Technical
Manual

LINEARSERV LM/UM Series
Technical Manual

1st edition

Introduction

Thank you very much for your purchase of the Linear servo actuator LINEARSERV. The LINEARSERV is a direct drive type linear servo actuator that achieves high speed and high precision. In combination with the DrvMII type driver, it can be used in a wide range of applications as an FA actuator for semiconductor manufacturing machines, automated assembly machines and high-precision positioning devices.
Please refer to this tecnical manual thoroughly when you use the product.

Precautions for Using this Technical Manual

1. Please make sure that this manual is handed out to the end user.
2. Please read this manual thoroughly and understand the contents fully before proceeding to the operation of the product.
3. Please note that the safety protection may be lost and the proper safety may not be guaranteed if the product is not used according to the instructions described in this manual.
4. Always make sure that this manual is handy for the operator when using this product. If it is stained or lost, we will distribute copies upon request, subject to charge.
5. This manual explains details of the features included in the product and does not guarantee to meet the specific purpose of the customers.
6. No part of this manual may be reprinted or reproduced in any form without permission.
7. The information in this document is subject to change without notice.
8. The information contained in this document is believed to be accurate at the time of publication, but if you notice any inaccuracies, errors, or omissions, please contact our sales or service staff.

Regarding the safe usage of this device

- This product has been marked with △ and △ Warnings signs so that it can be used safely. Ignoring precautions and prohibitions related to these signs and using this product in an incorrect way may cause danger to the life and body of the operator. Always follow the precautions and observe the prohibitions explained below.
- Please make sure to understand the information given below completely before you start reading the technical manual.
- Please keep the technical manual and this sheet handy while using the product. In addition, make sure that they are handed out to the operator of the product.

⚠ Warnings

● Operation warning:

The motor's slider part of this device moves at a very high speed. People and objects should be kept outside the moving area when a load is attached.

● Warning about electric shock:

Make sure to connect the device to ground to avoid electric shock.
Make sure to turn the power off and pull out the power supply plug when connecting cables to the driver part.
Make sure to turn the power off and pull out the power supply plug when removing the cover of the driver part while performing adjustment operations, etc.

● Fire and electric shock warning:

If any abnormalities such as abnormal noise, bad smell, or release of fumes that coming from the device are detected while it is in operation, turn the power off immediately, pull out the power supply plug, and contact us.
If the device is dropped or given a strong impact, stop the operation immediately, turn the power off, and contact us.
Do not operate at power supply voltages other than the one indicated on the device.

● Fire and electric shock warning:

Avoid dropping or inserting metal shards or combustible materials, or allowing water to get into the opening parts of the device (e.g., the clearance between the slider part and stator or the air vent of the driver part). In such an eventuality, turn the power off immediately and contact us.
The cables coming out from the slider part should not be forcibly bent, twisted, pulled, heated, or placed under a heavy object.
Never try to remodel or repair the device by yourself.

■ This product has been marked with △ and △Precautions signs so that it can be used safely. Ignoring precautions and prohibitions related to these signs and using this product in an incorrect way may cause danger to the life and body of the operator. Always follow the precautions and observe the prohibitions explained below.

⚠ Precautions

- Make sure to read the technical manual before using the device.
  Operational mistakes and faulty wiring may result in damages and failure of the device.
- Make sure to check the wiring once more before turning the power on.
  Faulty wiring may result in fire, electric shock, or damage of the device.
- Confirm that the proper combination of motor and driver parts is used. Using the device with an incorrect configuration may result in failure. (Be sure to confirm the model--MODEL--on the rating nameplates.)
- Make sure the conditions of temperature, humidity, dust, etc. are as specified for the installation and storage environments.
- Do not block the air vent of the device. Keep the specified open space around the device as well. Poor ventilation may cause overheating, leading to failure.
- Some of the motor parts are very heavy; please pay sufficient attention to this when carrying and installing the parts. If the weight is more than 10 kg (22.04 lbs), carrying or lifting tools should be used as much as possible.
- Both the motor and driver parts should be installed in the specified orientation.

Handling Precautions

1. Each head amp is calibrated with its corresponding motor. Do not change the specified motor-head amp combination.
   (Before using the device, check that the serial numbers on the motor and head amp parts match.)

2. Do not install the motor in reverse direction in such a way that the slider part of the motor is fixed and the base part moves.

3. Make sure to turn the power off before removing the side panel of the driver to set jumpers, etc. Touching the high voltage part inside the device is dangerous.

4. Make sure to ground the ground terminal to earth.

5. Secure the cable coming out from the slider part to the side of the slider part to prevent disconnection.

6. When attaching a load to the slider part, make sure to use load attachment screws that are shorter than the effective depth of the thread in the slider part.

7. The motor surface is magnetized; do not place things that can be affected by magnetism close to it.

8. The motor is neither dust-, drip- nor water (oil)-proof; the motor should be installed in carefully chosen environments.

9. The motor part contains glass material. Avoid applying mechanical vibration or impact.

10. After removing the mounting bracket attached to the driver, never use a screw whose length is 6 mm or longer when reattaching the bracket.

11. A coating has been applied on the base surface of the motor in order to prevent rust. When starting to use the product, wipe off the coating completely using a petroleum or chlorine solvent.

12. Never attempt to disassemble or remodel any part of this product.
    If such service is necessary, please contact us.
    We assume no responsibility for products that have been disassembled or remodeled without permission.

Table of Contents

Introduction 1

Chapter 1 Overview of the Product 1-1

1.1 About the LINEARSERV LM/UM Series 1-2
1.2 About the DrvMII Type Driver 1-3
1.3 Components of the Standard Product 1-4
1.4 Model Symbols 1-5
1.5 Name and Function of Each Part 1-6
1.6 System Configuration Diagram 1-9

Chapter 2 Installation 2-1

2.1 Installation of the Motor 2-2
2.2 Installation of the Driver 2-3

Chapter 3 Connection and Wiring 3-1

3.1 Diagram of Overall Connection 3-2
3.2 Cable Specification List 3-3
3.3 Connection between Motor and Driver 3-4
3.4 Wiring of Motor, AC Power Supply, and Ground Cable 3-5
3.5 Wiring of Encoder Cable 3-6
3.6 Wiring of Amp Cable 3-7
3.7 Wiring of Contact I/O (1) Cable 3-8
3.8 Wiring of Contact I/O (2) Cable 3-9
3.9 Wiring of Sensor Brake Terminal 3-10
3.10 Wiring of Regenerative Alarm Contact (CAN) 3-11

Chapter 4 Basic Operation (Let's Operate First) 4-1

4.1 Procedure (Flowchart) 4-2
4.2 Preoperation check 4-3
4.3 Installing the PC Utility on the PC 4-4
4.3.1 Procedure 4-4
4.3.2 Startup 4-6
4.4 Preparation 4-7
4.4.1 Selecting Communication Port 4-7
4.4.2 Selecting Channels 4-7
4.4.3 Displaying Communication Strings 4-8
4.4.4 Main Menu 4-9

4.5 Setting the Status to Servo ON 4-10
4.6 Auto-tuning 4-12
4.7 Performing Homing Operation 4-14
4.8 Performing Program Operation 4-16
4.8.1 Verifying Operation Conditions 4-16
4.8.2 Creating a Sample Program 4-17
4.8.3 Starting the Sample Program 4-19
Chapter 5 Functions 5-1
5.1 Operation Mode 5-2
5.2 Parameters and Monitors 5-3
5.2.1 General Parameters 5-3
5.2.2 Mechanical Setting Parameters 5-3
5.2.3 Monitors 5-4
5.2.4 Registration Parameters 5-4
5.2.5 Volatile Parameters 5-4
5.2.6 Points Set (Value and Option) 5-4
5.3 Operation Functions 5-5
5.3.1 Jog Operation 5-5
5.3.2 Test Operation 5-6
5.3.3 Auto-Tuning Operation 5-6
5.3.4 Homing Operation 5-7
5.3.5 Program Operation 5-10
5.3.6 Signal Search Operation 5-11
5.3.7 MDI Operation 5-11
5.3.8 Index Type A Operation 5-12
5.3.9 Index Type B Operation 5-13
5.3.10 Table Reference Operation 5-14
5.3.11 Mechanical Setting Mode 5-14
5.4 Coordinate System 5-15
5.4.1 Scaling Conversion (Conversion from Command Units to Pulse Units) 5-15
5.4.2 Conversion from Operation Units to Command Units 5-16
5.4.3 Rotational Coordinates 5-17
5.4.4 Linear Coordinates 5-17
5.4.5 Operation Coordinates Unsettled Status 5-18
5.5 Acceleration/Deceleration Function 5-19
5.5.1 Velocity Override Function 5-19
5.5.2 Trapezoidal Motion Profile 5-19
5.5.3 Cam Profile Move 5-21
5.6 Programming Language 5-24
5.6.1 NC Executable Statements 5-24
5.6.2 Parameter Statements 5-26
5.6.3 Control Parameters 5-26
5.6.4 Comment Statements 5-27
5.6.5 Explanation of Built-in Programs 5-28
5.7 M Function 5-30

5.8 Control System 5-31
5.8.1 Velocity Control Part 5-32
5.8.2 Position Control Part 5-33
5.8.3 Feed Forward 5-33
5.8.4 Servo Stiffness Parameter 5-33
5.9 Other Functions 5-34
5.9.1 Settling Wait, Position Settling Status, and Positioning Status 5-34
5.9.2 Velocity Monitor and Analog Monitor 5-35
5.9.3 Brake Signal 5-35
5.9.4 Area Signals 5-36
5.10 Special Parameter Processing 5-37
5.10.1 Internal Generation of Parameter Initial Values 5-37
5.10.2 Limiting and Checking the Maximum Velocity When Changing Scaling Data and Maximum Velocity Parameter 5-37
5.10.3 Auto Conversion and Clear Functions When Changing Scaling Data 5-38
5.10.4 Limiting the Maximum Parameter Values 5-38

Chapter 6 Control Interfaces 6-1

6.1 RS232C Interface 6-2
6.1.1 Overview 6-2
6.1.2 Connection and Setting 6-2
6.1.3 Communication Specifications 6-4
6.1.4 @ Commands 6-6
6.1.5 Parameter Commands 6-12
6.2 PLC Interface 6-13
6.2.1 Overview 6-13
6.2.2 Connection, Setting, and I/O Mapping 6-13
6.2.3 Operation 1 6-15
6.2.4 Operation 2 6-23

Chapter 7 DrvMII PC Utility 7-1

7.1 Overview 7-2
7.1.1 Overview of the Operation Menu 7-2
7.1.2 Overview of the Action Menu 7-2
7.1.3 Overview of the Data Management Menu 7-2
7.2 Installation 7-3
7.2.1 Installation under Windows 95/98/98SE/Me/NT4.0/2000 7-3
7.2.2 Starting the PC Utility 7-5
7.3 Preparation 7-6
7.3.1 Selecting a Communication Port 7-6
7.3.2 Selecting Channels 7-6
7.3.3 Displaying Communication Strings 7-7
7.3.4 Main Menu 7-8
7.4 Operation Menu 7-9
7.4.1 Terminal 7-9
7.4.2 Servo Tuning 7-14
7.4.3 Oscilloscope 7-16

