Supply

When the connected driver is equipped with a power supply, connect it as shown in Figure 3.5.

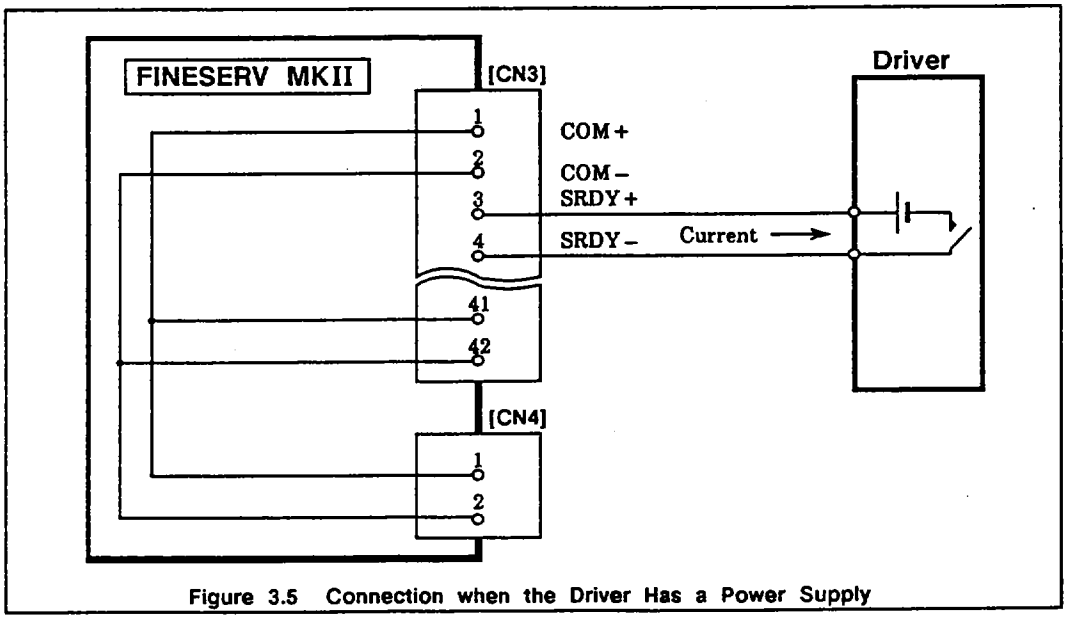


Figure 3.5 Connection when the Driver Has a Power Supply

3.8 Stepping Motor Connection

(1) Stepping Motor Driver Connection through CN3 - □

Pin No.	Name of Signal	Function / Meaning	Type	Electrical Connection Specification
1	UP	UP pulse	Pulse output	Open-collector (74LS06) +5V 5V DC/40mA max. #2 #1
3	DOWN	DOWN pulse		
15 16	RDY ※ GND	Driver ready	Status input	C-MOS logic 5V DC/5mA max. +5V 1kΩ #15 #16
27 28	TIM ※ GND	Drive timing signal		
47 48	OVL ※ GND	Overload		
23	H · OFF ※	Cut off of output current	Status output	The same as the above pulse output
43 44	Z+ Z−	Origin pulse input	Pulse input	Photocoupler for open-collector drive 24V DC/ON: 7mA/OFF: 0.1mA max. #43 3.3kΩ 4.7kΩ #44
2, 4, 12, 24	+5V	5V DC power supply output for I/F		
50	GND	Power supply GND for I/F		

※: Negative logic

(2) Stepping Motor Connection, and Origin Sensor and Oscilloscope Connection through CN4 - □

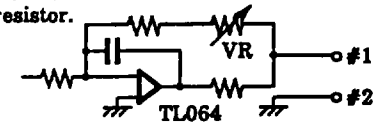
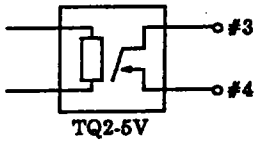
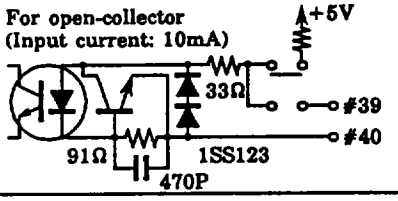
Purpose	Pin No.	Name of Signal	Function / Meaning	Type	Electrical Connection Specification
Origin Sensor Connection	3	ORGDG+ ※	Origin proximate sensor signal +	Status input	Input current: 7mA when ON and 0.1mA when OFF at max. 3.3kΩ 0.1μF 1SS123 4.7kΩ PC3Q17
	4	ORGDG− ※	Origin proximate sensor signal −		
	5	OTU+ ※	Hard limit signal + in CW direction		
	6	OTU− ※	Hard limit signal − in CW direction		
	7	OTD+ ※	Hard limit signal + in CCW direction		
	8	OTD− ※	Hard limit signal − in CCW direction		
Only for Monitoring on Oscilloscope	11	AGND	Analog ground	Analog output	±10V Never use these for any devices other than an oscilloscope.
	12	VIN	Speed instruction output monitoring		
	15	GND	Digital ground	Status output	
	16	ERST ※	Axis board reset signal monitoring		
	17	GND	Digital ground		
	18	SRVRDY ※	Servo ready signal monitoring		
	19	GND	Digital ground		
	20	COIN	COIN signal monitoring		

※: Negative logic

Note: Use contact B for ORGDG, OTU and OTD signals.

3.9 Inverter Connection

(1) Inverter and Encoder Connection through CN3 - □

Purpose	Pin No.	Name of Signal	Function / Meaning	Type	Electrical Connection Specification
Inverter	1 2	AOUT AGND	Speed instruction output	Analog output	Output voltage: 0 to 10V DC at max. The maximum value can be adjusted from 6 to 10V DC with a variable resistor. 
	3 4	SRVOFF+ SRVOFF−	Speed instruction, disable	Contact output	Contact rating: 30V DC/1A 
	5 6	CCW+ CCW−	CCW motor rotating direction		
	7 8	CW+ CW−	CW motor rotating direction		
Encoder	39 40	Z+ Z−	Origin pulse	Pulse input	For open-collector (Input current: 10mA) 
	41 42	A+ A−	A phase input		
	43 44	B+ B−	B phase input		
	45 46	+5V OUT	5V DC power supply for the encoder		
	49 50	GND OUT			

4. OPERATION

This chapter gives detailed descriptions you need for operation of the FINESERV MKII. We suggest that you read this chapter thoroughly in order to use and operate the FINESERV MKII with ease.

CONTENTS

HOW TO READ THIS CHAPTER

This chapter explains how to operate the controller. First, the outline of this chapter will be described. In the text, "FINESERV MKII Dedicated Controller PC1 Series" is called "controller".

This chapter is divided into five sections as follows:

(1) Preparation
(2) Operation for Installation
(3) Operation
(4) Commands
(5) Programming Language

Following lists the contents described in each section.

4.1 Preparation

This section shows connections to the <CN2> connector terminals from the personal computer, the operation display panel, or the operation display pendant using an RS-232C cable. This is required for initial setting of the RS-232C communication interface. A communication program shall be created on the personal computer using the BASIC language, and the created program is to be used to control the controller from the personal computer.

This section also shows connections to the <CN1> connector terminals from the sequencer or the switches using cables. The controller is to be controlled from the sequencer or the switches through the cables.

4.2 Operation for Installation

This section describes operations required for installation. This is to be carried out on the personal computer, or by the peripherals such as the operation display panel (optional unit: PA100AP) or the operation display pendant (also optional unit: PC000AT).

Before carrying out usual operations, the parameters and the programs required for operations shall be registered on the controller. This section describes the functions, operating procedures, notes, and so on concerning the items to be registered.

4.3 Operation

This section explains operations required for using the controller. Operations are carried out using the sequencer or the switches.

This section includes the functions, operating procedures, notes and so on concerning daily operations.

4.4 Commands

This section shows the commands which are available on the controller. This section includes functions of the commands, and the forms and how to use the commands.

4.5 Programming Language

This section explains the programming language which is available on the controller. The programming language used on the controller conforms to NC language. This section includes functions, forms, and related descriptions of the programming language.

Notes on Symbols in the Text

■ Operation: Means the operations you must carry out. Carefully follow the instructions.

◇ Change : Means a change of the controller or a change by the controller. Check the operations changed by the controller.

! Caution : Means something you must pay attention to before, during, or after carrying out operations.

Reference : Means the place to be referred to when additional information is necessary.

Memo : Means supplementary descriptions. Read it if you think it is necessary.

4.1 Preparation

This section shows connections to the <CN2> connector terminals from the personal computer, operation display panel, or operation display pendant using an RS-232C cable. This is required for initial setting of RS-232C communication interface. A communication program shall be created on the personal computer using the BASIC language, and the created program is to be used to control the controller from the personal computer.

This section also shows connections to the <CN1> connector terminals from the sequencer or the switches using cables. The controller is to be controlled from the sequencer or the switches through the cables.

See "3. Installation and Connections" for connections of the cables.

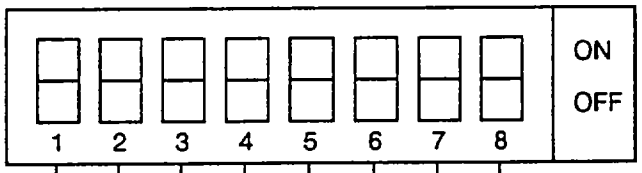
RS-232C Communication Specifications

- **Transmission System** : RS-232C asynchronous
- **Baud Rate** : Selectable (9600 / 4800 / 2400 / 1200 [BPS])
- **Stop Bit** : 1bit (Fixed)
- **Data Length** : 8bits (Fixed)
- **Parity** : Selectable (Odd / even / non-parity / parity)
- **Termination** : CR or CR / LF
- **Communication System** : Non protocol

Setting of <SW1>

<SW1> Carries out initial setting of the RS-232C communications and the related operation of the system.

<SW1> Is located on the front panel of the controller.



Baud Rate	1200	2400	4800	9600
8	ON	ON	OFF	OFF
7	ON	OFF	ON	OFF

Baud rate

Parity ON : Odd / OFF : Even

Parity on / off ON : Without parity / OFF : With parity

Termination ON : CR / OFF : CR, LF

<CN2> ON : Personal computer or not connected
OFF : Operation display panel or operation display pendant

Reserved (This must be set to ON.)

Initialize ON : All settings are set back to the factory-set condition when power is turned on or when resetting the controller.
OFF : Normal position.

Example of a Program Written in BASIC Using a IBM PC Personal Computer

```
10  '*****************************************************************
20  '*     FINESERV MKII OPERATING SAMPLE PROGRAM FOR IBM PC-AT      *
30  '*             9/11/1991  BY YOKOGAWA                            *
40  '*****************************************************************
50  '
60  OPEN "COM:2400, N, 8, 1, LF" AS #1   'Makes available the RS-232C port.
70  WHILE INKEY$ <> "" : WEND            'Clears the key buffer.
80  INPUT "              ··";CMND$       'Inputs a command.
90  PRINT #1,CMND$                       'Sends the command to FINESERV MKII.
100 IF LOC(1) <> 0 THEN 140              'Jumps to 140 when FINESERV answers.
110 ANYKEY$ = INKEY$                     '
120 IF ANYKEY$ <> " " THEN 70            'Jumps to 70 if space key is pressed.
130 GOTO 100                             '
140 LINE INPUT#1,ANS$                    'Reads data from FINESERV MKII.
150 PRINT ANS$                           'Displays the data on the screen.
160 FOR I = 1 TO 200 : NEXT              'Wait
170 IF LOC(1) <> 0 THEN GOTO 140         'Reads data until the buffer is empty.
180 GOTO 70                              '
190 CLOSE                                'Closes RS-232C port.
200 END
```

[Example]

```
RUN
     ·· @:40                      :  Calls the setting value of @40.
R02  mtr_type:00H:00H:00H:00H     :  These are the status conditions of 1st axis through 4th
                                     axis.
     ·· @40::1                    :  Sets the motor type of 2nd axis to 1.
R00  ready
     ·· @:40                      :  Recalls the setting value of @40.
R02  mtr_type:00H:01H:00H:00H     :  This shows that the motor type of 2nd axis is set to 1.
     ··
```

Return Data Format

Return data format is the message format returned from the controller to the personal computer, operation display panel, or the operation display pendant.
The return data includes return data messages and error messages.

(1) Return Data

R00_ready CR(,LF)

R01_ [A message of 8 characters] : ○○H CR(,LF)

R02_ [A message of 8 characters] : ○○H:○○H:○○H:○○H CR(,LF)

R04_ [A message of 8 characters] : ○○○○○○○○:○○○○○○○○

: ○○○○○○○○:○○○○○○○○ CR,LF

R09_DI/O_in_○○_○○_○○_○○_out_○○_○○ CR(,LF)

[!] ○○H of @R01 and R02 is displayed in 1-byte hexadecimal notation.
○○○○○○○○ of R04 is displayed in 4-byte hexadecimal notation.
"H" which means hexadecimal notation does not follow the data of R04 and R09.

(2) Error Messages

E○○_ [A message of 8 characters] CR(,LF)

└── Error number

4.2 Operation for Installation

Operation for installation is carried out in the RS-232C mode.

Before carrying out usual operations, the parameters and the programs required for operations shall be registered on the controller. This section describes the functions, operating procedures, notes, and so on concerning the items to be registered.

How to Read "Operation for Installation" Section.

- **Function** Explains functions simply.
- **Preparation** Explains preparation required before using this function.
- **Operation** Describes the use of this function in detail.
  Follow the command description format below when using and entering the commands on the following pages.
  (1) Items of @ and numbers shall be entered as they are. Do not make them short.
  (2) Items to be entered in the brackets "< >" are numbers of integer.
  The range that can be entered differs from command to command, and it is shown with parentheses "( )."
  (3) Items in the brackets "[ ]" can be omitted. When omitted, the default values (the values registered primarily to the controller) or the values which was designated last are used.
  There are more than one parameter separated with a colon ":". These can be omitted in the following cases.
  ① When omitting the parameters after a certain parameter. The colon can also be omitted in this case.
  ② When omitting the parameters in between parameters and designating following parameter, the colons before the omitted parameters cannot be omitted.
- **Timing** This shows the timing chart of operations with these functions.
- **Notes** This explains notes on the use with these functions.

RS-232C Mode — Transmission Mode

● **Function** In this mode, the controller is controlled from the personal computer, the operation display panel, or the operation display pendant.

● **Preparation** Is preparation for RS-232C communication finished? "4.1 Preparation"

● **Operation** In the RS-232C mode, registration of parameters and programs to the controller is carried out from the personal computer, the operation display panel, or the operation display pendant. Operation for installation is carried out in the RS-232C mode.

① **Transmission Mode**

■ Set the transmission mode to the RS-232C mode. Set the 1st parameter of the transmission mode command (@5) to 0.

@5 : 0

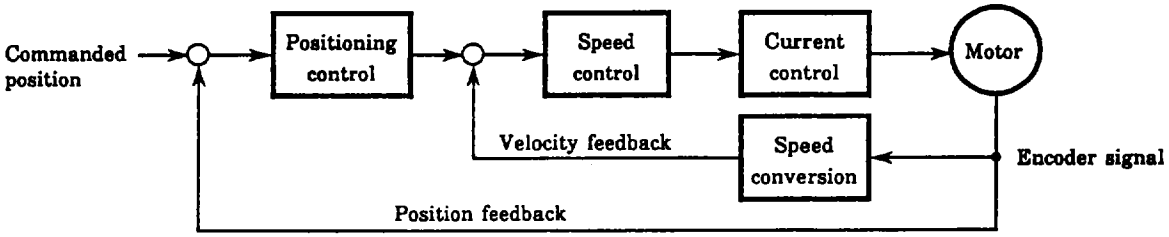
◇ The PLC lamp on the front panel of the controller goes out.

Memo Besides the RS-232C mode, parallel transmission mode is available as the transmission mode. Usual operations are carried out in the parallel transmission mode. "4.3 Operation / Parallel Transmission Mode."

Control System Identification

This controller can control the DYNASERV, AC / DC servo motor, stepping motor, and inverter. The DYNASERV and the AC / DC servo motor are controlled in a servo system while the stepping motor and the inverter are controlled in a non-servo system. Following describes the control functions and setting of these devices.

- **Servo System :** Controlled in a closed-loop circuit using the signal fed back from the motor.



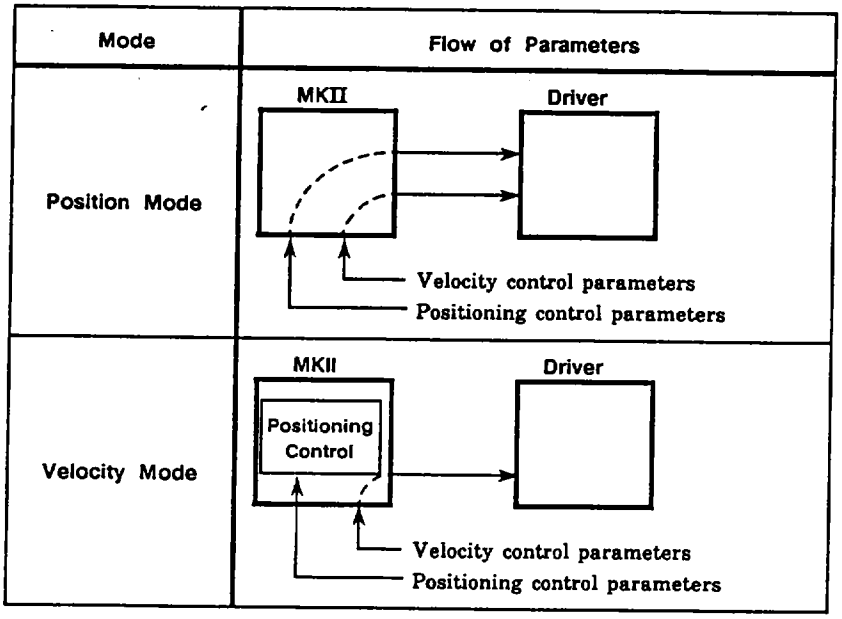
- **Non-Servo System :** Controlled in an open-loop circuit without feedback.
  Parameter setting such as gain for the positioning and speed controller is required for servo system while non-servo system doesn't require.

System	Motor Type	Control Parameter
Servo System	DYNASERV and AC/DC servo motor	To be set
Non-Servo System	Stepping motor and induction motor (inverter)	Not to be set

The servo system is grouped in two types of modes, the position mode and the velocity mode, depending on whether the positioning controller part above is controlled from the driver or from thr controller. The positioning controller driver shall be connected in the position mode, and the speed controller driver or the speed controller part (interface must be available in this case) of the positioning controller driver shall be connected in the velocity mode.

Mode		Control Function Block Diagram	Device to Be Connected
Servo System	Position Mode	MKII: (blank) — Driver: Positioning Control, Velocity Control, Current control	Positioning controller driver
	Velocity Mode	MKII: Positioning Control — Driver: Velocity Control, Current control	• Positioning controller driver • Speed controller part of the positioning controller driver

There are two types of control parameters to be set for the controller. One is the parameter of positioning control (positioning control parameter) and the other is that of speed control (velocity control parameter). In the positioning instruction mode, all control parameters are used for remote setting to the driver. In the velocity mode, speed control parameters are used for remote setting, and positioning control parameters are used for control operations in the controller.



The standard specifications of the controller are as follows:

- The DYNASERV is set in the position mode.
- The AC / DC servo motor is set in the velocity mode.

CAUTION

Some part of this instruction manual explains the combination of the use of the velocity mode for DYNASERV and the position mode for the AC / DC servo motor. However, operation of this combination is available only for those machines which are made so especially at the factory.

Position Mode and Velocity Mode Control Function

Positioning Control Parameter in the Position Mode

(1) COIN width switch

The COIN width switch is effective on the axis control of the DYNASERV, and is used to set remotely the positioning alignment check width of the DYNASERV driver.
The check width can be selected in 16 steps in the same way as for the switch setting of the DYNASERV driver.
For the remote setting with the COIN width switch, set the POSW switch of the DYNASERV driver to "0", "4", "8" or "C".

(2) COIN switch

The COIN switch enables or disables the positioning alignment check width with the (1) COIN width switch.

(3) fc remote switch

The fc remote switch is effective on the axes of the DYNASERV, and is used, instead of the switch of the driver, to set remotely the bandwidth of the positioning control loop of the DYNASERV driver. The setting value can be selected in 16 steps in the same way as for the switch setting of the switch of the DYNASERV driver.
For the remote setting with the fc remote switch, set the fc switch of the DYNASERV driver to "0".

Speed Control Parameter in the Position Mode

(1) Integral / proportional control

This switch sets remotely on and off of the interface PACT contact to the driver.
Refer to the instruction manual of the driver for the operation logic and the meaning.

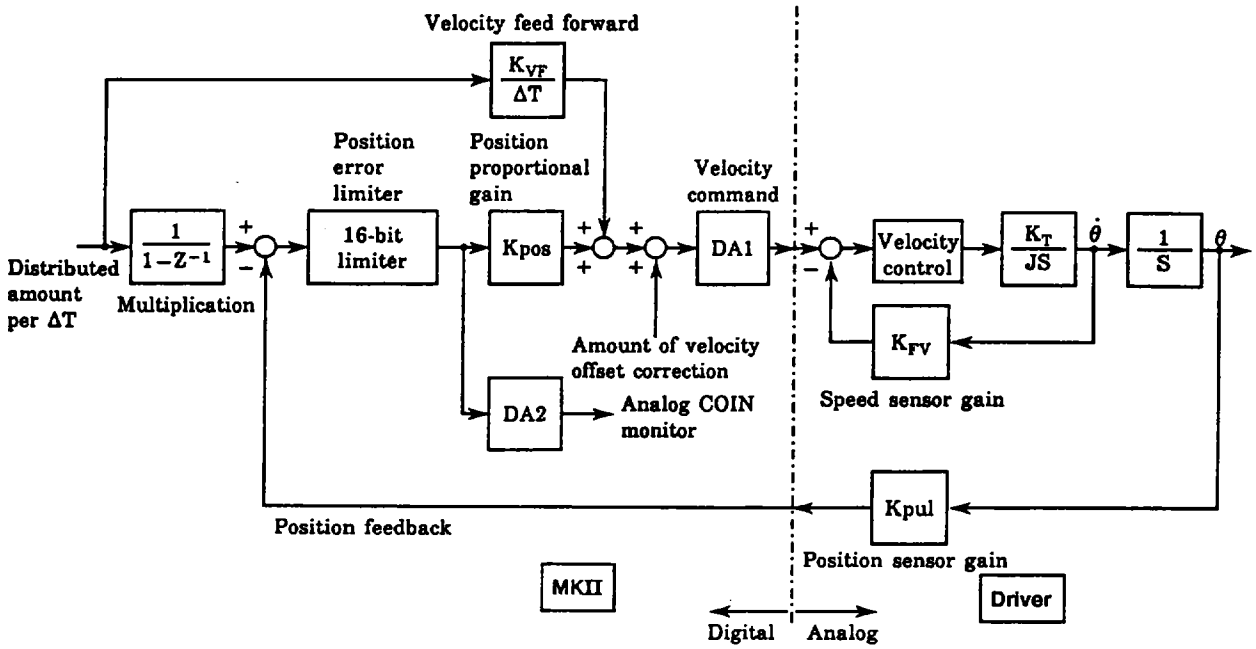
(2) DC gain

This is effective on the axis control of the DYNASERV, and is used, instead of the switch of the driver, to set remotely the bandwidth of the positioning control loop of the P (proportional) operation. When affected with a change of inertial force, this shall be set for the best value at minimum inertial force.

(3) Integral reset switch

This switch sets remotely on and off of the interface IRST contact of the driver.
Refer to the instruction manual of the driver for the operation logic and meaning.

Positioning Control Parameter in the Velocity Mode



(1) Position proportional gain control

Position proportional gain can be controlled with the position proportional gain control and the rotary switch at the front of the axis board.

- First, set coarse position proportional gain with the position proportional gain control commands (@28). The values of the position proportional gain switch setting and the corresponding gains are as follows:

Setting value	Coarse gain
0	×0.51
1	×1
2	×2
3	×4
4	×8
5	×16
6	×32
7	×64

- Then, adjust the gain finely with the rotary switch at the front of the axis board. This adjustment is available either when the axis of the motor does not rotate or when the axis of the motor is rotating. The range of setting is in 16 steps, from 1/16 through 16/16.

(2) Input sensitivity voltage

The input sensitivity voltage (mV) means the voltage of 1rps. Set the appropriate value for this parameter because it is used when controlling axes using a feedback loop.

(3) Velocity feed forward

The amount of velocity feed forward can be adjusted with the velocity feed forward ratio (%). The range of velocity feed forward setting is from 0 to 120 (%).

! The desired amount of velocity feed forward cannot be obtained when the appropriate value has not been set for resolution of rotation of the motor and the input sensitivity voltage.

(4) Velocity offset correction switch

The undesired analog voltage generated outside the speed loop is eliminated with the velocity offset correction function. The velocity offset correction switch enables or disables the velocity offset correction function. When enabled, speed correction is carried out when rotation instruction is not given to the motor.

! The rotation of the axis of the motor may become a little fluctuated when this switch is set to enabled position (velocity offset correction function is enabled), compared with when it is set to disabled position.

(5) Internal COIN width

The internal COIN width (p) means the COIN width for position feedback control from the controller. After a rotation instruction is given to the motor, positioning alignment check is carried out by comparing the internal COIN width with the positioning deviation of the position feedback control.

(6) COIN switch

The COIN switch enables or disables the positioning alignment check which has been set by the procedures of under (5) internal COIN width setting.

Velocity Control Parameter in the Velocity Mode

(1) Integral / proportional operation control

This switch sets remotely on and off of the interface PACT contact to the driver.
Refer to the instruction manual of the driver for the operation logic or the meaning.

(2) Integral reset switch

This switch sets remotely on and off of the interface IRST contact to the driver.
Refer to the instruction manual of the driver for the operation logic or the meaning.

Memo Following two monitoring functions are available for position feedback control of the controller.

- Monitoring of over deviation of position
  An error occurs when the amount of position deviation exceeds ±32768 pulses.
- Monitoring of over correction of speed
  An error occurs when the value of speed correction exceeds ±127mV for the DYNASERV and ±159mV for the AC / DC servo motor, with the speed correction function enabled.

If an error occurs by monitoring of over deviation of positioning or over correction of speed, an error message is displayed and the servo control to the corresponding axis will be turned off.

Memo Conditions of alignment can be observed on an oscilloscope while position feedback control is carried out with the controller. The lower 8 bits of the amount of position deviation is output with plus or minus sign.
However, if the amount includes the deviation of more than 8 bits (256 pulses or more), the output does not correspond to the amount or it has no meaning.

- Output range : "1.4 Specifications (3) Connection Specifications"
- Output rate : 62.5mV per pulse
- Output pin : AOUT output terminal (Analog COIN output monitor)

List of Servo Settings

	Item	Command Code	Related G Code	DYNA-SERV Positioning Instruction	AC / DC Servo Speed Instruction	Stepping Motor	Inverter	Reference	
								DYNA-SERV Velocity Command	AC / DC Servo Position Command
Common Parameter	Type of the Motor	@40	–	○	–	–	–	○	–
	Resolution of the Motor	@55	–	–	○	○	–	–	○
	Rotating Direction of the Motor	@37	–	○	○	○	○	○	○
	Servo Interlock Switch	@7	–	○	○	○	○	○	○
	Power On (Servo on and Off)	@20	–	○	○	○	○	○	○
Speed Control Loop Parameter	DC Gain	@22	G107	○	–	–	–	○	–
	fc Remote Switch	@23	G104	–	–	–	–	–	–
Positioning Control Loop Parameter	COIN Switch	@8	G110 G111	○	○	–	–	○	○
	COIN Width Switch	@24	G105	○	–	–	–	–	–
	Internal COIN Width	@59	–	–	○	–	–	○	–
	Position Proportional Gain	@28	–	–	○	–	–	○	–
	Velocity Offset Correction Switch	@29	–	–	○	–	–	○	–
	Input Sensitivity Voltage	@62	–	–	○	–	–	○	–
	Velocity Feed Forward Ratio	@63	–	–	○	–	–	○	–

The following section describes the parameter setting of the motors of each type. Please skip over the pages where you do not need to read.

Setting of the DYNASERV (Position Mode) — Auxiliary Function

● **Function** . This sets the parameters related to the DYNASERV (position mode).

● **Operation** In the following procedures, the operations of the commands with parameters of each axis shall be carried out only for the parameters of the axis where the DYNASERV (position mode) is connected. Omit the other parameters.

① **Type of the Motor**

■ Set the motor type.
This command is a dedicated command to the DYNASERV, and the related parameters are automatically set.
The motor type command's (@40) 1st parameter sets the motor type of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.

@40 [:<Motor type of axis X (0-8) >]
[:<Motor type of axis Y (0-8) >]
[:<Motor type of axis Z (0-8) >]
[:<Motor type of axis W (0-8) >]

▼

Memo Setting values of motor type and contents

Setting values of motor type and contents	Parameters automatically set
0 : DM1***B	· Resolution of the motor (@55)
1 : DM1***A	· Low feeding speed (@50)
2 : DM8***B	· High feeding speed (@51)
3 : DM8***A	· Maximum feeding speed (@52)
4 : DR1***B	· Jog feeding speed (@64)
5 : DR1***E	
6 : DR1***A	
7 : DR5***B	
8 : DR5***A	

② **Rotating Direction of the Motor**

■ Set the rotating direction of the motor.
The rotating direction shall be the rotation of the axis of the motor viewed from the top.
The motor rotating direction command's (@37) 1st parameter sets the rotating direction of the motor of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W. Set "0" to designate (+) direction to CW (clockwise direction) and set "1" to CCW (counterclockwise direction).

▼

@37 [:<Rotating direction of motor of axis X (0, 1)>]
[:<Rotating direction of motor of axis Y (0, 1) >]
[:<Rotating direction of motor of axis Z (0, 1) >]
[:<Rotating direction of motor of axis W (0, 1) >]

③ **Servo Interlock SW**

■ Set the servo interlock switch.
The servo interlock switch turns the servo control on and off by the power on and off of the controller.
The initial 0 command's (@7) 1st parameter sets the servo interlock switch. Set to "0" to turn off the servo control when power of the controller is turned on. Set to "1" to turn on the servo control when power of the controller is turned on.

@7:0 or @7:1

④ **Power On (Servo On and Off)**

■ Turn power to the servo motor on (servo on).
When the servo control is set to off with the ③ Servo interlock SW setting above, the servo on / off command's (@20) 1st parameter sets the power of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.
Set "1" to turn on power.

@20 [:1] [:1] [:1] [:1]

! Leave the SERVO ON / OFF (#3) input terminal set to on. When the SERVO ON / OFF (#3) input terminal changes from on to off or vice versa, this command has no effect, and the conditions of the SERVO ON / OFF (#3) input terminal is used to set all the parameters of every axis.

⑤ **Integral / Proportional Control**

■ Set the integral / proportional control of the servo motor. The integral / proportional control controls on and off of the interface PACT contact to the driver.
The integral / proportional control command's (@21) 1st parameter designates the integral / proportional control of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W. Setting "0" sets the switch open. Setting "1" sets the switch closed.

@21 [:<Integral / proportional control of axis X (0, 1)>]
[:<Integral / proportional control of axis Y (0, 1)>]
[:<Integral / proportional control of axis Z (0, 1)>]
[:<Integral / proportional control of axis W (0, 1)>]

! Refer to the instruction manual of the driver for the logic and the meaning of the interface PAC contact.

Memo Designation using the program:

G106 [X<Integral / proportional control of axis X (0, 1)>]
[Y <Integral / proportional control of axis Y (0, 1) >]
[Z <Integral / proportional control of axis Z (0, 1) >]
[W<Integral / proportional control of axis W (0, 1) >]

⑥ DC Gain

■ Set the DC gain of the DYNASERV.
The DC gain command's (@22) 1st parameter designates the DC gain of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W. The DC gain differs among the types of the motors.

@22 [: <DC gain of axis X (0-7) >]
[: <DC gain of axis Y (0-7) >]
[: <DC gain of axis Z (0-7) >]
[: <DC gain of axis W (0-7) >]

Memo Designation using the program:

G107 [X <DC gain of axis X (0-7) >]
[Y <DC gain of axis Y (0-7) >]
[Z <DC gain of axis Z (0-7) >]
[W <DC gain of axis W (0-7) >]

▼

Memo Setting value of the DC gain and the contents:

Type (0 to 3) of the motor (for DM series)	Type (4 to 8) of the motor (for DR series)
0 : ×1	0 : ×1
4 : ×10	1 : ×4
	2 : ×7
	3 : ×10
	4 : ×13
	5 : ×16
	6 : ×19
	7 : ×22

⑦ fc Remote

■ Set the fc remote switch of the DYNASERV.
Normally, set "0" with this command because the control band width is set and adjusted with the fc switch on the front panel of the DYNASERV.
The fc remote switch command's (@23) 1st parameter designates the fc remote switch of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.

@23 [: <fc remote switch of axis X (0-15) >]
[: <fc remote switch of axis Y (0-15) >]
[: <fc remote switch of axis Z (0-15) >]
[: <fc remote switch of axis W (0-15) >]

▼

Memo Designation using the program:

G104 [X <fc remote switch of axis X (0-15) >]
[Y <fc remote switch of axis Y (0-15) >]
[Z <fc remote switch of axis Z (0-15) >]
[W <fc remote switch of axis W (0-15) >]

⑧ COIN SW

■ Set the COIN switch.
The COIN switch selects whether or not to perform the positioning alignment check.
The initial 1 command's (@8) 4th parameter sets the COIN switch. Setting "0" disables the COIN function and setting "1" enables the COIN function.
@8: : : :0 or @8: : : :1

Memo Designation using the program:
G110 or G111

⑨ COIN Width

■ Set the COIN switch of the DYNASERV.
When the COIN function is enabled with procedure ⑧ COIN SW above, the COIN width switch command's (@24) 1st parameter designates the COIN width switch of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.
@24 [:<COIN width switch of axis X (0-3) >]
[:<COIN width switch of axis Y (0-3) >]
[:<COIN width switch of axis Z (0-3) >]
[:<COIN width switch of axis W (0-3) >]

Memo Designation using the program:
G105 [X <COIN width switch of axis X (0-3)>]
[Y <COIN width switch of axis Y (0-3) >]
[Z <COIN width switch of axis Z (0-3) >]
[W <COIN width switch of axis W (0-3) >]

Setting values of the COIN width switch and the contents:

	POSW1	POSW0
0:	H	H
1:	H	L
2:	L	H
3:	L	L

⑩ Integral Reset

■ Set the integral reset switch of the DYNASERV.
The integral reset switch turns on and off the interface IRST contact to the driver.
The integral reset switch command's (@26) 1st parameter designates the integral reset switch of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W. Setting "0" sets the switch open. Setting "1" sets the switch closed.
@26 [:<Integral reset switch of axis X (0, 1) >]
[:<Integral reset switch of axis Y (0, 1) >]
[:<Integral reset switch of axis Z (0, 1) >]
[:<Integral reset switch of axis W (0, 1) >]

! Refer to the instruction manual of the driver for the logic and the meaning of the interface IRST contact.

Setting of the AC / DC Servo Motor (Velocity Mode) Auxiliary Function

● **Function** This sets the parameters related to the AC / DC servo motor (velocity mode).

● **Operation** In the following procedures, the operations of the commands with parameters of each axis shall be carried out only for the parameters of the axis where the AC / DC servo motor (velocity mode) is connected. Omit the other parameters.

① **Resolution of the Motor** ■ Set the resolution of the motor.

> Note
> Set 4 times the encoder's resolution as the motor's resolution, because Fineserv automatically reduces the parameter into 1/4.

▼

The motor resolution command's (@55) 1st parameter sets the resolution of the motor of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.

@55 [:<Resolution of motor of axis X (0-999999999) >]
[:<Resolution of motor of axis Y (0-999999999) >]
[:<Resolution of motor of axis Z (0-999999999) >]
[:<Resolution of motor of axis W (0-999999999) >]

② **Input Sensitivity Voltage** ■ Set the input sensitivity voltage (mV).
The input sensitivity voltage means the input voltage of 1rps. The appropriate value shall be set because this parameter is required for the positioning feedback control.
The input sensitivity voltage command's (@62) 1st parameter sets the input sensitivity voltage of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.

@62 [:<Input sensitivity voltage of axis X (0-9999) >]
[:<Input sensitivity voltage of axis Y (0-9999) >]
[:<Input sensitivity voltage of axis Z (0-9999) >]
[:<Input sensitivity voltage of axis W (0-9999) >]

③ **Rotating Direction of the Motor** ■ Set the rotating direction of the motor.
The rotating direction shall be the rotation of the axis of the motor viewed from the top.
The motor rotating direction command's (@37) 1st parameter sets the rotating direction of the motor of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W. Set "0" to designate (+) direction to CW (clockwise direction) and set "1" to CCW (counterclockwise direction).

▼

@37 [:<Rotating direction of motor of axis X (0, 1)>]
[:<Rotating direction of motor of axis Y (0, 1) >]
[:<Rotating direction of motor of axis Z (0, 1) >]
[:<Rotating direction of motor of axis W (0, 1) >]

④ **Servo Interlock SW**

■ Set the servo interlock switch.
The servo interlock switch turns the servo control on and off by the power on and off of the controller.
The initial 0 command's (@7) 1st parameter sets the servo interlock switch. Set to "0" to turn off the servo control when power of the controller is turned on. Set to "1" to turn on the servo control when power of the controller is turned on.

@7 : 0 or @7 : 1

⑤ **Power On (Servo On and Off)**

■ Turn power to the servo motor on (servo on).
When the servo control is set to off with the ④ Servo interlock SW setting above, the servo on / off command's (@20) 1st parameter sets the power of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.
Set "1" to turn on power.

@20 [: 1] [: 1] [: 1] [: 1]

! Leave the SERVO ON / OFF (#3) input terminal set to on. When the SERVO ON / OFF (#3) input terminal changes from on to off or vice versa, this command has no effect, and the conditions of the SERVO ON / OFF (#3) input terminal is used to set all the parameters of every axis.

⑥ **Integral / Proportional Control**

■ Set the integral / proportional control of the servo motor.
The integral / proportional control controls on and off of the interface PACT contact to the driver. The integral / proportional control command's (@21) 1st parameter designates the integral / proportional control of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.
Setting "0" sets the switch open. Setting "1" sets the switch closed.

@21 [:<Integral / proportional switch of X axis (0, 1)>]
[:<Integral / proportional control of axis Y (0, 1) >]
[:<Integral / proportional control of axis Z (0, 1) >]
[:<Integral / proportional control of axis W (0, 1) >]

! Refer to the instruction manual of the driver for the logic and the meaning of the interface PAC contact.