7.5 Action Menu 7-19
7.5.1 Homing 7-20
7.5.2 Signal Search Move 7-22
7.5.3 Index Type A Operation 7-24
7.5.4 Index Type B Operation 7-26
7.5.5 Table Reference Operation 7-28
7.5.6 Jog Move 7-30
7.6 Data Management Menus 7-32
7.6.1 Parameter Manager 7-32
7.6.2 Program Manager 7-35
7.6.3 Index Menu 7-43
7.6.4 Point Set 7-52
7.6.5 Parts 7-53
7.6.6 I/O Set 7-59
Chapter 8 Operation Display Pendant 8-1
8.1 Overview 8-2
8.2 Features and Part Names 8-2
8.3 Switching Displays 8-3
8.4 Terminal Mode Display 8-4
8.5 Parameter Monitor Display 8-5
8.6 Parameter Settings Display 8-6
8.7 I/O Monitor Display 8-7
8.8 Special Command Display 8-8
8.9 Program Menu Display 8-9
8.9.1 Program Edit Displays 8-9
8.9.2 Copying a Program 8-13
8.9.3 Deleting a Program 8-14
Chapter 9 Maintenance and Inspection 9-1
9.1 Motor Part 9-2
9.2 Driver Part 9-2
9.3 Replacement of Battery for Memory Back-Up 9-3
9.4 Backup and Restore Operations of Driver Memory Contents 9-4
9.4.1 Backup Operation 9-4
9.4.2 Restore Operation 9-5
9.5 Motor Problems and Corrective Actions 9-6
Chapter 10 Specifications 10-1
10.1 Standard Specifications 10-2
10.2 Speed - Force Characteristics 10-4
10.3 External Dimensions (Unit: mm) 10-5

Chapter 1
Overview of the Product

1.1 About the LINEARSERV LM/UM Series

1.2 About the DrvMII Type Driver

1.3 Components of the Standard Product

1.4 Model Symbols

1.5 Name and Function of Each Part

1.6 System Configuration Diagram

1.1 About the LINEARSERV LM/UM Series

LINEARSERV LM series are linear positioning actuators adopting direct drive method.
The most significant points of the product are our proprietary interpolation-type optical linear encoder that uses a glass scale, thus achieving full-close high-precision positioning control and smooth driving. Other points that illustrate the high-precision positioning capability of the product include:
<Absolute accuracy: 10 μm or lower/1,000 mm)>
<Positioning repeatability accuracy: 0.1 to 0.5 μm>
<Low vibration: velocity ripple 1%>
<Force-to-weight ratio: 7 G>
By incorporating linear guides whose mechanical reliability has been field-proven, the product ensures high-rigidity linear guidance under all possible load conditions.

1.2 About the DrvMII Type Driver

The DrvMII type driver is a multipurpose driver with a built-in controller developed as the successor to the conventional M type driver. Not only have the functions been improved, but also the driver box volume has been made smaller, and it can support the DYNASERV rotation type motors, as well as the LINEARSERV series motors that are of the direct drive type.

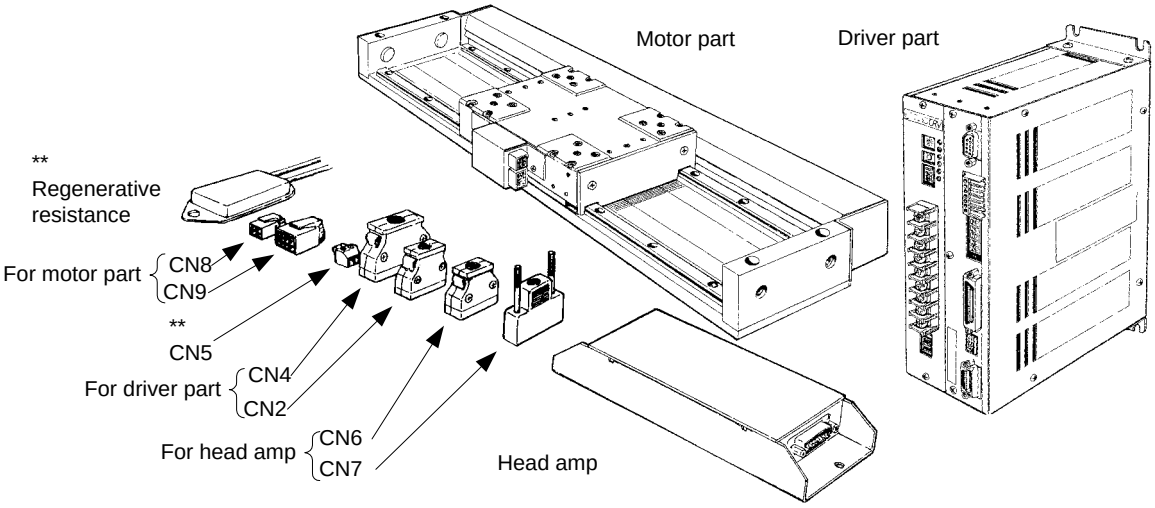
The features include the following:

(1) In addition to the contact I/O, <DeviceNet>, <CC-Link>, and <ProfiBus> can be selected as interfaces from the Fieldbus. Also, <CANopen> is under development.
(2) With the built-in controller function, the size of the driver is reduced to approximately half of the previous size (comparison within our company).
(3) The resolution is increased by a factor of four for the DM series and a factor of two for the DR and LM series.
(4) It can now support most of the models of the DYNASERV and LINEARSERV series.
(5) A sophisticated utility is now available and an oscilloscope function has been included as well.

1.3 Components of the Standard Product

The following is a list of parts included in the product. When you receive the product, you should first verify that the model and type of the product are correct and whether all the accessories are included.

Part name			Number	
Main body	Motor part		1	
	Driver part		1	
	Head amp part		1	
Standard accessory	Motor	Connector for CN8 Connector for CN9	1 1	AMP (housing), 4-pin/#172167-1 AMP (housing), 12-pin/#172170-1 AMP (socket) 170365-3 x 20 pieces, 170366-3 x 10 pieces
	Driver	Connector for CN4 *	1	Made by Honda Tsushin Kogyo (connector) PCR-S50FS (cover) PCR-LS50LA
		Connector for CN2	1	Made by Honda Tsushin Kogyo (connector) PCR-S20FS (cover) PCR-LS20LA1
		Connector for CN5 *	1	PHOENIX CONTACT MC1, 5/6-ST-3, 81
		Connector for CNA **	1	PHOENIX CONTACT MC1, 5/2-ST-5, 08
	H-amp	Connector for CN6	1	Made by Honda Tsushin Kogyo (connector) PCR-S20FS (cover) PCR-LS20LA1
		Connector for CN7	1	Made by JAE (connector) DA-15PF-N (cover) DA-C8-J10-F4-1
Regenerative resistance **			1	80 W 60 Ω (100 V) or 80 W 200 Ω (200 V)

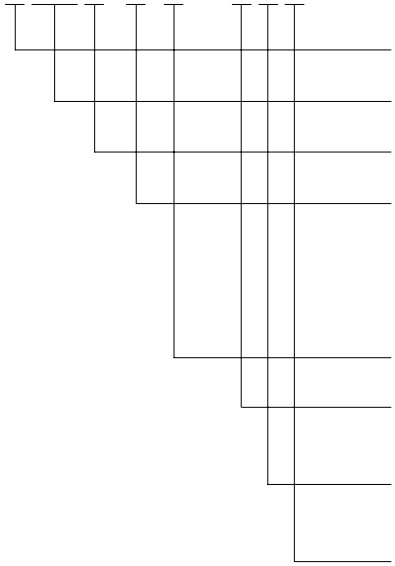


* For I/O contact type interfaces
** Supplied only for models with regenerative terminals

1.4 Model Symbols

(1) Motor unit part number

LM□□□□0□0□ - 0□□□



■ Design version (1: standard/2: high-rigidity/3: high-rigidity + high-speed/5: high-speed)

■ Static holding capacity (05: 50N/10: 100N/30: 300N/40: 400N)

■ Slider number (0: none/1 to 6)

■ Stroke length (D: 50 mm, E: 100 mm, F: 150 mm, G: 200 mm, H: 300 mm, J: 400 mm, K: 500 mm, L: 600 mm, M: 700 mm, N: 800 mm, P: 900 mm, Q: 1000 mm, R: 1100 mm, S: 1200 mm, T: 1300 mm, U: 1400 mm, V: 1500 mm, W: 1600 mm, X: 1700 mm, Y: 1800 mm)

■ Encoder scale (G: glass/M: metal)

■ Surface finish (0: standard/L: raydent finish on the base area/
B: raydent finish on the base area, black alumite finish on the aluminum area)

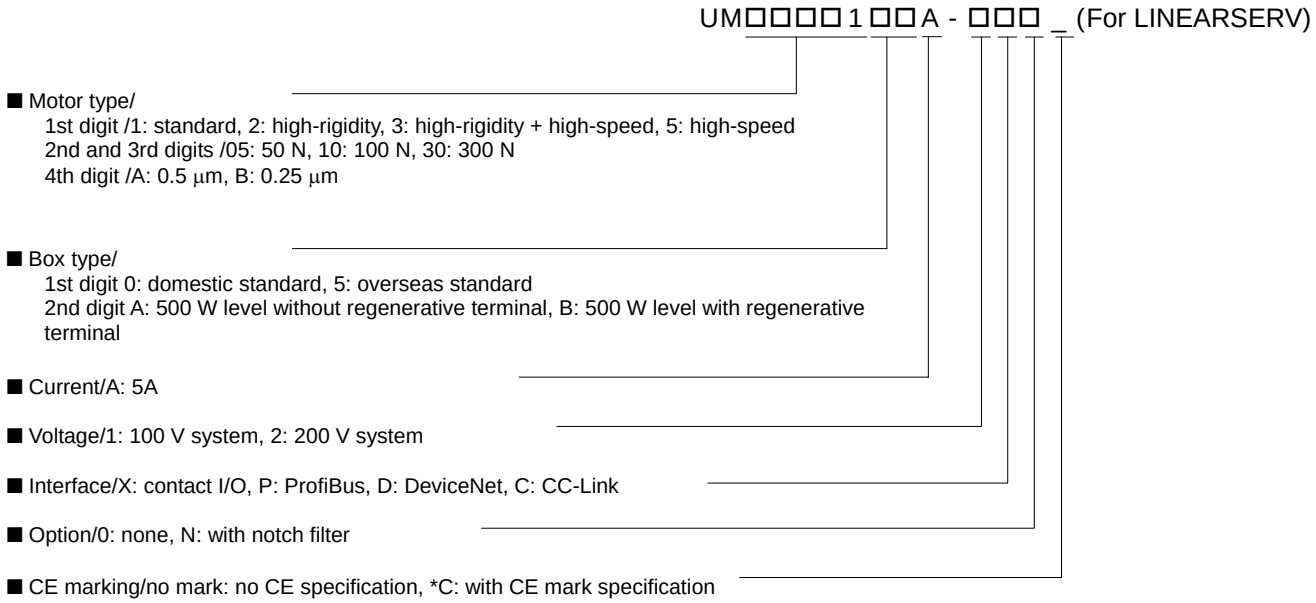
■ Sensor location (0: no sensors/Q: internal B contact/R: external B contact/
S: internal A contact/T: external A contact) *

■ Cable position (2: comes out to the right/4: comes out to the left)

* Sensor: Q/R: EE-Sx670
S/T: EE-Sx670A
Select B contact for combination with DrvMII

(2) Driver unit part number

UM□□□□1□□A - □□□ _ (For LINEARSERV)



■ Motor type/
1st digit /1: standard, 2: high-rigidity, 3: high-rigidity + high-speed, 5: high-speed
2nd and 3rd digits /05: 50 N, 10: 100 N, 30: 300 N
4th digit /A: 0.5 μm, B: 0.25 μm

■ Box type/
1st digit 0: domestic standard, 5: overseas standard
2nd digit A: 500 W level without regenerative terminal, B: 500 W level with regenerative terminal

■ Current/A: 5A

■ Voltage/1: 100 V system, 2: 200 V system

■ Interface/X: contact I/O, P: ProfiBus, D: DeviceNet, C: CC-Link

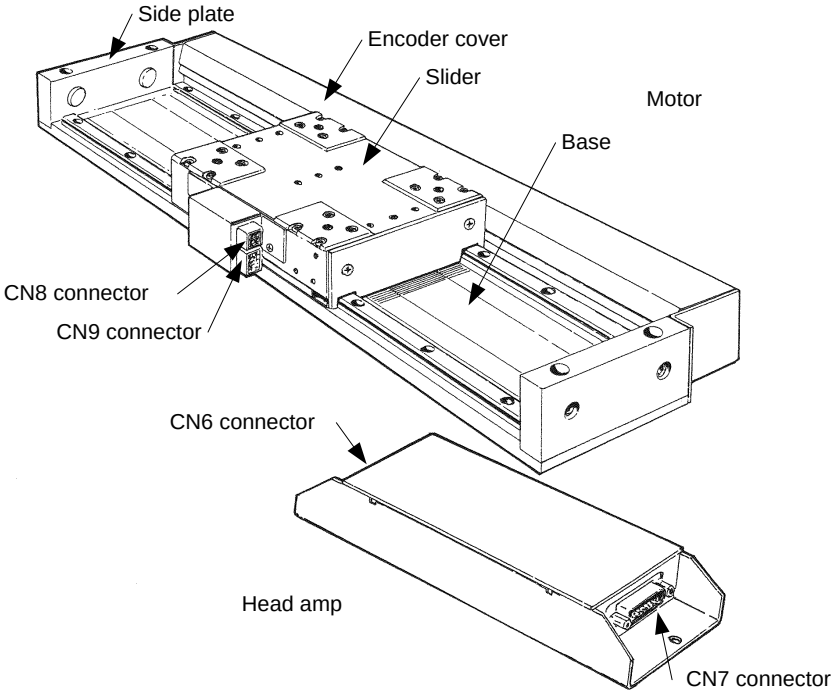
■ Option/0: none, N: with notch filter

■ CE marking/no mark: no CE specification, *C: with CE mark specification

Note: Motor and head amp units are calibrated after they have been assembled. Do not change the motor-head amp combination. As for the driver units, only identical models are compatible.