Memo Designation using the program :

G106 [X<Integral / proportional control of axis X (0, 1)>]
[Y <Integral / proportional control of axis Y (0, 1) >]
[Z <Integral / proportional control of axis Z (0, 1) >]
[W <Integral / proportional control of axis W (0, 1) >]

⑦ **Position Proportional Gain**

■ Set the coarse position proportional gain.
The position proportional gain control command's (@28) 1st parameter designates the position proportional gain control of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.

@28 [:<Position proportional gain control of axis X (0-7)>]
[:<Position proportional gain control of axis Y (0-7) >]
[:<Position proportional gain control of axis Z (0-7) >]
[:<Position proportional gain control of axis W (0-7) >]

▼

Memo Setting values of the position proportional gain and the contents:

0 : ×0.5
1 : ×1
2 : ×2
3 : ×4
4 : ×8
5 : ×16
6 : ×32
7 : ×64

▼

Memo Setting values of the position proportional gain can be adjusted finely in 16 steps from 1/16 to 16/16, using the rotary switch located on the front of the axis board. This fine-adjustment can be carried out either when the axis of the motor is not rotating or when the axis of the motor is rotating.

⑧ **Speed Offset Correction**

■ Set the speed offset correction switch. The speed offset correction switch corrects the undesired analog voltage generated outside the speed loop. This switch is used to select whether or not to carry out the speed offset correction.
The speed offset correction switch command's (@29) 1st parameter designates the speed offset correction switch of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W. Setting "0" disables the switch. Setting "1" enables the switch.

▼

@29 [:<Speed offset correction switch of axis X (0, 1)>]
[:<Speed offset correction switch of axis Y (0, 1) >]
[:<Speed offset correction switch of axis Z (0, 1) >]
[:<Speed offset correction switch of axis W (0, 1) >]

⑨ **COIN SW**

▼

■ Set the COIN switch.
The COIN switch selects whether or not to perform the positioning alignment check.
The initial 1 command's (@8) 4th parameter sets the COIN switch. Setting "0" disables the COIN function and setting "1" enables the COIN function.

@8: : : :0 or @8: : : :1

Memo Designation using the program:
G110 or G111

⑩ **Internal COIN Width**

▼

■ Set the internal COIN width (p) of the controller.
The internal COIN width means the COIN width when positioning feedback control is carried out from the controller. Alignment check is carried out depending on the internal COIN width compared with the positioning deviation of the positioning feedback control.
When the COIN function is enabled with procedure ⑨ COIN SW above, the internal COIN width switch command's (@59) 1st parameter designates the internal COIN width of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.

@59 [: <Internal COIN width of axis X (0-32767) >]
[: <Internal COIN width of axis Y (0-32767) >]
[: <Internal COIN width of axis Z (0-32767) >]
[: <Internal COIN width of axis W (0-32767) >]

⑪ **Feed Forward Ratio**

■ Set the velocity feed forward ratio (%).
The feed forward ratio controls the amount of velocity feed forward in a range of 0 to 120 (%).
The velocity feed forward ratio command's (@63) 1st parameter designates the velocity feed forward ratio of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.

@63 [: <Velocity feed forward ratio of axis X (0-120) >]
[: <Velocity feed forward ratio of axis Y (0-120) >]
[: <Velocity feed forward ratio of axis Z (0-120) >]
[: <Velocity feed forward ratio of axis W (0-120) >]

! The desired amount of velocity feed forward cannot be obtained when the appropriate values have not been set by procedure ① Resolution of the motor and procedure ② Input sensitivity voltage.

Setting of the Stepping Motor

Auxiliary Function

● **Function** This sets the parameters related to the stepping motor.

● **Operation** In the following procedures, the operations of the commands with parameters of each axis shall be carried out only for the parameters of the axis where the stepping motor is connected. Omit the other parameters.

① **Resolution of Rotation of the Motor**

▼

■ Set the resolution (ppr) of rotation of the motor.
The motor resolution command's (@55) 1st parameter sets the resolution of rotation of the motor of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.

@55 (:<Resolution of motor of axis X (0-999999999) >)
(:<Resolution of motor of axis Y (0-999999999) >)
(:<Resolution of motor of axis Z (0-999999999) >)
(:<Resolution of motor of axis W (0-999999999) >)

② **Rotating Direction of the Motor**

▼

■ Set the rotating direction of the motor.
The rotating direction shall be the rotation of the axis of the motor viewed from the top.
The motor rotating direction command's (@37) 1st parameter sets the rotating direction of the motor of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W. Set "0" to designate (+) direction to CW (clockwise direction) and set "1" to CCW (counterclockwise direction).

@37 (:<Rotating direction of motor of axis X (0, 1) >)
(:<Rotating direction of motor of axis Y (0, 1) >)
(:<Rotating direction of motor of axis Z (0, 1) >)
(:<Rotating direction of motor of axis W (0, 1) >)

③ **Servo Interlock SW**

▼

■ Set the servo interlock switch.
The servo interlock switch turns the servo control on and off by the power on and off of the controller.
The initial 0 command's (@7) 1st parameter sets the servo interlock switch. Set to "0" to turn off the servo control when power of the controller is turned on. Set to "1" to turn on the servo control when power of the controller is turned on.

@7 : 0 or @7 : 1

④ **Power on (Servo on and off)**

■ Turn power to the servo motor on (servo on).
When the servo control is set to off with the ⑬ Servo interlock SW setting, the servo on / off command's (@20) 1st parameter sets the power of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.
Set "1" to turn on power.

@20 [:1] [:1] [:1] [:1]

! Leave the SERVO ON / OFF (#3) input terminal set to on. When the SERVO ON / OFF (#3) input terminal changes from on to off or vice versa, this command has no effect, and the conditions of the SERVO ON / OFF (#3) input terminal is used to set all the parameters of every axis.

Setting of the Inverter — Auxiliary Function

● **Function** This sets the parameters related to the inverter.

● **Operation** In the following procedures, the operations of the commands with parameters of each axis shall be carried out only for the parameters of the axis where the inverter is connected. Omit the other parameters.

① **Rotating Direction of the Motor**

▼

■ Set the rotating direction of the motor.
The rotating direction shall be the rotation of the axis of the motor viewed from the top.
The motor rotating direction command's (@37) 1st parameter sets the rotating direction of the motor of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W. Set "0" to designate (+) direction to CW (clockwise direction) and set "1" to CCW (counterclockwise direction).

@37 [:<Rotating direction of motor of axis X (0, 1) >]
[:<Rotating direction of motor of axis Y (0, 1) >]
[:<Rotating direction of motor of axis Z (0, 1) >]
[:<Rotating direction of motor of axis W (0, 1) >]

② **Servo Interlock SW**

▼

■ Set the servo interlock switch.
The servo interlock switch turns the servo control on and off by the power on and off of the controller.
The initial 0 command's (@7) 1st parameter sets the servo interlock switch. Set to "0" to turn off the servo control when power of the controller is turned on. Set to "1" to turn on the servo control when power of the controller is turned on.

@7:0 or @7:1

③ **Power On (Servo On and Off)**

▼

■ Turn power to the servo motor on (servo on).
When the servo control is set to off with the ② Servo interlock SW setting, the servo on / off command's (@20) 1st parameter sets the power of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W. Set "1" to turn on power.

@20[:1][:1][:1][:1]

! Leave the SERVO ON / OFF (#3) input terminal set to on. When the SERVO ON / OFF (#3) input terminal changes from on to off or vice versa, this command has no effect, and the conditions of the SERVO ON / OFF (#3) input terminal is used to set all the parameters of every axis.

● **Note**

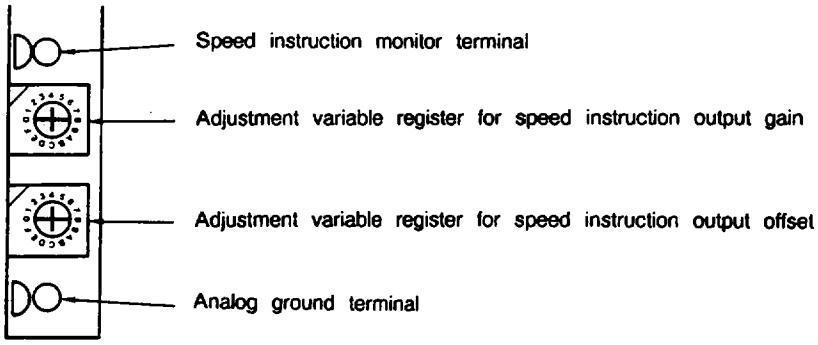
Rotating speed of the main axis compared with that of the inverter's axis is instructed in programmed operation or in MDI operation. Rotating direction and analog speed can be set in a range of −100 to +100%.

"4.5 Programming Language / main axis function"

④ **Speed Output Adjustment**

■ Readjust the maximum output voltage applied to the driver of the inverter's axis which is factory-set to 6V (100%). This maximum output voltage can be adjusted between 6 and 10V.
When adjusted to 10V, 100% corresponds to 10V.
Adjustment can be carried out using the variable resistor and the signal terminals, located at the inverter axis controller on the front panel of the controller. Adjustment procedures are as follows:

- Connect an oscilloscope to the <AOUT> (speed instruction monitor) signal terminal and to the <AGND> (analog ground) signal terminal.
- Adjust the <speed instruction output gain control> for the desired maximum output voltage, while observing the screen of the oscilloscope.



Coordination System — Auxiliary Function

The motor has a special position called "mechanical origin". Mechanical origin is the reference of all coordination systems. Besides the mechanical origin, there is "mechanical coordination origin" which is the reference of operation when controlling the motors from the controller. The mechanical origin never changes while the mechanical coordination origin can change according to parameters.

The coordination system that this controller uses is determined by the four following elements in addition to the mechanical origin.

(1) Offset value of origin return
The offset amount (p) of mechanical coordination origin position from mechanical origin position is called "origin return offset value". The origin return offset value is designated using the origin return offset value command (@46). When setting the origin return offset value to 0, the mechanical origin position coincides exactly with the mechanical coordination origin position.

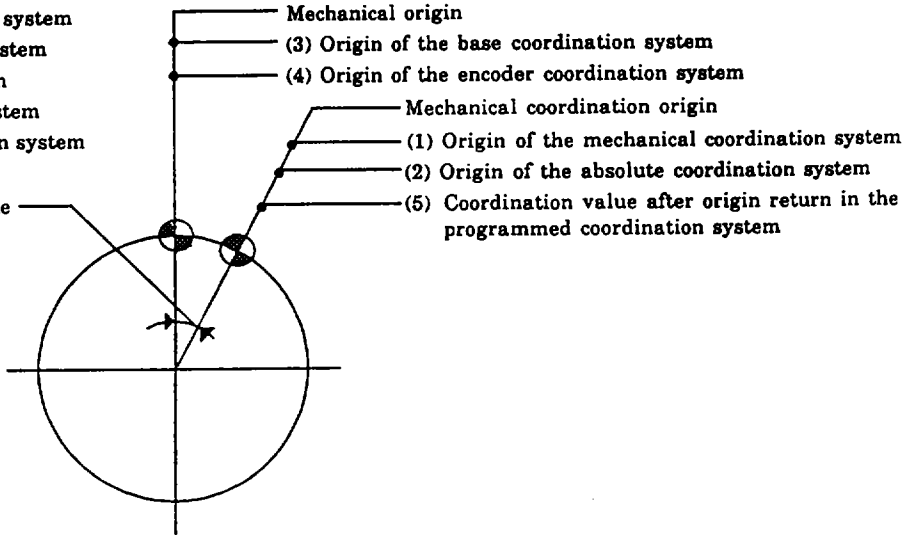
(2) Coordination value after origin return
The coordination value (p) of the mechanical coordination origin return can be designated. Input of positioning data is carried out using the coordination system determined by designating the coordination value. The coordination command (@47) after origin return designates the coordination value of the mechanical coordination origin position.

(3) Input unit system
The unit system for inputting positioning data is called "input unit system". The input unit system commands (@41) designates the desired type of input unit system. There are four types of input unit system as follows:
① Pulse input unit system
② Degree input unit system
③ Division input unit system
④ Feeding length input unit system

(4) Rotating direction of the motor
The rotating direction (+) of the motor is factory-set to CW (clockwise) and the rotating direction (−) of the motor is factory-set to CCW (counterclockwise). This setting of rotating direction is designated as normal setting.
This can be reversed using the motor rotating direction command (@37). In this case, (+) direction is for CCW and (−) for CW, and this setting is designated as reverse setting.

This controller controls in the five following coordination systems.

(1) Mechanical coordination system
(2) Absolute coordination system
(3) Base coordination system
(4) Encoder coordination system
(5) Programmed coordination system



Mechanical Coordination System

(1) Mechanical coordination system is a coordination system whose origin is the mechanical coordination origin position.
(2) The current position display shows the coordination value of the number of output pulses of axis feeding instruction, and the unit is in the number of pulses.
(3) The current position display is not cleared to 0 when the motor rotates once. It is accumulated.
(4) This coordination system is affected by rotating direction of the motor. When rotating direction of the motor is reversed, polarity changes and (+) direction is set for CCW (counterclockwise) direction.

Absolute Coordination System

(1) Absolute coordination system is a coordination system whose origin is the mechanical coordination origin position.
(2) The current position display shows the coordination value of output pulses of axis feeding instruction, and the unit is in the number of pulses.
(3) This current position display is cleared to 0 when the motor rotates once.
(4) This coordination system is affected by rotating direction of the motor. When rotating direction of the motor is reversed, polarity changes and (+) direction is set for CCW (counterclockwise) direction.

Base Coordination System

(1) Base coordination system is a coordination system whose origin is the mechanical coordination origin position.
(2) The current position display shows the coordination value of the number of output pulses of axis feeding instruction, and the unit is in the number of pulses. The displayed value is 16 times for the stepping motor's axis compared with that for the other motors' shaft.
(3) The current position display is not cleared to 0 when the motor rotates once. It is accumulated.
(4) This coordination system is affected by rotating direction of the motor. When rotating direction of the motor is reversed, polarity changes and (+) direction is set for CCW (counterclockwise) direction.

Encoder Coordination System

(1) Encoder coordination system is a coordination system whose origin is the mechanical coordination origin position.
(2) The current position display shows the coordination value of the amount of feedback from the encoder, and the unit is in the number of pulses. The displayed value is, however, the base coordination value for the stepping motor.
(3) The current position display is not cleared to 0 when the motor rotates once. It is accumulated.
(4) This coordination system is not affected by rotating direction of the motor. When up or down pulse is input, polarity is (+) for up (increasing) direction. When phase A or phase B is input, polarity is (+) for preceding phase A input.

Programmed Coordination System

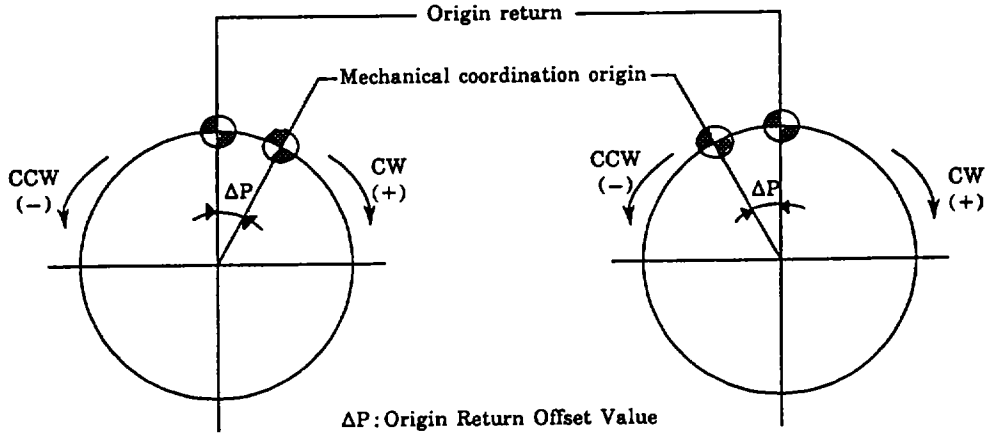
(1) Programmed coordination system is a coordination system whose origin value after origin return is the mechanical coordination origin position.
(2) Coordination value and unit of the current position display is the same as the input unit system.
(3) The current position display in the pulse input unit system and in the feeding length input unit system is not cleared to 0 when the motor rotates once. It is accumulated. However, it is cleared to 0 in the degree input unit system and division input unit system when the motor rotates once.
(4) This coordination system is affected by rotating direction of the motor. When rotating direction of the motor is reversed, polarity changes and (+) direction is set for CCW (counterclockwise) direction.

Origin Return Offset Value

(1) Origin return offset value is not affected by rotating direction of the motor.

(2) The origin return offset value is (+) when rotating direction of the motor is set for CW (clockwise) direction. It is (−) when rotating direction of the motor is set for CCW (counterclockwise) direction

≪When Origin Return Offset Value is Positive≫ ≪When Origin Return Offset Value is Negative≫



ΔP: Origin Return Offset Value

Current Position Display in a Special Case

(1) The table below shows the status of the current position display in each coordination system when power on or an error occurs.

	Mechanical Coordination System	Absolute Coordination System	Base Coordination System	Encoder Coordination System	Programmed Coordination System
Power on/off	0	0	0	0	0
Servo off	0	0	0	0	0
Reset command (@1) RESET (#10) input signal	No change	No change	No change	No change	No change
RESET SW (front panel)	0	0	0	0	0
An error resulting in servo off	0	0	0	0	0
An error not resulting in servo off	No change	No change	No change	No change	No change
Clear position (G160 code) Coordination system clear command (@4)	0	0	0	0	0
Origin return operation Hardware origin return (G27 code)	0	0	Offset value (@46)	Offset value (@46)	Coordination after origin return (@47)

! (1) By setting the clear switch of the coordination system with initial 1 command's (@8) 3rd parameter, all coordination systems will be set to 0 by reset command (@1) and also by the RESET (#10) input signal.

(2) Concerning the values of the base coordination system of origin return operation and hardware origin return (G27 code), sign of the origin return offset value (@46) will be reversed when rotating direction of the motor is reversed with the motor rotating direction command (@37).

Input Unit System : Auxiliary Function

This can designate the desired input unit system for positioning data input.
The following five input unit systems are available.

(1) Type 0 : Pulse input unit system
This input unit system uses the number of pulses (p) for positioning data.

(2) Type 1 : Degree input unit system
This input unit system uses angle (°) from the center of the motor for positioning data.

(3) Type 2 : Division input unit system
This input unit system uses division number (dividing 360° by a setting number) for positioning data.

(4) Type 3 : Feeding length input unit system
This input unit system uses feeding length of the motor for positioning data. Use for uni-directioal feeding application only.

(5) Types 5 : Degree input unit system (1 rotation)
This input unit system is almost the same as that for Type 1. However, the range is limited in 0° to 360°.

(6) Type 6 : Similar to type 3 with bi-directional capability
(Application includes leed-screw feeding etc.)

How to Designate

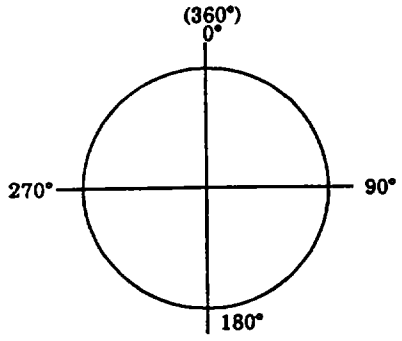
(1) Setting through the RS-232C mode

@41 [:<axis X' input unit system type (0-5)>] [:<axis Y' input unit system type (0-5)>]
[:<axis Z' input unit system type (0-5)>] [:<axis W' input unit system type (0-5)>]

(2) Setting by using the program

Type	Setting
Pulse input unit system	G140 [X] [Y] [Z] [W]
Degree input unit system (multiple rotation)	G141 [X] [Y] [Z] [W]
Division input unit system	G142 [X] [Y] [Z] [W]
Feeding length input unit system	G143 [X] [Y] [Z] [W]

Degree Input Unit System (Multiple Rotation)



(1) The minimum setting unit is 0.001°. For example, the value of 45° for axis X is X45000.

(2) Absolute instruction (G90 code) and incremental instruction (G91 code) can be used.

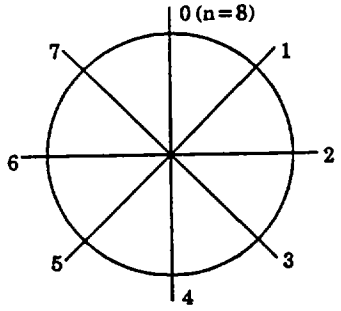
(3) The input range of the absolute instruction is ±360° (±360000).
An error occurs if the value exceeds this range.
A positive (+) value is for normal rotation of the shaft, and a negative (−) value for reverse rotation. For example, with the current position set at 20°, instructing 315° sets the axis at the 315° position by normal rotation. Instructing −45° sets the axis also at the 315° position but by reverse rotation.

(4) The input range of the incremental instruction can be exceeded ±360°, and rotation with this setting is called "multiple rotation". The current position is displayed in a range within 360°. When it is at the 360° degree position, 0° is displayed.

(5) Software origin return (G28 code) sets the position to 0° with reverse rotation.

(6) When resolution of the motor cannot be divided out by the minimum setting unit, the position will be set with the minimum error. The error is not accumulated.

Division Input Unit System



(1) The maximum division number for division input is 1024.

(2) Absolute instruction (G90 code) and incremental instruction (G91 code) can be used.

(3) + or − indicates the feeding direction of the shaft.

(4) The maximum input for absolute instruction is the division number (with sign) for division input. An error occurs if the input exceeds the division number for division input.

(5) The input range of the incremental instruction can be exceeded the division number for division input, and rotation with this setting is called "multiple rotation". The current position is displayed in a range within the division number for division input. When it is at the same position as the division number for division input, 0 is displayed.

(6) Software origin return (G28 code) sets the position to 0 with reverse rotation.

(7) When resolution of the motor cannot be divided out by the minimum setting unit, the position will be set with the minimum error. The error is not accumulated.

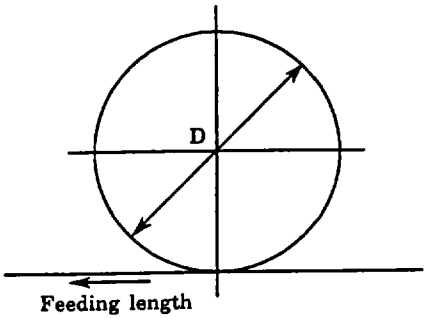
Feeding Length Input Unit System

(1) Feeding length factor is used for converting the input positioning data into the number of instruction pulse to the motor. In the coordination system determined by specifying the feeding length factor, inputting the positioning data value the same as the feeding length factor rotates the motor once. Therefore, the feeding length factor shall be set the same as the positioning data value for motor rotation of one time. For example, when the feeding length factor of axis X is set to 1000, instructing "X1000" rotates the motor once.

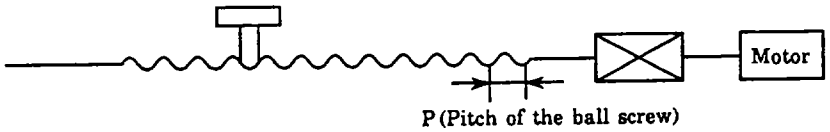
(2) Positioning data is converted into the number of instruction pulse to the motor, using the following formula.

$$\text{Instruction pulse (p)} = \frac{\text{Resolution of the motor (ppr)}}{\text{Feeding length factor}} \times \text{positioning data}$$

(3) In case of roll feeding, setting the feeding length factor with the following formula makes the positioning data the same as the length of arc of the drive roller's circumference.
Feeding length factor = Ratio of the circumference of a circle to the diameter (π) × Diameter (D) of the drive roller of the motor
Input unit of the positioning data is the same as that of the value for "D" in the formula above.
For example:
where D = 280mm and $\pi D = \pi \times 280$:
the unit is in mm,
and, $\pi D = \pi \times 280000$:
the unit is in μm (0.001mm).



(4) When using a ball screw, setting the pitch (μm) of the ball screw for feeding length factor makes the positioning data in rectangular coordinate system of the ball screws.



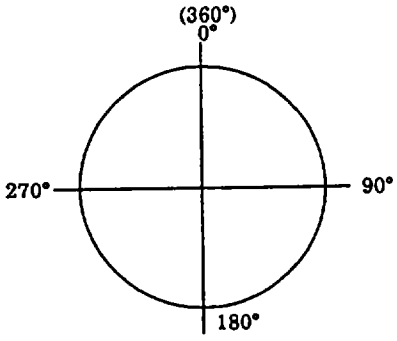
Feeding length factor = Pμm

Degree Input Unit System (One Rotation)

(1) The minimum setting unit is 0.001°. For example, the value of 45° for axis X is X45000.

(2) Absolute instruction (G90 code) and incremental instruction (G91 code) can be used.

(3) The input range of the absolute instruction is 0° to 360° (0 to 360000). An error occurs if the degree value is more than 360° or if it is minus (−). The position for 0° is the same as that for 360°. With the current position set at 0° degrees, instructing 360° rotates the axis once. Then, instructing 0° successively rotates the axis once again in reverse direction.



(4) The input range of the incremental instruction is ±360° (±360000).
The target value in this case is the input value plus the value of the current position. Therefore, an error occurs when the total value is out of the specified range 0° to 360° (0 to 360000).

(5) Software origin return (G28 code) sets the position to 0° with reverse rotation.

(6) When resolution of the motor cannot be divided out by the minimum setting unit, the position will be set with the minimum error. The error is not accumulated.

! The difference of this degree input unit system of one rotation from that of multiple rotation is as follows:
There are two positions for one rotation, position of 360° and that of 0°.
There is only one position for multiple rotation, so position of 360° is the same as that of 0°.

Pulse Input Unit System Auxiliary Function

● **Function** This input unit system uses the number of pulses (p) for positioning data.

● **Operation**

① **Input Unit System**

■ Set the input unit system in the pulse input unit system.
The input unit system command's (@41) 1st parameter sets the input unit system type of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W. Set the parameter of the axis of the pulse input unit system to "0".

@41 [:0] [:0] [:0] [:0]

Memo Designation using the program:
G140 [X] [Y] [Z] [W]

Degree Input Unit System Auxiliary Function

● **Function** This input unit system uses the angle from the center of the motor (°) for positioning data.

● **Operation**

① **Input Unit System**

■ Set the input unit system in the degree input unit system.
The input unit system command's (@41) 1st parameter sets the input unit system type of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W. Set the parameter of the axis of the degree input unit system to "1".

@41 [:1] [:1] [:1] [:1]

Memo Designation using the program:
G141 [X] [Y] [Z] [W]

! Setting for the degree input unit system of one rotation is:

@41 [:5] [:5] [:5] [:5]

Designation using the program is not available.

Division Input Unit System Auxiliary Function

● **Function** This input unit system uses the division number obtained by dividing 360° by a setting number.

● **Preparation** Is motor type registered?
☞"4.2 Operation for Installation / Type by type setting of the motor"
Is resolution of the motor rotation registered?
☞"4.2 Operation for Installation / Type by type setting of the motor"

● **Operation**

① **Input Unit System**

▼

■ Set the input unit system in the division input unit system.
The input unit system command's (@41) 1st parameter sets the input unit system type of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W. Set the parameter of the axis of the division input unit system to "2".

@41 [:2] [:2] [:2] [:2]

Memo Designation using the program:
G142 [X] [Y] [Z] [W]

② **The Number of Divisions for Division Input**

■ Set the number of divisions of motor rotation.
The division input division number command's (@56) 1st parameter sets the division numbers of division input of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.
Set the division number of division input as the parameter of the axis of the division input unit system.

@56 [:<The number of divisions for division input of axis X (0-1024)>]
[:<The number of divisions for division input of axis Y (0-1024) >]
[:<The number of divisions for division input of axis Z (0-1024) >]
[:<The number of divisions for division input of axis W (0-1024) >]

Memo Designation using the program:
G101 [X<The number of divisions for division input of axis X (0-1024)>]
[Y <The number of divisions for division input of axis Y (0-1024)>]
[Z <The number of divisions for division input of axis Z (0-1024)>]
[W<The number of divisions for division input of axis W (0-1024)>]

Feeding Length Input Unit System Auxiliary Function

● **Function** This input unit system uses the feeding length of the motor rotation for positioning data.

● **Preparation** Is the motor type registered?
◇"4.2 Operation for Installation / Type by type setting of the motor"
Is resolution of the motor rotation registered?
◇"4.2 Operation for Installation / Type by type setting of the motor"

● **Operation**

① **Input Unit System**

▼

■ Set the input unit system in the feeding length input unit system.
The input unit system command's (@41) 1st parameter sets the input unit system type of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W. Set the parameter of the axis of the feeding length input unit system to "3".

@41 [:3] [:3] [:3] [:3]

Memo Designation using the program:
G143 [X] [Y] [Z] [W]

② **Feeding Length Factor**

■ Set the feeding length factor of motor rotation.
Feeding length factor is used for converting the input positioning data into the number of instruction pulse to the motor. In the coordination system determined by specifying the feeding length factor, inputting the positioning data value the same as the feeding length factor rotates the motor once. Therefore, the feeding length factor shall be set the same as the positioning data value for motor rotation of one time.
The feeding length factor command's (@60) 1st parameter sets the feeding length factor of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.
Set the feeding length factor as the parameter of the axis of the feeding length input unit system.

@60 [:<Feeding length factor of axis X (1-999999999)>]
[:<Feeding length factor of axis Y (1-999999999)>]
[:<Feeding length factor of axis Z (1-999999999)>]
[:<Feeding length factor of axis W (1-999999999)>]

Memo Positioning data is converted into the number of instruction pulse to the motor, using the following formula.

$$\text{Instruction pulse (p)} = \frac{\text{Resolution of the motor (ppr)}}{\text{Feeding length factor}} \times \text{positioning data}$$

Origin Return Operation

Origin return operation is the movement of the motor to the mechanical coordination origin. Origin return operation of the controller is determined with the two following types of parameters.

(1) Origin return direction
Direction of origin return operation is called "origin return direction".
Origin return direction can be designated to normal direction (CW : clockwise rotation) or reverse direction (CCW : counterclockwise rotation), using the origin return command (@30).

(2) Origin return type
Types of practical origin return operation is called "origin return type".
Origin return type can be selected from four following types, using the origin return type command (@35).

① Type 0 : This type detects the edge of the motor origin signal by referring to the edge of the origin proximate sensor from origin return direction.
- First, the edge of the origin proximate sensor is searched for at low feeding speed to the origin return direction.
- When detecting the edge, decelerates and stops.
- Rotating reversely at very low speed until the origin proximate sensor becomes out of detection.
- Then, rotating normally at very low speed for detecting the origin proximate sensor.
- The edge of the motor origin signal is searched for and detected at the minimum speed.

② Type 1 : This type detects also the edge of the motor origin signal by referring to the edge of the origin proximate sensor from origin return direction.
- First, the edge of the overtravel sensor is searched for in the opposite direction of origin return direction at low feeding speed.
- When detecting the edge, decelerates and stops.
- After that, origin return operation is carried out in the same way as for Type 0.

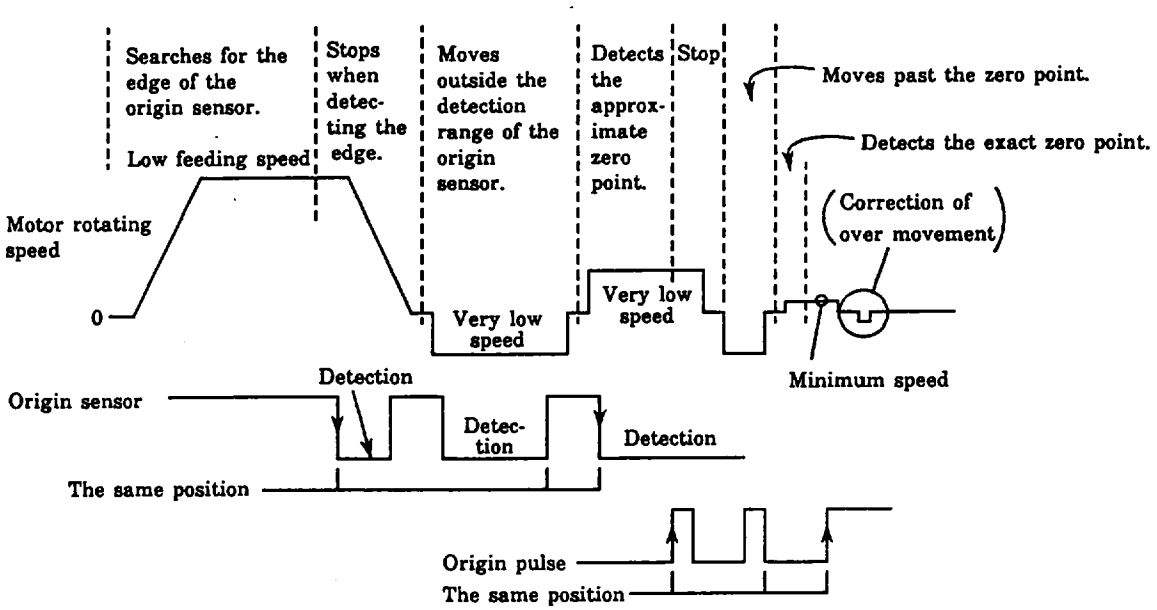
③ Type 2 : This type detects the edge of the motor origin signal by referring to the origin proximate sensor.
- First, the edge of the origin proximate sensor is searched for at low feeding speed to the origin return direction.
- When detecting the edge, decelerates and stops.
- Rotating normally at very low speed until the origin proximate sensor becomes out of detection.
- The edge of the motor origin signal is searched for and detected at the minimum speed.

④ Type 3 : This type detects also the edge of the motor origin signal by referring to the origin proximate sensor.
- First, the edge of the overtravel sensor is searched for in the opposite direction of origin return direction at low feeding speed.
- When detecting the edge, decelerates and stops.
- After that, origin return operation is carried out in the same way as for Type 2.

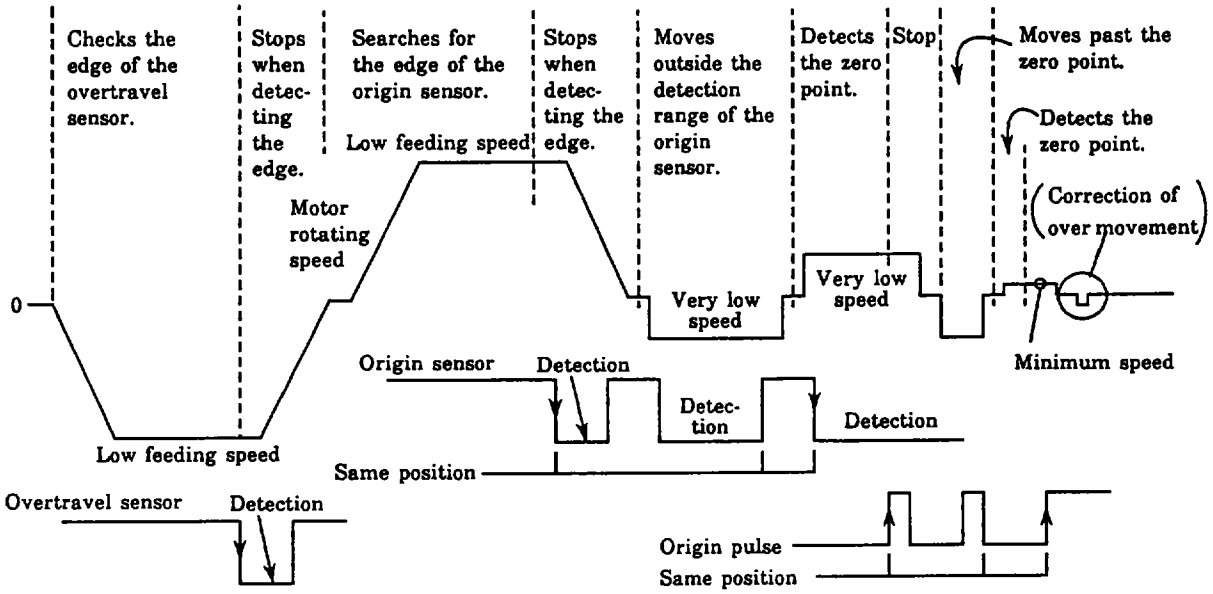
CAUTION

When the operating axis are the AC/DC servo type of control. kindly do not use origin types 2 & 3. Using these types 2 & 3 may induce positioning errors.

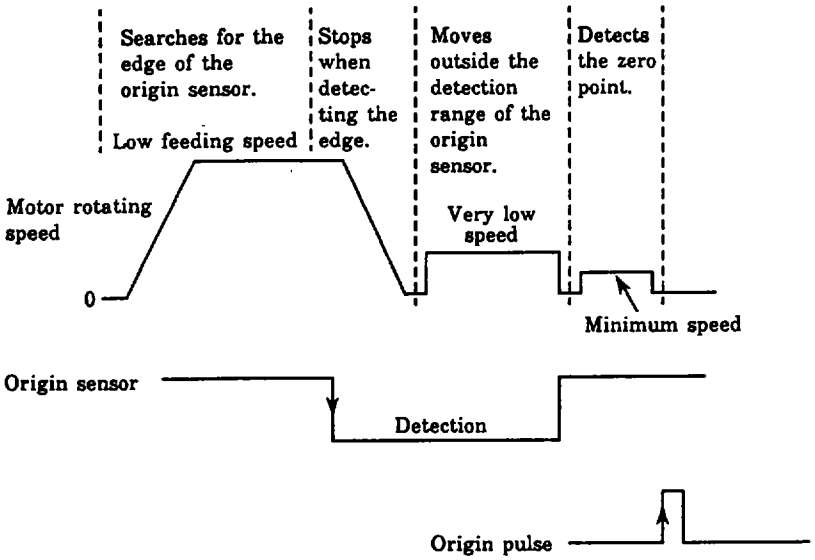
Origin Return Operation of Origin Return Type 0



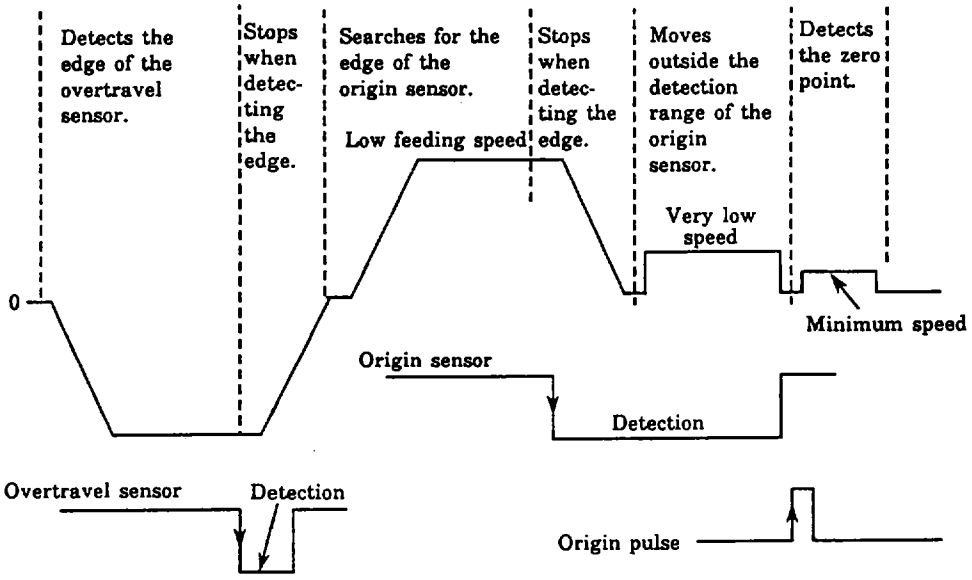
Origin Return Operation of Origin Return Type 1



Origin Return Operation of Origin Return Type 2



Origin Return Operation of Origin Return Type 3



Origin Return Parameters

Origin Return Function

● **Function** This registers the parameters related to origin return.