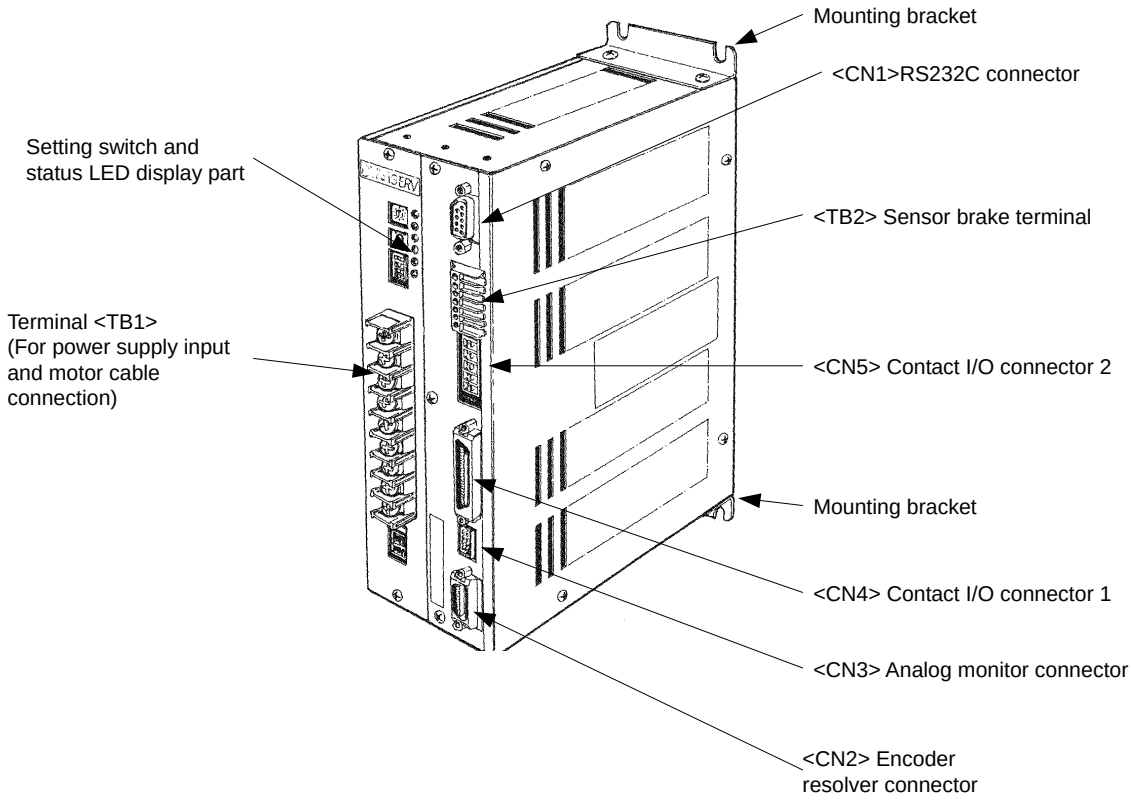
1.5 Name and Function of Each Part

(1) Motor Part



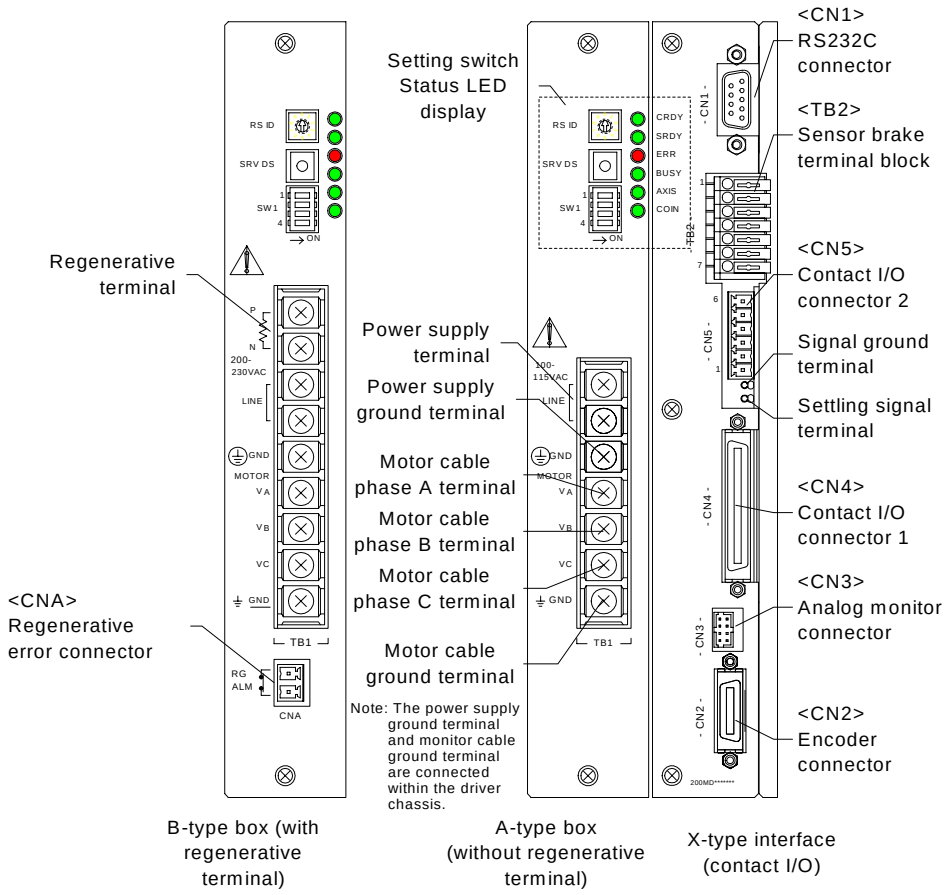
(2) Driver Part

500 W contact I/O type driver is shown.

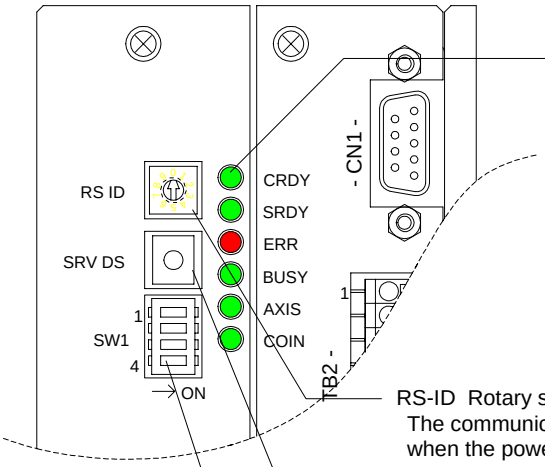


(3) Explanation of the front panel

[500 W driver]



[Details of Setting Switches and Status Display LEDs]



Status display LEDs

CRDY	CPU ready	Indicates that the driver finished its initial processing and went into its normal status.
SRDY	Servo ready	
ERR	Error status	
BUSY	Busy	Indicates that the driver is currently operating.
AXIS	Axis is operating	Indicates that the axis is currently operating (including dwell operation)
COIN	Settling status	Indicates that the axis is in its position settling status.

RS-ID Rotary switch

The communication method of the RS232C interface is set according to the status of this switch when the power is turned on.

0	Single channel communication
1 to 9	Multi-channel communication. The value corresponds to the ID of the slave station.

SRV-DS Servo disable switch

The servo is turned off for as long as this switch is pressed, regardless of the command status of the RS232C interface and PLC interface.

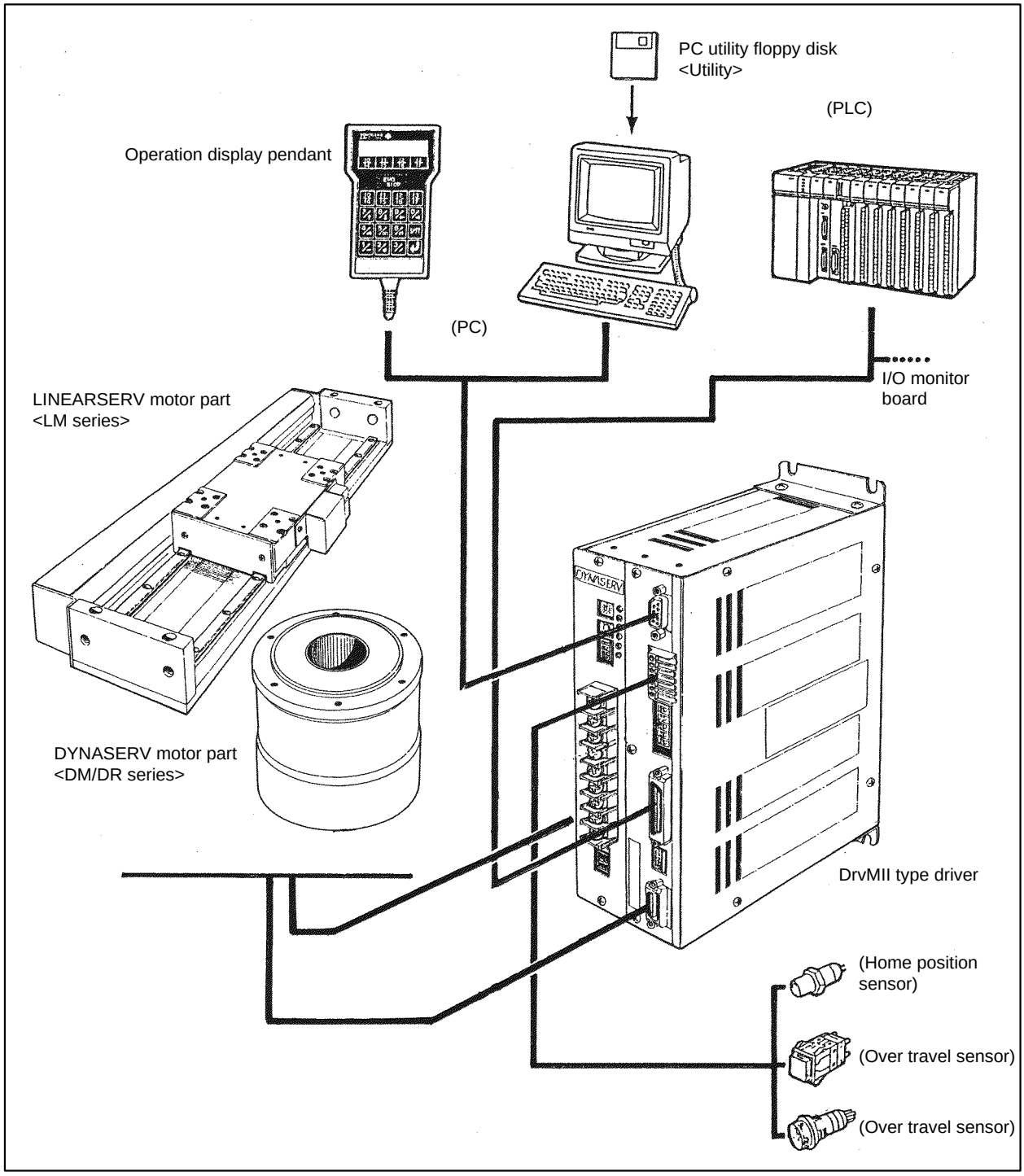
SW1 Slide switch

When the power is turned on, the operation status of the driver is determined by the status of these switches.