● **Preparation**
- Is the input unit system registered?
  "4.2 Operation for Installation / Input Unit System"

● **Operation**

① **Origin Return Direction**

■ Set the origin return direction.
The origin return direction shall be the direction of the movement of the origin return operation.
The origin return direction command's (@30) 1st parameter sets the origin return direction of the motor of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.
Set "0" to designate (+) direction to CW (clockwise direction) and set "1" to CCW (counterclockwise direction).

▼

@30 [:<Origin return direction of axis X (0, 1)>]
[:<Origin return direction of axis Y (0, 1)>]
[:<Origin return direction of axis Z (0, 1)>]
[:<Origin return direction of axis W (0, 1)>]

② **Origin Return Type**

■ Set the origin return type.
The origin return type shall be the type of the movement of the origin return operation.
The origin return type command's (@35) 1st parameter sets the origin return type of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.

▼

@35 [:<Origin return type of axis X (0-3)>]
[:<Origin return type of axis Y (0-3)>]
[:<Origin return type of axis Z (0-3)>]
[:<Origin return type of axis W (0-3)>]

③ **Offset Value**

■ Set the origin return offset value (p).
The origin return offset value shall be the offset amount (p) of the mechanical origin position from the mechanical coordination origin position.
The origin return offset value command's (@46) 1st parameter sets the origin return offset value of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.

▼

@46 [:<Offset value of axis X (−999999999- +999999999)>]
[:<Offset value of axis Y (−999999999- +999999999)>]
[:<Offset value of axis Z (−999999999- +999999999)>]
[:<Offset value of axis W (−999999999- +999999999)>]

④ **Origin Coordination**

■ Set the coordinate after origin return.
The coordinate after origin return determines the position (mechanical coordination origin) after origin return operation in the programmed coordination system. Unit of setting and the input range depend on the input unit system.
The coordination after origin return command's (@47) 1st parameter sets the coordination of axis X after origin return, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.

▼

@47 [:<Coordination of axis X>]
[:<Coordination of axis Y>]
[:<Coordination of axis Z>]
[:<Coordination of axis W>]

⑤ **Return Completion SW**

■ Set the origin return completion switch.
Normally, origin return operation (explained later) shall be carried out in the beginning of daily work.
Therefore, programmed operation (also explained later) is carried out after the origin return operation. By setting the origin return completion switch to enabled condition inhibits the programmed operation before the origin return operation.
The initial 0 command's (@7) 2nd parameter determines whether the origin return control is enabled or disabled. Setting the parameter to 0 disables the switch. Setting the parameter to 1 enables the switch.

@7::0 or @7::1

● **Note**

In order to designate the origin return Type 2 or 3, searching speed of the origin sensor set with the low feeding speed command (@50) shall be appropriately slow so that the searching can be stopped when detecting the origin sensor. If the speed is set too high and searching cannot be stopped at the origin sensor position, an error message (errors 60 to 63) is displayed in the mid-way of origin return operation, and the system is put in a halt.

Feeding Speed Parameters

Operation Function

● **Function** This sets the parameters related to feeding speed of the axis of the motor.

● **Operation**

① **Jog Feeding Speed**

▼

■ Set the jog feeding speed (kpps).
The jog feeding speed is the high jog feeding speed of the jogging operation. The jog feeding speed command's (@64) 1st parameter sets the jog feeding speed of the motor of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.

@64 [:<Jog feeding speed of axis X(1-1360)>]
[:<Jog feeding speed of axis Y (1-1360)>]
[:<Jog feeding speed of axis Z (1-1360)>]
[:<Jog feeding speed of axis W (1-1360)>]

② **Low Feeding Speed**

▼

■ Set the low feeding speed (kpps).
The low feeding speed is different between the origin return operation and the jog operation. The low feeding speed of the origin return operation is the feeding speed when the origin sensor is searched for. The low feeding speed of the jog operation is the low feeding speed when the jog operation is carried out. The low feeding speed command's (@50) 1st parameter sets the low feeding speed of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.

@50 [:<Low feeding speed of axis X (1-1360)>]
[:<Low feeding speed of axis Y (1-1360)>]
[:<Low feeding speed of axis Z (1-1360)>]
[:<Low feeding speed of axis W (1-1360)>]

③ **Maximum Feeding Speed**

▼

■ Set the maximum feeding speed (kpps).
The maximum feeding speed limits the speed of the axis feeding by the controller. This prevents the rotating speed of the motor from becoming out of control (it may happen by incorrect programming). The maximum feeding speed is obtained from the following formula.

$$\text{Maximum feeding speed (kpps)} \leq \frac{\text{Rated motor rotating speed (rps)} \times \text{Resolution of motor rotation (ppr)}}{1000}$$

The maximum feeding speed command's (@52) 1st parameter sets the maximum feeding speed of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.

@52 [:<Maximum feeding speed of axis X (1-1360)>]
[:<Maximum feeding speed of axis Y (1-1360)>]
[:<Maximum feeding speed of axis Z (1-1360)>]
[:<Maximum feeding speed of axis W (1-1360)>]

④ **High Feeding Speed** ■ Set the high feeding speed (kpps).
The high feeding speed is the feeding speed of fast forward (G00 code) instruction.
The high feeding speed is obtained from the following formula.

High feeding speed (kpps) ≤ Maximum feeding speed (kpps)

The high feeding speed command's (@51) 1st parameter sets the high feeding speed of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.

▼

@51 [:<High feeding speed of axis X (1-1360)>]
[:<High feeding speed of axis Y (1-1360)>]
[:<High feeding speed of axis Z (1-1360)>]
[:<High feeding speed of axis W (1-1360)>]

⑤ **Override** ■ Set the feeding speed override (%).
The feeding speed override means the values of all feeding speed override except for the override value of feeding speed for Operation ③ Maximum feeding speed.
The feeding speed override command's (@69) 1st parameter sets the feeding speed override of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.

@69 [:<Feeding speed override of axis X (0-100)>]
[:<Feeding speed override of axis Y (0-100)>]
[:<Feeding speed override of axis Z (0-100)>]
[:<Feeding speed override of axis W (0-100)>]

● **Notes**

(1) Use the motor type command (@40) in order to set automatically the jog feeding speed (@64), low feeding speed (@50), maximum feeding speed (@52), and high speed feeding speed (@51) of the DYNASERV (positioning instruction mode or speed instruction mode) axes. In order to set the different values from those set automatically, carry out the operations ①, ②, ③, or ④ above.

(2) Instructing a feeding speed which exceeds the value set for the maximum feeding speed displays an error message (error 58) on the personal computer, the operation display panel, or the operation display pendant. In this case, feeding speed follows the maximum feeding speed which has been set and operation continues.

Acceleration / Deceleration Control

Operation Function

Controlling the speed curve of the motor as desired is called "acceleration / deceleration control". The type of acceleration / deceleration control is called "acceleration / deceleration type". There are two groups of types of acceleration / deceleration as follows:

(1) Types 0 and 1 : 3rd-order spline curve and trapezoid curve
For these types, acceleration / deceleration speed is controlled with the speed curve of the motor so that it becomes 3rd-spline or trapezoid curve.

(2) Types 2 to 11 : Cam curve
The cam curve means the acceleration curve of the motor. For these types, acceleration / deceleration speed is controlled with the speed curve of the motor.

The following twelve types of "acceleration / deceleration type" are available.

(1) Type 0 : 3rd-order spline curve
(2) Type 1 : Trapezoid curve
(3) Type 2 : Modified sine wave curve
(4) Type 3 : Modified trapezoid curve
(5) Type 4 : Modified constant velocity curve
(6) Type 5 : Modified constant velocity curve II
(7) Type 6 : Trapecroid curve
(8) Type 7 : Trapecroid curve II
(9) Type 8 : Trapecroid curve III
(10) Type 9 : Constant acceleration curve
(11) Type 10 : User defined curve I
(12) Type 11 : User defined curve II

! User defined curve I of Type 10 and curve II of Type 11 can be created using the cam curve creation UTY in the tool box utility (optional).
Refer to "Instruction manual of cam curve creation utility" for creation of the user defined curves I and II.

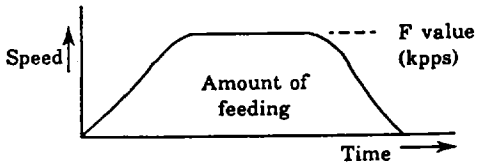
How to Designate the Desired Acceleration / Deceleration

(1) Designation through the RS-232C mode
@42 [:<Acceleration / deceleration type of axis X (0-11)>]
[:<Acceleration / deceleration type of axis Y (0-11)>]
[:<Acceleration / deceleration type of axis Z (0-11)>]
[:<Acceleration / deceleration type of axis W (0-11)>]

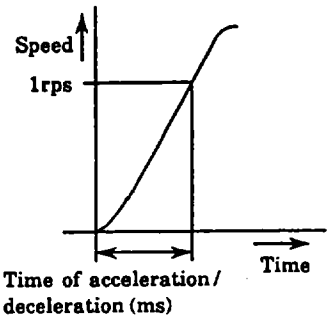
(2) Designation by using the program
G102 [X<Acceleration / deceleration type of axis X (0-11)>]
[Y <Acceleration / deceleration type of axis Y (0-11)>]
[Z <Acceleration / deceleration type of axis Z (0-11)>]
[W<Acceleration / deceleration type of axis W (0-11)>]

Acceleration / Deceleration Types 0 and 1

(1) Feeding speed instruction (F, A, B, C, and D codes) of the program is to be designated with frequency (kpps).

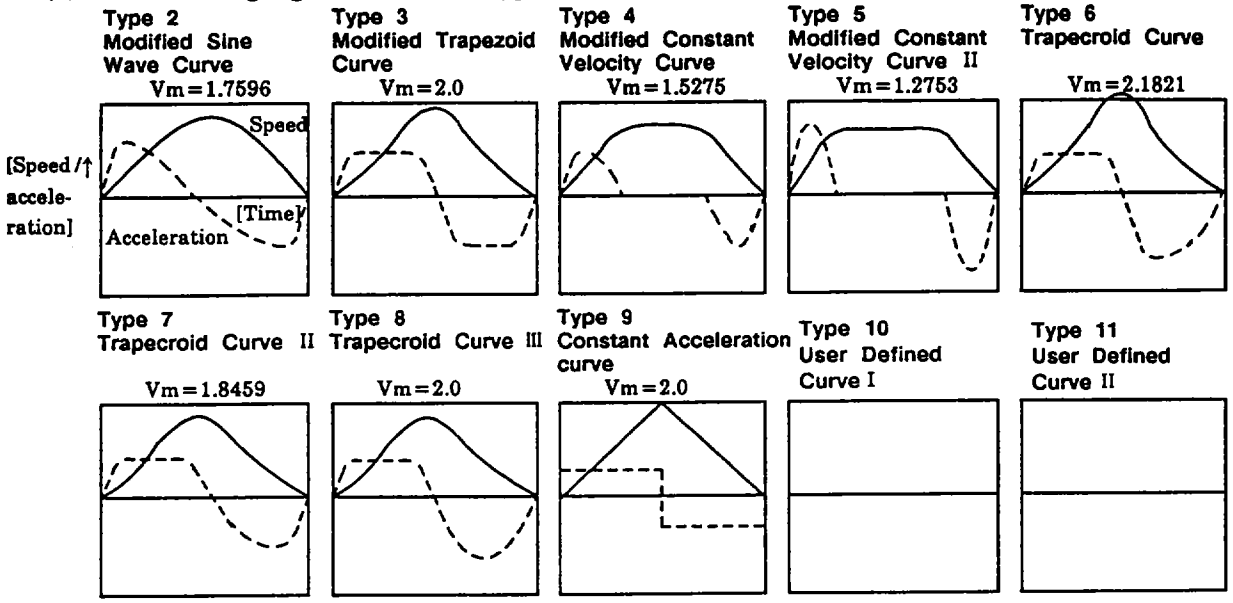


(2) Time of acceleration / deceleration (ms) is to be designated by rise up feeding time for rotation of the motor of 1rps, not by time of feeding for rotation of the motor of the rated speed.

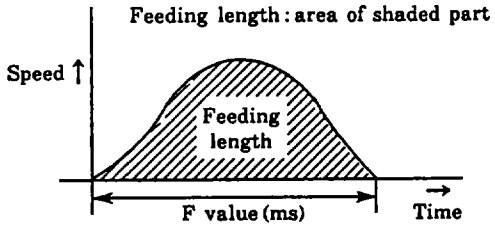


Acceleration / Deceleration Types 2 to 11 (Cam Curve)

(1) The following figures show the types of cam curves. Vm is a maximum speed of no-order.



(2) Feeding speed (F, A, B, C, and D codes) instruction of the program are to be defined in feeding time duration (ms).



(3) Feeding speed can be obtained from feeding time duration using the following formula.

$$\text{Feeding speed (pps)} = \frac{\text{Feeding length (p)}}{\text{Feeding time (s)}} \times \text{No-order maximum speed}$$

Therefore, a minimum feeding time duration which can be instructed with the program can be obtained using the following formula.

$$\text{Minimum feeding time (s)} = \frac{\text{Feeding length (p)}}{\text{Maximum feeding speed (pps)}} \times \text{No-order maximum speed}$$

(4) Cam curve unit system (The unit system of feeding time instruction) is normally specified in ms. This can be multiplied by 10 or 100.

(5) Cam curve deceleration factor is the factor of deceleration when axis feeding stop signal (emergency stop, reset, interlock halt, and error occurrence) is input. The larger the factor, feeding speed is more decelerated.

(6) The minimum feeding time duration (ms) of a cam curve is the minimum feeding time duration when axis feeding is restarted.

Acceleration / Deceleration Types 0 and 1 Operation Function

● **Function** This sets the acceleration / deceleration type of the speed curve of the motor to 3rd-order spline curve or trapezoid curve.

● **Preparation** Is resolution of the motor rotation registered?
"4.2 Operation for Installation / Type by type setting of the motor"

● **Operation**

① **Acceleration / Deceleration Type**

■ Set the acceleration / deceleration type to 3rd-spline curve or trapezoid curve.
The acceleration / deceleration type command's (@42) 1st parameter sets the acceleration / deceleration type of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.
Set the parameter of the axis to be set to 3rd-spline curve to "0".
@42 [:0] [:0] [:0] [:0]
▼
Set the parameter of the axis to be set to trapezoid curve to "1".
@42 [:1] [:1] [:1] [:1]

Memo Designation using the program:
G102 [X0] [Y0] [Z0] [W0] or
G102 [X1] [Y1] [Z1] [W1]

② **Acceleration / Deceleration Time Duration**

■ Set the acceleration / deceleration time duration (ms).
Time of acceleration / deceleration (ms) is to be designated by rise up feeding time for rotation of the motor of 1rps, not by time of feeding for rotation of the motor of the rated speed.

Memo Designate time duration of acceleration / deceleration of the DYNASERV using the following formula.

$$\text{Acceleration / deceleration time duration (ms)} \geq \frac{2\pi \times \text{load inertial force}}{0.8 \times \text{Maximum torque of the motor}}$$

The acceleration / deceleration time duration command's (@53) 1st parameter sets the acceleration / deceleration time duration of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.
Set the acceleration / deceleration time duration as the parameter of the axis to be controlled with 3rd-spline curve or trapezoid curve.

@53 [: <Acceleration / deceleration time duration of axis X (1-65535)>]
[:<Acceleration / deceleration time duration of axis Y (1-65535)>]
[:<Acceleration / deceleration time duration of axis Z (1-65535)>]
[:<Acceleration / deceleration time duration of axis W (1-65535)>]

Memo Designation using the program:

G103 [X<Acceleration / deceleration time duration of axis X (1-65535)>]
[Y <Acceleration / deceleration time duration of axis Y (1-65535)>]
[Z <Acceleration / deceleration time duration of axis Z (1-65535)>]
[W<Acceleration / deceleration time duration of axis W (1-65535)>]

Acceleration / Deceleration Types 2 to 11

Operation Function

● **Function** This sets the rotating speed of the motor to the acceleration / deceleration type controlled with the acceleration curve of cam curve.

● **Preparation** Is the motor type registered?
"4.2 Operation for Installation / Type by type setting of the motor"
Is resolution of the motor rotation registered?
"4.2 Operation for Installation / Type by type setting of the motor"

● **Operation**

① **Acceleration / Deceleration Type**

■ Set the acceleration / deceleration type to cam curve (types 2 to 11).
The acceleration / deceleration type command's (@42) 1st parameter sets the acceleration / deceleration type of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W. Set the parameter of the axis to be set to the desired cam curve to "2 to 11".

▼

@42 [:<Acceleration / deceleration type of axis X (2-11)>]
[:<Acceleration / deceleration type of axis Y (2-11)>]
[:<Acceleration / deceleration type of axis Z (2-11)>]
[:<Acceleration / deceleration type of axis W (2-11)>]

Memo Designation using the program:

G102 [X<Acceleration / deceleration type of axis X (2-11)>]
[Y <Acceleration / deceleration type of axis Y (2-11)>]
[Z <Acceleration / deceleration type of axis Z (2-11)>]
[W<Acceleration / deceleration type of axis W (2-11)>]

② **Unit System**

■ Set the cam curve unit system.
The cam curve unit system is effective on the axis whose acceleration / deceleration is to be controlled with cam curve.
The cam curve unit system selects the unit system of feeding time duration instruction (F, A, and B codes of the program). The unit system is specified normally in ms. However, this can be multiplied by 10 or 100.
The cam curve unit system command's (@43) 1st parameter sets the cam curve unit system of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.

▼

Set the parameter to "0" for selecting the normal unit system.

@43 [:0] [:0] [:0] [:0]

Set the parameter to "1" for the unit system of 10 times.

@43 [:1] [:1] [:1] [:1]

Set the parameter to "2" for the unit system of 100 times.

@43 [:2] [:2] [:2] [:2]

③ **Deceleration Factor**

■ Set the cam curve deceleration factor.
The cam curve deceleration factor is the factor of deceleration when axis feeding stop signal is input. The larger the factor, the speed is more decelerated. This is effective on the axis whose acceleration / deceleration is to be controlled with cam curve.

▼

The cam curve deceleration factor command's (@54) 1st parameter sets the cam curve deceleration factor of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.

@54 (:<Cam curve deceleration factor of axis X (0-4095)>)
(:<Cam curve deceleration factor of axis Y (0-4095)>)
(:<Cam curve deceleration factor of axis Z (0-4095)>)
(:<Cam curve deceleration factor of axis W (0-4095)>)

④ **Minimum Feeding Time Duration**

■ Set the cam curve minimum feeding time duration (ms).
The cam curve minimum feeding time duration is the factor of minimum feeding time duration when axis feeding is restarted. This is effective on the axis whose acceleration / deceleration is to be controlled with cam curve.
The cam curve minimum feeding time duration command's (@67) 1st parameter sets the cam curve minimum feeding time duration of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.

@67 (:<Cam curve minimum feeding time duration of axis X(1-65535)>)
(:<Cam curve minimum feeding time duration of axis Y (1-65535)>)
(:<Cam curve minimum feeding time duration of axis Z (1-65535)>)
(:<Cam curve minimum feeding time duration of axis W (1-65535)>)

Synchronization Control Operation Function

The synchronization control is to control, using the axis feeding instruction, the axis feeding start timing with respect to the start signal input. The synchronization control function is available with the programmed AUTO operation, programmed STEP operation, programmed CONT operation, and MDI operation. The type of synchronization control is called "synchronization type". The following four synchronization types are available.

(1) Type 0 : Synchronization disabled type
Axis feeding starts when the start signal is input.

(2) Type 1 : Start delay synchronization type
Start of axis feeding is delayed by the registered time duration (synchronization data) with respect to the start signal input.

(3) Type 2 : Time synchronization type (Effective on axis Y, axis Z, and axis W)
Start of axis Y, Z, or W feeding is delayed by the registered time duration (synchronization data) with respect to the start signal input of X axis which has been driven by the start signal input.

(4) Type 3 : Position synchronization type (Effective on axis Y, axis Z, and axis W)
Axis Y, Z, or W feeding starts when axis X is set at the registered position (synchronization data) after the start signal input of axis X which was driven by the start signal input has started.

[!] In the following operation modes, synchronization type (Types 1 to 3) are disabled, and axis feeding starts in the operation mode of Type 0 (synchronization disabled type).
(1) Origin return operation
(2) Jog operation
(3) Origin calibration operation

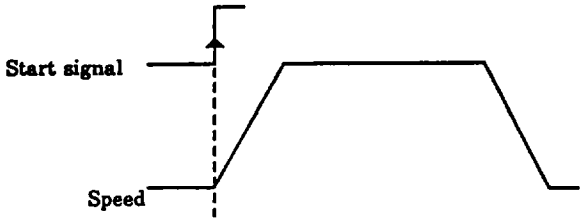
How to Designate

(1) Designation through the RS-232C mode
@44 [:<Synchronization type of axis X (0, 1)>]
[:<Synchronization type of axis Y (-3)>]
[:<Synchronization type of axis Z (-3)>]
[:<Synchronization type of axis W (-3)>]

(2) Designation by using the program
① Synchronization disabled type
G150 [X] [Y] [Z] [W]
② Start delay synchronization type
G151 [X<Synchronization data of axis X (0-999999999)>]
[Y <Synchronization data of axis Y (0-999999999)>]
[Z <Synchronization data of axis Z (0-999999999)>]
[W<Synchronization data of axis W (0-999999999)>]
③ Time synchronization type
G152 [Y<Synchronization data of axis Y (0-999999999)>]
[Z <Synchronization data of axis Z (0-999999999)>]
[W<Synchronization data of axis W (0-999999999)>]
④ Position synchronization type
G152 [Y<Synchronization data of axis Y (0-999999999)>]
[Z <Synchronization data of axis Z (0-999999999)>]
[W<Synchronization data of axis W (0-999999999)>]

Synchronization Disabled Type

(1) There is no condition of synchronization at the start of axis feeding.

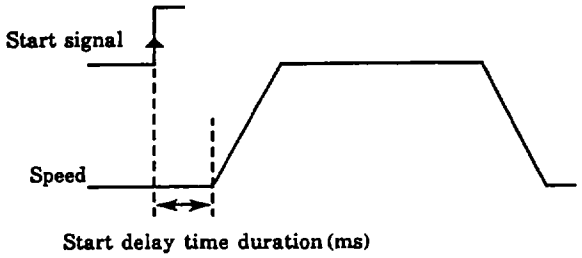


(2) This type is selected in the following operation modes.
- ① Origin return operation
- ② Jog operation
- ③ Origin calibration operation

(3) Any other type selected is disabled and this type is enabled with the input of interlock halt signal when start of axis feeding is ready in that (selected) synchronization type.

Start Delay Synchronization Type

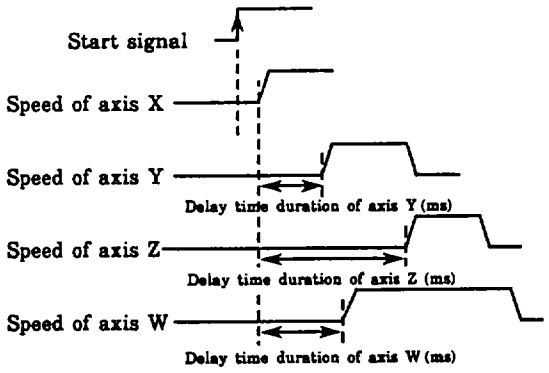
(1) Start delay time duration (ms) is used as the synchronization data.
Shaft feeding starts delayed by the synchronization data, with respect to the start signal input.



Time Synchronization Type

(1) Delay time duration (ms) of axis feeding start of axis Y, Z, or W is used as the synchronization data.
Shaft feeding of axis Y, Z, or W starts delayed by the synchronization data, with respect to the axis feeding start signal of axis X after the start signal has been input.

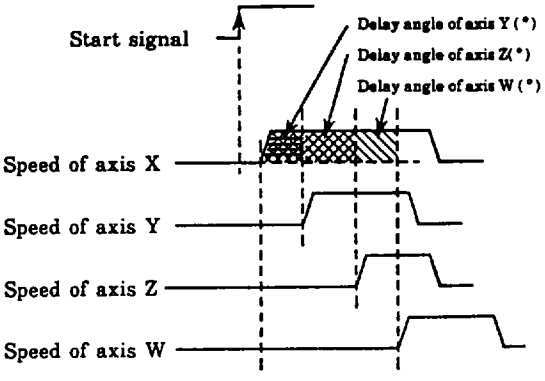
(2) This function is effective on axes Y, Z, and W.



Position Synchronization Type

(1) Delay angle (°) of axis feeding start of axis Y, Z, or W is used as the synchronization data.
Shaft feeding of axis Y, Z, or W starts when axis X is the same as the synchronization data with respect to the start signal input of axis X after the start signal input of axis X was driven by the start signal.

(2) This function is effective on axes Y, Z, and W.



Synchronization Disabled Type

● **Function** This starts axis feeding using the axis feeding instruction, with the start signal input.

● **Operation**

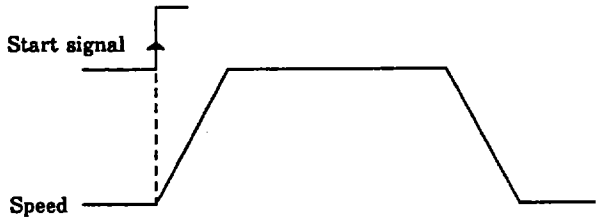
① Synchronization Type ■ Set the synchronization type to synchronization disabled type.
The synchronization type command's (@44) 1st parameter sets the synchronization type of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.
Set the parameter of the axis whose synchronization is to be disabled to "0".

@44〔:0〕〔:0〕〔:0〕〔:0〕

Memo Designation by using the program:

G150〔X〕〔Y〕〔Z〕〔W〕

● **Timing**



Start Delay Synchronization Type

● **Function** This delays the start of axis feeding instructed by the axis feeding instruction by the registered time duration with respect to the start signal input.

● **Operation**

① **Synchronization Type** ■ Set the synchronization type to start delay synchronization type.

▼

The synchronization type command's (@44) 1st parameter sets the synchronization type of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.
Set the parameter of the axis whose synchronization is to be set fro start delay synchronization type to "1".

@44 [:1] [:1] [:1] [:1]

② **Synchronization Data** ■ Set the synchronization data.

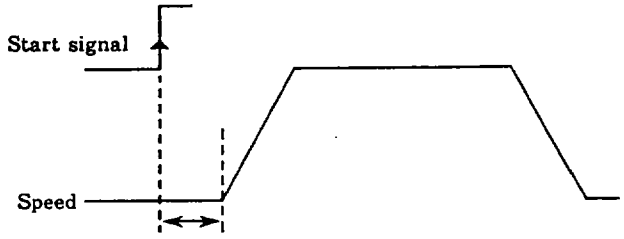
The synchronization data of the start delay synchronization type is designated by the start delay time duration (ms).
The synchronization data command's (@71) 1st parameter sets the start delay time duration of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.
Set the start delay time duration as the parameter of the axis for the start delay synchronization type.

@71 [:<Start delay time duration of axis X (0-999999999)>]
[:<Start delay time duration of axis Y (0-999999999)>]
[:<Start delay time duration of axis Z (0-999999999)>]
[:<Start delay time duration of axis W (0-999999999)>]

Memo Designation by using the program:

G151 [X<Start delay time duration of axis X (0-999999999)>]
[Y <Start delay time duration of axis Y (0-999999999)>]
[Z <Start delay time duration of axis Z (0-999999999)>]
[W<Start delay time duration of axis W (0-999999999)>]

● **Timing**



Time Synchronization Type — Operation Function

● **Function**

Start of axis feeding of axis Y, Z, or W with axis feeding instruction is delayed by the registered time duration (synchronization data), with respect to the start signal input of the axis feeding of axis X. This is effective only on axes Y, Z, and W.

● **Operation**

① **Synchronization Type**

▼

■ Set the synchronization type to time synchronization type.
The synchronization type command's (@44) 2nd parameter sets the synchronization type of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.
Set the parameter of the axis to be set for time synchronization type to "2".

@44 : [:2] [:2] [:2]

② **Synchronization Data**

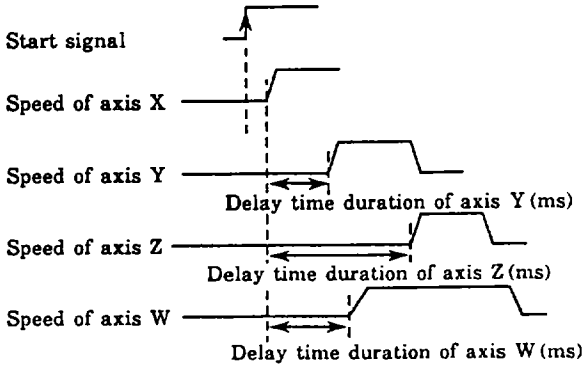
■ Designate the synchronization data.
The delay time duration (ms) is to be used for synchronization data of time synchronization type.
The synchronization data command's (@71) 2nd parameter sets the delay time duration of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.
Set the delay time duration as the parameter of the axis for the time synchronization type.

@71 : [:<Delay time duration of axis Y (0-999999999)>]
[:<Delay time duration of axis Z (0-999999999)>]
[:<Delay time duration of axis W (0-999999999)>]

Memo Designation using the program :

G152 [Y <Delay time duration of axis Y (0-999999999)>]
[Z <Delay time duration of axis Z (0-999999999)>]
[W <Delay time duration of axis W (0-999999999)>]

● **Timing**



Position Synchronization Type — Operation Function

● **Function**

Shaft feeding of axis Y, Z, or W with axis feeding instruction starts when axis X becomes the registered position (synchronization data) after axis feeding start signal input of axis X which was started with the start signal input. This is effective on axes Y, Z, and W.

● **Operation**

① **Synchronization Type**

▼

■ Set the synchronization type to position synchronization type.
The synchronization type command's (@44) 2nd parameter sets the synchronization type of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.
Set the parameter of the axis to be set for position synchronization type to "3".

@44 : [:3] [:3] [:3]

② **Synchronization Data**

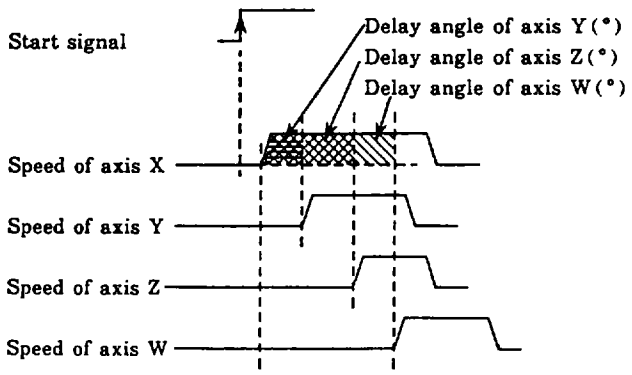
■ Designate the synchronization data.
The delay angle (°) is to be used for synchronization data of position synchronization type.
The synchronization data command's (@71) 2nd parameter sets the delay angle of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.
Set the delay angle as the parameter of the axis for the position synchronization type.

@71 : [:<Delay angle of axis Y (0-999999999)>]
[:<Delay angle of axis Z (0-999999999)>]
[:<Delay angle of axis W (0-999999999)>]

Memo Designation by using the program :

G153 [Y <Delay angle of axis Y (0-999999999)>]
[Z <Delay angle of axis Z (0-999999999)>]
[W <Delay angle of axis W (0-999999999)>]

● **Timing**



Overtravel Processing Function

Operation Function

● **Function** Rotation of the motor is decelerated and stopped when axis feeding of the motor goes in the inhibit area (within a stroke limit).

● **Operation**

① **Mechanical Direction (−)**

▼

■ Register the enabled / disabled condition of the overtravel processing function in mechanical limit (−) direction.
The overtravel (−) direction command's (@31) 1st parameter sets the enabled / disabled condition of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.
Set to "0" to disable the overtravel processing function.
Set to"1" to enable the overtravel processing function.

@31 [:<Overtravel of axis X (0, 1)>]
[:<Overtravel of axis Y (0, 1)>]
[:<Overtravel of axis Z (0, 1)>]
[:<Overtravel of axis W (0, 1)>]

② **Mechanical Direction (+)**

▼

■ Register the enabled / disabled condition of the overtravel processing function in mechanical limit (+) direction.
The overtravel (+) direction command's (@32) 1st parameter sets the enabled / disabled condition of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.
Set to "0" to disable the overtravel processing function.
Set to "1" to enable the overtravel processing function.

@32 [:<Overtravel of axis X (0, 1)>]
[:<Overtravel of axis Y (0, 1)>]
[:<Overtravel of axis Z (0, 1)>]
[:<Overtravel of axis W (0, 1)>]

③ **Soft Limit Direction (−)**

▼

■ Register the enabled / disabled condition of the soft limiter processing function in (−) direction.
The soft limit (−) direction command's (@33) 1st parameter sets the enabled / disabled condition of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.
Set to "0" to disable the overtravel processing function.
Set to "1" to enable the overtravel processing function.

@33 [:<Soft limit of axis X (0, 1)>]
[:<Soft limit of axis Y (0, 1)>]
[:<Soft limit of axis Z (0, 1)>]
[:<Soft limit of axis W (0, 1)>]

④ **Soft Limit Direction (+)**

■ Register the enabled / disabled condition of the soft limiter processing function in (+) direction.
The soft limit (+) direction command's (@34) 1st parameter sets the enabled / disabled condition of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.
Set to "0" to disable the overtravel processing function.
Set to "1" to enable the overtravel processing function.

@34 [:<Soft limit of axis X (0, 1)>]
[:<Soft limit of axis Y (0, 1)>]
[:<Soft limit of axis Z (0, 1)>]
[:<Soft limit of axis W (0, 1)>]

⑤ **Limit Value (−)**

■ Register the soft limit value of (−) direction.
The soft limit value (−) direction command's (@48) 1st parameter sets the soft limit value of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.

@48 [:<Soft limit value of axis X (−999999999-+999999999)>]
[:<Soft limit value of axis Y (−999999999-+999999999)>]
[:<Soft limit value of axis Z (−999999999-+999999999)>]
[:<Soft limit value of axis W (−999999999-+999999999)>]

⑥ **Limit Value (+)**

■ Register the soft limit value of (+) direction.
The soft limit value (+) direction command's (@49) 1st parameter sets the soft limit value of axis X, 2nd parameter sets that of axis Y, 3rd parameter sets that of axis Z, and 4th parameter sets that of axis W.

@49 [:<Soft limit value of axis X (−999999999-+999999999)>]
[:<Soft limit value of axis Y (−999999999-+999999999)>]
[:<Soft limit value of axis Z (−999999999-+999999999)>]
[:<Soft limit value of axis W (−999999999-+999999999)>]

M Output Interface — Operation Function

Using the auxiliary functions (M code and O code) in the program, BCD 2-digit code signal and strobe signal can be output to the sequencer and other external input and output devices. These signals are used to control on and off from the external input and output devices. There are two codes, M code and O code, of the auxiliary functions. When they are executed, M-output is sent to the external devices. The M-code signal is output after completion of axis feeding. The O code is output during axis feeding and after axis feeding. The output port for M code and that for O code are M OUT 0 to 7 (#42 to #49) and they are the same for each code. Timing with the external input and output devices can be adjusted using M ENABLE (#41) output terminal and M ANSWER (#12) input terminal.

Handshaking (communication using timing-controlled signals) with these external input and output devices is called "M output interface".

M Output Interface

(1) The codes which can be used for the M output interface are M codes and O codes of the auxiliary function. M codes consist of 2-digit codes of M00 to M99. O codes consist of 2-digit codes of O00 to O99.

[!] The other M codes available are M888 code and M999 code. However, these M codes are to be used for branch function, and are not to be used for handshake of M output.

"4.5 Programming Language / Branch Function"

(2) The following M codes are used for a special purpose, in addition to be used for handshaking to the external input and output devices. The initial 2 command (@9) enables or disables M00, M01, and M30 codes.

① M00 : Program halt
The block instructed with the M00 code is executed. Then, programmed operation stops. Restart operation starts the programmed operation again.

② M01 : Optional stop
The same way as the M00 code functions, the block instructed with the M01 code is executed. Then, programmed operation stops. Restart operation starts the programmed operation again.

③ M30 / M02 : End of program
These show the end of the program. Programmed operation stops.

(3) Either only the M code output (Mout signal) or both the M code and O code outputs (both Mout and Oout signals) can be selected as the signal output of the M OUT 0 to 7 (#42 to #49) output connector.
The output of these codes are a signal of BCD 2-digit code number.

#49	#48	#47	#46	#45	#44	#43	#42
M OUT 7	M OUT 6	M OUT 5	M OUT 4	M OUT 3	M OUT 2	M OUT 1	M OUT 0
Upper 4bits				Lower 4bits			

[!] BCD is abbreviated from binary coded decimal. BCD uses binary code for decimal code. Each digit of decimal code needs 4bits of binary code.

(4) The M ANSWER (#12) signal is the input signal of M completion to the M output signal. During programmed operation and after M output is executed, the program is in standby condition until the M ANSWER (#12) signal is input.

M Output Parameters — Operation Function

● **Function** This registers the parameter for M output interface.

● **Operation**

① **M Output Type** ■ Register the M signal output type.
The operation mode command's (@6) 4th parameter registers the M signal output type. Set to "0" to obtain only the Mout signal. Set to "1" to obtain both the Mout and Oout signals.
@6::::0 or @6::::1

▼

② **M Output Switch** ■ Register the M output switch.
The M output switch enables or disables the M output interface. When the M output interface is enabled, timing of the sequencer or external input and output devices can be adjusted, using the input and output terminal.
The initial 2 command's (@9) 1st parameter registers the M output switch. Set to "0" to disable the M output. Set to "1" to enable the M output.
@9:0 or @9:1

▼

③ **M01 Switch** ■ Register the M01 switch.
The M01 switch enables or disables the M01 code of the program. The M01 code is called "optional stop".
The initial 2 command's (@9) 2nd parameter registers the M01 switch. Set to "0" to disable the M01 code. Set to "1" to enable the M01 code.
@9::0 or @9::1

▼

④ **M00 Switch** ■ Register the M00 switch.
The M00 switch enables or disables the M00 code of the program. The M00 code is called "program stop".
The initial 2 command's (@9) 3rd parameter registers the M00 switch. Set to "0" to disable the M00 code. Set to "1" to enable the M00 code.
@9:::0 or @9:::1

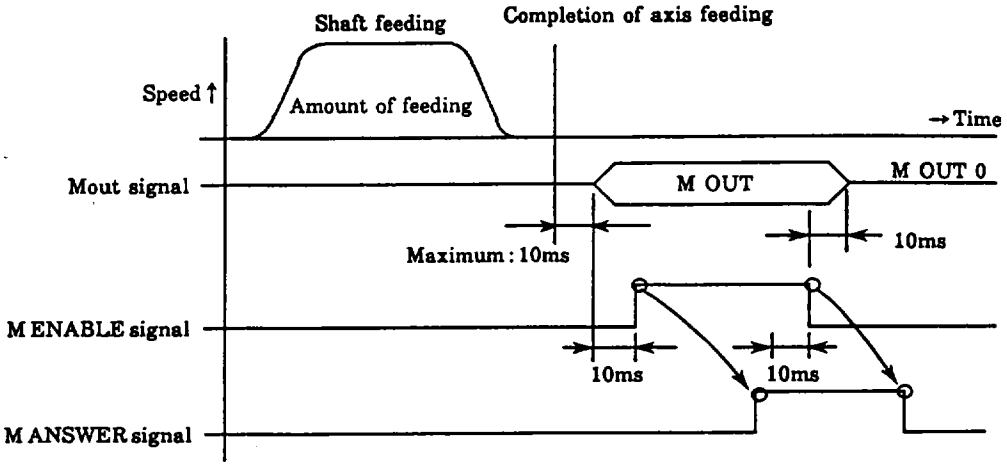
▼

⑤ **M30 Switch** ■ Register the M30 switch.
The M30 switch enables or disables the M30 code of the program. The M30 code is called "end of program".
The initial 2 command's (@9) 4th parameter registers the M30 switch. Set to "0" to disable the M30 code. Set to "1" to enable the M30 code.
@9::::0 or @9::::1

! Be sure to add the M02 code at the end of the program when the M30 code is disabled. An error occurs by ending the program with M30 code.

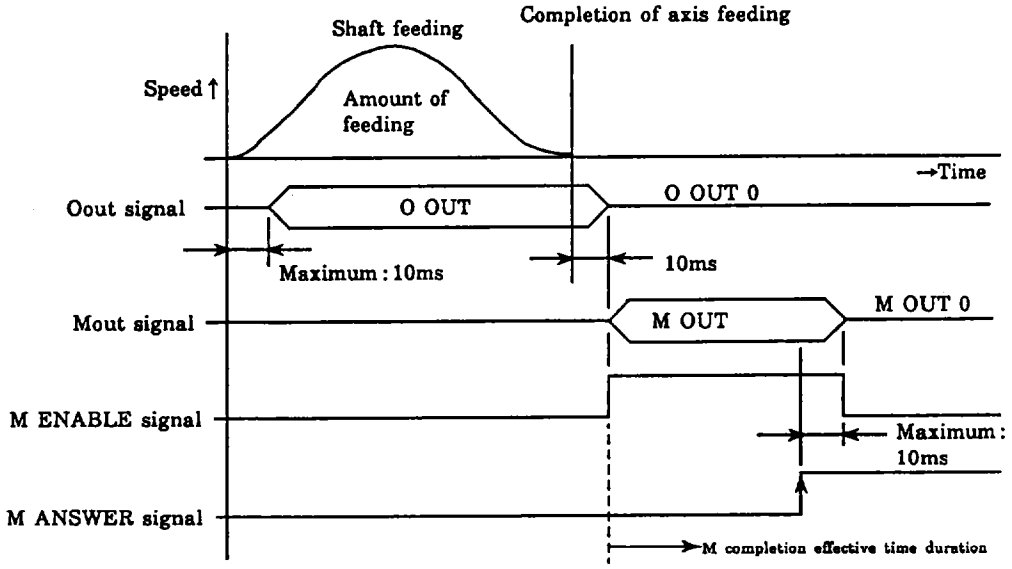
● Timing

≪Outputting Only Mout Signal≫

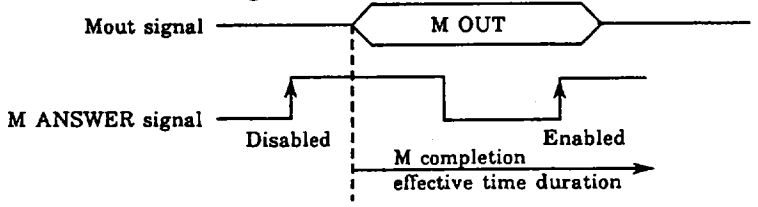


! (1) M ANSWER signal is a level-sensing signal.
(2) Set M ANSWER signal to ON after M ENABLE signal is turned ON.
(3) Set M ANSWER signal to OFF after M ENABLE signal is turned OFF.
(4) Program does not proceed to the next block until M ANSWER signal is turned OFF.

≪Outputting Both Mout and Oout Signals≫



! (1) M ANSWER signal is a edge-sensing signal.
(2) Set M ANSWER signal to ON after Mout signal is output.



(3) Program does not proceed to the next block until M ANSWER signal is turned OFF.

Operation Modes

This controller can be used in a variety of operation modes.
The seven following operation modes are available.

(1) Mode 0 : Origin return operation
Determines the mechanical coordination origin which is the reference of positioning.

(2) Mode 1 : Programmed AUTO operation
Executes the designated program from the beginning to the end.

(3) Mode 2 : Programmed STEP operation
Executes the designated program block by block from the beginning to the end.

(4) Mode 3 : Programmed CONT operation
Executes the designated program from the beginning to the end repeatedly.

(5) Mode 4 : Jog operation
Executes jog feeding to the motor.

(6) Mode 8 : Origin calibration operation
Measures the distance (the number of pulses) between the edge of the origin proximate sensor and the edge of the motor origin signal.

(7) Mode 10 : MDI operation
Inputs the program by 1 block and executes that block of the program.

! Operation modes 0 to 4 are used during movement. Refer to the following sections.
- "4.3 Operation / Origin Return Mode"
- "4.3 Operation / Programmed AUTO Mode"
- "4.3 Operation / Programmed STEP Mode"
- "4.3 Operation / Programmed CONT Mode"
- "4.3 Operation / Jog Mode"

How to Designate

(1) Designation through the RS-232C mode
@6 : <Operation modes (0-10)>

(2) Designation through the parallel transmission mode

	#9 MODE 3	#8 MODE 2	#7 MODE 1	#6 MODE 0
Origin return operation	No	No	No	No
Programmed AUTO operation	No	No	No	Yes
Programmed STEP operation	No	No	Yes	No
Programmed CONT operation	No	No	Yes	Yes
Jog operation	No	Yes	No	No
Origin calibration operation	Yes	No	No	No
MDI operation	Yes	No	Yes	No

Yes : Signal ON
No : Signal OFF

Origin Return Operation

(1) This is the operation to determine the mechanical coordination origin which is the reference of positioning. This shall be done daily at the beginning of operation.

(2) Width (mm) of the origin sensor shall be in accordance with the following formula.

$$\text{Width of origin sensor (mm)} \geq 30\,(\text{ms}) \times \frac{\text{Searching speed of origin sensor (kpps)}}{\text{Resolution of the motor (ppr)}} \times \text{circumference of the motor axis (mm/r)}$$

! The low feeding speed command (@50) registers searching speed (kpps) of the origin sensor.

(3) For origin return operation of origin return types 2 and 3, searching speed to the origin sensor shall be appropriately low so that operation can stop as desired by detecting the origin sensor.

(4) Origin calibration value (p) shall be set to 750 pulses or more. Origin calibration value means the distance between the edge of the origin proximate sensor and the edge of the motor origin signal. Origin calibration value shall be measured with origin calibration operation.

Programmed Operation

(1) Operation of mechanisms according to the program is called "programmed operation". The three following programmed operations are available.
- ① Programmed AUTO operation
- ② Programmed STEP operation
- ③ Programmed CONT operation

(2) Creating a program is called "programming". Programming is necessary before starting programmed operation. Programming can be carried out using the personal computer, operation display panel, or operation display pendant. Created program shall be stored in the controller.

"4.2 Operation for Installation / Managing Programs"

(3) Following procedures are followed by in order to change the mode of operation from programmed operation to another operation mode.

① Changing at the end of the program

In programmed AUTO operation, go to the end of the program, and then change.
In programmed STEP operation, step to the end of the program, and then change.
In programmed CONT operation, change to the programmed AUTO operation, go to the end of the program, and then change.

② Stopping programmed operation in the mid-way of programmed operation and changing to another operation mode

Stop the programmed operation in the mid-way of operation using the RESET (#10) signal, and then change.

Use the reset command (@1) in the RS-232C Mode.

(4) It is possible to change to another programmed operation during a certain programmed operation.

Origin Calibration Operation

(1) Origin calibration operation is used to measure the distance between the edge of the origin proximate sensor and the edge of the motor origin signal.

(2) The points to be measured by origin calibration operation differs from one origin return type to another. First, set the origin return type. Then, start origin calibration operation.

(3) Check the measured result of origin calibration value (p), using the message command (@0).

(4) Set the origin calibration value (p) to 750 pulses or more. If it is less than 750, align the mounting position of the origin proximate sensor, and then measure the number of pulses again.

(5) Feeding (rotation) of the motor with origin calibration operation is very slow. So, measurement shall be started close to the origin proximate sensor.

Origin Calibration Operation

Operation Mode

● **Function**

Origin calibration value means the distance between the edge of the origin proximate sensor and the edge of the motor origin signal. Adjust the position of the origin proximate sensor so that origin calibration value (p) shall be set to 750 pulses or more.

● **Preparation**

Is origin return direction registered?

"4.2 Operation for Installation / Input Unit System"

Is origin return type registered?

"4.2 Operation for Installation / Origin Return Parameters"

● **Operation**

① **Operation Mode**

■ Set the operation mode in the origin calibration operation.
Set the operation mode command's (@6) 1st parameter to "8".
@6 : 8

② **Start of Axis X**

■ Start the origin calibration operation of axis X.
Set the start command's (@3) 1st parameter to "1".
@3 : 1

③ **Completion of Axis X**

◇ This starts feeding by the motor, and measures the origin calibration value of axis X. Then, operation automatically stops.

④ **Measured Value**

■ Read the measured origin calibration value of axis X.
Set the message command's (@0) 1st parameter to "86".
@0 : 86
The read measured origin calibration value of axis X is displayed on the screen of the personal computer terminal.

! If the read value is less than 750, readjust the position of the origin proximate sensor, and then go back to step ② Start of axis X.

⑤ **Start of Axis Y**

■ Start the origin calibration operation of axis Y.
Set the start command's (@3) 2st parameter to "1".
@3 : : 1

⑥ **Completion of Axis Y**

◇ This starts feeding by the motor, and measures the origin calibration value of axis Y. Then, operation automatically stops.

⑦ **Measured Value**

■ Read the measured origin calibration value of axis Y.
Set the message command's (@0) 1st parameter to "86".
@0:86
The read measured origin calibration value of axis Y is displayed on the screen of the personal computer terminal.

! If the read value is less than 750, readjust the position of the origin proximate sensor, and then go back to step ⑤ Start of axis Y.

⑧ **Start of Axis Z** ■ Start the origin calibration operation of axis Z.
Set the start command's (@3) 3rd parameter to "1".
@3 : : : 1

▼

⑨ **Completion of Axis Z** ◇ This starts feeding by the motor, and measures the origin calibration value of axis Z. Then, operation automatically stops.

⑩ **Measured Value** ■ Read the measured origin calibration value of axis Z.
Set the message command's (@0) 1st parameter to "86".
@0 : 86
The read measured origin calibration value of axis Z is displayed on the screen of the personal computer terminal.

▼

[!] If the read value is less than 750, readjust the position of the origin proximate sensor, and then go back to step ⑧ Start of axis Z.

⑪ **Start of Axis W** ■ Start the origin calibration operation of axis W.
Set the start command's (@3) 4th parameter to "1".
@3 : : : : 1

▼

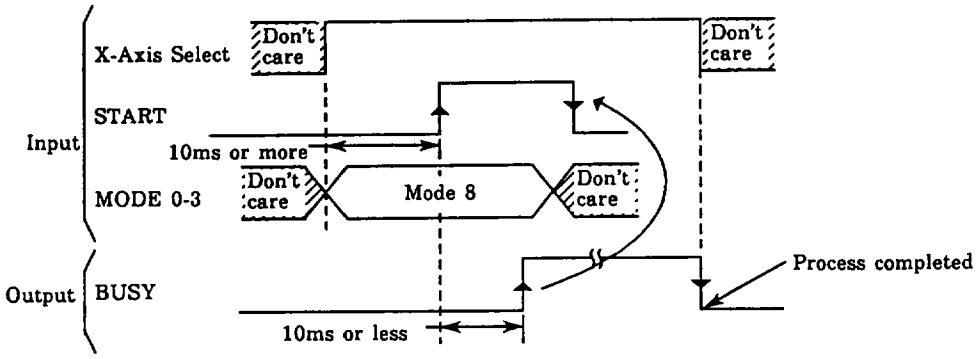
⑫ **Completion of Axis W** ◇ This starts feeding by the motor, and measures the origin calibration value of axis W. Then, operation automatically stops.

▼

⑬ **Measured Value** ■ Read the measured origin calibration value of axis W.
Set the message command's (@0) 1st parameter to "86".
@0 : 86
The read measured origin calibration value of axis W is displayed on the screen of the personal computer terminal.

[!] If the read value is less than 750, readjust the position of the origin proximate sensor, and then go back to step ⑪ Start of axis W.

● Timing



● Notes

(1) Feeding (rotation) of the motor with origin calibration operation is very slow. So, measurement shall be started close to the origin proximate sensor.

(2) The point to be measured for obtaining origin calibration operation value differs from one origin return type to another. First, set the origin return type. Then, start origin calibration operation.

MDI Operation — Operation Mode

● **Function** This is executed by entering a block of the program.

● **Preparation** Is servo set to ON?
◇ "4.2 Operation for Installation / Type by type setting of the motor"

● **Operation**

① [Operation Mode]
▼
■ Set the operation mode in the MDI operation.
Set the operation mode command's (@6) 1st parameter to "10".
@6 : 10

② [Starting]
▼
■ Load the program.
Load a block of the program using the keyboard.
◇ When loading of the program is completed, MDI operation starts automatically.

③ [Ending]
◇ That block of the program is executed, and operation stops automatically.
In order to continue MDI operation, go back to step ② Start.

● **Note** MDI operation cannot be controlled in parallel transmission mode.

Managing Programs

An instruction using NC language is given to the controller, and operates the servo motor and other devices. A set of instruction is called "program". Creating a program is called "programming". Programming is carried out using the personal computer, operation display panel, or operation display pendant. The programs created are to be stored in the controller.

The controller has a program management function. With this function, 128 types of programs can be managed using numbers. The four following types of program management operations are available.

(1) Reading programs : Reads the registered programs on the personal computer, operation display panel, or operation display pendant.
(2) Registering programs : Registers the program created with a designated program number in the controller.
(3) Deleting programs : Deletes the undesired program.
(4) Listing programs : Lists the registered program.

Reading Programs

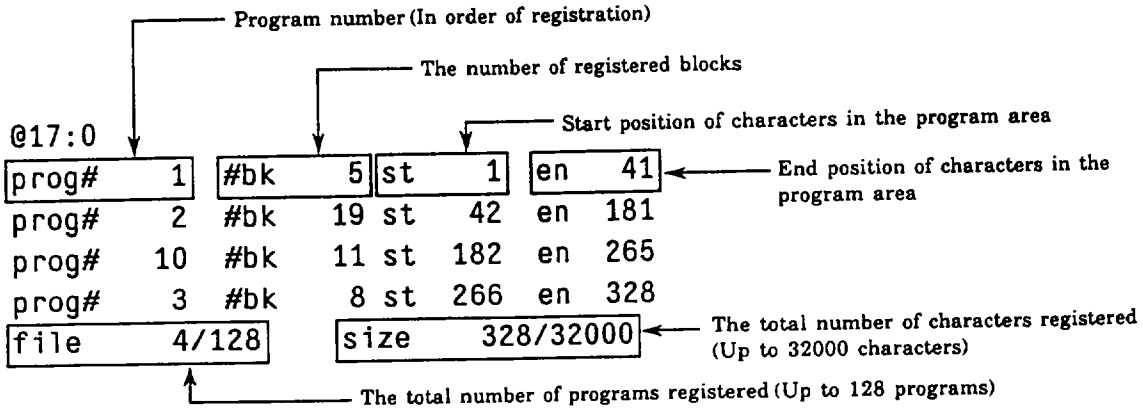
(1) Using the RS-232C mode, 128 types of programs (program number 1 to 128) can be read.
(2) Using the parallel transmission mode, 99 types of programs (program number 1 to 99) can be read.
The programs whose numbers are 100 to 128 and stored using the RS-232C mode cannot be read using the parallel transmission mode.

Registering Programs

(1) Total memory capacity of the program area is 32K-bytes (about 32000 characters). Up to 128 types of programs can be managed. However, if a program is created using 32000 characters, there is no memory space and another program cannot be created.
(2) The number of blocks of a program is not limited. However, it is confined to the space of memory capacity.

Listing Programs

(1) When designating 1 to 128 for program numbers, status of the designated programs are listed. When 0 is designated (or the number is omitted), status of all the programs are listed.
For example, designating "0" for the program number lists the following.



Listing Programs — Managing Program

● **Function** This reads the registered programs on the personal computer, operation display panel, or operation display pendant.

● **Preparation** Is program registered? "4.2 Operation for Installation / Saving Programs"

● **Operation**

① **Starting to Read**

▼

■ Designate the program number, and start reading the program.
The program number is designated with 1st parameter, and reading is started using the program read command (@14).
@14 : <Program number (1 to 128)>
◇ Starts loading of the designated program.

② **Ending**

◇ When loading of the designated program is complete, program reading ends automatically.

Saving Programs

● **Function**

This creates a program with the designated number, and registers the program in the controller, with the designated number.

● **Operation**

① **Starting to Register**

■ Designate the desired program number, and start registration of the program.
The program number is designated with 1st parameter, and registration is started using the program registration command (@15).
@15 : <Program number (1-128)>

◇ It is ready to type the desired program.

▼

② **Creating**

■ Enter the program.
The desired program can be entered block by block using the keyboard.

▼

③ **Ending**

■ End the registration of the program.
Type as follows from the keyboard.
END

◇ Operation exits from the program input status.

● **Note**

Once the controller is set in the program input status by the program registration command (@15), any characters entered from the keyboard are regarded as a part of the program until you finish the registration of the program and exit from the program. If you use a command before ending the program registration, the command is regarded as a block of the program.

Deleting Programs — Managing Program

● **Function** This deletes the programs which are not necessary.

● **Preparation** Is program registered? "4.2 Operation for Installation / Saving Programs"

● **Operation**

① **Starting to Delete**

▼

■ Designate the program number with which you want to delete the program, and start deletion.
The program number is designated with 1st parameter, and deletion is started using the program deletion command (@16).
@16 : <Program number (1-128)>
◇ This starts deletion of the designated program.

② **Ending**

◇ When the designated program has been deleted, operation of program deletion ends automatically.

Program Directory — Managing Program

● **Function** This lists the status of the registered programs.

● **Preparation** Is program registered? ◇"4.2 Operation for Installation / Saving Programs"

● **Operation**

① Starting to List

■ Designate the desired program number, and start to list the programs.
The program number is designated with 1st parameter, and listing is started using the program list command (@17).
To list the status of a program, designate the program number of that program to 1st parameter. To list the status of all the registered programs, designate "0" (or omit parameter designation).

@17 : <Program number (1-128)>

▼

◇ This starts transferring of the designated program or all the programs to be listed.

② Ending

◇ When the designated program or all the programs have been transferred, operation of program listing ends automatically.

4.3 Operation

This section explains functions, notes, and how to operate the controller.

Operation is carried out normally in the parallel transmission mode. So, how to operate is explained mainly in the parallel transmission mode. However, the controller can be operated in the RS-232C mode. Operation in the RS-232C mode is explained only by using command descriptions.

How to Read the Explanations of Operation

- **Function** Describes the function of operations simply.
- **Preparation** Explains necessary preparation before using the function.
- **Operation** Describes how to use the function in detail.
  Description conventions of parallel transmission mode are as follows:
  (1) The number that follows "#" is the pin number of connector <CN1> of the controller.
  (2) The name of the signal assigned to the pin is shown in the raw below the pin number in tables.
  (3) In tables, "ON" means that the signal is to be set to ON, and "OFF" means that the signal is to be set to OFF.
- **Timing** Shows the timing chart of operations of the function.
- **Note** Explains notes on the use of the function.

Parallel Transmission Mode

● **Function** This operates the controller through the parallel transmission line or the switches.

● **Preparation** Are parameters and programs registered? "4.2 Operation for Installation"

● **Operation** Using the parallel transmission mode, each operation of the motor can be controlled from the sequencer or the switches. Operation is normally carried out using the parallel transmission mode.

① **Transmission Mode**

■ Set the transmission mode in the parallel transmission mode.
Set the transmission mode command's (@5) 1st parameter to "1".

@5 : 1

◇ The PLC lamp on the front panel of the controller lights up.

Memo Besides the parallel transmission mode, the RS-232C mode is available. Operation for installation shall be carried out using the RS-232C mode.

"4.2 Operation for Installation"

Origin Return Mode — Operation Mode

● Function

This determines the mechanical coordination origin which is the reference of positioning.
In order to determine the origin, the origin sensor and the origin pulse output (Z phase output) from the motor are used.

● Notes

Is origin return direction registered?
◇"4.2 Operation for Installation / Origin Return Parameters"

Is origin return type registered?
◇"4.2 Operation for Installation / Origin Return Parameters"

Is origin return offset value registered?
◇"4.2 Operation for Installation / Origin Return Parameters"

Is coordination after origin return registered?
◇"4.2 Operation for Installation / Origin Return Parameters"

Is origin sensor search speed registered?
◇"4.2 Operation for Installation / Feeding Speed Parameters"

Is resolution of the motor registered?
◇"4.2 Operation for Installation / Type by Type Setting of the Motor"

● Operation

① **Operation Mode**

■ Set the operation mode to origin return mode.
This operation mode can be designated by the following MODE 0 to 3 (#6 to #9) signals combination.

#9	#8	#7	#6
MODE 3	MODE 2	MODE 1	MODE 0
OFF	OFF	OFF	OFF

@6:0 for the RS-232C Mode

▼

② **Selection of Origin Return Axis**

■ Select the axis subject to the origin return operation.
Any number of axes, one or four, can be selected at the same time for origin return operation. Use the X-AXIS SELECT (#14), Y-AXIS SELECT (#15), Z-AXIS SELECT (#16), and W-AXIS SELECT (#17) signals to select the desired axes.
The X-AXIS SELECT (#14) signal selects axis X.

#14
X-AXIS SELECT
ON

The Y-AXIS SELECT (#15) signal selects axis Y.

#15
Y-AXIS SELECT
ON

The Z-AXIS SELECT (#16) signal selects axis Z.

#16
Z-AXIS SELECT
ON

The W-AXIS SELECT (#17) signal selects axis W.

#17
W-AXIS SELECT
ON

▼

③ **Starting**

■ Start the origin return operation.
The START (#4) signal starts the origin return operation.

#4
START
ON

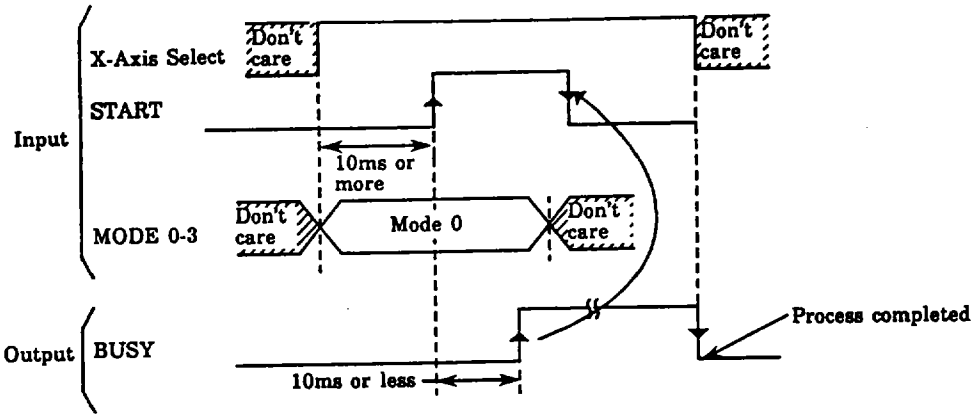
@3 [: 1] [: 1] [: 1] [: 1] for the RS-232C Mode.

▼

④ **Ending**

◇ The motor starts rotating and the origin of the axis selected in step ② Operation is determined. Then, operation ends automatically.

● Timing



● Note

For operation of origin return type 2 and 3, designate an appropriately low searching speed to the origin sensor so that the motor can stop as desired by detecting the origin sensor.
If you designate the speed with which the origin sensor cannot stop at the origin sensor, an error message (errors 60 to 63) is displayed during origin return operation and operation stops.

Programmed AUTO Mode

Operation Mode

● **Function** Executes the designated program to the end of the program (M30 code).

● **Preparation** Is servo turned ON?
"4.2 Operation for Installation / Type by type setting of the motor"
Is program registered? "4.2 Operation for Installation / Registering programs"

● **Operation**

① **Operation Mode**

■ Set the operation mode in the programmed AUTO mode.
This operation mode can be designated by the following MODE 0 to 3 (#6 to #9) signals.

#9	#8	#7	#6
MODE 3	MODE 2	MODE 1	MODE 0
OFF	OFF	OFF	ON

@6:1 for the RS-232C Mode

② **Program Number**

■ Select the desired program number.
The range of the program numbers which can be designated is 1 to 99 of 2-digit BCD.
The DATA 0 to 7 (#26 to #33) signals designate the desired program number.

#33	#32	#31	#30	#29	#28	#27	#26
DATA 7	DATA 6	DATA 5	DATA 4	DATA 3	DATA 2	DATA 1	DATA 0
				Higher 4bits			
Lower 4bits							

Memo BCD is abbreviated from binary coded decimal. Decimal is described using 4-bit of BCD for 1 digit of decimal.

③ **Starting**

■ Start programmed operation.
The START (#4) signal starts the programmed operation.

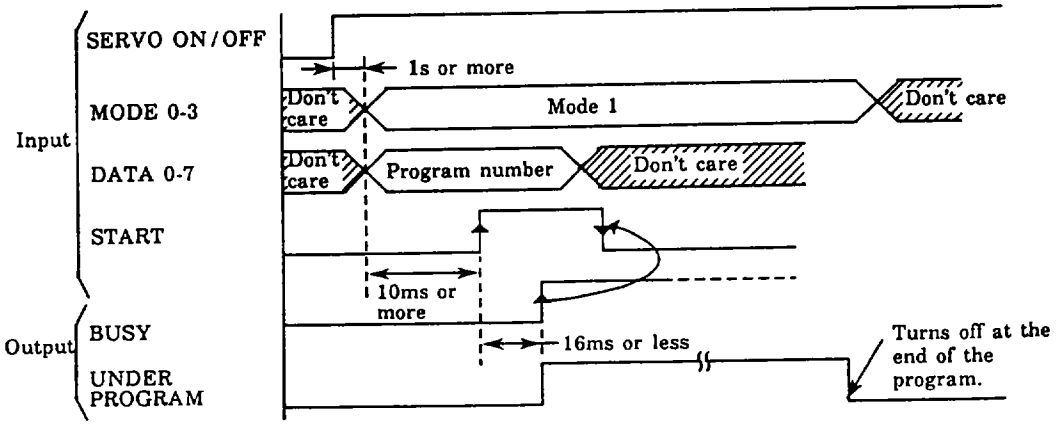
#4
START
ON

@3:<Program Number (1-128)> for the RS-232C Mode

④ **Ending**

◇ Executing the end of program (M-30 code) stops operation automatically.

● **Timing**



Programmed STEP Mode

Operation Mode

● **Function**

Executes the designated program block by block.
By executing the end of program (M30 code), the program returns to the beginning of the program.

● **Preparation**

Is servo turned ON?
◇"4.2 Operation for Installation / Type by type setting of the motor"
Is program registered? ◇"4.2 Operation for Installation / Saving Programs"

● **Operation**

① **Operation Mode**

■ Set the operation mode in the programmed STEP mode.
This mode can be designated by the following signal combination of MODE 0 to 3 (#6 to #9).

#9	#8	#7	#6
MODE 3	MODE 2	MODE 1	MODE 0
OFF	OFF	ON	OFF

@6:2 for the RS-232C Mode

▼

② **Program Number**

■ Select the desired program number.
The range of the program numbers which can be designated is 1 to 99 of 2-digit BCD. The DATA 0 to 7 (#26 to #33) signals designate the desired program number.

#33	#32	#31	#30	#29	#28	#27	#26
DATA 7	DATA 6	DATA 5	DATA 4	DATA 3	DATA 2	DATA 1	DATA 0
				Lower 4bits			Higher 4bits

▼

Memo BCD is abbreviated from binary coded decimal. Decimal is described using 4-bit of BCD for 1 digit of decimal.

③ **Starting**

■ Start programmed operation.
The START (#4) signal starts the programmed operation.

#4
START
ON

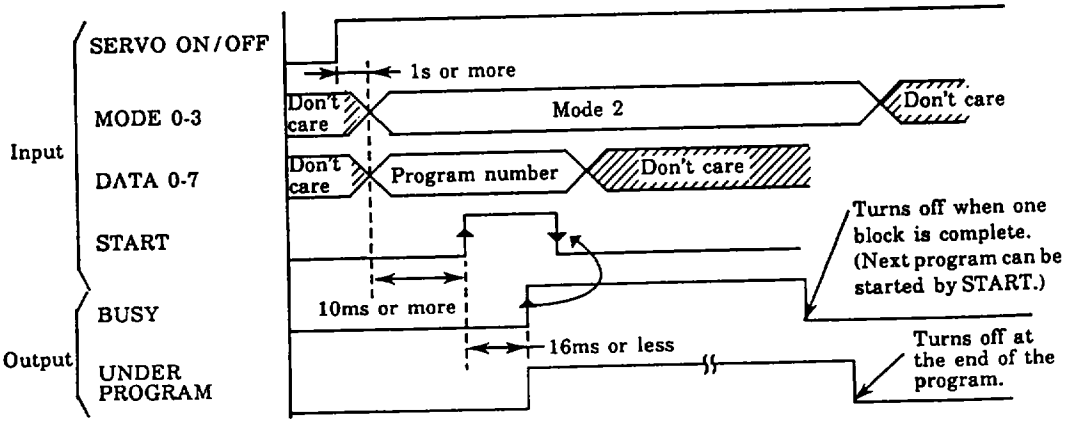
@3 : <Program Number (1-128)> for the RS-232C Mode

▼

④ **Ending**

◇ Executing the end of program (M-30 code) stops operation automatically.
If this is during programmed operation, the program goes back to step ③ of Operation.

● **Timing**



Programmed CONT Mode

Operation Mode

● **Function** Executes the designated program repeatedly from the beginning to the end (M-30 code).

● **Preparation** Is servo turned ON?
Is program registered?
◇"4.2 Operation for Installation / Type by type setting of the motor"
◇"4.2 Operation for Installation / Registering programs"

● **Operation**

① **Operation Mode**

■ Set the operation mode in the programmed CONT mode.
This operation mode can be designated by the following MODE 0 to 3 (#6 to #9) signals.

#9	#8	#7	#6
MODE 3	MODE 2	MODE 1	MODE 0
OFF	OFF	ON	ON

@6 : 3 for the RS-232C Mode

② **Program Number**

■ Select the desired program number.
The range of the program numbers which can be designated is 1 to 99 of 2-digit BCD.
The DATA 0 to 7 (#26 to #33) signals designate the desired program number.

#33	#32	#31	#30	#29	#28	#27	#26
DATA 7	DATA 6	DATA 5	DATA 4	DATA 3	DATA 2	DATA 1	DATA 0

Lower 4bits Higher 4bits

Memo BCD is abbreviated from binary coded decimal. Decimal is described using 4-bit of BCD for 1 digit of decimal.

③ **Starting**

■ Start programmed operation.
The START (#4) signal starts the programmed operation.

#4
START
ON

@3 : <Program Number (1-128)> for the RS-232C Mode

④ **Ending**

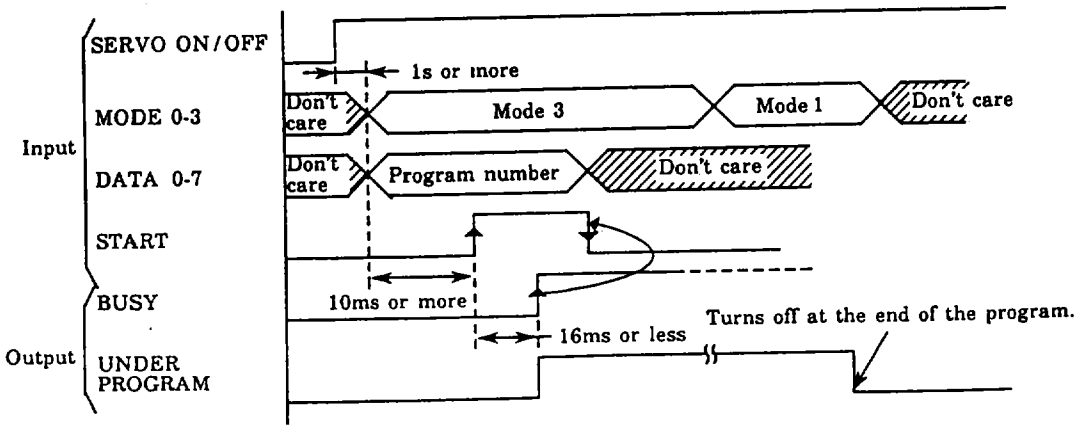
■ While the program is being executed, set the operation mode in the programmed AUTO operation.
The MODE 0 to 3 (#6 to #9) signals designate the desired operation mode.

#9	#8	#7	#6
MODE 3	MODE 2	MODE 1	MODE 0
OFF	OFF	OFF	ON

@6 : 1 for the RS-232C Mode

◇ Executing the end of program (M-30 code) stops operation automatically.

● **Timing**



Jog Mode — Operation Mode

● **Function** This executes jog feeding by the motor.

● **Preparation** Is low feeding speed registered?
"4.2 Operation for Installation / Feeding Speed Parameters"
Is high feeding speed registered?
"4.2 Operation for Installation / Feeding Speed Parameters"

● **Operation**

① **Operation Mode**

■ Set the operation mode in the jog mode.
The MODE 0 to 3 (#6 to #9) signals in the table below designate the job mode.

#9 MODE 3	#8 MODE 2	#7 MODE 1	#6 MODE 0
OFF	ON	OFF	OFF

@6 : 1 for the RS-232C Mode

▼

② **Feeding Speed**

■ Select feeding speed.
The feeding speed can be selected from two speeds, high feeding speed (HIGH) and low feeding speed (LOW).
The JOG HIGH / LOW (#13) signal selects the feeding speed.

	#13 JOG HIGH/LOW
High speed feeding	ON
Low speed feeding	OFF

Setting of the Jog Feeding Speed Command (@64) is Used for the RS-232C Mode

▼

③ **Starting**

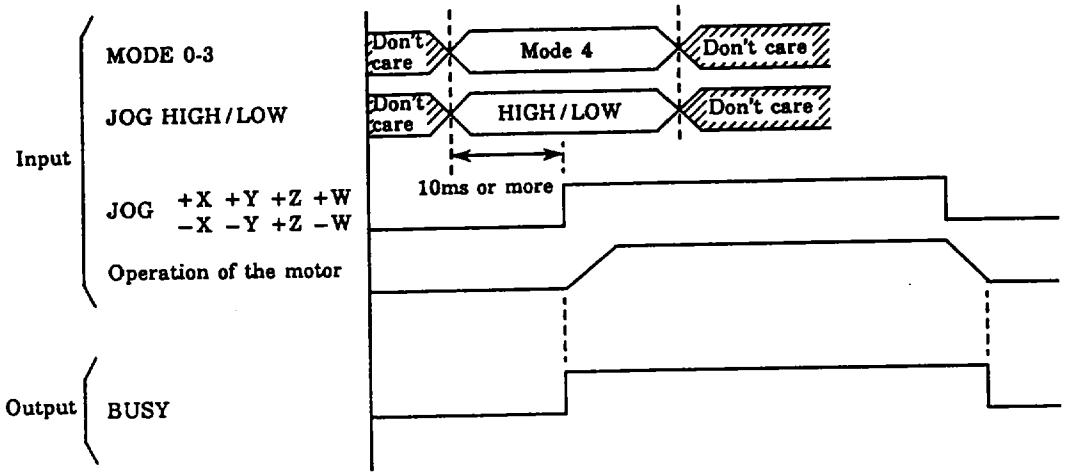
■ Select the axis and direction subject to jog feeding, and then start the jog operation.
Select one of the followings : JOG +X (#18), JOG −X (#19), JOG +Y (#20), JOG −Y (#21), JOG +Z (#22), JOG −Z (#23), JOG +W (#24), or JOG −W (#25), for determining the axis and direction of jog feeding.
Selecting the axis and direction of jog feeding starts operation.

	#18 JOG +X	#19 JOG −X	#20 JOG +Y	#21 JOG −Y	#22 JOG +Z	#23 JOG −Z	#24 JOG +W	#25 JOG −W	For the RS-232C Mode
Axis X + direction	ON	OFF	OFF	OFF	OFF	OFF	OFF	OFF	@3 : 1
Axis X − direction	OFF	ON	OFF	OFF	OFF	OFF	OFF	OFF	@3 : −1
Axis Y + direction	OFF	OFF	ON	OFF	OFF	OFF	OFF	OFF	@3 : , 1
Axis Y − direction	OFF	OFF	OFF	ON	OFF	OFF	OFF	OFF	@3 : , −1
Axis Z + direction	OFF	OFF	OFF	OFF	ON	OFF	OFF	OFF	@3 : , , 1
Axis Z − direction	OFF	OFF	OFF	OFF	OFF	ON	OFF	OFF	@3 : , , −1
Axis W + direction	OFF	OFF	OFF	OFF	OFF	OFF	ON	OFF	@3 : , , , 1
Axis W − direction	OFF	OFF	OFF	OFF	OFF	OFF	OFF	ON	@3 : , , , −1

④ **Ending** ■ Turn off the axis / direction signal of feeding selected at step ❸ of operation.