Normally, bit4 should be set to off. When bit4 is on, the driver is set in maintenance operation status.

bit1	Reset all	If this bit is on when the power is turned on, all driver information is reset to the default status at shipping.
bit2	Main operation mode setting (PLC main operation mode/RS232C main operation mode)	If this bit is on when the power is turned on, the driver goes into PLC main operation mode. If it is off when the power is turned on, the driver goes into RS232C main operation mode.
bit3	PLC DI emulation setting	If this bit is on when the power is turned on, any input signals from the PLC are ignored and the driver can be operated by the PC utility as if the input signals are transmitted from the PLC. Normally, this bit should be set to off.
bit4	Maintenance operation setting	If this bit is on when the power is turned on, the driver is set in maintenance operation status. Normally, this bit should be set to off.

1.6 System Configuration Diagram



Chapter 2
Installation

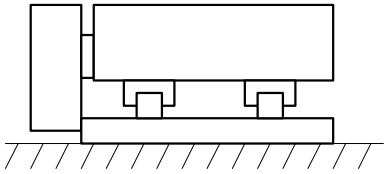
2.1 Installation of the Motor

2.2 Installation of the Driver

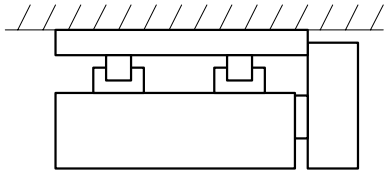
When you receive the product, you should first verify that the model and type of the product are correct, whether all the accessories are included, and that the combination of a motor and a driver is correct before you begin installation and wiring.

2.1 Installation of the Motor

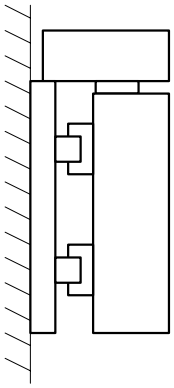
The motor part can be installed and used in either a horizontal (face-up/face-down) or a vertical position. If installed in a wrong way or position, the life of the motor may be shortened or the motor may fail. Always follow the instructions explained below.



Horizontal position (face-up)



Horizontal position (face-down)



Vertical position

(1) Installation Position

The motor part is designed based on the assumption that it is used indoors. Therefore, choose the location of installation so that it satisfies the following conditions:

- It should be indoors and not in a place where it can be exposed to corrosive and/or volatile gases.
- The ambient air temperature should be from 0 to 45 °C.
- There should not be too much dust or particles, the ventilation should be good, and the humidity should be low.

Note: The LINEARSERV is not drip- or water (oil)-proof. If it is used in such an environment, a proper drip- or water (oil)-proof cover should be applied.

(2) Mechanical Installation

- The surface flatness to which a load is attached must be 0.01 mm or less.
- Since there is only 0.1 mm clearance between the slider part and the base part, be careful not to leave metal powder or dust on the base surface.
- A coating has been applied on the base installation surface of the motor in order to prevent rust. When starting to use the product, wipe off the coating completely.
  If any of the coating remains, it may affect the mechanical precision.
  Use petroleum or chlorine solvent to remove the coating.
- The clamping torque for the screws used to install the base part and to attach a load to the slider part should be equal to or less than the value indicated below. In addition, make sure to apply a screw lock (Loctite 601 or equivalent product) when tightening the screws.
  Slider part: 130 kgcm or less, base part: 40 kgcm or less
- Precision of the motor part will be affected by the flatness of the base part installation surface. Therefore, it is very important to improve the flatness of the installation surface.

2.2 Installation of the Driver

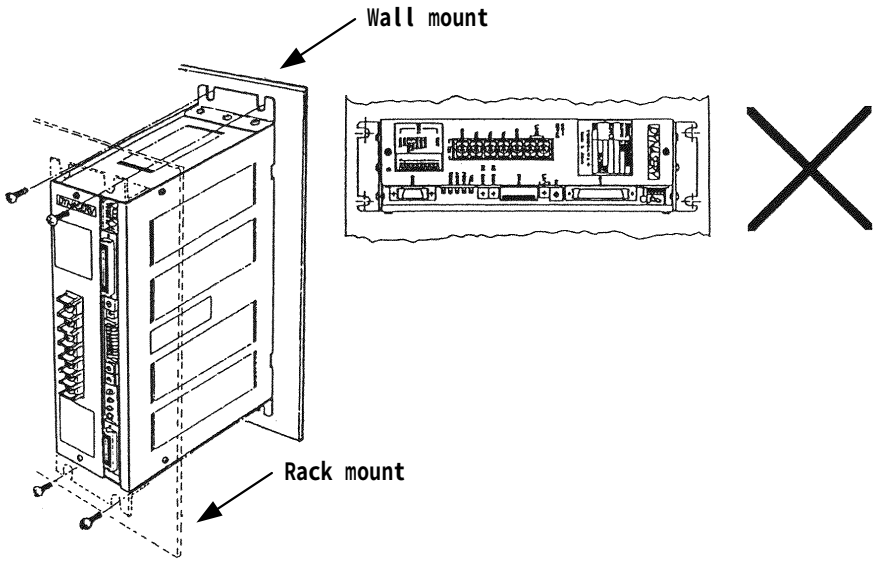
The standard installation method for the driver is either to mount it on a rack or a wall.

(1) Installation Position

- If there is a heating source near by, the temperature should be prevented from increasing by installing a shielding cover, etc.; the temperature around the driver should not exceed 50 °C.
- If there is a source of vibration near by, the driver should be installed via a vibration absorption material.
- In addition to the above, it should be avoided to install the driver in surroundings that are high in temperature and humidity, filled with dust, metal powder, corrosive gas, etc.

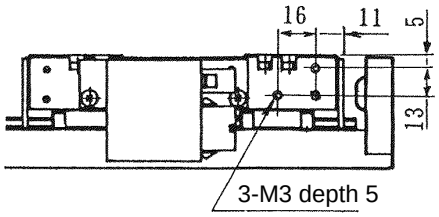
(2) Installation Method

- The standard installation method is to mount the driver on a rack, aligning the top and bottom with the front panel in the front. You can also install the driver so that the panel will face upward. However, you may not install the unit to have the panel into a sideways position. (See the figure below.)
- The driver box employs a natural air cooling system. Make sure to secure spaces of at least 25 mm above and below and at least 10 mm right and left for ventilation. (See the figure below.) The power loss of the driver is approximately 30 W.
- As for the screws used to secure the brackets to the driver box, do not use a screw whose length is 6 mm or longer since it may damage the boards inside the unit.



(3) Fixing the slider unit cables

Power and encoder cables are coming out of the two connector areas of the slider part. Secure these cables to the side of the slider during installation. There are two installation holes on the slider's side surface for this purpose.
(See the figure to the right for the <LM105> type with cables coming out to the right. See external dimensions for other models.)



Chapter 3
Connection and Wiring

3.1 Diagram of Overall Connection

3.2 Cable Specification List

3.3 Connection between Motor and Driver

3.4 Wiring of Motor, AC Power Supply, and Ground Cable

3.5 Wiring of Encoder Cable

3.6 Wiring of Amp Cable

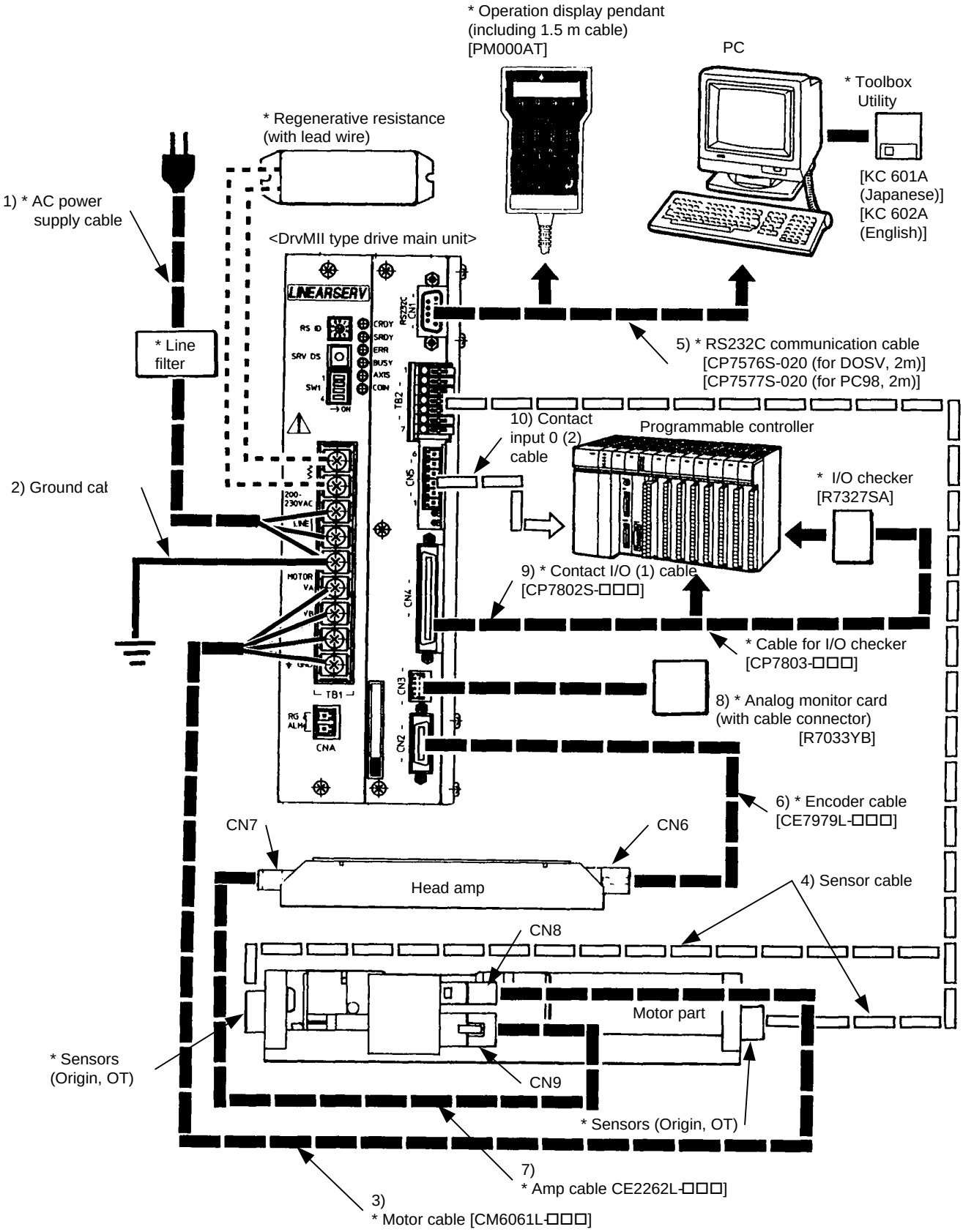
3.7 Wiring of Contact I/O (1) Cable

3.8 Wiring of Contact I/O (2) Cable

3.9 Wiring of Sensor Brake Terminal

3.10 Wiring of Regenerative Alarm Contact (CAN)

3.1 Diagram of Overall Connection



* Optional parts (see separate wiring section for motor and encoder cables.)