For the RS-232C Mode
(1) Axis X Feeding ···@3[,0]
(2) Axis Y Feeding ···@3[,,0]
(3) Axis Z Feeding ···@3[,,,0]
(4) Axis W Feeding ···@3[,,,,0]

● **Timing**



Interlock Halt — Operation Function

● **Function**

When interlock halt occurs during axis feeding of the motor, rotation of the motor decelerates and stops.
When interlock halt is released, axis feeding of remaining amount is carried out.

● **Operation**

≪Interlock Halt During Axis Feeding≫

① **Starting**

■ Interlock halt starts during axis feeding.
The INTERLOCK (#11) signal starts interlock halt.

▼

#11
INTERLOCK
ON

◇ Shaft feeding decelerates and stops.

② **Ending**

■ Interlock halt is released.
The INTERLOCK (#11) signal releases interlock halt.

#11
INTERLOCK
OFF

◇ Remaining axis feeding (amount which should be fed in normal condition) restarts.

≪Interlock Halt when Axis Feeding is Stopped≫

① **Starting**

■ Interlock halt starts when axis feeding is stopped.
The INTERLOCK (#11) signal starts interlock halt.

▼

#11
INTERLOCK
ON

② **Starting of Axis Feeding**

■ Shaft feeding starts with programmed operation, etc.
The START (#4) signal starts axis feeding.

▼

#4
START
ON

◇ Shaft feeding does not start yet.

③ **Ending**

■ Release the interlock halt.
The INTERLOCK (#11) signal releases interlock halt.

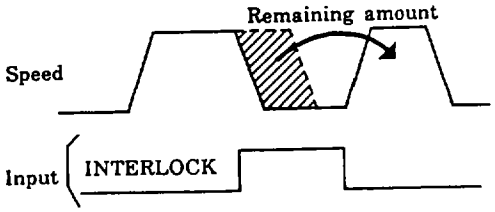
#11
INTERLOCK
OFF

◇ Shaft feeding starts.

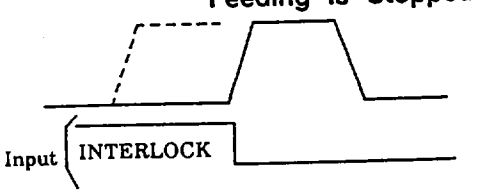
Memo Even when synchronization types are set to 1 to 3, they are not effective. Shaft feeding starts when interlock halt is released.

● **Timing**

≪Interlock Halt During Axis Feeding≫



≪Interlock Halt when Axis Feeding is Stopped≫



Emergency Stop — Operation Function

● **Function**

Emergency stop stops all operations and turns off servo.
During execution of axis feeding of the motor, rotation of the motor is decelerated and stops. Servo is also turned off.

● **Operation**

① **Starting** ■ The EMG (#2) signal activates emergency stop.

#2
EMG
ON

▼

◇ All operations stop. During axis feeding, operation decelerates and stops. Then, servo is turned off.

◇ The error display LED on the front panel flashes.

② **Ending** ■ Emergency stop is released.

The EMG (#2) signal and the RESET (#10) signal release emergency stop.
First, turn off the EMG (#2) signal. Then, turn on the RESET (#10) signal once, and turn it off.

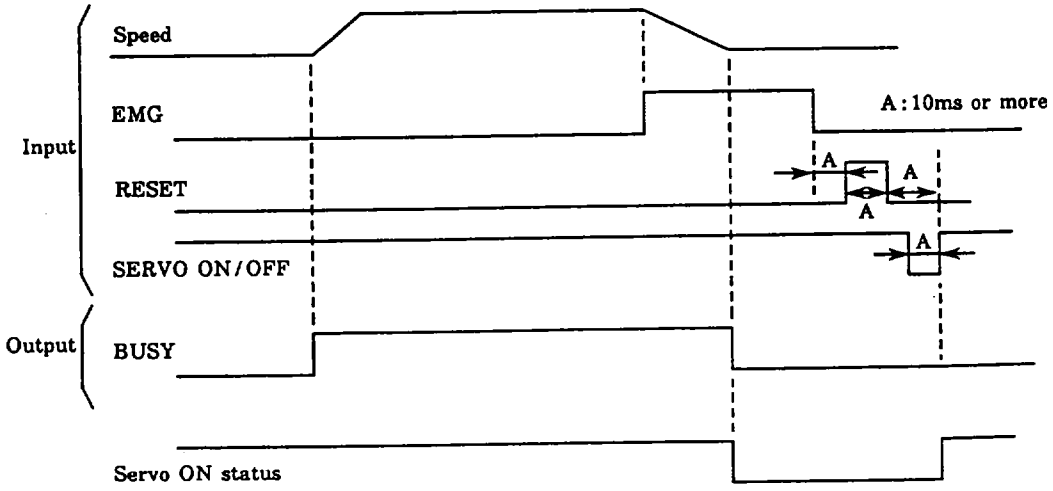
#2		#10		#10
EMG	⇨	RESET	⇨	RESET
OFF		ON		OFF

◇ Condition is reset to initial condition. Servo remains off disregarding the status of the SERVO ON / OFF (#3) signal.

Memo In order to turn on servo, turn off the SERVO ON / OFF (#3) signal once. Then, turn on it.

#3		#3
SERVO ON/OFF	⇨	SERVO ON/OFF
OFF		ON

● **Timing**



4.4 Commands

This section explains the commands available with the controller. It explains functions, format, descriptions, etc. of the commands. When a command is input, a return message is sent back.

The messages started with "R" mean operation is successfully completed. The messages started with "E" mean operation is not completed resulting in errors.

"4.4 Commands" "Appendix (2) List of Errors and Alarms"

Return message [R00 ready] means the desired operation has been completed with correct execution of the command.

How to Read Explanations of Commands

Function Explains the functions of the commands simply.

Format Explains how to script the commands. Follow the following rules when entering a command.

(1) Items of @ and numbers shall be entered as they are. They cannot be omitted.

(2) Items in "< >" are numbers of integer. The range of the numbers differs from command to command. The range is shown by the item in "( )".

(3) Items in "[ ]" can be omitted. When omitted, default values (the values initially set in the controller) or last entered values will be used.
There are multiple parameters separated with colon " : ". Colon can be omitted in the following cases.

① When omitting parameters that follow a certain parameter, those parameters, including colon, can be omitted.

② When omitting the parameters in between parameters, the colon before the omitted parameters cannot be omitted.

Example Shows a simple example of input.

Initial Value Shows the initial value (default value) of the parameter. This also shows examples of input.

Description Explains the functions of the commands in detail.

Note Describes notes on the use of the commands.

Reference Shows items related to the commands.

Messages

Command

Function Calls the status of the internal parameters of the controller.

Format @[0] : < Parameter number (0-9, 20-99)>

Example @0:5
@:40

Description The internal parameters of the controller can be called in order to check the status of the controller.
This command number can be omitted.
The data, sent from the controller, is called "return data". Return data is displayed on screen of the personal computer terminal, operation display panel, or operation display pendant. Format of the return data is one of the following formats, depending on the types of internal parameters called. Numeric data is displayed in hexadecimal notation.

(1) R01__ [8 characters of message] : OOH CR(, LF)
(2) R02__ [8 characters of message] : OOH:OOH:OOH:OOH CR(, LF)
(3) R04__ [8 characters of message] : OOOOOOOO:OOOOOOOO
: OOOOOOOO:OOOOOOOO CR(,LF)
(4) R09__DI/O___in__OO__OO__OO__OO__out__OO__OO CR(,LF)

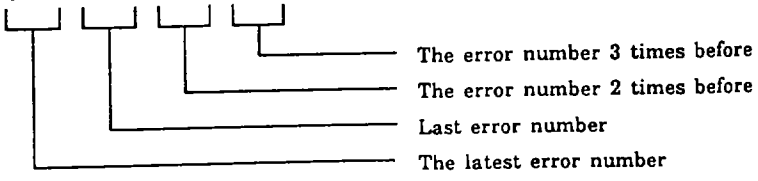
<Parameter number> shows the types of internal parameters. Setting value, meaning, and format of <Parameter number> are as follows:

Setting Value	Meaning	Format
0	Error number when halt by errors	R02
1	Enabled axes	R01
2	Axes of origin return completion	R01
3	Operation condition	R02
4	(Reserved)	---
5 to 9 20 to 79	Status of the parameter of command number	R02, R04
80	Mechanical coordination	R04
81	Programmed coordination	R04
82	(Reserved)	---
83	Absolute coordination	R04
84	Base coordination	R04
85	Encoder coordination	R04
86	Origin calibration value	R04
87 to 98	(Reserved)	---
99	DI/O status	R09

Each return data is as follows:

(1) Error number display request by (@ [0] :0) when operation halt by errors

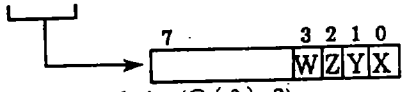
R02__error__hs:OOH:OOH:OOH:OOH



(2) Enabled axes (@ (0) : 1)

R01__valid__ax : ○○H

This can be checked by converting hexadecimal number to binary number.

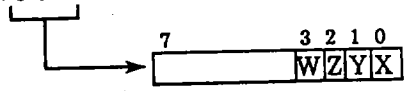


0 : Disabled
1 : Enabled

(3) Axes of origin return completion (@ (0) : 2)

R01__org__fini : ○○H

This can be checked by converting hexadecimal number to binary number.



0 : Disabled
1 : Enabled

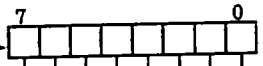
(4) Operation condition (@ (0) : 3)

R02__status____ : ○○H : ○○H : ○○H : ○○H

Error stop condition
An error number is displayed in an error condition when it is necessary to reset. Otherwise, "0" is displayed.

Operation mode change type
0: Changeable
1: Changeable during programmed operations
2: Changeable after execution completed
3: Not changeable

Operation mode number



Operation condition
Request to the next block during programmed operation
(Reserved)
In execution (For all operation modes)
In programmed operation
Stop condition in programmed operation
Jog operation / high speed feeding
(Reserved)
In programmed operation / data set

(5) Coordination value call (@ (0) : < Types of coordination (80-85) >)

R02__xxxxxxxx : ○○○○○○○○ : ○○○○○○○○ : ○○○○○○○○ : ○○○○○○○○

Coordination value of axis X
Coordination value of axis Y
Coordination value of axis Z
Coordination value of axis W

(6) Origin calibration value (@ (0) : 86)

R02__calib__dt : ○○○○○○○○ : ○○○○○○○○ : ○○○○○○○○ : ○○○○○○○○

Origin calibration value of axis X
Origin calibration value of axis Y
Origin calibration value of axis Z
Origin calibration value of axis W

(7) DI / O status (@ (0) : 99)

R09__DI / O____in__○○__○○__○○__○○__out__○○__○○ ..... Displayed by groups of 2 digits in hexadecimal number

① ② ③ ④ ⑤ ⑥

Input signal (#2-#33) of connector <CN1>
Output signal (#34-#49) of connector <CN1>

① #9-#2
② #17-#10
③ #25-#18
④ #33-#26
⑤ #41-#34
⑥ #49-#42

(Higher pin number is for high-order bit.)

Reference

"4.4 Commands / Clearing Coordination Systems (@4)"
"4.4 Commands / Operation Mode (@6)"
"4.4 Commands / Initial 0 (@7)"
"4.2 Operation for Installation / Coordination System"
"4.2 Operation for Installation / Origin Calibration Operation"
"4.3 Daily Operation / Origin Return Operation"
"Appendix (1) List of Commands"
"Appendix (2) List of Errors and Alarms"

Resetting Error Halt

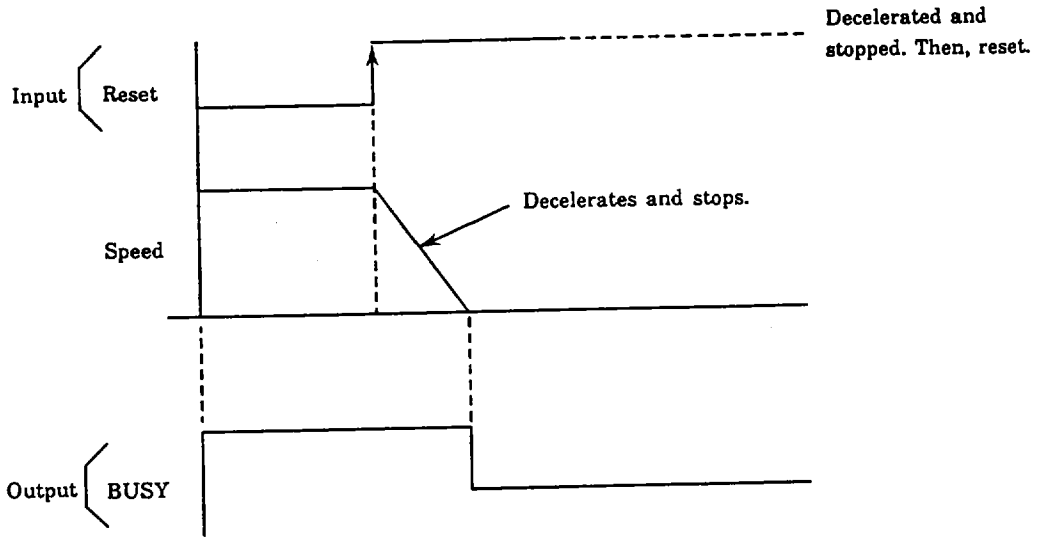
Command

Function Resets the controller from the error halt status.

Format @ 1

Description The controller goes into a halt status when an error occurs. To release the controller from the error halt status, use the reset command or use power-on reset (turn on power after turn off power once).
The controller is put in initial status by the reset command.
The coordination system clear switch (the initial 1 command's (@8) 3rd parameter) can set whether to clear the coordination system or not when the reset command is executed.

≪ Resetting during Feeding ≫



Reference "4.4 Commands / Initial 1 (@8)",
"4.2 Operation for Installation / Coordination System"
"Appendix (2) List of Errors and Alarms"

Stopping Operation

Command

Function Stops the operation when this command is executed after completing operation of the block of that time, during execution of the program.

Format @2

Initial This command is effective in programmed AUTO mode or in programmed CONT mode.

When this command is executed in the program, the operation stops after completing operation of the block being operation. Use the start command (@3) to restart the programmed operation.

Reference
- "4.4 Commands / Starting Operation (@3)"
- "4.2 Operation for Installation / Operation Mode"
- "4.3 Operation / Programmed AUTO Mode"
- "4.3 Operation / Programmed CONT Mode"

Starting Operation

Command

Function Starts operation.

Format
(1) @3 [: <Designation of axis X (1)>] [: <Designation of axis Y (1)>]
[: <Designation of axis Z (1)>] [: <Designation of axis W (1)>]
(2) @3 [: <Program number (1-128)>]
(3) @3 [: <Designation of axis X and feeding direction (−1, 0, 1)>]
[: <Designation of axis Y and feeding direction (−1, 0, 1)>]
[: <Designation of axis Z and feeding direction (−1, 0, 1)>]
[: <Designation of axis W and feeding direction (−1, 0, 1)>]

Example
(1) @3: :1
(2) @3: 15
(3) @3: −1

Description This command is effective in origin return operation, programmed AUTO operation, programmed STEP operation, programmed CONT operation, jog operation, and origin calibration operation.
Use this command to start operation (feeding) after designating the desired operation mode and completing necessary preparation.

Use Format (1) for origin return operation and origin return calibration operation. Setting <Designation of axis X> to 1 starts the origin return operation or origin calibration operation of axis X. Setting <Designation of axis Y>, <Designation of axis Z>, or <Designation of axis W> to 1 starts the origin return operation or origin calibration operation of axis Y, Z, or W respectively.

Use Format (2) for programmed operations (programmed AUTO mode, programmed STEP mode, and programmed CONT mode). This starts operation of the program registered with <Program number>. <Program number> is stored in the controller. When <Program number> is not designated, last designated <Program number> will be used.

Use format (3) for jog mode. Designate the desired axis with <Designation of axis and feeding direction>. More than one axis cannot be designated at the same time. Following shows the setting values and operation of <Designation of axis and feeding direction>.

1	:	Starts jog operation in + (normal) direction.
−1	:	Starts jog operation in − (reverse) direction.
0 or none	:	Stops jog operation.

Reference
"4.2 Operation for Installation / Operation Modes"
"4.2 Operation for Installation / Origin Calibration Operation"
"4.3 Operation / Origin Return Operation"
"4.3 Operation / Programmed AUTO Mode"
"4.3 Operation / Programmed STEP Mode"
"4.3 Operation / Programmed CONT Mode"
"4.3 Operation / Jog Mode"

Clearing Coordination Systems Command

Function Clears all the coordination systems of the controller.

Format
@4 [: <Clears coordination system of axis X (0, 1)>]
[: <Clears coordination system of axis Y (0, 1)>]
[: <Clears coordination system of axis Z (0, 1)>]
[: <Clears coordination system of axis W (0, 1)>]

Example @4:1:1:1:1

Description This command clears all the coordination systems of the controller, and sets the position designated by this command to the origin of all the coordination systems. This command is a one-shot command.
The setting values and operation of <Clear of coordination system of axis> are as follows:

0: Does not clear the coordination systems.
1: Clears the coordination systems.

The coordination systems to be cleared are as follows:

(1) Mechanical coordination
(2) Absolute coordination
(3) Base coordination
(4) Encoder coordination
(5) Programmed coordination

Reference "4.4 Commands / Messages (@ 0)"
"4.2 Operation for Installation / Coordination System"
"4.5 Programming Language / Clearing Positions"

Transmission Mode

Command

Function Designates the transmission mode of the controller.

Format @5 : <Transmission mode (0, 1)>

Initial Value @5 : 1

Description The controller can be operated through two types of transmission modes. <Transmission mode> shall be the type of the transmission mode to be selected. The setting value, type, and operation of <Transmission mode> are as follows :

Setting Value	Type	Operation
0	RS-232C mode	To be operated from the personal computer, operation display panel, or operation display pendant.
1	Parallel transmission mode	To be operated from the sequencer or switches.

Operation for installation shall be carried out using the RS-232C mode, and usual operation shall be carried out using the parallel transmission mode.

Reference "4.2 Operation for Installation / RS-232C Mode"
"4.3 Operation / Parallel Transmission Mode"

Operation Mode

Command

Function Designates the operation mode of the controller.

Format @6[:<Operation mode (0-10)>]::[:<Type of M signal outpu (0, 1)>]

Initial Value @6:0:::0

Description The controller can control up to seven modes of operations.
<Operation mode> designates the desired operation mode. The setting value, modes, and operation of <Operation mode> are as follows:

Setting Value	Mode	Operation
0	Origin return mode	This determines the origin for the reference of positioning.
1	Programmed AUTO mode	This executes the designated program from the beginning to the end.
2	Programmed STEP mode	This executes the designated program block by block from the beginning.
3	Programmed CONT mode	This executes the designated program repeatedly between the beginning and the end.
4	Jog mode	This executes the jog feeding of the motor.
8	Origin calibration mode	This measures the number of pulses between the origin sensor and the edge of the origin pulse.
10	MDI mode	This enters the program by one block, and executes it at the same time.

	#9 MODE 3	#8 MODE 2	#7 MODE 1	#6 MODE 0
Origin return mode	OFF	OFF	OFF	OFF
Programmed AUTO mode	OFF	OFF	OFF	ON
Programmed STEP mode	OFF	OFF	ON	OFF
Programmed CONT mode	OFF	OFF	ON	ON
Jog mode	OFF	ON	OFF	OFF
Origin calibration mode	ON	OFF	OFF	OFF
MDI mode	ON	OFF	ON	OFF

<Type of M signal output> selects the type of M signal output. The setting value and operation of <Type of M signal output> are as follows:

0 : Allows to output Mout signal only. This does not allow to output Oout signal.

1 : Allows to output both Mout and Oout signals.

Reference "4.2 Operation for Installation / Operation Modes"
"4.2 Operation for Installation / M output Interface"
"4.2 Operation for Installation / M output Parameters"

Initial 0

Command

Function Selects the parameter of the initial 0 of the controller.

Format @7 [: < Servo interlock switch (0, 1)>]
[: <Origin return completion switch (0, 1)>]

Initial Value @7 : 1 : 1

Description This command sets both <Servo interlock switch> and <Origin return completion switch>.

Servo can be activated with power-on of the controller. <Servo interlock switch> is used for that purpose.
The setting and operation of <Servo interlock switch> are as follows:

0 : Turns off the servo with power-on of the controller.
1 : Turns on the servo with power-on of the controller.

Normally, origin return operation is carried out at the beginning of daily work. Therefore, programmed operation is to be carried out after origin return operation. The controller memorizes whether origin return operation has been carried out or not. <Origin return completion switch> enables or disables the memory for this purpose. By setting <Origin return completion switch> to enabled status, starting of programmed operation before completing origin return operation can be inhibited.
The setting value and operation of <Origin return completion switch> are as follows:

0 : Disabled (Programmed operation can be carried out without completing origin return operation.)
1 : Enabled (Programmed operation cannot be carried out without completing origin return operation.)

Reference "4.2 Operation for Installation / Type by type setting of the motor"
"4.2 Operation for Installation / Origin Return Operation"
"4.2 Operation for Installation / Origin Return Parameters"

Initial 1

Command

Function Selects the parameter of the initial 1 of the controller.

Format @8::〔:<Coordination system clear switch (0, 1)>〕〔:<COIN switch (0, 1)>〕

Initial Value @8:::0:1

Description This command sets both <Coordination system clear switch> and <COIN switch> to enabled or disabled status.

<Coordination system clear switch> selects whether or not to clear all the coordination systems of the controller when the reset command (@1) is executed or when the controller is reset with the RESET (#10) input signal.
The setting and operation of <Coordination system clear switch> are as follows:

0 : Disabled status (Coordination systems remain as they are even when reset operation is executed.)
1 : Enabled status (Coordination systems are reset when reset operation is executed.)

<COIN switch> enables or disables positioning completion check after axis feeding.
This is effective to DYNASERV (position control mode), DYNASERV (velocity control mode), AC / DC servo motor (position control mode), and AC / DC servo motor (velocity control mode) in origin return operation, programmed AUTO operation, programmed STEP operation, programmed CONT operation, or MDI operation.
The setting value and operation of <COIN width switch> are as follows:

0 : Disabled status (No positioning completion check)
1 : Enabled status (Positioning completion check)

Memo Designation using the program:
G110 or G111

Memo Refer to the instruction manual of the DYNASERV for positioning width of the DYNASERV.
Setting of POSW1 and POSW2 to the DYNASERV shall be open.

Reference
"4.2 Operation for Installation / Coordination System"
"4.4 Commands / Internal COIN width Switch (@24)"
"4.4 Commands / Internal COIN width Switch (@59)"
"4.2 Operation for Installation / Type by type setting of the motor"
"4.5 Programming Language / COIN"

Initial 2

Command

Function Selects the parameter of the initial 2 of the controller.

Format @9 [: <M output switch M(0, 1)>] [: <M01 switch (0, 1)>]
[: <M00 switch (0, 1)>] [: <M30 switch (0, 1)>]

Initial Value @9 : 1 : 1 : 1 : 1

Description This command sets <M output switch>, <M01 switch>, <M00 switch>, and <M30 switch> to enabled or disabled status.

<M output switch> enables or disables the M output interface. By setting the M output interface to enabled status, operation can be synchronized with the external devices (such as the sequencer), through the controller's input / output terminal (using the output signal of M ENABLE (#41) and M OUT 0-7 (#42-49), and the input signal of M ANSWER (#12)).
The codes which can be used for M output interface are 2-digit M codes of M00 to M99. M output to the external device is executed when the M code of the program or the MDI operation is executed.
The setting value and operation of <M output switch> are as follows:

0 : Disabled M output
1 : Enabled M output

<M01 switch> enables or disables the M01 code in the program. M01 code is called "optional stop". The setting value and operation of <M01 switch> are as follows:

0 : Disabled
1 : Enabled

<M00 switch> enables or disables the M00 code in the program. M00 code is called "program stop". The setting value and operation of <M00 switch> are as follows:

0 : Disabled
1 : Enabled

<M30 switch> enables or disables the M30 code in the program. M30 code is called "end of program". The setting value and operation of <M30 switch> are as follows:

0 : Disabled
1 : Enabled

! Be sure to enter the M02 code at the end of the program when setting <M30 switch> to disabled status. Otherwise (ending with the M30 code), an error occurs.

Reference "4.2 Operation for Installation / M output Interface"
"4.2 Operation for Installation / M output Parameters"
"4.5 Programming Language / Auxiliary Function"

Listing Program

Function Reads the registered program on the personal computer, operation display panel, or operation display pendant.

Format @14 : <Program number (1-128)>

Example @14 : 1

Description This command is effective in programmed AUTO mode, programmed STEP mode, or programmed CONT mode, and it reads the program of registered <Program number> on the personal computer, operation display panel, or operation display pendant.

Reference
"4.2 Operation for Installation / Managing Programs"
"4.2 Operation for Installation / Listing Programs"
"4.2 Operation for Installation / Operation Modes"
"4.3 Operation / Programmed AUTO Mode"
"4.3 Operation / Programmed STEP Mode"
"4.3 Operation / Programmed CONT Mode"

Saving Program

Function Creates a program with a designated program number, and registered the program to the controller.

Format @15 : <Program number (1-128)>

Example @15 : 2

Description This is effective to programmed AUTO mode, programmed STEP mode, and programmed CONT mode.
This command registers the created program to the controller with <Program number>.
When executing this command, the personal computer, operation display panel, or operation display pendant is set in program input status. In this status, input a program using the keyboard. When completing to input the program, exit from the program input status using the following command.

END

The number of blocks in the program is not limited. However, the total memory capacity which can be used for the program is 32K bytes (about 32000 characters).

Note When the controller is set in the program input status using this command, any characters entered from the keyboard are treated as a part of the program until completing the program registration and exiting from the program input status. Note that if you enter a command without exiting from the program registration, the command is treated as a block of the program.

Reference "4.2 Operation for Installation / Managing Programs"
"4.2 Operation for Installation / Listing Programs"
"4.2 Operation for Installation / Operation Modes"
"4.3 Operation / Programmed AUTO Mode"
"4.3 Operation / Programmed STEP Mode"
"4.3 Operation / Programmed CONT Mode"

Deleting Program

Function Deletes a program.

Format @16 : <Program number (1-128)>

Example @16 : 3

Description This command is effective in programmed AUTO mode, programmed STEP mode, and programmed CONT mode.
This command deletes the undesired program with <Program number>.

Reference "4.2 Operation for Installation / Managing Programs"
"4.2 Operation for Installation / Deleting Programs"
"4.2 Operation for Installation / Operation Modes"
"4.3 Operation / Programmed AUTO Mode"
"4.3 Operation / Programmed STEP Mode"
"4.3 Operation / Programmed CONT Mode"

Listing Programs

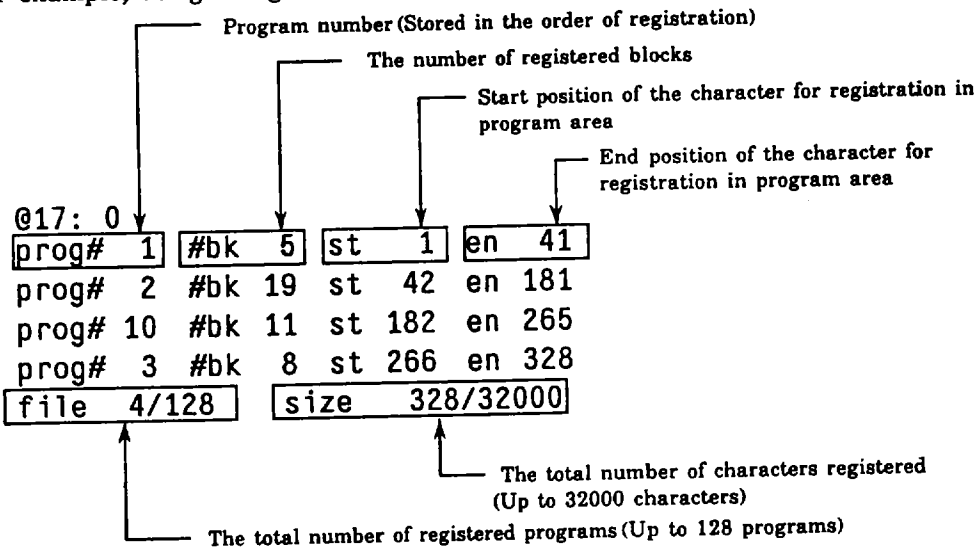
Function Lists the registered programs.

Format @17:<Program number (0-128)>

Example @17:4

Description This command is effective in programmed AUTO mode, programmed STEP mode, and programmed CONT mode.
This command lists the conditions of the programs of the registered <Program number>. Designating 1 to 128 for <Program number> lists the condition of the designated program. Designating 0 (or if you don't designate any number) lists the conditions of all the registered programs.

For example, designating 0 for <Program number> lists the following.



Reference
"4.2 Operation for Installation / Managing Programs"
"4.2 Operation for Installation / Listing Programs"
"4.2 Operation for Installation / Operation Modes"
"4.3 Operation / Programmed AUTO Mode"
"4.3 Operation / Programmed STEP Mode"
"4.3 Operation / Programmed CONT Mode"

Servo ON and OFF

Function Turns power to the servo motor on and off.

Format
@20 [: < Servo ON / OFF of axis X (0, 1)>]
[: < Servo ON / OFF of axis Y (0, 1)>]
[: < Servo ON / OFF of axis Z (0, 1)>]
[: < Servo ON / OFF of axis W (0, 1)>]

Initial Value @20 : 1 : 1 : 1 : 1

Description This command turns on and off the power to the servo motor.
Setting <Servo ON / OFF> selects power on or off of the servo motor.
The setting value and operation are as follows:

0 : OFF (Not powered)
1 : ON (Powered)

Note
(1) Logic operation of the SERVO ON / OFF (#3) input terminal is ANDed with this command. So, if the status of the SERVO ON / OFF (#3) input terminal is not ON, this command cannot turn on the servo.
(2) When the status of the SERVO ON / OFF (#3) input terminal changes between ON and OFF, this command is disregarded, and the status of the SERVO ON / OFF (#3) input connector is set to the parameters of all axes.
(3) When servo on, coordination system is cleared.

Reference "4.4 Commands / Initial 0 (@7)"
"4.2 Operation for Installation / Type by type setting of the motor"

Integral / Proportional Control

Servo Setting

Function Changes the selection of the integral / proportional control of the servo motor.

Format @21 [: < Integral / proportional control of axis X (0, 1) >]
[: < Integral / proportional control of axis Y (0, 1) >]
[: < Integral / proportional control of axis Z (0, 1) >]
[: < Integral / proportional control of axis W (0, 1) >]

Initial Value @21 : 0 : 0

Description This command is effective to the DYNASERV (position mode), DYNASERV (velocity mode), AC / DC servo motor (position mode), and AC / DC servo motor (velocity mode). This command changes the selection of the integral / proportional control of the servo motor.
<Integral / proportional control> turns on and off the interface PACT contact connected to the driver.
The setting value and operation of <Integral / proportional control> are as follows:

0 : Contact is open.
1 : Contact is closed.

Memo Designation by using the program:
G106 [X < Integral / proportional control of axis X (0, 1) >]
[Y < Integral / proportional control of axis Y (0, 1) >]
[Z < Integral / proportional control of axis Z (0, 1) >]
[W < Integral / proportional control of axis W (0, 1) >]

Note Refer to the instruction manual of the driver for the logic status or meaning of the interface PACT contact.

Reference
"4.2 Operation for Installation / Position Mode and Velocity Mode"
"4.2 Operation for Installation / Type by type setting of the motor"
"4.5 Programming Language / Preparatory Function (G Function)"
"4.5 Programming Language / Data Registration"

CAUTION

When the system is set for proportional control, the coin switch(enable/disable) will automatically be ignored.

DC Gain

Servo setting

Function Designates the DC gain of the DYNASERV.

Format @22〔:<DC gain of axis X(0-7)>〕〔:<DC gain of axis Y(0-7)>〕
〔:<DC gain of axis Z(0-7)>〕〔:<DC gain of axis W(0-7)>〕

Initial Value @22:0:0

Description This command is effective to the DYNASERV (position mode) axis and DYNASERV (velocity mode). This command designates the DC gain of the DYNASERV.
<DC gain> differs among the types of motors. The setting values and the meanings are as follows:

For motor types 0 to 3 (DM series)

0	:	×1
4	:	×10

For motor types 4 to 8 (DR series)

0	:	×1
1	:	×4
2	:	×7
3	:	×10
4	:	×13
5	:	×16
6	:	×19
7	:	×22

Memo Designation using the program:
G107〔X<DC gain of axis X(0-7)>〕
〔Y<DC gain of axis Y(0-7)>〕
〔Z<DC gain of axis Z(0-7)>〕
〔W<DC gain of axis W(0-7)>〕

Reference
"4.4 Commands/Motor Type (@40)"
"4.2 Operation for Installation/Setting of the DYNASERV (Position Mode)"
"4.5 Programming Language/Preparatory Function (G Function)"
"4.5 Programming Language/Data Registration"

fc Remote Switch

Function Designates the fc remote switch of the DYNASERV.

Format @23 [: <fc remote switch of axis X (0-15)>]
[: < fc remote switch of axis Y (0-15)>]
[: < fc remote switch of axis Z (0-15)>]
[: < fc remote switch of axis W (0-15)>]

Initial Value @23 : 0 : 0 : 0 : 0

Description This command is effective to the DYNASERV (position mode).
This command changes the selection of the fc remote switch of the DYNASERV.
Normally, position loop bandwidth is selected with the fc switch on the front panel of the DYNASERV. So, set this command to "0" in that case.
Use this command when setting position control loop bandwidth remotely from the controller. In this case, set the fc switch on the front panel of the DYNASERV to "0".
The setting values and the meanings are as follows:

Setting Value	<fc Remote Switch>				<Positioning Control Bandwidth> [Hz]	
	FN3	FN2	FN1	FN0	DM Driver	DR Driver
0	H	H	H	H	5	1
1	H	H	H	L	6	2
2	H	H	L	H	7	3
3	H	H	L	L	8	4
4	H	L	H	H	9	5
5	H	L	H	L	10	6
6	H	L	L	H	11	7
7	H	L	L	L	12	8
8	L	H	H	H	13	9
9	L	H	H	L	14	10
10	L	H	L	H	15	11
11	L	H	L	L	16	12
12	L	L	H	H	17	13
13	L	L	H	L	18	14
14	L	L	L	H	19	15
15	L	L	L	L	20	16

When the fc switch on the front panel of the DYNASERV is set to "0".

Memo Designation using the program:
'G104 [X < fc remote switch of axis X (0-15)>]
[Y <fc remote switch of axis Y (0-15)>]
[Z <fc remote switch of axis Z (0-15)>]
[W <fc remote switch of axis W (0-15)>]

Reference "4.2 Operation for Installation / Setting of the DYNASERV (Position Mode)"
"4.5 Programming Language / Preparatory Function (G Function)"
"4.5 Programming Language / Data Registration"

COIN width Switch

Servo Setting

Function Designates the COIN width switch of the DYNASERV.

Format
@24 [: < COIN width switch of axis X (0-3)>]
[: < COIN width switch of axis Y (0-3)>]
[: < COIN width switch of axis Z (0-3)>]
[: < COIN width switch of axis W (0-3)>]

Initial Value @24:0:0:0:0

Description This command is effective to the DYNASERV (position control mode).
This command changes the selection of the COIN width switch of the DYNASERV. The COIN width means the positioning alignment width. <COIN width switch> means the POSW0 and POSW1 remote control signals of the DYNASERV.
Set the POSW digital switch on the DYNASERV control board to "0", "4", "8", or "C".
The setting values and the meanings of <COIN width switch> are as follows:

Setting Value	<COIN Width Switch>		POSW Switch of the DYNASERV	Alignment Width (p)
	POSW1	POSW0		
0	H	H	0	1
			4	2
			8	4
			C	8
1	H	L	0	5
			4	10
			8	20
			C	40
2	L	H	0	20
			4	40
			8	80
			C	160
3	L	L	0	100
			4	200
			8	400
			C	800

Memo Designaion using the program:
G105 [X < COIN width switch of axis X (0-3)>]
[Y < COIN width switch of axis Y (0-3)>]
[Z < COIN width switch of axis Z (0-3)>]
[W < COIN width switch of axis W (0-3)>]

Reference
"4.4 Commands / Initial 1 (@8)"
"4.2 Operation for Installation / Setting of the DYNASERV (Position Mode)"
"4.5 Programming Language / Preparatory Function (G Function)"
"4.5 Programming Language / COIN"
"4.5 Programming Language / Data Registration"

CAUTION

When the system is set for proportional control, the coin switch(enable/disable) will automatically be ignored.

Integral Reset Switch

Servo Setting

Function Designates the integration reset switch in the velocity loop of the DYNASERV (DR Series only).

Format @26 [: < Integration reset switch of axis X (0, 1)>]
[: < Integration reset switch of axis Y (0, 1)>]
[: < Integration reset switch of axis Z (0, 1)>]
[: < Integration reset switch of axis W (0, 1)>]

Initial Value @26 : 0 : 0

Description This command is effective to the DYNASERV (position mode) and the DYNASERV (velocity mode).
This command changes the status of the integration reset switch of the DYNASERV.
<Integration reset switch> can close or open the interface IRST contact of the DYNASERV.
The setting values and the meanings of <Integration reset switch> are as follows:

0 : Contact is open (Normal Operation).
1 : Contact is closed (Intergral Reset).

Note Refer to the instruction manual of the driver for the operation logic and the meanings of the interface IRST contact. This command is available only for DYNASERV DR series.

Reference "4.2 Operation for Installation / Setting of the DYNASERV (Position Mode)"

Proportional Gain Control in Position Loop

Servo Setting

Function Designates the proportional gain control in the position control loop.

Format @28 [: <Position proportional gain control of axis X(0-7)>]
[: <Position proportional gain control of axis Y (0-7)>]
[: <Position proportional gain control of axis Z (0-7)>]
[: <Position proportional gain control of axis W (0-7)>]

Initial Value @28 : 0 : 0 : 0 : 0

Description This command is effective to the DYNASERV (velocity mode) and the AC / DC servo motor (velocity mode).
<Position proportional gain control> sets coarse proportional gain in the position control loop.
The setting values and the meanings of <Position proportional gain control> are as follows :

0: ×0.5
1: ×1
2: ×2
3: ×4
4: ×8
5: ×16
6: ×32
7: ×64

Memo Position proportional gain which has been set with this command can be adjusted further finely in 16 steps (1/16 to 16/16), using the rotary switch on the front of the axis board.

Reference "4.2 Operation for Installation / Position Mode and Velocity Mode"
"4.2 Operation for Installation / Setting of the AC / DC Servo Motor (Velocity Mode)"

Velocity Offset Correction Switch

Function Disables or enables the velocity offset correction function.

Format @29 [: < velocity offset correction switch of axis X (0, 1)>]
[: < Velocity offset correction switch of axis Y (0, 1)>]
[: < Velocity offset correction switch of axis Z (0, 1)>]
[: < Velocity offset correction switch of axis W (0, 1)>]

Initial Value @29 : 0 : 0 : 0 : 0

Description This command is effective to the DYNASERV (velocity mode) and the AC / DC servo motor (velocity mode).
The correction function of the analog voltage generated outside the velocity loop is called "velocity offset correction function". <Velocity offset correction switch> enables or disables the velocity offset correction function.
The setting values and the meanings of <Velocity offset correction switch> are as follows :

0 : Disabled
1 : Enabled

Reference "4.2 Operation for Installation / Position Mode and Velocity Mode"
"4.2 Operation for Installation / Setting of the AC / DC Servo Motor (Velocity Mode)"

Origin Return Direction

Flag Setting

Function Changes the direction of origin return.

Format
@30 [: < Origin return direction of axis X (0, 1)>]
[: < Origin return direction of axis Y (0, 1)>]
[: < Origin return direction of axis Z (0, 1)>]
[: < Origin return direction of axis W (0, 1)>]

Initial Value @30 : 0 : 0 : 0 : 0

Description This command is effective in origin return operation or in origin calibration operation.
<Origin return direction> changes the return direction of origin return.
The setting values and the meanings of <Origin return direction> are as follows:

0 : Normal (CW : clockwise)
1 : Reverse (CCW : counterclockwise)

Memo When setting to origin return type 1 or 3 using the origin return type command (@35), the axis rotates in the reverse direction to <Origin return direction> which has been set with this command until the edge of the overtravel sensor is detected.

Reference
"4.2 Operation for Installation / Origin Return Operation"
"4.2 Operation for Installation / Origin Return Parameters"
"4.2 Operation for Installation / Operation Modes"
"4.2 Operation for Installation / Origin Calibration Operation"
"4.3 Operation / Origin Return Mode"

(–) Direction of Overtravel

Flag Setting

Function Enables or disables the overtravel function in (–) direction.

Format @31 [: < Overtravel of axis X (0, 1)>] [: < Overtravel of axis Y (0, 1)>]
[: < Overtravel of axis Z (0, 1)>] [: < Overtravel of axis W (0, 1)>]

Initial Value @31 : 0 : 0 : 0 : 0

Description The setting values and the meanings are as follows:

0 : Disabled
1 : Enabled

Reference "4.4 Commands / (+) Direction of Overtravel (@32)"
"4.2 Operation for Installation / Overtravel Processing Function"

(+) Direction of Overtravel

Flag Setting

Function Enables or disables the overtravel function in (+) direction.

Format @32 [: < Overtravel of axis X (0, 1)>] [: < Overtravel of axis Y (0, 1)>]
[: < Overtravel of axis Z (0, 1)>] [: <Overtravel of axis W (0, 1)>]

Initial Value @32 : 0 : 0 : 0 : 0

Description The setting values and the meanings are as follows :

0	:	Disabled
1	:	Enabled

Reference "4.4 Commands / (−) Direction of Overtravel (@31)"
"4.2 Operation for Installation / Overtravel Processing Function"

(−) Direction of Soft Limit — Flag Setting

Function Enables or disables the soft limit function in (−) direction.

Format @33 [: < Soft limit of axis X (0, 1)>] [: < Soft limit of axis Y (0, 1)>]
[: < Soft limit of axis Z (0, 1)>] [: <Soft limit of axis W (0, 1)>]

Initial value @33 : 0 : 0 : 0 : 0

Description This command is effective in programmed AUTO mode, programmed STEP mode, programmed CONT mode, and MDI mode.
The setting values and the meanings are as follows :

0 : Disabled
1 : Enabled

Reference "4.4 Commands / (+) Direction of Soft Limit (@34)"
"4.4 Commands / (−) Direction of Soft Limit Value (@48)"
"4.2 Operation for Installation / Overtravel Processing Function"

(+) Direction of Soft Limit — Flag Setting

Function Enables or disables the soft limit function in (+) direction.

Format @34 [: < Soft limit of axis X (0, 1)>] [: < Soft limit of axis Y (0, 1)>]
[: < Soft limit of axis Z (0, 1)>] [: <Soft limit of axis W (0, 1)>]

Initial Value @34 : 0 : 0 : 0 : 0

Description This command is effective in programmed AUTO mode, programmed STEP mode, programmed CONT mode, and MDI mode.
The setting values and the meanings are as follows:

0 : Disabled
1 : Enabled

Reference "4.4 Commands / (−) Direction of Soft Limit (@33)"
"4.4 Commands / (+) Direction of Soft Limit Value (@49)"
"4.2 Operation for Installation / Overtravel Processing Function"

Origin Return Type

Parameter Setting

Function Designates the type of origin return operation.

Format @35 [: < Origin return type of axis X (0-3) >]
[: < Origin return type of axis Y (0-3) >]
[: < Origin return type of axis Z (0-3) >]
[: < Origin return type of axis W (0-3) >]

Initial Value @35 : 0 : 0 : 0 : 0

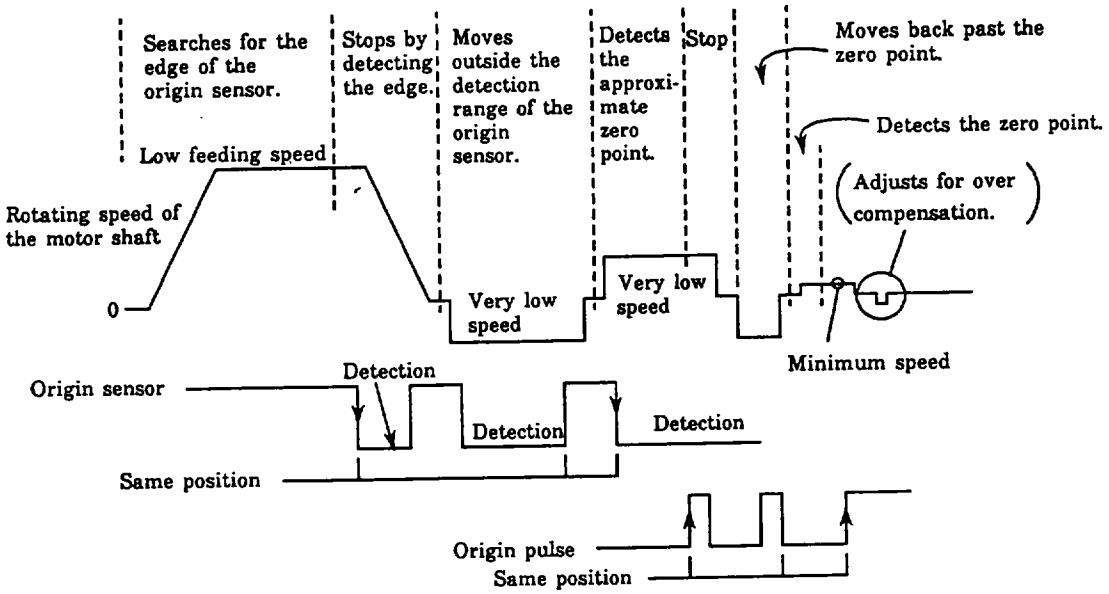
Description This command is effective in origin return mode and in origin calibration mode. The type of actual operation of origin return is called <Origin return type>. The setting values and the meanings of <Origin return type> are as follows :

Setting Value	Meaning
0	Detects the edge of the motor origin signal using the edge of the origin proximate sensor in the origin return direction as the reference. • First, searches the edge of the origin proximate sensor to the origin return direction at low feeding speed. • When detecting the edge, decelerates and stops. • Reversely rotates and returns to the out-of-detection range of the origin proximate at very low speed. • Then, goes to the origin proximate sensor again at very low speed. • Searches and detects the edge of the motor origin signal at minimum speed.
1	Detects the edge of the motor origin signal using the edge of the origin proximate sensor in the origin return direction as the reference. • First, searches the edge of the overtravel sensor in the opposite direction of the origin return direction at low feeding speed. • When detecting the edge, decelerates and stops. • Then, performs origin return operation in the same way as type 0.
2	Detects the edge of the motor origin signal using the origin proximate sensor as the reference. • First, searches the edge of the origin proximate sensor to the origin return direction at low feeding speed. • When detecting the edge, decelerates and stops. • Goes further in the same direction to the out-of-detection range of the origin proximate at very low speed. • Searches and detects the edge of the motor origin signal at minimum speed.
3	Detects the edge of the motor origin signal using the origin proximate sensor as the reference. • First, searches the edge of the overtravel sensor in the opposite direction of the origin return direction at low feeding speed. • When detecting the edge, decelerates and stops. • Then, performs origin return operation in the same way as type 2.

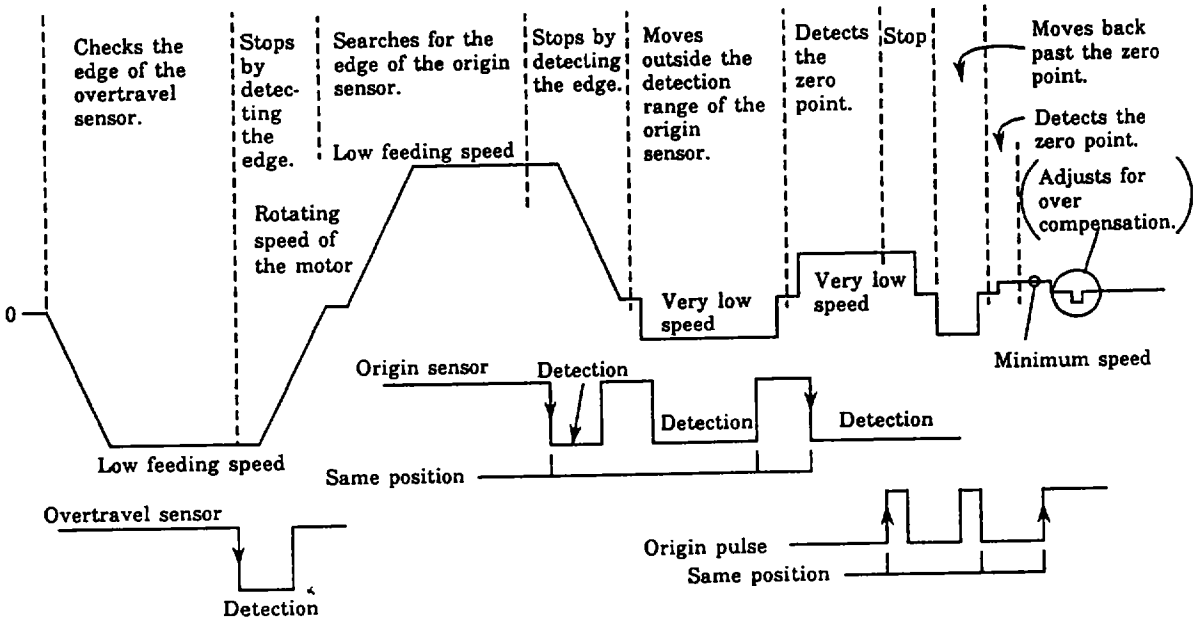
CAUTION

When the operating axis are the AC/DC servo type of control, kindly do not use origin types 2 & 3. Using these types 2 & 3 may induce positioning errors.

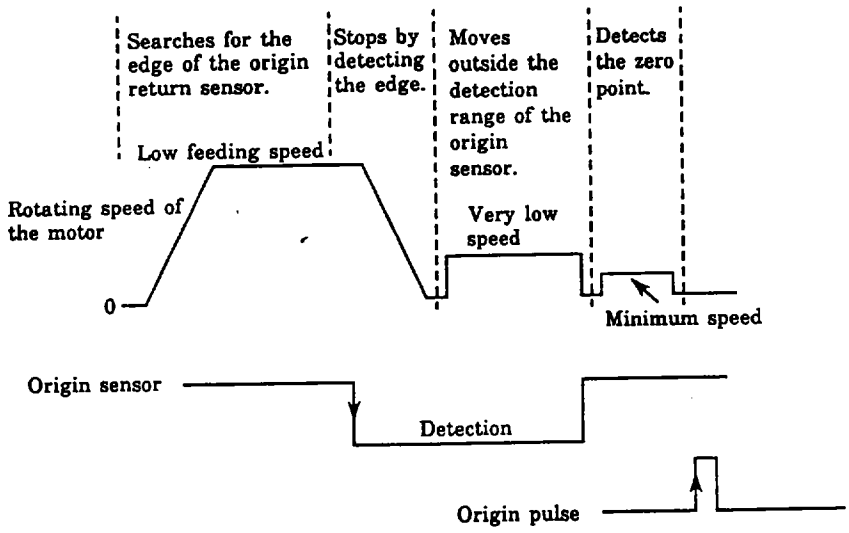
Origin Return Mode of Origin Return Type 0



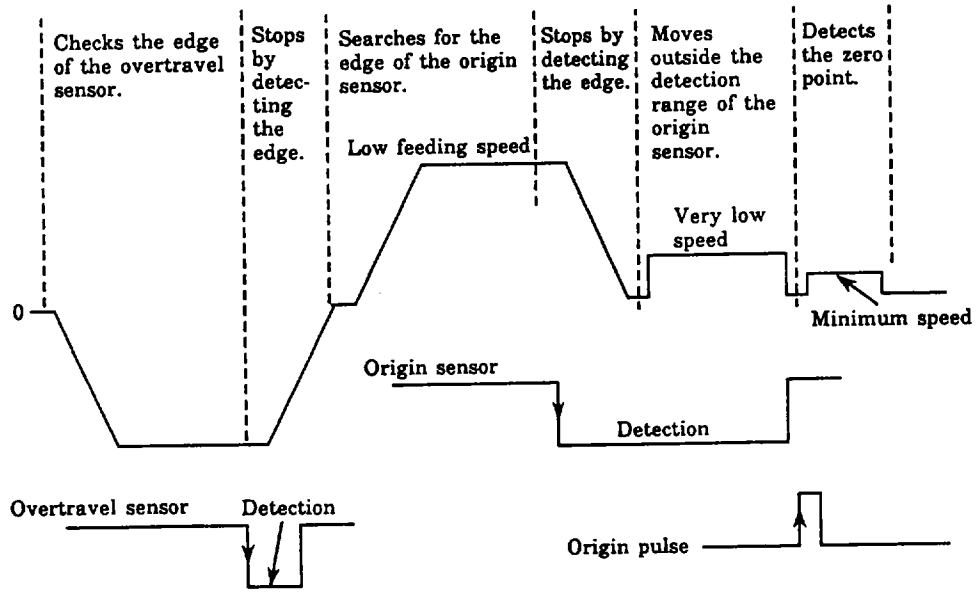
Origin Return Mode of Origin Return Type 1



Origin Return Mode of Origin Return Type 2



Origin Return Mode of Origin Return Type 3



Note When designating origin return type 2 or 3, set the searching speed (low feeding speed) to the origin sensor appropriately so that the operation can stop when detecting the origin sensor. If a higher speed than which the operation can stop at the origin sensor is set, an error message (error 60 to 63) appears during origin return mode, and the operation stops.

Reference
- "4.4 Commands / Origin Return Direction (@30)"
- "4.4 Commands / Low Feeding Speed (@50)"
- "4.2 Operation for Installation / Origin Return Operation"
- "4.2 Operation for Installation / Origin Return Parameters"
- "4.2 Operation for Installation / Operation Modes"
- "4.2 Operation for Installation / Origin Calibration Operation"
- "4.3 Daily Operation / Origin Return Mode"

Rotating Direction of Motors

Servo Setting

Function Changes the rotating direction (normal / reverse) of the axis of the motor.

Format
@37 [: < Motor axis rotating direction of axis X (0, 1)>]
[: < Motor axis rotating direction of axis Y (0, 1)>]
[: < Motor axis rotating direction of axis Z (0, 1)>]
[: < Motor axis rotating direction of axis W (0, 1)>]

Initial Value @37 : 0 : 0 : 0 : 0

Description This command is used only when installation.
<Motor axis rotating direction> is the rotating direction of the axis of the motor viewed from the top.
The setting values and the meanings are as follows :

0 : Sets (+) direction as CW (clockwise) rotating direction
1 : Sets (−) direction as CCW (counterclockwise) rotating direction

Note The rotating direction of the axis of the motor shall be set when installation. If this is reset, be sure to power off the controller once. Then, power on again.

Reference "4.2 Operation for Installation / Type by type setting of the motor"
"4.2 Operation for Installation / Coordination System"

Motor Type

Function Designates the motor type.

Format @40 [: < Motor type of axis X (0-8) >] [: < Motor type of axis Y (0-8) >]
[: < Motor type of axis Z (0-8) >] [: < Motor type of axis W (0-8) >]

Initial Value @40 : 0 : 0 : 0 : 0

Description This command is effective to the DYNASERV (position mode) axis and the DYNASERV (velocity mode).
This command is a dedicated command to the DYNASERV, and with this command, the related parameters are set automatically.
<Motor type> means the type of the motor of the DYNASERV.
The setting values of <Motor type>, type of the motor, parameters automatically set, and values to be registered are as follows:

Setting Value	Type	Parameters to be Set Automatically and Values to be Registered				
		Resolution of the Motor (@55)	Low Feeding Speed (@50)	High Feeding Speed (@51)	Maximum Feeding Speed (@52)	Jog Feeding Speed (@64)
0	DM1•••B	655360	13	655	1310	655
1	DM1•••A	1024000	10	512	1024	512
2	DM8•••B	2097152	8	524	1048	1048
3	DM8•••A	2097152	8	524	1048	1048
4	DR1•••B	507904	5	508	1016	508
5	DR1•••E	614400	6	614	1228	614
6	DR1•••A	819200	8	614	1228	614
7	DR5•••B	278528	8	278	1360	278
8	DR5•••A	425984	8	425	1360	425

Reference
"4.4 Commands / Resolution of the Motor (@55)"
"4.4 Commands / Low Feeding Speed (@50)"
"4.4 Commands / High Feeding Speed (@51)"
"4.4 Commands / Maximum Feeding Speed (@52)"
"4.4 Commands / Jog Feeding Speed (@64)"
"4.2 Operation for Installation / Setting of the DYNASERV (Position Mode)"
"4.2 Operation for Installation / Feeding Speed Parameters"

Input Unit System

Parameter Setting

Function Designates the input unit system.

Format @41 [:< Inpút unit system of axis X (0-5)>]
[:< Input unit system of axis Y (0-5)>]
[:< Input unit system of axis Z (0-5)>]
[:< Input unit system of axis W (0-5)>]

Initial Value @41:0:0:0:0

Description <Input unit system> selects the desired unit system for input of the positioning data.
The setting values and the meanings of <Input unit system> are as follows:

Setting Value	Type	Meaning
0	Pulse input unit system	Indicates the positioning date using the number of pulses.
1	Degree input unit system (Multi-rotation)	Indicates the positioning date using the degree from the center of the motor.
2	Division input unit system	Indicates the positioning date using the division number obtained by dividing 360° by the specified number.
3	Feeding length input unit system (Uni-directional)	Indicates the positioning date using the feeding length of the motor.
5	Degree input unit system (1 rotation)	The same as for setting value 1 except for that the range is from 0° ~ 360° .
6	Feeding length input unit system (Bi-directional)	Similar to type 3 with bi-directional capability (Application includes leed-screw feeding etc.)

Memo Designation using the program:

Type	Designated Value
Pulse input unit system	G140 (X) (Y) (Z) (W)
Degree input unit system (Multi-rotation)	G141 (X) (Y) (Z) (W)
Division input unit system	G142 (X) (Y) (Z) (W)
Feeding length input unit system	G143 (X) (Y) (Z) (W)

! No setting is available for the degree input unit system (1 rotation).
No setting is available for the feeding length input unit system (Bi-directional)

Reference "4.4 Commands / Number of Divisions (@56)"
"4.4 Commands / Feeding Length Factor (@60)",
"4.2 Operation for Installation / Coordination System",
"4.5 Programming Language / Preparatory Function (G Function)",
"4.5 Programming Language / Input Unit System"

Acceleration / Deceleration Type — Parameter Setting

Function Designates the acceleration / deceleration type of acceleration / deceleration control.

Format
@42 [:< Acceleration / deceleration type of axis X (0-11)>]
[:< Acceleration / deceleration type of axis Y (0-11)>]
[:< Acceleration / deceleration type of axis Z (0-11)>]
[:< Acceleration / deceleration type of axis W (0-11)>]

Initial Value @42:0:0:0:0

Description To control the speed curve of the motor for the desired curve is called "acceleration / deceleration control".
<Acceleration / deceleration type> is the type of acceleration / deceleration control. This command selects the desired acceleration / deceleration type.
The setting values, types, and the meanings of <Acceleration / deceleration type> are as follows:

Setting Value	Type	Operation
0	3rd order spline curve	This type makes the acceleration / deceleration curve of the motor to the 3rd-order spline curve or trapezoid curve.
1	Trapezoid curve	
2	Modified sine wave curve	Type 2 to 11 are called cam curve. The cam curve means the acceleration curve of the motor. As a result of the curve, speed curve of the motor is determined in the acceleration / deceleration curve. **Memo** Type 6 is recommended when moving a short distance in a short period of time. **Memo** Types 10 and 11 are created using the cam curve creation UTY of the tool box utility (optional).
3	Modified trapezoid curve	
4	Modified constant velocity curve	
5	Modified constant velocity curve II	
6	Trapecroid curve	
7	Trapecroid curve II	
8	Trapecroid curve III	
9	Constant acceleration curve	
10	User defined curve I	
11	User defined curve II	

Memo Designation using the program:
G102 [X < Acceleration / deceleration type of axis X (0-11)>]
[Y < Acceleration / deceleration type of axis Y (0-11)>]
[Z < Acceleration / deceleration type of axis Z (0-11)>]
[W < Acceleration / deceleration type of axis W (0-11)>]

Reference
"4.4 Commands / Cam Curve Unit System (@43)"
"4.4 Commands / Acceleration / Deceleration Time Duration (@53)"
"4.4 Commands / Deceleration Factor of Cam Curve (@54)"
"4.4 Commands / Minimum Feeding Time Duration (@67)"
"4.2 Operation for Installation / Acceleration/Deceleration Control"
"4.5 Programming Language / Preparatory Function (G Function)"
"4.5 Programming Language / Data Registration"

Cam Curve Unit System

Parameter Setting

Function Designates the cam curve unit system of acceleration / deceleration types 2 to 11.

Format @43 [: < Cam curve unit system of axis X (0-2) >]
[: < Cam curve unit system of axis Y (0-2) >]
[: < Cam curve unit system of axis Z (0-2) >]
[: < Cam curve unit system of axis W (0-2) >]

Initial Value @43 : 0 : 0 : 0 : 0

Description This command is effective when the acceleration / deceleration control set for either of acceleration / deceleration types 2 to 11.
<Cam curve unit system> is the unit system of feeding time duration instruction. Normally, unit is in ms. However, this can be multiplied 10 times or 100 times when necessary.
The setting values and the meanings of <Cam curve unit system> are as follows:

0 : ms
1 : 10ms (10 times)
2 : 100ms (100 times)

Reference "4.4 Commands / Acceleration/Deceleration Type (@42)"
"4.2 Operation for Installation / Acceleration/Deceleration Control"
"4.2 Operation for Installation / Acceleration/Deceleration Types 2 to 11"

Synchronization Type

Function Designates the synchronization type of synchronization control.

Format
@44 [:< Synchronization type of axis X (0,1)>]
[:< Synchronization type of axis Y (0-3)>]
[:< Synchronization type of axis Z (0-3)>]
[:< Synchronization type of axis W (0-3)>]

Initial Value @40 : 0 : 0 : 0 : 0

Description This command is effective in programmed AUTO mode, programmed STEP mode, programmed CONT mode, and MDI mode.
Synchronization control is to control the timing of axis feeding start to the start input signal, using the axis feeding instruction. <Synchronization type> is the type of synchronization control. This command selects the desired synchronization type. <Synchronization types> 2 and 3 are effective only to axes Y, Z, and W.
The setting values and the meanings of <Synchronization type> are as follows :

Setting Value	Synchronization Type	Meaning
0	Synchronization disabled type	Starts immediately with the start signal input.
1	Start time synchronization type	Starts delayed by the registered time duration with respect to the start signal input.
2	Time synchronization type	Start of axis Y, Z, or W is delayed by the registered time duration with respect to the start signal input to the axis X after the start signal is input.
3	Position synchronization type	Start of axis Y, Z, or W is when axis X is placed at the registered position after the start signal input to the axis X after the start signal is input is complete.

Memo Designation using the program :
(1) Synchronization disabled type
G150 [X] [Y] [Z] [W]
(2) Start time synchronization type
G151 [X< Synchronization data of axis X (0-999999999)>]
[Y < Synchronization data of axis Y (0-999999999)>]
[Z < Synchronization data of axis Z (0-999999999)>]
[W < Synchronization data of axis W (0-999999999)>]
(3) Time synchronization type
G152 [Y< Synchronization data of axis Y (0-999999999)>]
[Z < Synchronization data of axis Z (0-999999999)>]
[W < Synchronization data of axis W (0-999999999)>]
(4) Position synchronization type
G153 [Y< Synchronization data of axis Y (0-999999999)>]
[Z < Synchronization data of axis Z (0-999999999)>]
[W < Synchronization data of axis W (0-999999999)>]

Note In the following operation modes, designating types 1 to 3 does not take effect. In this case, axis feeding starts with type 0 (synchronizing disabled type).
(1) Origin return operation
(2) Jog operation
(3) Origin calibration operation

Reference
"4.4 Commands / Synchronization Data (@71)"
"4.2 Operation for Installation / Synchronization Control"
"4.5 Programming Language / Synchronization Control"
"4.5 Programming Language / Data Registration"

Offset Value of Origin Return

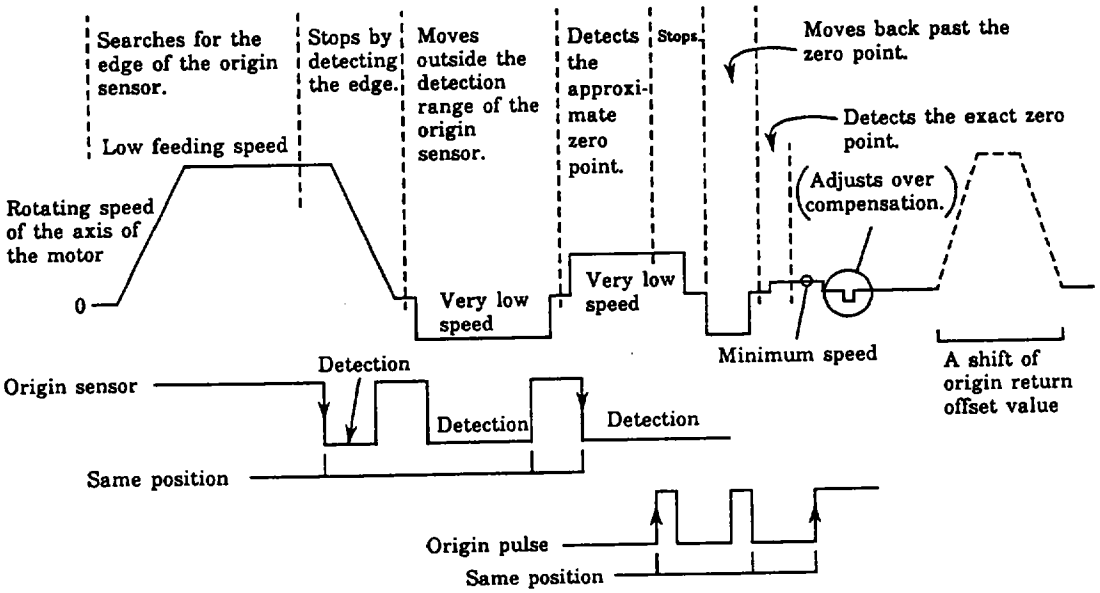
Parameter setting

Function Designates the origin return offset value (p).

Format @46 [:<Origin return offset value of axis X (−999999999-+999999999)>]
[:< Origin return offset value of axis Y (−999999999-+999999999)>]
[:< Origin return offset value of axis Z (−999999999-+999999999)>]
[:< Origin return offset value of axis W (−999999999-+999999999)>]

Initial Value @46:0:0:0:0

Description This command is effective in origin return operation.
<Origin return offset value> is the amount of offset (p) of mechanical coordination origin point from the mechanical origin point.



Reference
"4.4 Commands / Coordination after Origin Return (@47)"
"4.4 Commands / Low Feeding Speed (@50)"
"4.2 Operation for Installation / Coordination System"
"4.2 Operation for Installation / Origin Return Operation"
"4.2 Operation for Installation / Origin Return Parameters"
"4.2 Operation for Installation / Feeding Speed Parameters"
"4.2 Operation for Installation / Operation Modes"
"4.3 Operation / Origin Return Mode"

Coordination after Origin Return

Parameter Setting

Function Designates the coordination value after origin return.

Format @47 [:< Coordination value of axis X after origin return (−999999999-+999999999)>]
[:< Coordination value of axis Y after origin return (−999999999-+999999999)>]
[:< Coordination value of axis Z after origin return (−999999999-+999999999)>]
[:< Coordination value of axis W after origin return (−999999999-+999999999)>]

Initial value @47:0:0:0:0

Description <Coordination value after origin return> is the coordination value of the mechanical coordination origin position in programmed coordination systems.
The unit and the range of input value of <Coordination value after origin return> depends on the selected input unit system.

Reference "4.4 Commands / Input Unit System (@41)"
"4.4 Commands / Origin Return Offset Value (@46)"
"4.2 Operation for Installation / Coordination System"
"4.2 Operation for Installation / Origin Return Operation"
"4.2 Operation for Installation / Origin Return Parameters"
"4.2 Operation for Installation / Input Unit System"

(−) Direction of Soft Limit Value

Function Designates the soft limit value (p) in (−) direction.

Format @48 [:< Soft limit value of axis X (−999999999-+999999999)>]
[:< Soft limit value of axis Y (−999999999-+999999999)>]
[:< Soft limit value of axis Z (−999999999-+999999999)>]
[:< Soft limit value of axis W (−999999999-+999999999)>]

Initial Value @48:0:0:0:0

Description <Soft limit value> is set in mechanical coordination sysem. The unit is the number of pulses.

Reference "4.4 Commands / (−) Direction of Soft Limit (@33)"
"4.4 Commands / (−) Direction of Soft Limit Value (@48)"
"4.2 Operation for Installation / Overtravel Processing Function"

(+) Direction of Soft Limit value — Parameter Setting

Function Designates the soft limit value (p) in (+) direction.

Format @49 [:< Soft limit value of axis X (−999999999-+999999999)>]
[:< Soft limit value of axis Y (−999999999-+999999999)>]
[:< Soft limit value of axis Z (−999999999-+999999999)>]
[:< Soft limit value of axis W (−999999999-+999999999)>]

Initial Value @49:0:0:0:0

Description <Soft limit value> is set in mechanical coordination system. The unit is the number of pulses.

Reference "4.4 Commands / (+) Direction of Soft Limit (@34)"
"4.4 Commands / (+) Direction of Soft Limit Value (@49)"
"4.2 Operation for Installation / Overtravel Processing Function"

Low Feeding Speed

Parameter Setting

Function Designates the low feeding speed (kpps).

Format
@50 [:< Low feeding speed of axis X (1-1360)>]
[:< Low feeding speed of axis Y (1-1360)>]
[:< Low feeding speed of axis Z (1-1360)>]
[:< Low feeding speed of axis W (1-1360)>]

Initial Value
(1) @50:13:13:13:13
(2) @50:0:0:0:0

Description
This command is effective in origin return operation or in jog operation.
<Low feeding speed> is used in origin return operation and also used for low speed feeding in jog operation.
For the axes of the DYNASERV, the initial value shall be (1) above. <Low feeding speed> is registered automatically when the motor type command (@40) is used. Use this command in order to change the value to the different value from the automatically registered one.
For the axes which are not of the DYNASERV, the initial value shall be (2) above.
<Low feeding speed> is the feeding speed (kpps) for searching the origin sensor.
<Low feeding speed> in jog operation is the low feeding speed (kpps) of jog feeding.

Reference
"4.4 Commands / Motor Type (@40)"
"4.4 Commands / Jog Feeding Speed (@64)"
"4.2 Operation for Installation / Setting of the DYNASERV (Position Mode)"
"4.2 Operation for Installation / Feeding Speed Parameters"
"4.2 Operation for Installation / Operation Modes"
"4.3 Operation / Origin Return Mode"
"4.3 Operation / Jog Mode"
"4.5 Programming Language / Origin Return Function"

High Feeding Speed

Parameter Setting

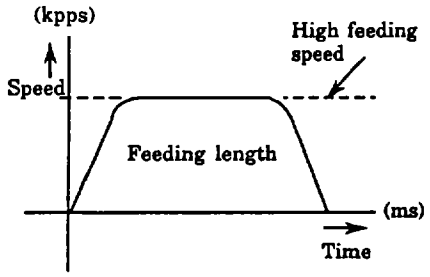
Function Designates the high feeding speed (kpps).

Format @51 [:< High feeding speed of axis X (1-1360)>]
[:< High feeding speed of axis Y (1-1360)>]
[:< High feeding speed of axis Z (1-1360)>]
[:< High feeding speed of axis W (1-1360)>]

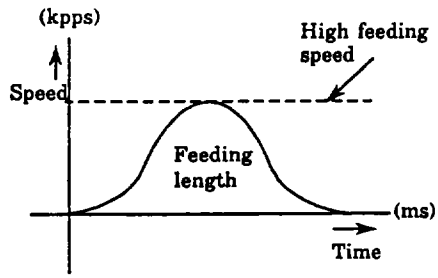
Initial Value (1) @51 : 655 : 655 : 655 : 655
(2) @51 : 0 : 0 : 0 : 0

Description <High feeding speed> is the feeding speed with the fast forward (G00 code) instruction.
For the axes of the DYNASERV, the initial value shall be (1) above. <High feeding speed> is registered automatically when the motor type command (@40) is used. The registered value is a half of the rated speed of the motor. Use this command in order to change the value to the different value from the automatically registered one.
For the axes which are not of the DYNASERV, the initial value shall be (2) above.

≪For Acceleration / Deceleration Types 0 and 1≫



≪For Acceleration / Deceleration Types 2 to 11≫



Reference "4.4 Commands / Motor Type (@40)"
"4.2 Operation for Installation / Setting of the DYNASERV (Position Mode)"
"4.2 Operation for Installation / Feeding Speed Parameters"
"4.5 Programming Language / Interpolation Function"

Maximum Feeding Speed — Parameter Setting

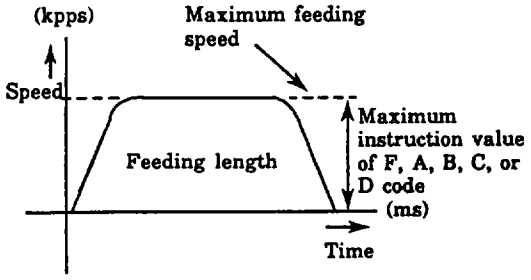
Function Designates the maximum feeding speed (kpps).

Format @52 [:< Maximum feeding speed of axis X (1-1360)>]
[:< Maximum feeding speed of axis Y (1-1360)>]
[:< Maximum feeding speed of axis Z (1-1360)>]
[:< Maximum feeding speed of axis W (1-1360)>]

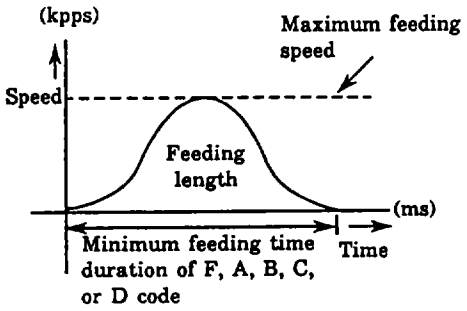
Initial Value (1) @52 : 1310 : 1310 : 1310 : 1310
(2) @52 : 0 : 0 : 0 : 0

Description <Maximum feeding speed> is the maximum feeding speed with the feeding speed (F, A, B, C, or D code) instruction.
For the axes of the DYNASERV, the initial value shall be (1) above. <Maximum feeding speed> is registered automatically when the motor type command (@40) is used. The registered value is the rated speed of the motor. Use this command in order to change the value different from the automatically registered value.
For the axes which are not of the DYNASERV, the initial value shall be (2) above.

≪For Acceleration / Deceleration Types 0 and 1≫



≪For Acceleration / Deceleration Types 2 to 11≫



Reference "4.4 Commands / Motor Type (@40)"
"4.2 Operation for Installation / Setting of the DYNASERV (Position Mode)"
"4.2 Operation for Installation / Feeding Speed Parameters"
"4.5 Programming Language / Feeding Function"

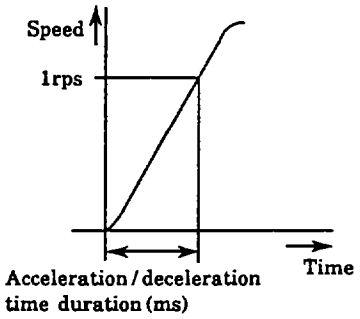
Acceleration / Deceleration Time Duration

Function Designates the acceleration / deceleration time duration of acceleration / deceleration types 0 and 1.

Format @53 [:< Acceleration / deceleration time duration of axis X (0-65535)>]
[:< Acceleration / deceleration time duration of axis Y (0-65535)>]
[:< Acceleration / deceleration time duration of axis Z (0-65535)>]
[:< Acceleration / deceleration time duration of axis W (0-65535)>]

Initial Value @53 : 350 : 350 : 350 : 350

Description This command is effective when the acceleration / deceleration control is set for acceleration / deceleration types 0 and 1.
<Acceleration / deceleration time duration> is not the feeding time duration required for the motor until the rotating speed of the motor becomes the rated speed. It is the feeding time duration until the speed of the motor becomes 1rps.