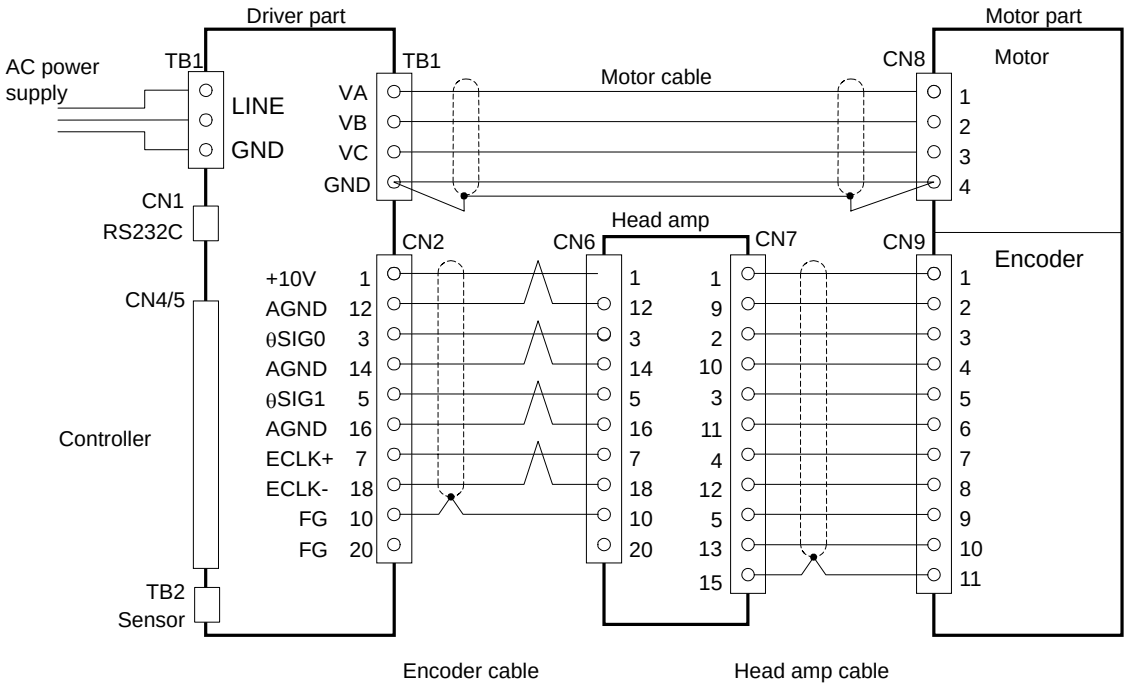
3.2 Cable Specification List

	Cable name	Electric cable size	Driver	Current (A)
1)	AC power supply cable	0.5 mm^2 or more, 30 m or less in length	TB1	5
2)	Ground cable (power supply)	0.5 mm^2 or more	TB1	
3)	Motor cable	0.5 to 0.89 mm^2, 15 m or less in length	TB1	5
4)	Sensor brake cable	0.3 to 0.75 mm^2	TB2	
5)	RS232C communication cable	Dedicated cable is required.	CN1	
6)	Encoder cable	0.2 mm^2 twisted pair, batch shielded cable, outer diameter ϕ 14 mm or less, 10 m or less in length	CN2	Maximum 100 mA DC
7)	Head amp cable	0.3 mm^2, batch shielded cable, outer diameter ϕ 9 mm or less, 5 m or less in length		
8)	Analog monitor card	Dedicated cable [R7033YB] is required.	CN3	
9)	Contact I/O (1) cable	0.2 to 0.5 mm^2, batch shielded cable, outer diameter ϕ 9 mm or less, 3 m or less in length	CN4	
10)	Contact I/O (2) cable	0.2 to 1.5 mm^2	CN5	

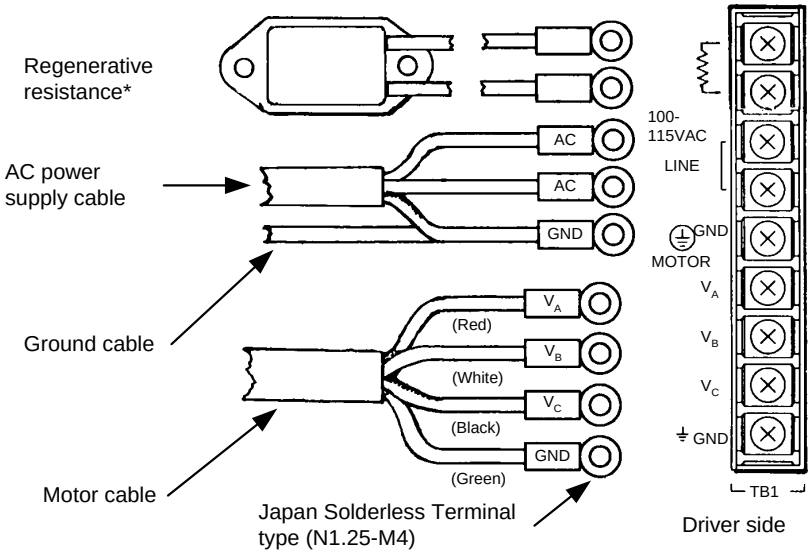
* 20A for the A (10”) and E (8”) types for both the DM and DR series
15A for the B (6”) type, and 10A for the DM1004B/1004C motors

3.3 Connection between Motor and Driver

Note: Shielding should be applied to each wire.



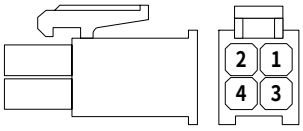
3.4 Wiring of Motor, AC Power Supply, and Ground Cable

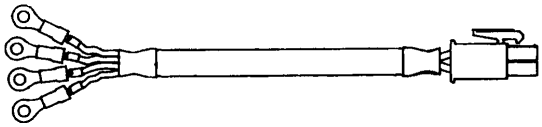


<CN8> Motor-side terminal for the motor cable

AMP
(Housing: 172167-1 or 172338-1
Contacter: 170366-3)
Manual tool: 755331-1

Pin #	Signal name
1	V_A
2	V_B
3	V_C
4	GND



Cable	Specification
AC power supply cable	■ 0.5 mm^2 or more, 30 m or less in length ■ Clamping torque of terminal: 12 [kgf-cm^2] (1.18 [N·m]) (terminal screw: M4 x 0.7) ■ Power supply filter, recommended part: Tokin Corporation #LF-200 series
Motor cable	■ 0.5 mm^2 or more, 15 m or less in length ■ Since this is a movable part, use robot cables that have higher flexibility. ■ Optional cable: CE6061L-□□□ 
Ground cable	■ 0.5 mm^2 or more (use as thick cable as possible) ■ Third grade ground (ground resistance 100Ω or less)
Regenerative resistance*	For 100 V: 80 W 60Ω
	For 200 V: 80 W 200Ω

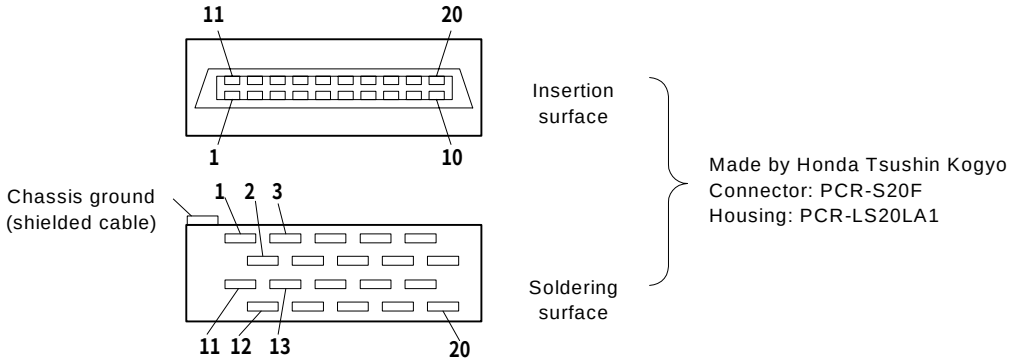
* Only for models with regenerative resistance

3.5 Wiring of Encoder Cable

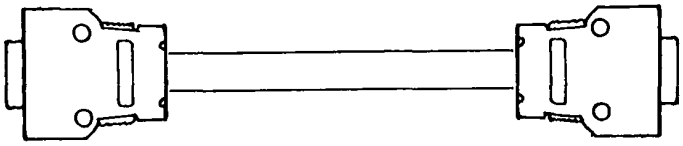
<CN2>/<CN6>

Pin #	Signal name	Pin #	Signal name
1	+ 10 V	11	-
2	-	12	GND
3	SIG 0	13	-
4	-	14	GND
5	SIG 1	15	-
6	-	16	GND
7	ECLK+	17	-
8	-	18	ECLK-
9	-	19	-
10	FG	20	FG

Terminal for <CN2>/<CN6>



Electric wire specification	■ 0.2 mm^2 multiple-core twisted pair batch shielded cable, 10 m or less in length ■ Optional cable: CE7979L-□□□



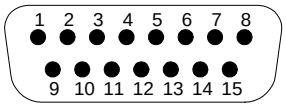
3.6 Wiring of Amp Cable

Terminal for <CN7>

JAE
Connector: DA-15PF-N
Housing: DA-C8-J10-F4-1

Pin #	Signal name
1	PDA 1
2	PDA 3
3	PDA 5
4	PDA 7
5	+ 10 V
6	-
7	-
8	-
9	PDA 2
10	PDA 4

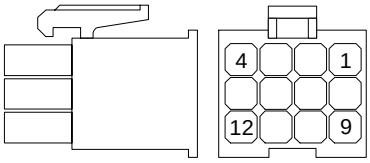
Pin #	Signal name
11	PDA 6
12	PDA 8
13	AGND
14	-
15	FG



Pin layout as seen from the insertion surface

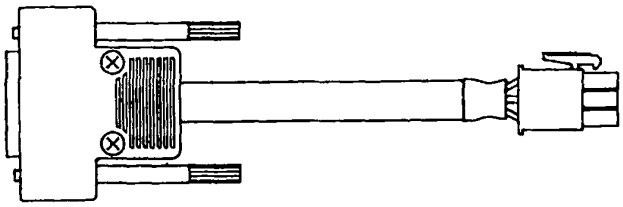
Terminal for <CN9>

AMP
Contacter: 170365-3
Housing: 172170-1 or 172341-1
(Manual tool: 755330-1)



Pin #	Signal name
1	PDA 1
2	PDA 2
3	PDA 3
4	PDA 4
5	PDA 5
6	PDA 6
7	PDA 7
8	PDA 8
9	+ 10 V
10	AGND
11	FG
12	-

Electric wire specification	■ 0.3 mm^2, multiple-core batch shielded cable, 5 m or less in length ■ Since this is a movable part, use robot cables that have higher flexibility. ■ Optional cable: CE2262L-□□□