Memo Designation using the program :
G103 [X<Acceleration / deceleration time duration of axis X (0-65535)>]
[Y < Acceleration / deceleration time duration of axis Y (0-65535)>]
[Z < Acceleration / deceleration time duration of axis Z (0-65535)>]
[W < Acceleration / deceleration time duration of axis W (0-65535)>]

Reference "4.4 Commands / Acceleration/Deceleration Type (@42)"
"4.2 Operation for Installation / Acceleration/Deceleration Control"
"4.2 Operation for Installation / Acceleratory / Deceleration Types 0 and 1"
"4.5 Programming Language / Preparatory Function (G Function)"
"4.5 Programming Language / Data Registration"

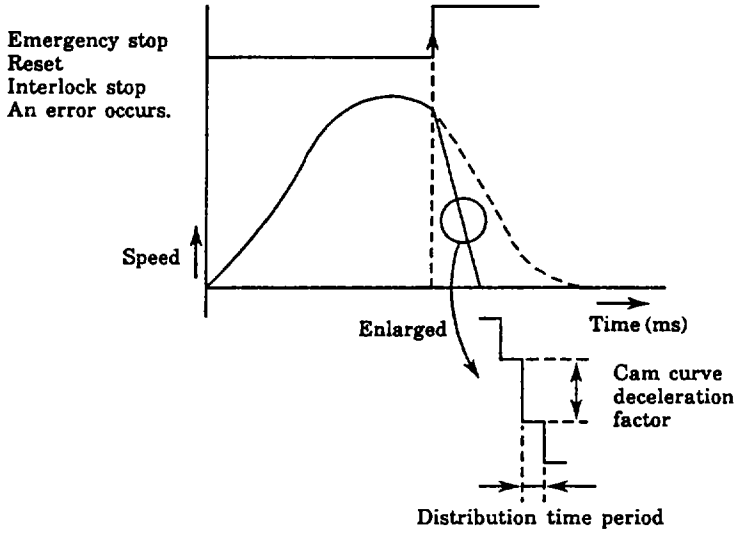
Deceleration Factor of Cam Curves Parameter Setting

Function Designates the deceleration factor of acceleration / deceleration types 2 to 11.

Format @ 54 [:< Cam curve deceleration factor of axis X (0-4095)>]
[:< Cam curve deceleration factor of axis Y (0-4095)>]
[:< Cam curve deceleration factor of axis Z (0-4095)>]
[:< Cam curve deceleration factor of axis W (0-4095)>]

Initial Value @ 54 : 30 : 30 : 30

Description This command is effective when the acceleration / deceleration control is set for acceleration / deceleration types 2 to 11 of cam curves.
<Cam curve deceleration factor> is a factor which is the degree of deceleration when the stop signal of axis feeding is input (when emergency stop, reset, interlock stop, or an error occurs). The higher the factor is, the more decelerated.



Reference
"4.4 Commands / Acceleration/Deceleration Type (@42)"
"4.2 Operation for Installation / Acceleration/Deceleration Control"
"4.2 Operation for Installation / Acceleration/Deceleration Types 2 to 11"
"4.5 Programming Language / Preparatory Function (G Function)"
"4.5 Programming Language / Data Registration"

Resolution of the Motor

Servo Setting

Function Designates resolution of the motor (ppr).

Format @55 [:< Resolution of rotation of the motor of axis X (0-999999999)>]
[:< Resolution of the motor of axis Y (0-999999999)>]
[:< Resolution of the motor of axis Z (0-999999999)>]
[:< Resolution of the motor of axis W (0-999999999)>]

Initial Value (1) @55 : 655360 : 655360 : 655360 : 655360
(2) @55 : 0 : 0 : 0 : 0

Description This command is effective to the DYNASERV (position mode), the DYNASERV (velocity mode), the AC / DC servo motor (position mode), the AC / DC servo motor (velocity mode), and the stepping motor.
<Resolution of the motor> is stepping resolution of rotation of the motor.
For the axes of the DYNASERV, the initial value shall be (1) above. <Resolution of the motor> is registered automatically when the motor type command (@40) is used. Check the value by referring to the specifications of the motor. Use this command in order to change the value different from the automatically registered value.
For the axes which are not of the DYNASERV, the initial value shall be (2) above.

Reference "4.4 Commands / Motor Type (@40)"
"4.2 Operation for Installation / Type by type setting of the motor"

Number of Divisions

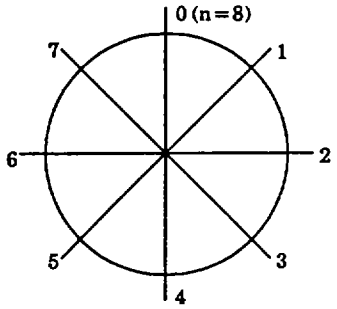
Parameter Setting

Function Designates the number of divisions in the division input unit system.

Format @56 [:< Division number of axis X (1-1024)>]
[:< Division number of axis Y (1-1024)>]
[:< Division number of axis Z (1-1024)>]
[:< Division number of axis W (1-1024)>]

Initial Value @56 : 1024 : 1024 : 1024 : 1024

Description This command is effective when input unit system is set in the division input unit system.
<Division number> determines the minimum setting unit in the division input unit system. First, 360° is divided by <Division number> in order to obtain the desired division number. Then, the desired position can be designated with the division number from 1 to <Division number>. So, use the division number to input the desired position data.
When resolution of rotation of the motor is not integer of <Division number>, the position for minimum error is selected. The error is not accumulated.



Memo Designation using the program :
G101 [X< Division number of axis X (1-1024)>]
[Y < Division number of axis Y (1-1024)>]
[Z < Division number of axis Z (1-1024)>]
[W < Division number of axis W (1-1024)>]

Note If the motor resolution which has been registered automatically by using the motor type command (@40) differs from the actual motor resolution, or if the value of motor resolution which has been registered by using the motor resolution command (@55) differs from the actual resolution of rotation of the motor, the division position is not correctly set and normally operation cannot be expected.

Reference
"4.4 Commands / Input Unit System (@41)"
"4.4 Commands / Motor Resolution (@55)"
"4.2 Operation for Installation / Input Unit System"
"4.2 Operation for Installation / Division Input Unit System"
"4.5 Programming Language / Preparatory Function (G Function)"
"4.5 Programming Language / Data Registration"

Internal COIN Width

Servo Setting

Function Designates the internal COIN width (p) of the controller.

Format
@59 [:< Internal COIN width of axis X (0-32767)>]
[:< Internal COIN width of axis Y (0-32767)>]
[:< Internal COIN width of axis Z (0-32767)>]
[:< Internal COIN width of axis W (0-32767)>]

Initial Value @59:0:0:0:0

Description This command is effective to the DYNASERV (velocity mode) and the AC / DC servo motor (velocity mode).
<Internal COIN width> is the COIN width (p) when feedback control is carried out from the controller.
Alignment check is carried out, after sending rotation instruction to the motor, by comparing <Internal COIN width> with the positioning deviation in the positioning feedback control.

Reference
"4.4 Commands / Initial 1 (@8)"
"4.2 Operation for Installation / Position Mode and Velocity Mode"
"4.2 Operation for Installation / Setting of the AC / DC Servo Motor (Velocity Mode)"
"4.5 Programming Language / COIN"

Feeding Length Factor

Parameter Setting

Function Designates the feeding length factor in the feeding length input unit system.

Format @60 [:< Feeding length factor of axis X (1-999999999)>]
[:< Feeding length factor of axis Y (1-999999999)>]
[:< Feeding length factor of axis Z (1-999999999)>]
[:< Feeding length factor of axis W (1-999999999)>]

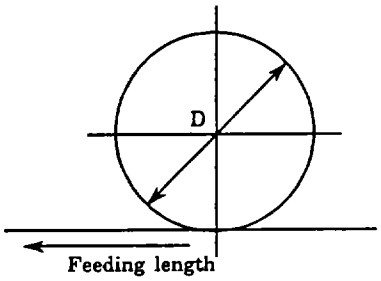
Initial Value @60 : 655360 : 655360 : 655360 : 655360

Description This command is effective when the input unit system is set in the feeding length input unit system.
<Feeding length factor> is a factor used when converting the input positioning data to the instruction pulse which is sent to the motor.
In the coordination systems which require designation of <Feeding length factor>, a motor rotates once by inputting the positioning data which is the same value as the <Feeding length factor>. So, the value of feeding length that rotates the motor once shall be designated for <Feeding length factor>. For example, when <Feeding length factor of axis X> is 1000, instructing "X1000" rotates the motor once. Positioning data is converted to the instruction pulse to the motor, by the following formula.

$$\text{Instruction pulse (p)} = \frac{\text{Resolution of the motor (ppr)}}{\text{Feeding length factor}} \times \text{Positioning data}$$

For Feed-to-length application, the positioning data can be used for arc of the drive roller by setting the feeding length factor as follows:

Feeding length factor = Pi (π) × Diameter of the drive roller (D)



The input unit of the positioning data is the same unit as used for the value for (D) in the formula above.
For example, where D = 280mm:
Input unit for the positioning data is mm where $\pi D = \pi \times 280$
Input unit for the positioning data is μm (0.001mm) where $\pi D = \pi \times 280000$
For feeding length using a ball screw, the positioning data is the rectangular coordinates of ball when setting the pitch (μm) of the ball screw for the feeding length factor.

Reference "4.4 Commands / Input Unit System (@41)"
"4.2 Operation for Installation / Input Unit System"
"4.2 Operation for Installation / Feeding Length Input Unit System"

Input Sensitivity Voltage

Servo Setting

Function Designates the input sensitivity voltage (mV).

Format @62 [:< Input sensitivity voltage of axis X (0-9999)>]
[:< Input sensitivity voltage of axis Y (0-9999)>]
[:< Input sensitivity voltage of axis Z (0-9999)>]
[:< Input sensitivity voltage of axis W (0-9999)>]

Initial Value @62 : 0 : 0 : 0 : 0

Description This command is effective to the DYNASERV (velocity mode) and the AC / DC servo motor (velocity mode).
<Input sensitivity voltage> is the input voltage for 1rps. Set the appropriate value for the input voltage because the value is used for position feedback control.

Reference "4.2 Operation for Installation / Position Mode and Velocity Mode"
"4.2 Operation for Installation / Setting of the AC / DC Servo Motor (Velocity Mode)"

Velocity Feed Forward

Servo Setting

Function Designates the velocity feed forward ratio (%).

Format @63 [:< Velocity feed forward ratio of axis X (0-120)>]
[:<Velocity feed forward ratio of axis Y (0-120)>]
[:<Velocity feed forward ratio of axis Z (0-120)>]
[:<Velocity feed forward ratio of axis W (0-120)>]

Initial Value @63 : 0 : 0 : 0 : 0

Description This command is effective to the DYNASERV (velocity mode) and the AC / DC servo motor (velocity mode).
<Velocity feed forward ratio> is used to adjust the amount of feed forward. The setting range of <Velocity feed forward ratio> is from 0 to 120 (%).

Note The proper feed forward cannot be obtained when the appropriate values are not set for resolution of the motor and for input sensitivity voltage.

Reference
"4.4 Commands / Resolution of the Motor (@55)"
"4.4 Commands / Input Sensitivity Voltage (@62)"
"4.2 Operation for Installation / Position Mode and Velocity Mode"
"4.2 Operation for Installation / Setting of the AC / DC Servo Motor (Velocity Mode)"

Jog Feeding Speed

Parameter Setting

Function Designates the jog feeding speed (kpps) in jog operation.

Format @64 [:<Jog feeding speed of axis X (1-1360)>]
[:< Jog feeding speed of axis Y (1-1360)>]
[:< Jog feeding speed of axis Z (1-1360)>]
[:< Jog feeding speed of axis W (1-1360)>]

Initial Value (1) @64 : 100 : 100 : 100 : 100
(2) @64 : 0 : 0 : 0 : 0

Description This command is effective in jog operation.
<Jog feeding speed> is the high feeding speed of jog feeding.
For the axes of the DYNASERV, the initial value shall be (1) above. <Jog feeding speed> is registered automatically when the motor type command (@40) is used. Use this command in order to change the value different from the automatically registered value.
For the axes which are not of the DYNASERV, the initial value shall be (2) above.

Reference "4.4 Commands / Motor Type (@40)"
"4.4 Commands / Low Feeding Speed (@50)"
"4.2 Operation for Installation / Setting of the DYNASERV (Position Mode)"
"4.2 Operation for Installation / Feeding Speed Parameters"
"4.2 Operation for Installation / Operation Modes"
"4.3 Operation / Jog Mode"

Minimum Feeding Time Duration of Cam Curves

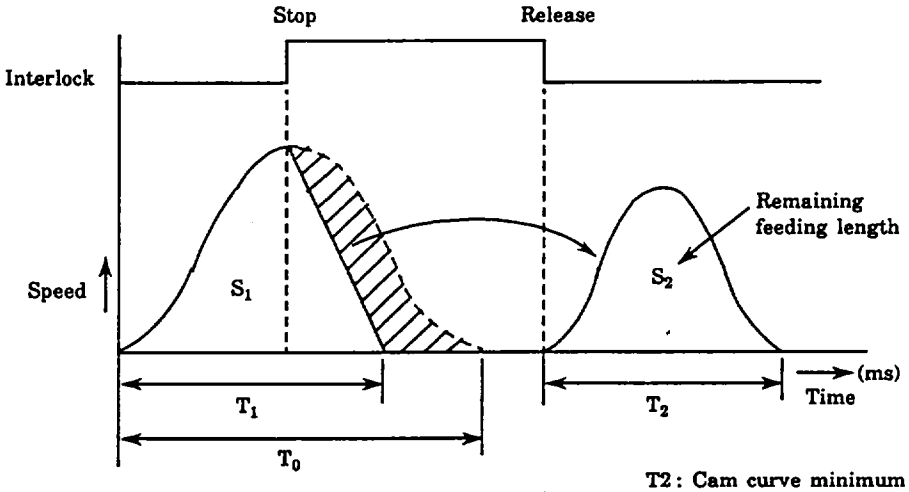
Parameter Setting

Function Designates the minimum feeding time duration (ms) for restarting of axis feeding when acceleration / deceleration control is set for cam curves.

Format @67 [:< Cam curve minimum feeding time duration of axis X (1-65535)>]
[:< Cam curve minimum feeding time duration of axis Y (1-65535)>]
[:< Cam curve minimum feeding time duration of axis Z (1-65535)>]
[:< Cam curve minimum feeding time duration of axis W (1-65535)>]

Initial Value @67 : 150 : 150 : 150 : 150

Description This command is effective when acceleration / deceleration control is set for acceleration / deceleration types 2 to 11 of cam curves.
<Cam curve minimum feeding time duration> is the minimum feeding time duration required for axis feeding of remaining length when the interlock halt is released. Therefore, this command is effective when the interlock halt has occurred.



T2 : Cam curve minimum feeding time duration

! Required feeding time duration (T2) for remaining feeding length when the interlock halt is released is obtained from the following formula.

$$T_2 = T_0 - \frac{S_1}{S_0} T_1$$

However, if T_2 is too small, feeding is not smooth (step by step feeding like feeding with vibration). Therefore, T_2 must be kept higher than a certain time duration. Designate <Cam curve minimum feeding time duration> by taking this effect into consideration.

Reference "4.4 Commands / Acceleration/Deceleration Type (@42)"
"4.2 Operation for Installation / Acceleration/Deceleration Control"
"4.2 Operation for Installation / Acceleration/Deceleration Types 2 to 11"

Override of Feeding Speed Parameter Setting

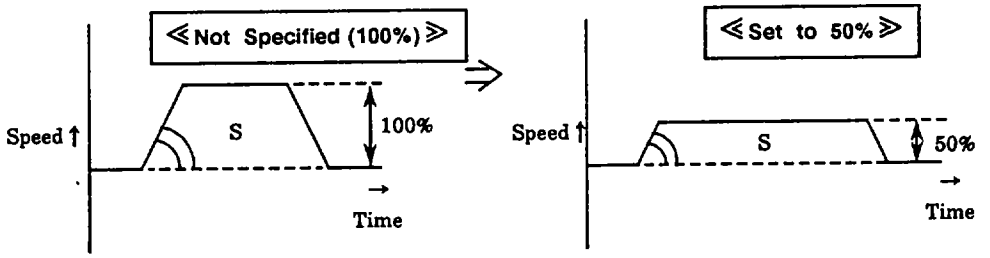
Function Designates the amount of override (%) of feeding speed.

Format @69 [:< Feeding speed override of axis X (0-100)>]
[:< Feeding speed override of axis Y (0-100)>]
[:< Feeding speed override of axis Z (0-100)>]
[:< Feeding speed override of axis W (0-100)>]

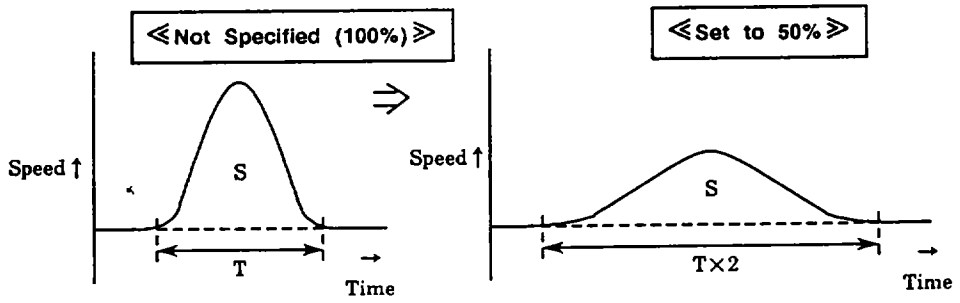
Initial Value @69 : 100 : 100: 100 : 100

Description <Feeding speed override> is the override amount of all feeding speeds except for the maximum feeding speed.
This command is to be used for slow feeding operation, such as for debugging or test operation. Using this command, <Feeding speed override> can be changed during axis feeding in origin return operation and in jog operation. For other mode of operations, <Feeding speed override> which has been set before starting axis feeding is used.

<Feeding speed override> for acceleration / deceleration types 0 and 1 is the amount of override for feeding speed, and in this case, feeding length and acceleration do not change.



<Feeding speed override> for acceleration / deceleration types 2 to 11 is not the amount of override for feeding speed. It is the amount of override for feeding time duration, and in this case, feeding length does not change.



Reference "4.4 Commands / Maximum Feeding Speed (@52)"
"4.2 Operation for Installation / Feeding Speed Parameters"
"4.2 Operation for Installation / Operation Modes"

Synchronization Data

Parameter Setting

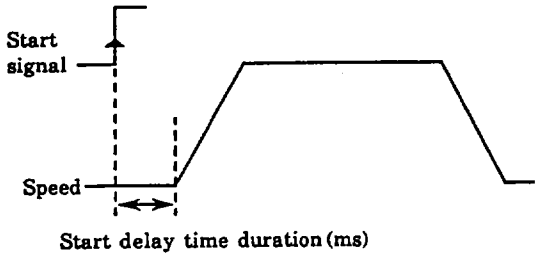
Function Designates the synchronization data of synchronization types 1 to 3.

Format @71 [:< Synchronization data of axis X (0-999999999)>]
[:< Synchronization data of axis Y (0-999999999)>]
[:< Synchronization data of axis Z (0-999999999)>]
[:< Synchronization data of axis W (0-999999999)>]

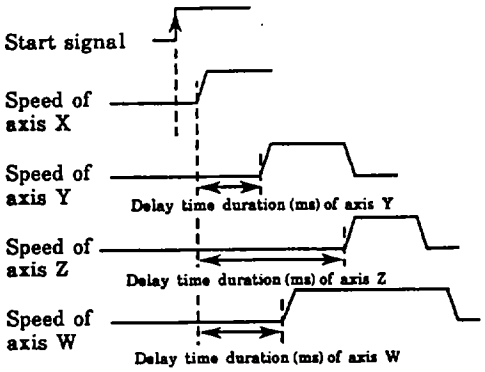
Initial Value @71 : 0 : 0 : 0 : 0

Description This command is effective when synchronization control is set for synchronization types 1 to 3.
The meaning of <Synchronization data> is different among synchronization types.

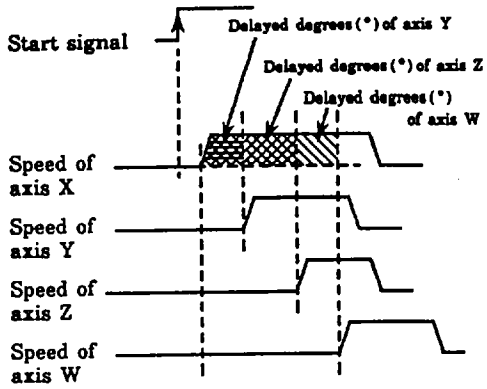
<Synchronization data> for the start delay synchronization type is the start delay time duration (ms). Axis feeding starts delayed by the time duration of <Synchronization data>, with respect to the input start signal.



<Synchronization data> for the time synchronization type is the start delay time duration (ms) for axis feeding of axis Y, Z, or W. Axis feeding of axis Y, Z, or W starts delayed by the time duration of <Synchronization data>, with respect to the axis feeding start signal of axis X after the start signal is input.
This is effective only to axes Y, Z, and W.



<Synchronization data> for the positioning synchronization type is the start delay degree (°) for axis feeding of axis Y, Z, or W. Axis feeding of axis Y, Z, or W starts when axis X becomes the same as <Synchronization data> after inputting the axis feeding start signal of axis X, following the input start signal.
This is effective only to axes Y, Z, and W.



Memo Designaion using the program:
G < Synchronization type number (151-153) >
[X < Synchronization data of axis X (0-999999999) >]
[Y < Synchronization data of axis Y (0-999999999) >]
[Z < Synchronization data of axis Z (0-999999999) >]
[W < Synchronization data of axis W (0-999999999) >]

Reference
"4.4 Commands / Synchronization Type (@44)"
"4.2 Operation for Installation / Synchronization Control"
"4.2 Operation for Installation / Start Delay Synchronization Type"
"4.2 Operation for Installation / Time Synchronization Type"
"4.2 Operation for Installation / Position Synchronization Type"

4.5 Programming Language

This section describes the programming language used for the controller. Function, format, description, etc. of the programming language will be explained.

How to Read the Explanation for the Programming Language

Function Explains simply the function of the programming language.

Format Shows the format of programming language. Note the followings when writing a program.

(1) Write the word (the address and some digits of numbers which follow the address) as usual. This cannot be omitted.
(2) Items in "< >" are numeric values of integral. The range of the value differs among the programming languages, and the range is shown in "( )".
(3) Items in "[ ]" can be omitted.
(4) One of the items shall be selected when more than one item are shown in "{ }".

Example Shows a sample program as an example.

Description Explains the functions of the programming language in detail.

Note Shows notes on use of the programming language.

Reference Shows the items related to the programming language.

Configuration of Programs

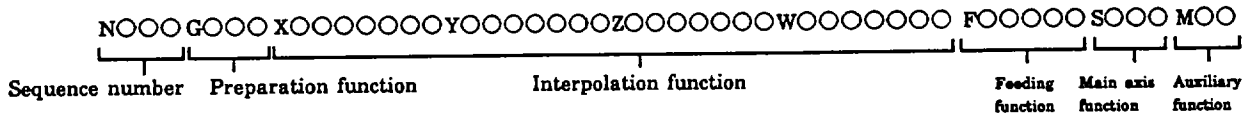
An instruction written in the NC language is given to the controller, and moves the mechanisms such as the servo motor. A group of the instructions are called a "program". The program shall be written in the order that it controls the mechanisms.

The program consists of groups of instructions. One group of instruction is called a "block".

The number to separate (to identify each block) blocks is called "sequence number".

Configuration of Block

(1) A block is configured as follows:



(2) A block consists of multiple words.

(3) M02 or M30 code shall be written at the end of a program.

Words and Addresses

(1) Words which make a block consists of an address and some digits of numeric values which follow the address. (Note that a word which is not followed by numeric values may also exist.)

(2) An alphabetical character is used for an address.

(3) The functions, addresses, and the meanings are as follows:

Function	Address	Meaning
Sequence number	N	Identification number of a block
Preparation function	G	Specifies the meaning of a block.
Interpolation function	X, Y, Z, W	Specifies the positioning data.
Feeding function	F	Specifies the feeding speed.
	A, B, C, D	Specifies the feeding speed of an individual axis.
Main axis function	S	Specifies the rotating speed of the inverter main axis.
Auxiliary function	M	Specifies the way of external control after axis feeding.
	O	Specifies the way of external control during axis feeding.
Dwell	X	Specifies the dwell time.
Designation of axis	X, Y Z, W	Specifies individual axis.
Axis data	X, Y Z, W	Data of individual axis.
Branch function	J	Specifies the destination of jump.
	P	Specifies another program number.
	Q	Specifies the sub-program number.
	M	Termination block of the sub-program
	L	Specifies the times of repetition.
	M	Termination block of repetition
Comment function	/	Specifies the part of comments.

(4) The number of digits of each word is as follows:

N3G3X-9Y-9Z-9W-9F5A5B5C5D5S-3O2M2J3P3Q3L3M3/

The numeric value that follows the word means the number of digits. The word with a (−) sign before its numeric value means the word with sign.

Sequence Numbers

Programming Language

Function Identifies the block.

Format N<Sequence number (1-999)>

Example N10

Description A sequence number can be placed at the beginning of a block. The sequence number is used to identify the desired block.
The order of the numbers of <Sequence number> is not specified particularly. And, <Sequence number> is not necessarily continuous.
The sequence number can be applied to all blocks. Or, if desired, it can be applied to necessary parts of the program.
A sequence number is necessary for the block of branch destination of J-code or Q-code for branch function.

Reference "4.5 Programming Language / Branch Function"

Preparatory Function (G Function)

Function Specifies the meaning of a block.

Format G < Preparation function (00-160) >

Example G00

Description The preparatory function (G function) specifies the meaning of the block. The preparatory function gives a special instruction to the controller. The preparatory function is normally called "G code".

The preparatory function (G code) is classified into three types as follows:

(1) One-shot G code : This type of G code is effective to the specified blocks only.

(2) Modal G code : This type of G code is effective until another G code in the same group is executed.

(3) Data registration G code : This type of G code is used to register data.

Multiple G codes can be used in the same block when the group is different in that block.

Types, operations, and formats (abbreviated) of the preparatory function are as follows:

G code	Group	Operation	Format
G00	1	Positioning	G00XOOOYOOOZOOOWOOO
G01		Linear interpolation	G01XOOOYOOOZOOOWOOO
G04	0	Dwell	G04XOOO
G27		Hardware origin return	G27XY
G28		Software origin return	G28XY
G90	2	Absolute instruction	G90
G91		Incremental instruction	G91
G92	0	Setting of coordination systems	G92XOOOYOOOZOOOWOOO
G101	6	Division input division number	G101XOOOYOOOZOOOWOOO
G102		Acceleration/deceleration type	G102XOOOYOOOZOOOWOOO
G103		Acceleration/deceleration time duration	G103XOOOYOOOZOOOWOOO
G104		Setting of fc remote switch	G104XOOOYOOOZOOOWOOO
G105		Setting of COIN width switch	G105XOOOYOOOZOOOWOOO
G106		Setting of integral/proportional control	G106XOOOYOOOZOOOWOOO
G107		Setting of DC gain	G107XOOOYOOOZOOOWOOO
G110	3	COIN disabled	G110
G111		COIN enabled	G111
G140	4	Pulse input unit system	G140XYZW
G141		Degree input unit system	G141XYZW
G142		Division input unit system	G142XYZW
G143		Feeding length input unit system	G143XYZW
G150	5	Synchronization disabled type	G150XYZW
G151		Start time synchronization type	G151XOOOYOOOZOOOWOOO
G152		Time synchronization type	G152YOOOZOOOWOOO
G153		Position synchronization type	G153YOOOZOOOWOOO
G160	0	Clear position	G160XOOOYOOOZOOOWOOO

! In the table, note the followings:
(1) G codes in group 0 are G codes of one-shot type.
(2) G codes in groups 1 to 5 are G codes of modal type.
(3) G codes in group 6 are G codes of data setting.
(4) Numbers are to be placed in OOO. However, it does not mean the number of digits.

Interpolation Function — Programming Language

Function Feeds the axis to the specified position.

Format { G00 / G01 } [X< Positioning data of axis X (−999999999-+999999999)>]
[Y < Positioning data of axis Y (−999999999-+999999999)>]
[Z < Positioning data of axis Z (−999999999-+999999999)>]
[W< Positioning data of axis W (−999999999-+999999999)>]

Example G00X1000

Description The interpolation function feeds the axis to the specified position. The preparation function which specifies this function is grouped in modal G code. There are two G codes in that group as follows:

(1) G00 code is used for positioning instruction. Positioning is carried out at fast forwarding speed to the position specified by <Positioning data>. The fast forwarding speed is to be registered to the controller using the high feeding speed command (@51). It cannot be controlled directly from the program.

(2) G01 code is used for linear interpolation. Positioning is carried out at the feeding speed which has been specified by the feeding function, to the position specified by <Positioning data>.

Note The controller does not carry out linear interpolation actually. However, "linear interpolation" is a general term used as NC language. Any other word is not suitable to mean the operation of G01 code; and therefore, we use that word for explaining operations of this system.

Reference "4.5 Programming Language / Absolute Instruction and Incremental Instruction"
"4.5 Programming Language / Feeding Function"
"4.4 Commands / High Feeding Speed (@51)"

Feeding Function

Programming language

Function Specifies the feeding speed.

Format [F< Feeding speed of all axes (0-65535)>]
[A < Feeding speed of axis X (0-65535)>]
[B < Feeding speed of axis Y (0-65535)>]
[C < Feeding speed of axis Z (0-65535)>]
[D < Feeding speed of axis W (0-65535)>]

Example F100

Description The feeding function specifies the speed of axis feeding. This is effective for axis feeding of linear interpolation.
The once-specified speed by the feeding function is effective until another speed is specified. Therefore, it is not necessary to specify the speed for all blocks which concern axis feeding.
The F code specifies the feeding speed of all axes. The A code specifies the feeding speed of axis X, the B code specifies that of axis Y, the C code specifies that of axis Z, the D code specifies that of axis W. The code A, B, C, or D has priority over the F code. When the code A, B, C, or D specifies a speed in a block in which the speed has already been specified by the F code, the speed specified by the code A, B, C, or D is used.
For example: When F1000B1310 is specified, 1000 is used for axes X, Z, and W, and 1310 is used for axis Y.
The unit of feeding speed is different between the acceleration types 0 and 1 and the acceleration types 2 to 11. The unit for the acceleration types 0 and 1 is pps, and that for the acceleration types 2 to 11 is ms.
An upper limit of the feeding speed can be set. When actual feeding speed is about to exceed the set upper limit, it is kept at the speed of the specified upper limit. The upper limit value is to be registered to the controller using the maximum feeding speed command (@52). It cannot be specified directly from the controller.

Reference "4.5 Programming Language / Interpolation Function"
"4.2 Operation for Installation / Feeding Speed Parameters"
"4.4 Commands / Maximum Feeding Speed (@52)"

Main Axis Function

Programming Language

Function Specifies the rotating speed (%) of the main axis of the inverter.

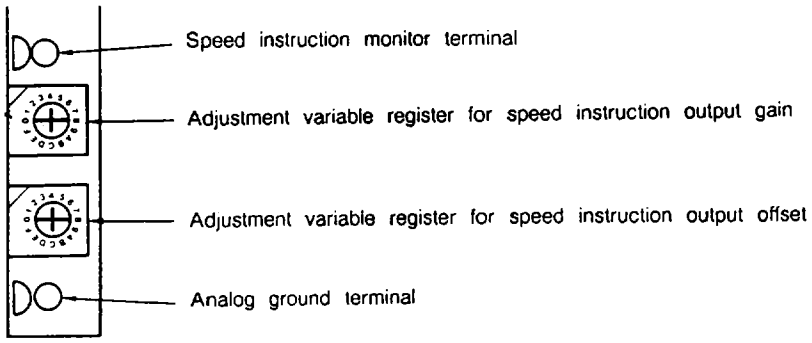
Format S <Rotating speed of main axis (−100-+100)>

Example S 50

Description This command is effective to the axis of the inverter.
The main axis function is the code to be used to control the axis of the inverter. It specifies the output speed to the axis of the inverter.
The unit of <Rotating speed of main axis> is percent (%), and it is used for rotating direction (contact output for normal / reverse rotation switching) and analog speed.
The plus or minus sign of <Rotating speed of main axis> specifies the rotating direction. Specifying <Rotating speed of main axis> with the plus sign rotates the axis in normal direction (CW : clockwise), and specifying <Rotating speed of main axis> with the minus sign rotates the axis in reverse direction (CCW : counterclockwise).
The value of <Rotating speed of main axis> specifies the analog speed. The value of <Rotating speed of main axis> (0 to 100%) specifies the voltage level to be applied to the inverter. 100% is for the voltage level of 6V (factory-set value).

The maximum voltage level is factory-set to 6V. This voltage can be readjusted between 6 and 10V.
When readjusted to 10V, 100% is for the voltage level of 10V.
The desired voltage level can be obtained by turning the variable register located on the axis controller for the inverter which is on the front panel of the controller, and by using the terminal also located there. Follow the steps below for adjustment.

(1) Connect an oscilloscope to the <AOUT> (speed instruction monitor) signal terminal and to the <AGND> (analog ground) signal terminal.
(2) Turn <Adjustment variable register for speed instruction output gain> for the maximum output voltage you desire, while observing the oscilloscope.



Reference "4.2 Operation for Installation / Setting of the Inverter"

Auxiliary Function

Programming Language

Function Outputs the 2-digit code signal of BCD and the strobe signal to the external I / O device.

Format 〔 O<function number O (00-99)> 〕〔 M< M function number (00-99)> 〕

Example M30

Description The auxiliary function outputs the 2-digit code signal of BCD and the strobe signal to the external input / output device such as a sequencer. These signals are used to control on and off by the external input / output device.
There are two types of code, M code and O code, for the auxiliary function. The M code outputs a signal after completing axis feeding. The O code outputs a signal during and after axis feeding. The same output port is used for the M code and the O code, and it is M OUT 0-7 (#42-49). The output signal of the auxiliary function can be enabled or disabled using the initial 2 command (@9). The operation mode command (@6) selects whether to enable the output by the M code only or to enable the output by the M code and O code.
Either one M code or one O code, or both is effective in one block.
Besides that the following <M function number> is used to output the signal to the external input / output device, it is used for a special purpose. And, it can be enabled or disabled using the initial 2 command (@9).

(1) M00 : Program stop
Stops the programmed operation after execution of the block to which the M00 code is used is complete. The restart operation restarts the programmed operation.

(2) M01 : Optional stop
Stops the programmed operation after execution of the block to which the M01 code is used is complete, in the same way as the M00 code does. The restart operation restarts the programmed operation.

(3) $\left.\begin{matrix}\text{M30}\\ \text{M02}\end{matrix}\right\}$: End of program
This means the end of the program, and the programmed operation is terminated.

Reference
"4.2 Operation for Installation / M Output Interface"
"4.2 Operation for Installation / M Output Parameters"
"4.4 Commands / Operation Modes (@6)"
"4.4 Commands / Initial 2 (@9)"

Dwell

Function Gives a waiting time (ms) to the block that follows.

Format G04X< Dwell time (0-600000)>

Example G04X200

Description The dwell instruction makes the waiting time specified with <Dwell time> to the block that follows. The dwell is the G code of one-shot type. The unit of <Dwell time> is ms.
Resolution of the dwell time is 5ms.

Setting of Coordination Systems — Programming Language

Function Changes the current coordination value in the programmed coordination system.

Format G92 [X< Coordination value of axis X (−999999999-+999999999)>]
[Y < Coordination value of axis Y (−999999999-+999999999)>]
[Z < Coordination value of axis Z (−999999999-+999999999)>]
[W < Coordination value of axis W (−999999999-+999999999)>]

Example G92X500Y0Z0W0

Description In order to set the desired coordination system, set <Coordination value of axis X> to the current coordination value of axis X, set <Coordination value of axis Y> to the current coordination value of axis Y, set <Coordination value of axis Z> to the current coordination value of axis Z, and set <Coordination value of axis W> to the current coordination value of axis W. The setting of coordination system is the G code of one-shot type. The unit of <Coordination value> is the same as that of the input unit system. The input unit system is to be specified using the input unit system command (@41) or is to be specified directly from the program (G140 to G143).

Note For multiple rotation of the motor, be sure to clear the absolute position using the following command, before the internal counter overflows.
G92X0Y0Z0W0

Reference "4.5 Programming Language / Input Unit System"
"4.2 Operation for Installation / Input Unit System"
"4.4 Commands / Input Unit System (@41)"

Absolute Instruction and Incremental Instruction

Function Specifies the way to instruct the desired feeding length to each axis.

Format {G90 / G91}

Description Either of the absolute instruction or the incremental instruction can be used to instruct the desired feeding length to each axis. These instructions are the G code of modal type in the same group.

(1) The G90 code is for the absolute instruction. In this case, each axis is instructed with an absolute value, and the final position of axis feeding is to be programmed.

(2) The G91 code is for the incremental instruction. In this case, each axis is instructed with a relative value, and the feeding length of the axis is to be programmed.

Reference "4.5 Programming Language / Interpolation Function"
"4.2 Operation for Installation / Input Unit System"

Origin Return Function

Programming Language

Function Returns to origin.

Format $\left\{\begin{matrix}G27\\G28\end{matrix}\right\}$ [X] [Y] [Z] [W]

Example G27XY

Description There are two types of origin return function, the hardware origin return and the software origin return. These types of origin return function are the G code of one-shot type. In order to carry out the origin return of axis X, set X, to carry out the origin return of axis Y, set Y, to carry out the origin return of axis Z, set Z, and to carry out the origin return of axis W, set W. Setting is to be carried out on addresses only.

(1) The G27 code is for the hardware origin return instruction. In this case, origin return is carried out to the origin position (0 point) in the mechanical coordination system.

(2) The G28 code is for the software origin return instruction. In this case, origin return is carried out to the origin position (0 point) in the programmed coordination system.

Reference "4.5 Programming Language / Setting of Coordination Systems"
"4.2 Operation for Installation / Operation Modes"
"4.2 Operation for Installation / Coordination System"
"4.3 Operation / Origin Return Mode"

Clearing Positions

Programming Language

Function Clears all positioning data to 0 after axis feeding is completed.

Format G160 〔 X< Positioning data of axis X (−999999999-+999999999)> 〕
〔 Y < Positioning data of axis Y (−999999999-+999999999)> 〕
〔 Z < Positioning data of axis Z (−999999999-+999999999)> 〕
〔 W < Positioning data of axis W (−999999999-+999999999)> 〕

Example G160X500Y1000Z2000W0

Description The clear position carries out axis feeding to the position specified by <Positioning data>. It clears all the positioning data to 0 after the axis feeding is completed. The clear position is the G code of one-shot type.
The operation of axis feeding is the same as that of linear interpolation of the G01 code.
Write as follows in order to set all the positioning data which are not subject to axis feeding to 0.