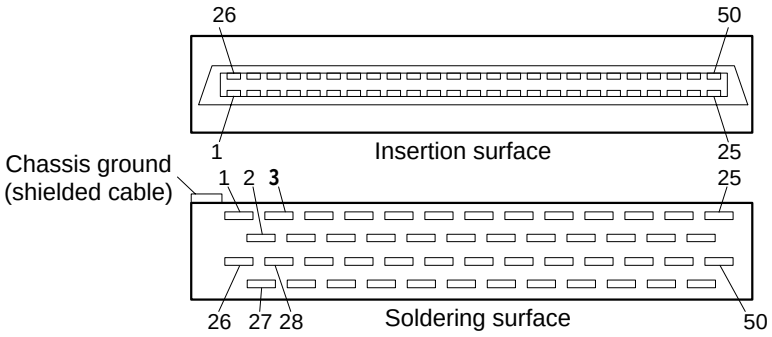
3.7 Wiring of Contact I/O (1) Cable

<CN4> terminal

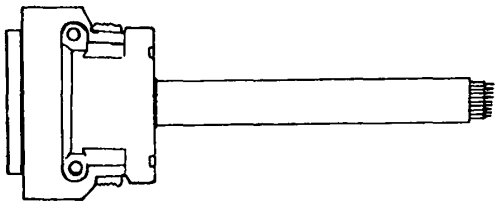
Pin #	Signal name	Pin #	Signal name	Pin #	Signal name	Pin #	Signal name	Pin #	Signal name
1	COMP1	11	IN_I_CODE.1	21	IN_ ERR_RESET	31	IN_ROTDIR_ STR_OPT.1	41	(RESERVE)
2	IN_EMG	12	IN_I_CODE.2	22	IN_M_ANS	32	IN_ABS_ STR_OPT	42	OUT_0_ CODE.0
3	IN_SERVO	13	IN_I_CODE.3	23	IN_ERRCODE _REQ	33	(RESERVE)	43	OUT_0_ CODE.1
4	IN_MODE_ START	14	IN_I_CODE.4	24	(RESERVE)	34	OUT_ CPURDY	44	OUT_0_ CODE.2
5	IN_MODE_ STOP	15	IN_I_CODE.5	25	IN_POS_INH	35	OUT_ SRDY	45	OUT_0_ CODE.3
6	IN_MODE.0	16	IN_I_CODE.6	26	IN_JOG_UP	36	OUT_ MODE_EXE	46	OUT_0_ CODE.4
7	IN_MODE.1	17	IN_I_CODE.7	27	IN_JOG_DN	37	OUT_ ERR	47	OUT_0_ CODE.5
8	IN_MODE.2	18	IN_PRG_ REWIND	28	IN_OVERRIDE _SEL	38	OUT_ ALARM	48	OUT_0_ CODE.6
9	IN_MODE.3	19	IN_ INTERLOCK	29	IN_SIGN_ INDEX	39	OUT_ M_EN	49	OUT_0_ CODE.7
10	IN_I_CODE.0	20	IN_ABORT	30	IN_ROTDIR_ STR_OPT.0	40	OUT_ERRCO DE OUT	50	COMN1

Terminal for <CN4>

Made by Honda Tsushin Kogyo
Connector : PCR-S50FS
Housing : PCR-LS50LA1



Electric wire specification	■ 0.2 to 0.5 mm^2 or more, multiple-core batch shielded cable, 3 m or less in length ■ Optional cable: CP7802S-□□□



3.8 Wiring of Contact I/O (2) Cable

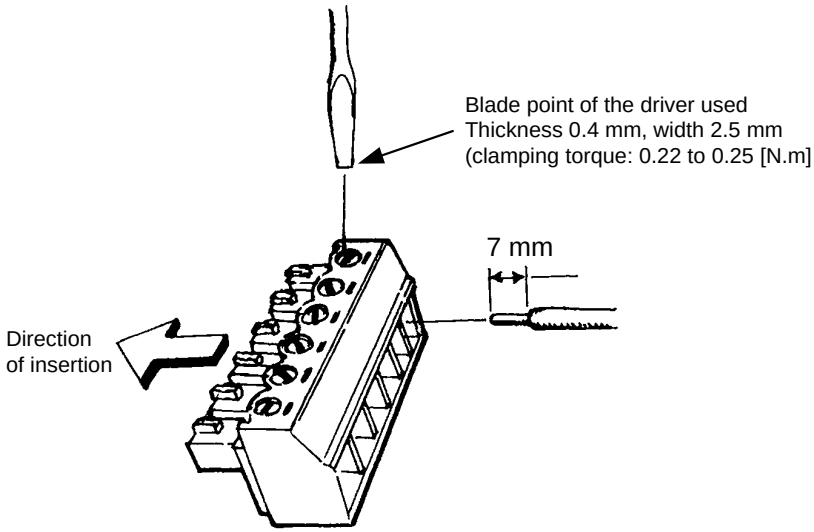
Pin #	Signal name
6	COMN2
5	OUT_AREA
4	OUT_AREA0
3	(reserve)
2	OUT_COIN
1	COMP2

Electric wire specification	■ 0.2 to 1.5 mm^2 or more, multiple-core batch shielded cable ■ Do not solder the core wire (twisted wire). It may cause a contact problem.

See the panel surface of the driver for pin numbers.

<CN5>

Made by Phoenix Contact
(Plug: MC1, 5/6-ST-3, 81)



3.9 Wiring of Sensor Brake Terminal

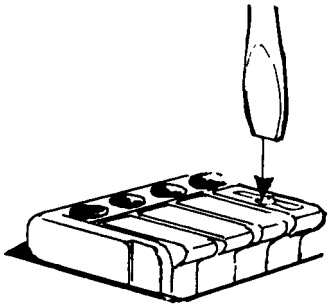
Pin #	Signal name
7	XBRKN
6	XBRKP
5	(NC)
4	XOTU
3	XOTD
2	XORG
1	COMP0

Electric wire specification	■ 0.3 to 0.75 mm^2, electric wire coating with 10 mm of the core exposed at the tip ■ If a twisted wire is used, the diameter of the strand should be ϕ 0.18 or larger.

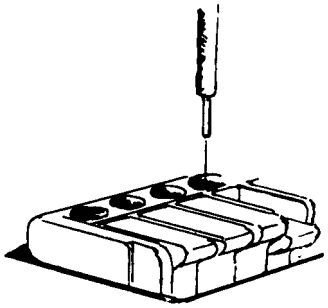
See the panel surface of the driver for the pin numbers.

<TB2> Made by Sato Parts (ML1900H)

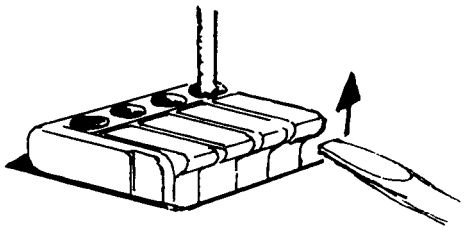
1) Push down the lever with a screwdriver.



2) Insert the wire deeply.



3) Push up the lever (until you hear the click)

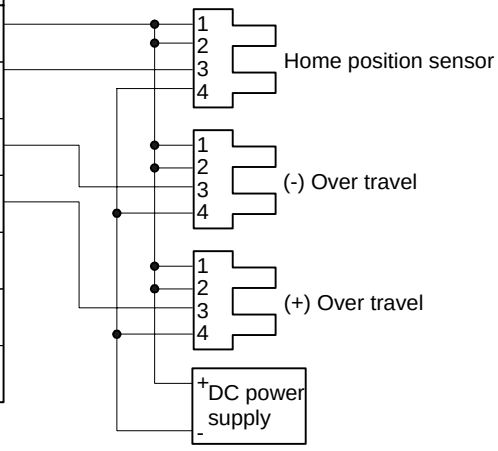


Example of sensor connection (sensor: EE-SX670 manufactured by Omron)

The recommended sensor logic is B contact.

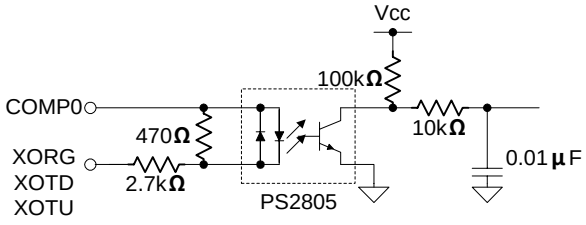
Set the sensor to OFF when the light is shielded. The sensor described above will be set to OFF when the light is shielded by the following result.

Signal name	Pin #
COMP0	1
XORG	2
XOTD	3
XOTU	4
(NC)	5
XBRKP	6
XBRKN	7



[Electrical specifications]

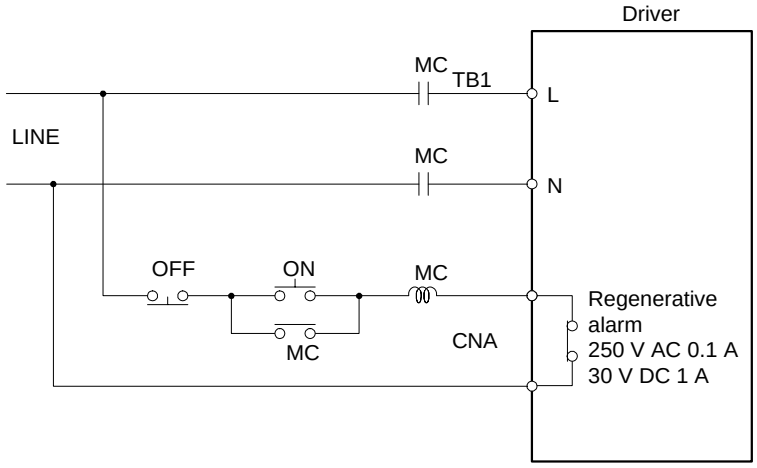
Input specifications	
Rated voltage	12～24VDC （±10%）
Rated input current	4.1 mA/point (at 12 VDC) 8.5 mA/point (at 24 VDC)
Input impedance	3.0kΩ
Operating voltage (relative to COMP*)	At OFF: 3.0 VDC or less At ON: 9.0 VDC or more
Allowable leakage current	OFF is guaranteed at 1.0 mA or less.



3.10 Wiring of Regenerative Alarm Contact <CNA>

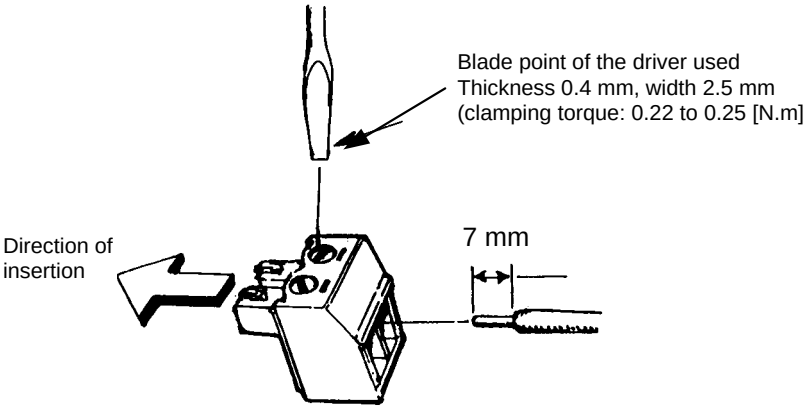
This driver (with regenerative terminal) is equipped with a regenerative circuit failure detection circuit. When connecting the regenerative circuit, build a sequence circuit as shown in the figure below in order to prevent burnout incidents.

Note: Build a sequence circuit so that it will turn off the power supply at alarm operation.



<CNA>

Made by Phoenix Contact
(Plug: MC1, 5/6-ST-3, 81)



Chapter 4 Basic Operation (Let’s Operate First)

This chapter describes an "Example of Stand-Alone Motor Operation," which should be used as the first step in understanding the "motor/driver/PC utility." The information is provided progressively, focusing on motor tuning, homing operation, program creation (moving to a position of 50 mm, moving with a velocity of 50 mm/sec, reciprocating operation), and startup method.
Make sure to perform the operations described in this chapter as a preliminary step before commencing device production.

4.1 Procedure (Flowchart)

4.2 Preoperation Check

4.3 Installing the PC Utility on the PC
- 4.3.1 Procedure
- 4.3.2 Startup

4.4 Preparation
- 4.4.1 Selecting Communication Port
- 4.4.2 Selecting Channels
- 4.4.3 Displaying Communication Strings
- 4.4.4 Main Menu

4.5 Setting the Status to Servo ON

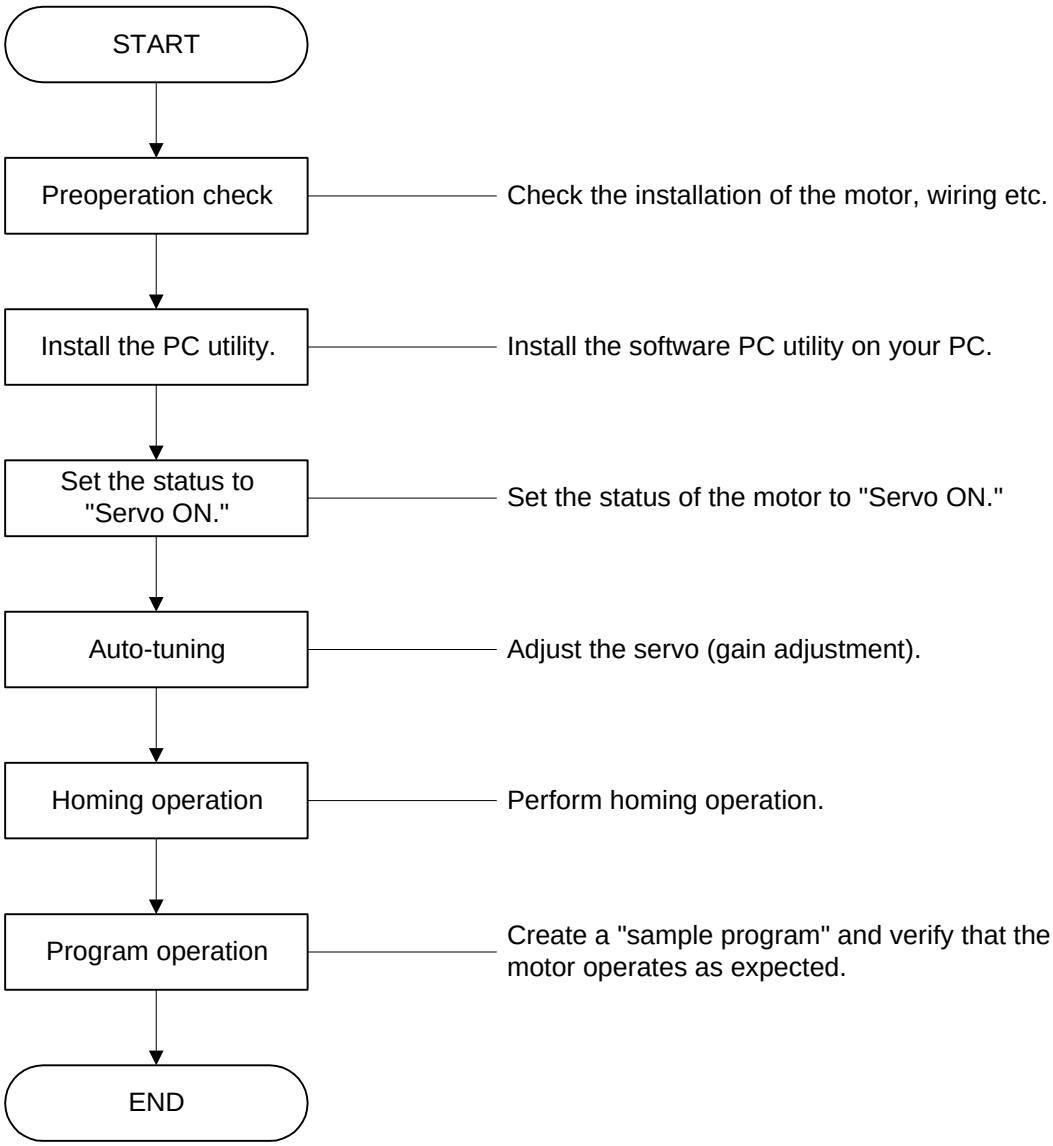
4.6 Auto-tuning

4.7 Performing Homing Operation

4.8 Performing Program Operation
- 4.8.1 Verifying Operation Conditions
- 4.8.2 Creating a Sample Program
- 4.8.3 Starting the Sample Program

4.1 Procedure (Flowchart)

In this section, we will operate the motor according to the procedure below.

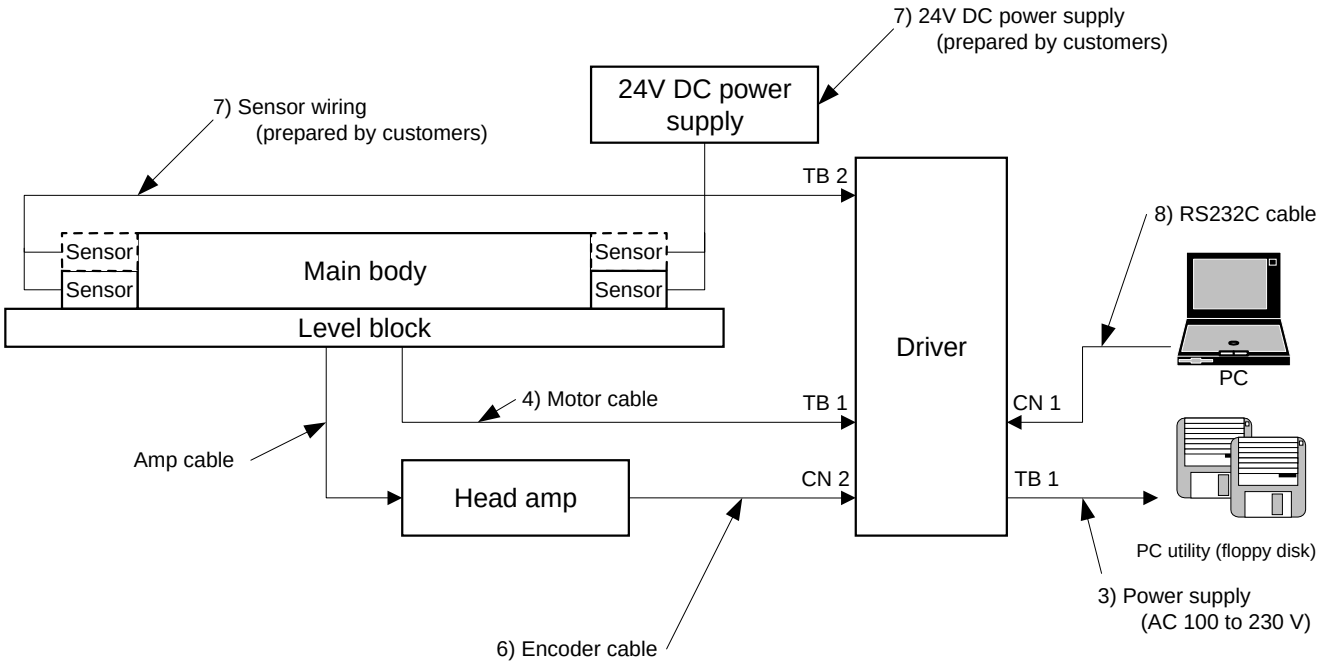


4.2 Preoperation Check

(1) Items to prepare

- Motor unit/driver/head amp
- PC utility (floppy disk)
- Level block for fixing the motor
- PC (with Windows 95/98/98SE/Me/NT4.0/2000 installed)
- Various cables

(2) Installation and Wiring



(3) Items to be checked

		Check
1)	Is the main body fixed on the level block?	☐
2)	Is the slider not interfering with peripherals?	☐
3)	Is the power supply line wired properly? (LINE, GND)	☐
4)	Is the motor cable wired properly? (VA, VB, VC. GND)	☐
5)	Is the amp cable wired properly?	☐
6)	Is the encoder cable wired properly?	☐
7)	Is the sensor wired properly? (Home and OT sensor)	☐
8)	Is the RS232C cable wired properly?	☐
9)	Is the operation mode set to RS232C? (Is bit 2 of SW1 of the driver set to OFF?)	☐

4.3 Installing the PC Utility on the PC

4.3.1 Procedure

Installation under Windows 95/98/98SE/Me/NT4.0/2000

The M2 PC utility (hereinafter referred to as the "PC utility") runs on Windows 95/98/98SE/Me/NT4.0/2000. It can be installed via "Add/Remove Programs" under the "Control Panel" in Windows. If an older version of the PC utility is present, delete it first and then install the new version.
Display the "Properties of Adding/Removing Programs" dialog box and click "*Set Up (I)*." Then proceed according to the instructions displayed on the screen. The PC utility setup program starts up.
Proceed with the setup according to the instructions on the screen. A dialog box for determining the directory in which to install the PC utility appears (see Figure 4.3.1).

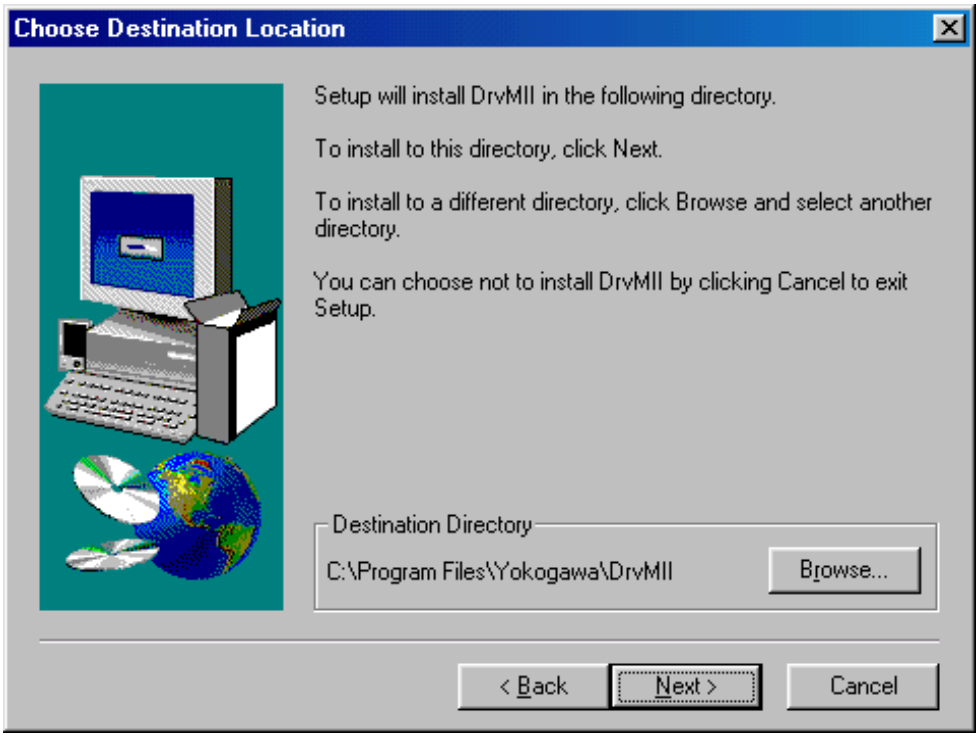


Figure 4.3.1 "Choose Destination Location" dialog box

Click "*Browse* " to display the "Select Directory" dialog box and select the desired drive and directory. Click "*Next* " to display "Select Program Folder" (see Figure 4.3.2).