G91G160X0

The positioning data to be cleared are as follows:

(1) Mechanical coordination value
(2) Absolute coordination value
(3) Base coordination value
(4) Encoder coordination value
(5) Programmed coordination value

Reference "4.5 Programming Language / Interpolation Function"
"4.2 Operation for Installation / Coordination System"
"4.4 Commands / Clearing Coordination System (@4)"

COIN

Function Enables or disables the COIN (Completion of Instruction).

Format { G110 / G111 }

Description The positioning status signal output from the driver of the motor is called "COIN". The COIN sets whether to carry out positioning accuracy check after completion of axis feeding. The COIN function is the G code of modal, and there are two types of G code in the same group.

(1) The G110 code disables the COIN.
(2) The G111 code enables the COIN.

The COIN can be set using the initial 1 command (@8).
Refer to the instruction manual of the DYNASERV for the positioning alignment width (COIN) of the DYNASERV.

Memo The POSW1 and the POSW2 to the DYNASERV are set open at the factory.

Reference "4.2 Operation for Installation / Type by type setting of the motor"
"4.4 Commands / Initial 1 (@8)"

Input Unit System

Programming Language

Function Determines the unit system for the positioning data to be input.

Format G < Input unit system number (140-143)> [X] [Y] [Z] [W]

Example G141XY

Description The input unit system determines what unit system to be used for inputting the positioning data. <Input unit system number> selects the type of the desired input unit system. The input unit system function is the G code of modal type, and there are four types of G code in the same group.

(1) The G140 code selects the pulse input unit system. The number of pulses (p) is used as the unit of the positioning data in the pulse input unit system.

(2) The G141 code selects the degree input unit system. The length in degrees from the center of the motor rotation is used as the unit of the positioning data in the degree input unit system.

(3) The G142 code selects the division input unit system. The division number obtained by dividing 360° by the specified number is used as the unit of the positioning data in the division input unit system.

(4) The G143 code selects the feeding length input unit system. The feeding length by the motor is used as the unit of the positioning data in the feeding length input unit system.

In order to set the desired input unit system to axis X using <Input unit system number>, set X, to set the desired input unit system to axis Y using <Input unit system number>, set Y, to set the desired input unit system to axis Z using <Input unit system number>, set Z, to set the desired input unit system to axis W using <Input unit system number>, set W. Setting is to be carried out on addresses only.

The input unit system can be designated using the input unit system command (@41).

Reference "4.5 Programming Language / Data Registration"
"4.2 Operation for Installation / Input Unit System"
"4.4 Commands / Input Unit System (@41)"

Synchronization Control — Programming Language

Function Specifies the type of synchronization control and the synchronization data.

Format

{G150 (X) (Y) (Z) (W)
G <Synchronization type number (151-153)> (X <Synchronization data of axis X (0-999999999)>)
(Y <Synchronization data of axis Y (0-999999999)>)
(Z <Synchronization data of axis Z (0-999999999)>)
(W <Synchronization data of axis W (0-999999999)>)}

Example G152Y100Z100W200

Description To control the start timing of axis feeding with respect to the start signal by the axis feeding instruction is called "synchronization control". The type of synchronization control is called "synchronization type", and the desired type is selected using <Synchronization type number>. The synchronization control function is the G code of modal type, and there are four types of G code in the same group.

(1) The G150 code selects the synchronization disabled type. Using this type starts axis feeding immediately when the start signal is input.
In order to specify this type for axis X, set X, to specify this type for axis Y, set Y, to specify this type for axis Z, set Z, to specify this type for axis W, set W. Setting is to be carried out on addresses only.

(2) The G151 code selects the start delay synchronization type. Using this type starts axis feeding delayed by the time duration of <Synchronization data>, with respect to the input start signal. <Synchronization data> means the start delay time duration (ms).

(3) The G152 code selects the time synchronization type. Using this type starts axis feeding of axis Y, Z, or W delayed by the time duration of <Synchronization data>, with respect to the axis feeding start signal of axis X after the start signal is input.
<Synchronization data> means the delay time duration (ms) to the start of axis feeding of axis Y, Z, or W.
This is effective only to axes Y, Z, and W.

(4) The G153 code selects the positioning synchronization type. Using this type starts axis feeding of axis Y, Z, or W when axis X becomes the same as <Synchronization data> after inputting the axis feeding start signal of axis X, following the input start signal. <Synchronization data> means the delay degrees (°) to the start of axis feeding of axis Y, Z, or W.
This is effective only to axes Y, Z, and W.

The synchronization control can be designated using the synchronization type command (@44) and the synchronization data command (@71).

Reference
"4.2 Operation for Installation / Synchronization Control"
"4.4 Commands / Synchronization Type (@44)"
"4.4 Commands / Synchronization Data (@71)"

Data Registration

Programming Language

Function Registers various data to the controller.

Format G < Data number (101-107) >

[X <Registered data of axis X>]
[Y <Registered data of axis Y>]
[Z <Registered data of axis Z>]
[W <Registered data of axis W>]

Example G101X512Y1024Z512W1024

Description This command registers the various data to the controller using the program. This function is effective until new data is registered using the same G code in the G code of data registration. There are seven types of G code in the same group.

(1) The G101 code registers the number of divisions for division input in the division input unit system. This is effective when the division input unit system is selected as the input unit system. <Registered data> means the number of divisions for division input. The range which can be specified is from 0 to 1024.

(2) The G102 code registers the acceleration / deceleration type of the acceleration / deceleration control. <Registered data> means the acceleration / deceleration type. The range which can be specified is from 0 to 11.

(3) The G103 code registers the acceleration / deceleration time duration of the acceleration / deceleration types 0 and 1. This is effective when the acceleration / deceleration types 0 and 1 are selected for the acceleration / deceleration control. <Registered data> means the acceleration / deceleration time duration (ms). The range which can be specified is from 0 to 65535.

(4) The G104 code registers the fc remote switch of the DYNASERV (positioning instruction mode) axis. <Registered data> means the fc remote switch. The range which can be specified is from 0 to 15.

(5) The G105 code registers the COIN width switch of the DYNASERV (positioning instruction mode) axis. <Registered data> means the COIN width switch. The range which can be specified is from 0 to 3.

(6) The G106 code registers the integral / proportional control of the DYNASERV (positioning instruction mode) axis, the DYNASERV (speed instruction mode) axis, the AC / DC servo motor (positioning instruction mode) axis, or the AC / DC servo motor (speed instruction mode) axis. <Registered data> means the intrgral / proportional control. The range which can be specified is from 0 to 1.

(7) The G107 code registers the DC gain of the DYNASERV (positioning instruction mode) axis or the DYNASERV (speed instruction mode) axis. <Registered data> means the DC gain. The range which can be specified is from 0 to 7.

Various data registration using commands is available.

Reference

"4.2 Operation for Installation / Division Input Unit System"
"4.4 Commands / Number of Divisions (@56)"
"4.2 Operation for Installation / Acceleration/Deceleration Control"
"4.4 Commands / Acceleration/Deceleration Type (@42)"
"4.2 Operation for Installation / Acceleration/Deceleration Types 0 and 1"
"4.4 Commands / Acceleration/Deceleration Time Duration (@53)"
"4.2 Operation for Installation / Type by type setting of the motor"
"4.4 Commaned / fc Remote Switch (@32)"
"4.4 Commands / COIN width Switch (@24)"
"4.4 Commands / Integ/Proportional Control (@21)"
"4.4 Commands / DC Gain (@22)"

Branch Function

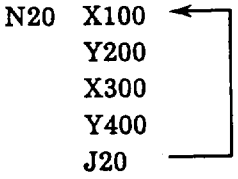
Function Branches the program.

Format
(1) J < Branch destination sequence number (1-999) >
(2) P <Branch destination program number (1-128)>
(3) Q <Sub-program number (1-999)>
(4) L < Repetition times (0-999)>

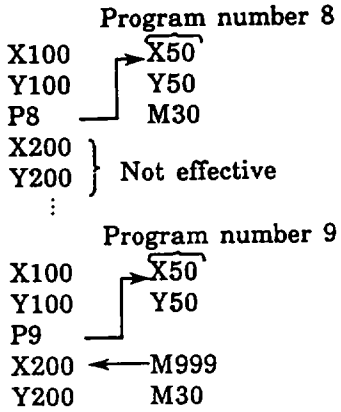
Example
(1) J20
(2) P8
(3) Q10
(4) L15

Description The branch function is used to branch the program. The branch function can be classified into three groups by the way of branch.

(1) **Jump Function (J Branch)**
Form (1) above is used for jump function. The jump function is used to continue the execution of the program by jumping the program to the line of <Branch sequence number>.

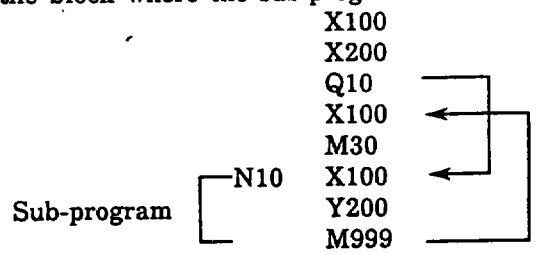


(2) **Program Jump Function (P Branch)**
Form (2) above is used for calling another program. When the called program ends with the M30 code, the programmed operation is terminated at that time. When it ends with the M999 code, the programmed operation returns to the original program.



(3) Sub-program Function (Q Branch)

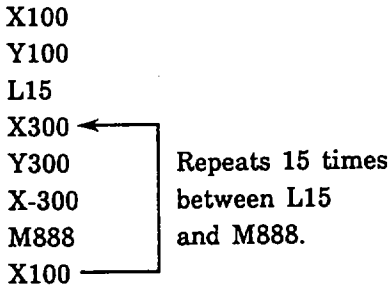
Form (3) above is used for calling the sub-program. The program which is independent from any other programs and which ends with the M999 code is called "sub-program". This function calls the sub-program which starts with <Sub-program number>. The M999 code is called "end of sub-program". When the M999 code is executed, the program goes to the block that follows the block where the sub-program is called (the block of Q code).



While using the sub-program, another sub-program can be called. This function is called "nesting of sub-program", and nesting is possible up to 32 levels.

(4) Repeating Function (L Branch)

Form (4) above is used for repeating function. The repeating function executes the program repeatedly between the block of L code and the block of M888 code. The M888 code is called "end of repeat" When the execution of the program is repeated specified times of <Repeating times>, the program goes to the block next to the block of M888 code.



In the program part subject to repetition, repeating function cannot be used, but the sub-program (3) above can be called. And, it is possible to use the repeating function in that sub-program.

Note The operation of the program cannot be guaranteed if the program jumps to the sub-program or to the repeated program part from outside, or vise versa.

Reference "4.5 Programming Language / Sequence Numbers"

Comment Function

Programming Language

Function Writes comments in the program.

Format / [<comment>]

Example /1310kpps

Description This command can write the comments in the program. With the comments written by this function, the program can be read with ease. <Comment> that follows / (slash) functions as the comments to the program only. It does not affect the execution of the program.

G91X1000F1310 / INC 1310kpps
G4X500 / timer
M30 / end of program

Note It is possible to make a block which contains only comments. However, if a number of blocks which contains comments only are created successively, the following axis feeding is occasionally delayed.

5. TROUBLES AND REMEDIES / MAINTENANCE AND CHECKS

5.1 Malfunction and Remedies

(1) Display and Output of Errors and Alarm

Display and output of errors and alarm of the controller are as follows:

Table 5.1 Display and Output of Errors and Alarm

			Output Alarming an Error	Output of Error Stop
Output type	LED Display on the Front Panel		<ERR> lights	<ERR> flashes
	<CN1> output		ERROR STOP (#35) output signal "OFF"	ERROR STOP (#35) output signal "ON"
	Communication Output	When an Error Occurs	Error message output (Exx)	
		Recall of the Error Condition	Display of hexadecimal number by message command (@0)	
Recover			Recovers by the following operation.	
	Parallel Transmission Mode		———	RESET (#10) input signal "ON"
	RS-232C Mode		———	Reset command (@1)
	Common		———	Set the reset switch on the front panel to "ON" or turn power on.

Note: See "Appendix (2) Error / Alarm list" for the meaning of error messages.

(2) The Meaning of the LED Display on the Front Panel

Table 5.2 The Meaning of the LED Display on the Front Panel

Symbol	Meaning of Lit Condition	Meaning of Flashing Condition
DRY	Controller preparation completed	Necessary to replace the lithium battery for memory backup
ERR	Serious error (Warning)	Error stop
BUSY	Operation	———
PROG	Programmed operation	———
M-EN	M signal output	———
ORGF	Origin return completed	———
COIN	Positioning completed	———
PLC	Setting of parallel transmission mode	———

(3) Follow the steps below for remedy of troubles without error and alarm output.

Table 5.3 Troubles without Error and Alarm Output

Trouble	Remedy
Personal computer communication is impossible.	◆ Check the communication cable. ◆ Check the setting of RS-232C connection to the personal computer. ◆ Check the setting of the DIP switch on the front panel.
An LED on the front panel does not light.	◆ Check the power supply voltage and the polarity.
All the LEDs on the front panel light.	◆ Check the power supply voltage.
No operation by the input to terminal. <CN1>	◆ Check the connection cables. ◆ Check the 24V power supply voltage. ◆ Check that the appropriate input terminal is assigned to the transmission mode command (@5).
Servo does not turn on by the servo ON / OFF command (@20).	◆ Check that the status of the SERVO ON / OFF (#3) input terminal is on.
The DYNASERV (DD motor) does not start by the start instruction.	◆ Check that there is any error or alarm being output. ◆ Check the setting and adjustment of the DYNASERV.
Condition is not set in axis ready when the DYNASERV (DD motor) operation is completed.	◆ Check for adjustment of the DYNASERV **(Note)** When set to position P mode, only the torque corresponding to the positioning deviation of the DD motor is output. So, arrangement cannot be done when arrangement width is too narrow. ◆ Check the M-signal interface when M-output is enabled.
The amount of motor movement is different from that instructed.	◆ Check the motor type command (@40). ◆ Check the feeding factor command (@60) when set in the feeding length input unit system.

5.2 Maintenance and Checks

(1) About the Lithium Battery for Memory Backup

A lithium battery for memory backup of parameters and data is connected in the box on the back of the controller.

The life of the lithium battery is about 20,000 hours in normal condition.

When necessary to replace the lithium battery, the <LED> on the front panel flashes and the status of the BATTERY ALARM (#39) of connector <CN1> turns on. In this case, replace the battery within two weeks.

CAUTION

In the above condition, if the battery is not replaced within an appropriate period of time, the controller will be set in an error stop condition and the parameters and data will be destroyed.

(2) Replacement of the Lithium Battery

Follow the following procedures and remember the notes also below for replacement of the lithium battery.

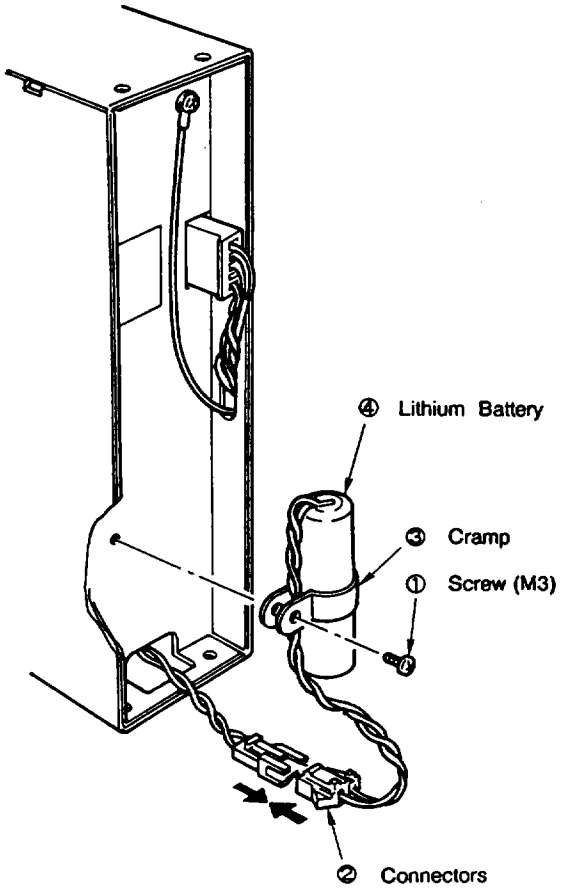


Figure 5.1 Replacement of the lithium Battery

CAUTIONS

- Keep the AC or DC power supply set to "ON" when replacing the lithium battery for memory backup.
- Complete the replacement within five minutes.
- The spare battery is supplied with a specific connector and cable. So, order the spare battery assembly to Yokogawa Precision with product name (R7304LT).

Follow the procedure below for replacement.

(1) Remove the screw (M3) ① and remove also the battery from the box.
(2) Disconnect the connectors ②.
(3) Detach the lithium battery assembly ④ from the cramp ③, and replace the battery with a new one.
(4) Connect the connectors, and fix the cramp firmly with the screw.
(5) Check that the connector is securely connected.

APPENDIX

(1) List of Commands

List of Commands (1/4)

Code No.	Name of Command	Contents / Range of the Setting of 1st Parameter	Contents / Range of the Setting of 2nd Parameter	Contents / Range of the Setting of 3rd Parameter	Contents / Range of the Setting of 4th Parameter
00	Messages	Parameter number 0: Error number when stopped by an error 1: Enabled axis 2: Origin return completion axis 3: Operating condition 4: (Reserved) 5 to 9 } Parameter setting 20 to 79 } of command number 80: Mechanical coordination 81: Programmed coordination 82: (Reserved) 83: Absolute coordination 84: Base coordination 85: Encoder coordination 86: Origin calibration value 87 to 98: (Reserved) 99: D I/O condition			
01	Reseting Error Halt				
02	Stopping Operation				
03	Starting Operation	Origin return operation / Origin calibration operation } 1: specifies axis X. Programmed operation: Program number (1 to 128) Jog operation: axis X and feeding direction −1: − direction 0: Stop 1: + direction	Origin return operation / Origin calibration operation } 1: specifies axis Y. Jog operation: axis Y and feeding direction −1: − direction 0: Stop 1: + direction	Origin return operation / Origin calibration operation } 1: specifies axis Z. Jog operation: axis Z and feeding direction −1: − direction 0: Stop 1: + direction	Origin return operation / Origin calibration operation } 1: specifies axis W. Jog operation: axis W and feeding direction −1: − direction 0: Stop 1: + direction
04	Clearing, Coordination Systems	1: Coordination system clear for axis X	1: Coordination system clear for axis Y	1: Coordination system clear for axis Z	1: Coordination system clear for axis W

List of Commands (2/4)

Code No.	Name of Command	Contents / Range of the Setting of 1st Parameter	Contents / Range of the Setting of 2nd Parameter	Contents / Range of the Setting of 3rd Parameter	Contents / Range of the Setting of 4th Parameter
05	Transmission Mode	0: RS-232C mode ※1: Parallel transmission mode			
06	Operation Mode	Operation mode ※0: Origin return mode 1: Programmed AUTO mode 2: Programmed STEP mode 3: Programmed CONT mode 4: Jog mode 8: Origin calibration mode 10: MDI mode			Type of M-signal output ※0: Only Mout signal 1: Both Mout signal and Oout signal
07	Initial 0	Servo control 0: OFF ※1: ON	Origin return completion switch 0: Disabled ※1: Enabled		
08	Initial 1			Coordination clear switch ※0: Disabled 1: Enabled	COIN switch 0: Disabled ※1: Enabled
09	Initial 2	Mout switch 0: Disabled ※1: Enabled	M01 switch 0: Disabled1 ※1: Enabled	M00 switch 0: Disabled ※1: Enabled	M30 switch 0: Disabled ※1: Enabled
14	Listing Program	Program number (1 to 128)			
15	Resistering Program	Program number (1 to 128)			
16	Deleting Program	Program number (1 to 128)			
17	Listing Programs	Program number (0 to 128)			

List of Commands (3/4)

Code No.	Name of Command (Unit)	Contents / Range of the Setting	Initial Value of 1st Parameter (for Axis X)	Initial Value of 1st Parameter (for Axis Y)	Initial Value of 1st Parameter (for Axis Z)	Initial Value of 1st Parameter (for Axis W)
20	Servo ON and OFF	0: OFF 1: ON	1	1	1	1
21	Integral / Proportional Control	0: Open 1: Close	0	0	0	0
22	DC Gain	0 to 7	0	0	0	0
23	fc Remote Switch	0 to 15	0	0	0	0
24	COIN Width Switch	0 to 3	0	0	0	0
26	Integral Reset Switch	0: Open 1: close	0	0	0	0
28	Proportional Gain Control in Position Loop	0 to 7	0	0	0	0
29	Velocity Offset Correction Switch	0: Disabled 1: Enabled	0	0	0	0
30	Origin Return Direction	0: Normal rotation 1: Reverse rotation	0	0	0	0
31	(−) Direction of Overtravel	0: Disabled 1: Enabled	0	0	0	0
32	(+) Direction of Overtravel	0: Disabled 1: Enabled	0	0	0	0
33	(−) Direction of Soft Limit	0: Disabled 1: Enabled	0	0	0	0
34	(+) Direction of Soft Limit	0: Disabled 1: Enabled	0	0	0	0

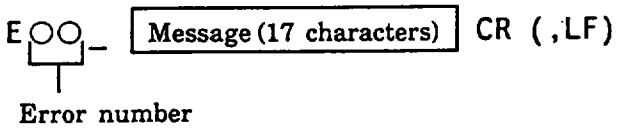
List of Commands (4/4)

Code No.	Name of Command (Unit)	Contents / Range of the Setting	Initial Value of 1st Parameter (for Axis X)	Initial Value of 1st Parameter (for Axis Y)	Initial Value of 1st Parameter (for Axis Z)	Initial Value of 1st Parameter (for Axis W)
35	Origin Return Type	0 to 3	0	0	0	0
37	Rotating Direction of Motors	0: Rotation in CW direction 1: Rotation in CCW direction	0	0	0	0
40	Motor Type	0 to 8	0	0	0	0
41	Input Unit System	0: Pulse input unit system 1: Degree input unit system (multi-rotation) 2: Division input unit system 3: Feeding length input unit system 5: Degree input unit system (1 rotation) 6: Feeding length input unit system (Bi-directional)	0	0	0	0
42	Acceleration / Deceleration Type	0: 3rd-order spline curve 1: Trapezoid curve 2 to 11: Cam curve	0	0	0	0
43	Cam Curve Unit System	0: ms 2: 100ms 1: 10ms	0	0	0	0
44	Synchronization Type	0: Synchronization disable type 1: Start time synchronization type 2: Time synchronization type 3: Position synchronization type	0	0	0	0
46	Offset Value of Origin Return (p)	−999,999,999 to +999,999,999	0	0	0	0
47	Coordination after Origin Return	−999,999,999 to +999,999,999	0	0	0	0
48	(−) Direction of Soft Limit Value	−999,999,999 to +999,999,999	0	0	0	0
49	(+) Direction of Soft Limit Value	−999,999,999 to +999,999,999	0	0	0	0
50	Low Feeding Speed (kpps)	1 to 1360	13	13	13	13
51	High Feeding Speed (kpps)	1 to 1360	655	655	655	655
52	Maximum Feeding Speed (kpps)	1 to 1360	1310	1310	1310	1310
53	Acceleration / Deceleration Time Duration (ms)	0 to 65535	350	350	350	350
54	Deceleration Factors of Cam Curve	0 to 4095	30	30	30	30
55	Resolution of the Motor	0 to 999,999,999	655360	655360	655360	655360
56	Number of Divisions	1 to 1024	1024	1024	1024	1024
59	Internal COIN Width	0 to 32767	0	0	0	0
60	Feeding Length Factor	1 to 999,999,999	655360	655360	655360	655360
62	Input Sensitivity Voltage	0 to 9,999	0	0	0	0
63	Velocity Feed Forward	0 to 120	0	0	0	0
64	Jog Feeding Speed (kpps)	1 to 1360	655	655	655	655
67	Minimum Feeding Time Duration of Cam Curve (ms)	1 to 65535	150	150	150	150
69	Override of Feeding Speed (%)	0 to 100	100	100	100	100
71	Synchronization Data	0 to 999,999,999	0	0	0	0

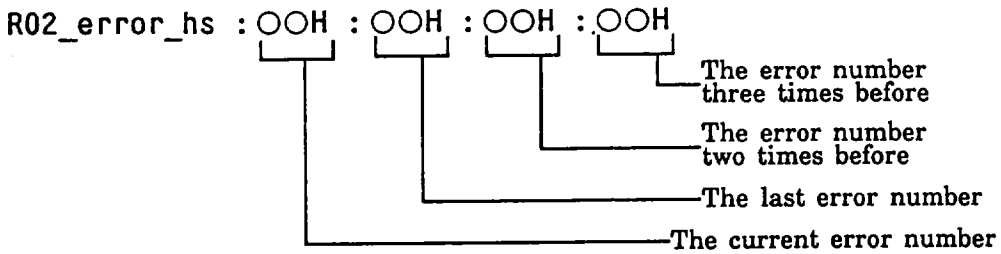
(2) List of Errors and Alarms

When an error occurs during operation of the controller or the motor, the error message is sent to the personal computer, operation display panel, or operation display pendant through the RS-232C communication interface, and it is displayed on the screen. This section describes the error messages, and causes and remedies of the errors.

Each type of errors has an error number which is a decimal number. The form of the error messages displayed on the personal computer, operation display panel, or operation display pendant is as follows.



The history of the error number when operation stops with error can be displayed in hexadecimal number, using the message command (@〔 0 〕:0).



When an error occurs, error processing is carried out. One of the five types of error processing is carried out according to the error occurred. Name of the error type and the type of error processing are as follows.

(1) Error Processing Type 0

① The error message is output.
② The ERROR lamp on the front panel lights.

(2) Error Processing Type 1

① The error message is output.
② The ERROR lamp on the front panel lights.
③ Operating conditions are reset.

(3) Error Processing Type 2

① The error message is output.
② The ERROR lamp on the front panel flashes.
③ Decelerates and stops.
④ Operating conditions are reset.
⑤ Turns in an error halt condition.

(4) Error Processing Type 3

① The error message is output.
② The ERROR lamp on the front panel flashes.
③ Decelerates and stops.
④ Servo is turned off.
⑤ Operating conditions are reset.
⑥ Turns in an error halt condition.

(5) Error Processing Type 4

① The error message is output.
② The ERROR lamp on the front panel flashes.
③ Decelerates and stops.
④ Servo is turned off.
⑤ Operating conditions are reset.
⑥ Turn in an error halt condition.

[!] Operation stops by the error which corresponds to error processing types 2 to 4. When operation stops by the error, resetting is required.
This condition (operation stop by the error) can be known from the flashing of the ERROR lamp on the front panel.

The number at the far left of the message is the error number (decimal notation) which is to be displayed on the screen of the personal computer. The number in parentheses "( )" is the error number (hexadecimal notation) of the return data generated by the message command.

The number in the brackets "[ ]" on the right of the message means the type of error processing. The error with "[1or2]" corresponds to error processing type 1 when not operating in programmed operation, and it corresponds to error processing type 2 when operating in programmed operation.

"Cause" explains simply the cause of the error.

"Remedy" explains how to remove the cause of the error, or how to avoid the error.

01 (01H) SYSTEM error [4]

Cause : The system error has occurred.
Remedy: This may have to be repaired. Contact your sales person.

02 (02H) SCU status error [2]

Cause : An SCU (SERIAL CONTROL UNIT) error has occurred during communications through the RS-232C.
Remedy: Reset the system, and then operate again.

03 (03H) EMG stop [3]

Cause : An emergency stop (EMG) signal has been input.
Remedy: Reset the system, turn on the servo, and then start from the origin return operation.

05 (05H) X axis limit [1or2]

Cause : The mechanical limit of axis X has been detected.
Remedy: (1) When operating in programmed operation, reset the system, feed the axis in the opposite direction of the limit in jog or MDI mode, and then start from the origin return operation.
(2) When not operating in programmed operation, feed the axis in the opposite direction of the limit.

06 (06H) Y axis limit [1or2]

Cause : The mechanical limit of axis Y has been detected.
Remedy: (1) When operating in programmed operation, reset the system, feed the axis in the opposite direction of the limit in jog or MDI mode, and then start from the origin return operation.
(2) When not operating in programmed operation, feed the axis in the opposite direction of the limit.

09 (09H) slave cal error [4]

Cause : An operation error has occurred in the slave MPU.
Remedy: This may have to be repaired. Contact your sales person.

10 (0AH) slave ram error [4]

Cause : An error has occurred when checking the slave RAM memory.
Remedy: This may have to be repaired. Contact your sales person.

11 (0BH) master ram error [4]

Cause : An error has occurred when checking the master RAM memory.
Remedy: This may have to be repaired. Contact your sales person.

12 (0CH) battery error [4]

Cause : The battery for memory backup is disconnected, or data to be backed up has been destroyed because of insufficient voltage.
Remedy: This may have to be repaired. Contact your sales person.

13 (0DH) axis ID error [4]

Cause : The ID number of the axis board does not match.
Remedy: This may have to be repaired. Contact your sales person.

14 (0EH) mode change error [0]

Cause : Change of operation mode is not appropriate.
Remedy: Once complete the current operation, and then change to another operation.

15 (0FH) slave set error [4]

Cause : A slave setting error has occurred.
Remedy: This may have to be repaired. Contact your sales person.

17 (11H) driver servo off [2]

Cause : An axis feeding instruction has been carried out when the driver is in servo off condition.
Remedy: Reset the system, and then set the driver in servo on condition.

18 (12H) driver error [3]

Cause : The driver has been in an error condition.
Remedy: Carry out diagnosis on the driver, and remove the cause of the error. Then, reset the system, and start from the origin return operation.

19 (13H) driver over load [2]

Cause : The driver has been in an overload condition.
Remedy: Remove the cause of the error (overload to the motor). Then, reset the system, and operate again.

22 (16H) not ready data [1or2]

Cause : The necessary data has not been prepared when starting operation.
Remedy: Carry out the following operation, and if the error still exists, contact the Company.
(1) When operating in programmed operation, reset the system and then start from the origin return operation.
(2) When not operating in programmed operation, try the same operation again.

23 (17H) CAM time error [1or2]

Cause : For acceleration/deceleration control of a cam curve, the desired cam curve cannot be created because the feeding time duration instruction is too long.
Remedy: Set the feeding time duration instruction short. Then, perform as follows.
(1) When operating in programmed operation, reset the system and then start from the origin return operation.
(2) When not operating in programmed operation, try the same operation again.

24 (18H) servo sw off [0]

Cause : Servo has been turned on using the servo ON / OFF command (@20) when the SERVO ON/OFF input connector is set to off.
Remedy: Set the SERVO ON/OFF input connector to on. Then, turn on the servo again.

25 (19H) under operation [0]

Cause : A prohibited @ command has been used during operation.
Remedy: Use the @ command after completing operation.

26 (1AH) illegal start [0]

Cause : Start has been activated (by command or input signal) in MDI operation or in jog operation of parallel transmission mode.
Remedy: Carry out correct operation.

27 (1BH) illegal @-cord [0]

Cause : An undefined @ command has been used.
Remedy: Carry out correct operation.

28 (1CH) @-cord format [0]

Cause : An error has bee found in the formula of the @ command or the parameter.
Remedy: Carry out correct operation.

29 (1DH) program starting [0]

Cause : A @ command which cannot be used during programmed operation has been used.
Remedy: Use the @ command after completing programmed operation.

30 (1EH) command inhibited [0]

Cause : For program registration, an @ command has been used in program input condition.
Remedy: Use the @ command after exiting from program input condition.

31 (1FH) NC data inhibited [0]

Cause : An NC data has been input during MDI operation or without in program registration condition.
Remedy: Carry out correct operation.

32 (20H) not EMG or error [0]

Cause : The reset command (@1) has been used when not in emergency stop or in an error halt condition.
Remedy: Carry out correct operation.

33 (21H) under PLC [0]

Cause : An @ command has been used in parallel transmission mode.
Remedy: Set the operation in the RS-232C mode (@5:0). Then, use the @ command.

34 (22H) illegal file no. [1or2]

Cause : A program which has not been registered has been called.
Remedy: Call the program which has been registered.

35 (23H) illegal axis no. [1or2]

Cause : A disabled axis (not connected axis) has been designated.
Remedy: Do not designate a disabled axis.

36 (24H) file not set [1or2]

Cause : The desired program has not been called yet.
Remedy: Call the desired program. Then, carry out correct operation.

37 (25H) CR not found [1or2]

Cause : A CR (,LF) character has not been found when receiving data through the RS-232C communication interface.
Remedy: Write a CR (,LF) character as the transmission terminator at the end of the transmission data.

38 (26H) unfit value [1or2]

Cause : There has been an incorrect value in the program.
Remedy: Make correction for the value.

39 (27H) unfit word [1or2]

Cause : There has been a word error in the program.
Remedy: Make correction for the word.

40 (28H) illegal G no. [1or2]

Cause : An undefined preparation function (G code) has been used in the program.
Remedy: Delete the undefined preparation function (G code).

41 (29H) origin not set [1or2]

Cause : Programmed operation has been carried out without performing origin return operation.
Remedy: Perform origin return operation. Then, carry out programmed operation.

42 (2AH) file dir overflow [1or2]

Cause : The number of registered programs has exceeded 128.
Remedy: Consider deleting of unnecessary programs.

43 (2BH) not found N [1or2]

Cause : The branch destination of the branch function has not been found in the program.
Remedy: Write the sequence number (N code) in the branch destination.

44 (2CH) under error stop [0]

Cause : An @ command has been used in the condition of error halt.
Remedy: Carry out correct operation.

45 (2DH) NC format error [1or2]

Cause : An expression (format) error of the program has been found when reading the program.
Remedy: Make correction to the format.

46 (2EH) not used massage [0]

Cause : An undefined parameter has been used with the @ command.
Remedy: Carry out correct operation.

47 (2FH) data area over [1or2]

Cause : The total number of the registered character of the program has exceeded 32 K-bytes.
Remedy: Consider deleting of the unnecessary programs.

48 (30H) illegal M no. [1or2]

Cause : An undefined auxiliary function (M code) has been used in the program.
Remedy: Delete the undefined auxiliary function (M code).

50 (32H) X axis soft limit [1or2]

Cause : Soft limit of axis X has been detected.
Remedy: (1) When operating in programmed operation, reset the system, and then start from the origin return operation.
(2) When not operating in programmed operation, feed the axis in the opposite direction to the limit.

51 (33H) Y axis soft limit [1or2]

Cause : Soft limit of axis Y has been detected.
Remedy: (1) When operating in programmed operation, reset the system, and then start from the origin return operation.
(2) When not operating in programmed operation, feed the axis in the opposite direction to the limit.

54 (36H) illegal return [1or2]

Cause : One of the following errors on the use of the end of repetition (M888 code) of branch function or on the use of the end of sub-program (M999 code) instruction has been found in the program.
(1) Another repetition is used in the repetition period of the repetition function.
(2) The end of repetition (M888 cod) block is used in the repetition function when there is no repetition start (L code) block.
(3) The end of sub-program (M999 code) block is used in the repetition function when there is no sub-program call (Q code) block.
Remedy: Check the flow of the program, and match correctly the start of repetition with the end of repetition, and also the call of sub-program with the end of sub-program, respectively.

55 (37H) over stack [1or2]

Cause : The depth of multiplication of the sub-program in the branch function of the program has exceeded 32.
Remedy: Adjust the depth of the multiplexed sub-program within 32.

56 (38H) over err cornter [3]

Cause : The amount of positioning deviation in the positioning controller part has exceeded ±32767 pulses.
Remedy: Check the setting of the control mode of the driver / controller, and check also the control parameters.

57 (39H) over vel offset [3]

Cause : The speed offset correction value in the positioning controller part has exceeded ±127mV for the axis of the DYNASERV, and ±159 mV for the axis of the AC / DC servo motor.
Remedy: Adjust the speed offset of the driver. When an external force is applied to the motor when the motor stops, disable the speed offset correction.

58 (3AH) velocity clamp [0]

Cause : A feeding speed which is more than maximum feeding speed has been instructed.
Remedy: The feeding speed is automatically adjusted to the maximum feeding speed, and operation is continued.

59 (3BH) disable mode [1]

Cause : An undefined operation mode has been designated.
Remedy: Designate correct operation mode.

60 (3CH) X origin over run [1or2]

Cause : During the origin return operation of axis X with origin return type 2 or 3, operation has not stopped at the origin sensor.
Remedy: Reregister lower feeding speed. Set the sensing time duration long, and acceleration / deceleration time duration short.

61 (3DH) Y origin over run [1or2]

Cause : During the origin return operation of axis Y with origin return type 2 or 3, operation has not stopped at the origin sensor.
Remedy: Reregister lower feeding speed. Set the sensing time duration long, and acceleration / deceleration time duration short.

62 (3EH) Z origin over run [1or2]

Cause : During the origin return operation of axis Z with origin return type 2 or 3, operation has not stopped at the origin sensor.
Remedy: Reregister lower feeding speed. Set the sensing time duration long, and acceleration / deceleration time duration short.

63 (3FH) W origin over run [1or2]

Cause : During the origin return operation of axis W with origin return type 2 or 3, operation has not stopped at the origin sensor.
Remedy: Reregister lower feeding speed. Set the sensing time duration long, and acceleration / deceleration time duration short.

68 (44H) X fb-pulse error [3]

Cause : The interface connector (<CN3-□>) of axis X has come off, or there is no feedback pulse because of disconnected (or cut out) cable.
Remedy: Check the connector and the cable.

69 (45H) Y fb-pulse error [3]

Cause : The interface connector (<CN3-□>) of axis Y has come off, or there is no feedback pulse because of disconnected (or cut out) cable.
Remedy: Check the connector and the cable.

70 (46H) Z fb-pulse error [3]

Cause : The interface connector (<CN3-□>) of axis Z has come off, or there is no feedback pulse because of disconnected (or cut out) cable.
Remedy: Check the connector and the cable.

71 (47H) W fb-pulse error [3]

Cause : The interface connector (<CN3-□>) of axis W has come off, or there is no feedback pulse because of disconnected (or cut out) cable.
Remedy: Check the connector and the cable.

(3) List of Connector Specifications

■ <CN2> RS-232C Connection Specifications

Pin No.	Name of Signal	Input / Output	Meaning of Signal
1	FG		Frame ground
2	T_xD	Output	Transmit data
3	R_xD	Input	Receive data
4	RTS	Output	Request to send
5	CTS	Input	Request ready
6			
7	SG		Signal ground
8			
9			
10			
11			
12			
13			
14			
15			
16			
17	+5V DC		+ side of 5V DC power supply
18	+5V DC		+ side of 5V DC power supply
19			
20			
21	TBRET	Input	(Reserved)
22			
23	EMG	Input	Emergency stop
24	GND		Ground
25	GND		Ground