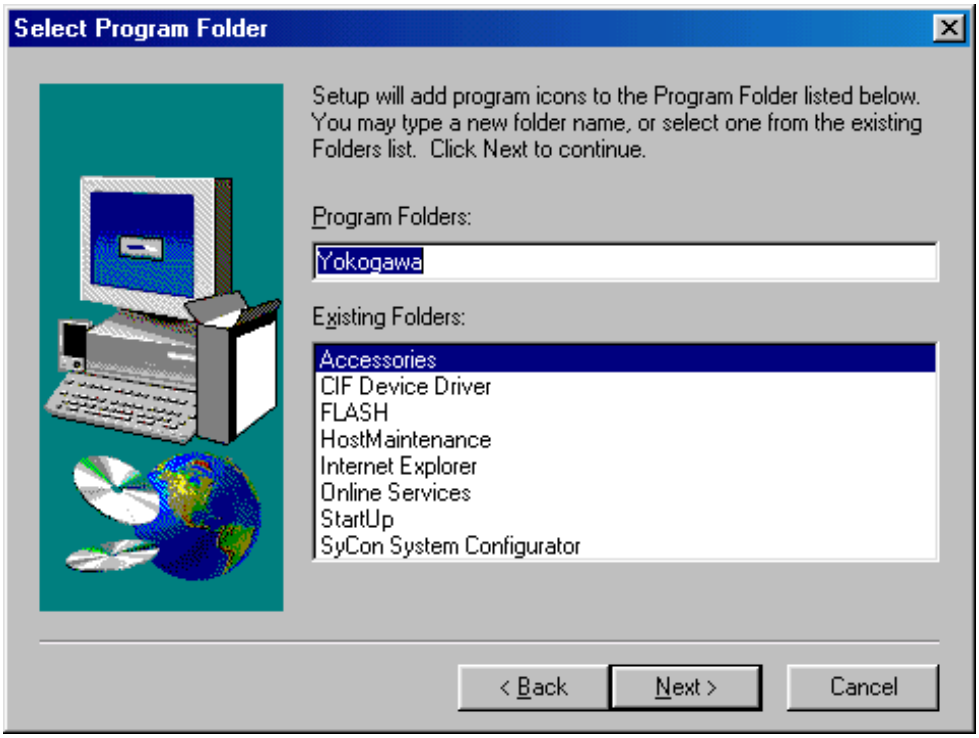


Figure 4.3.2 "Select Program Folder" dialog box

Select a program folder and click "*Next* " The installation begins. Follow the instructions on the screen and change disks. When the setup is completed, the "Setup Complete" dialog box appear (see Figure 4.3.3).

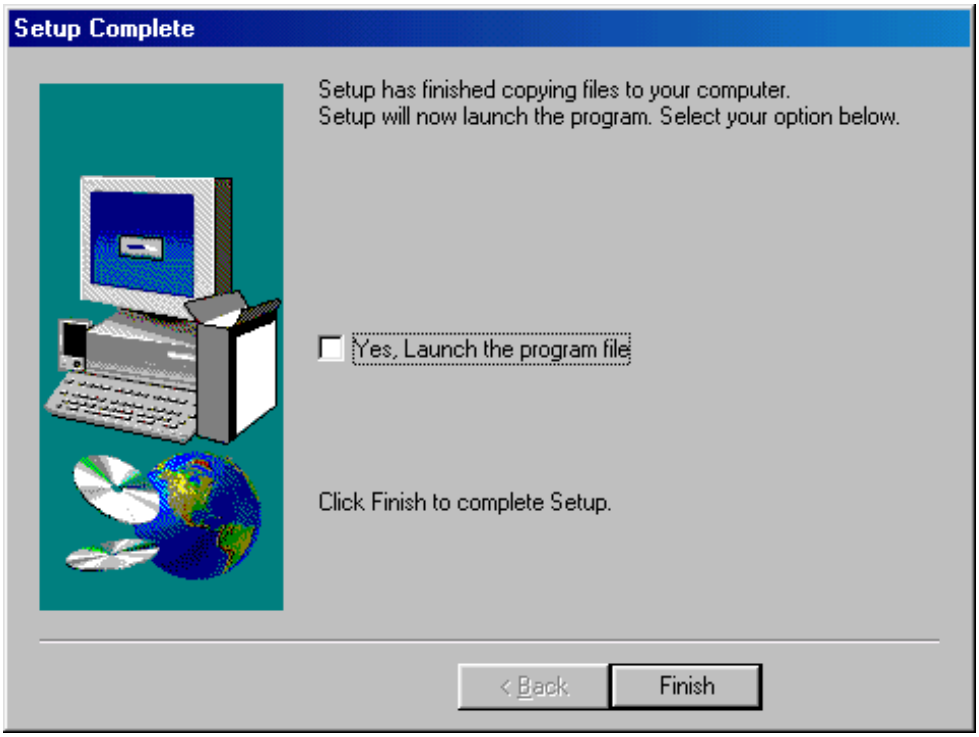


Figure 4.3.3 "Setup Complete" dialog box

To start the program, select "Launch the program file" and click "*Finish*." If you do not want to start the program, just click "*Finish*." If you are prompted to restart the computer, simply follow the message and restart it.

Note: Remove the floppy disk before restarting the computer.

4.3.2 Startup

1) To start the PC utility, click “Start,” “Program” menu, “DrvMII,” and then "DrvMII."

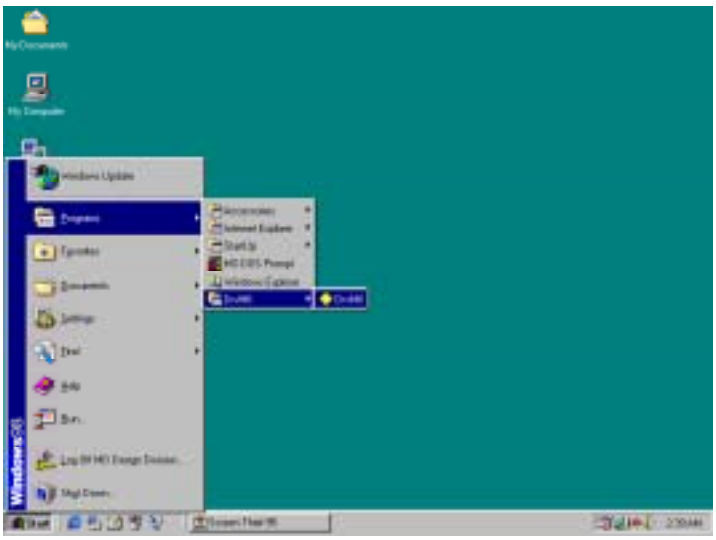


Figure 4.3.4 “Startup”

2) An “Opening” dialog box is displayed for several seconds and then the PC utility starts up.

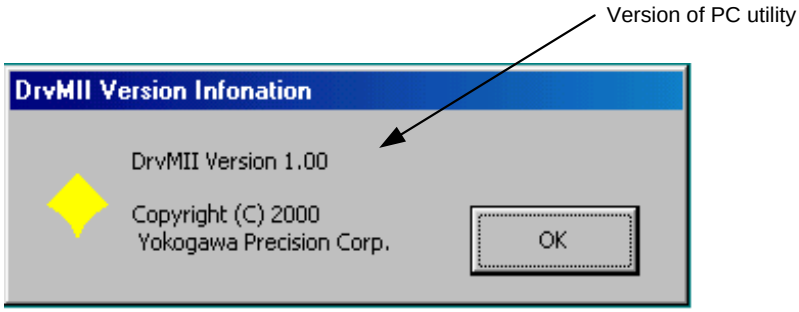


Figure 4.3.5 “Opening” dialog box

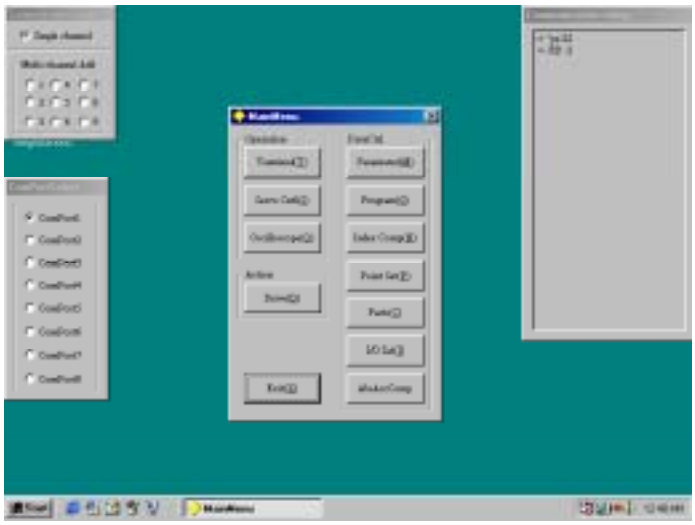


Figure 4.3.6 "After starting up the PC utility"

4.4 Preparation

Connect the serial port of the PC with the serial port of the driver with a dedicated cable. (Do not use commercial off-the-shelf cables. There is a terminal for which 5 V is output from the driver for connecting with the teaching box.)

4.4.1 Selecting Communication Port

When you start the PC utility, the “ComPortSelect” dialog box appears in the left side of the screen (see Figure 4.4.1). Change the setting according to the communication port of the connected PC.

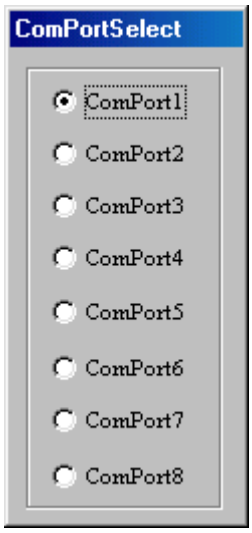


Figure 4.4.1 “ComPortSelect”

Note: Settings made in the “ComPortSelect” dialog box are stored in a file. It is not necessary to make settings from the next time you start the PC utility. Change the setting as necessary.

4.4.2 Selecting Channels

When you start the PC utility, the “Communication mode” dialog box appears in the upper left corner of the screen (see Figure 4.4.2). If you are using one driver, select a single channel, and if you are using multiple drivers, select multi-channel addresses. (See Chapter 6 for how to make setting on the driver side.)

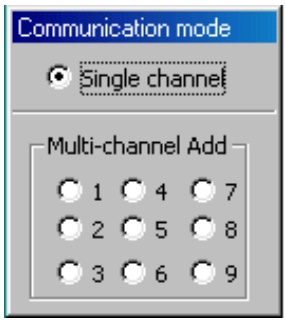


Figure 4.4.2 “Communication mode”

Note: The settings made in the “Communication mode” dialog box are not stored. When the PC utility is started up, a single channel is always set.

4.4.3 Displaying Communication Strings

When you start the PC utility, the “Communication string” dialog box appears in the upper right corner of the screen. (See Figure 4.4.3.) Any strings that the PC utility sends to the driver as well as any strings received from the driver are displayed regardless of the menu.

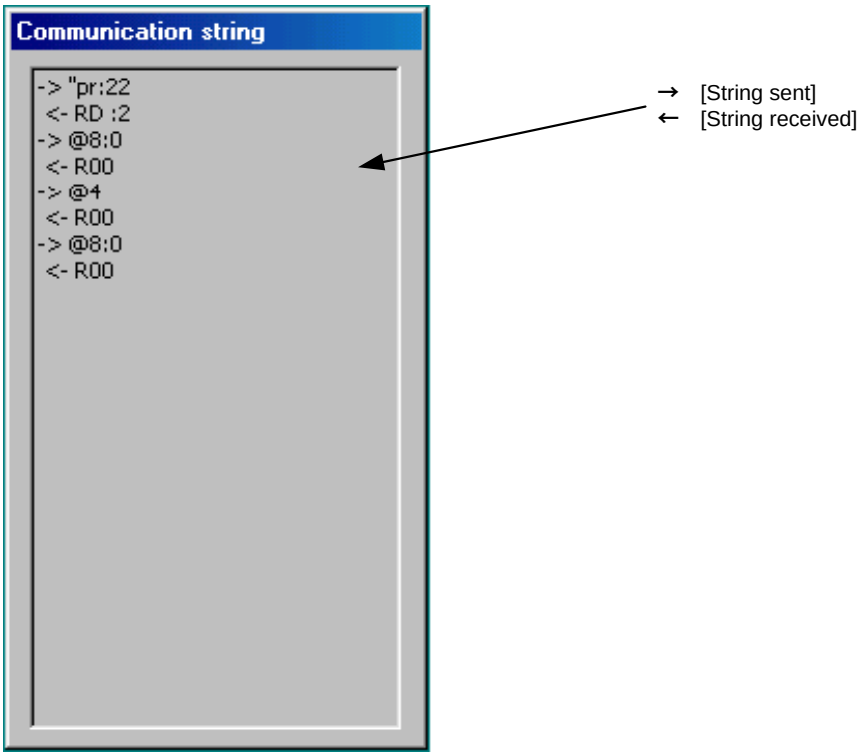


Figure 4.4.3 “Communication string”

4.4.4 Main Menu

When you start the PC utility, the “MainMenu” dialog box appears (see Figure 4.4.4). See the following chapters for how to start the actual operation.

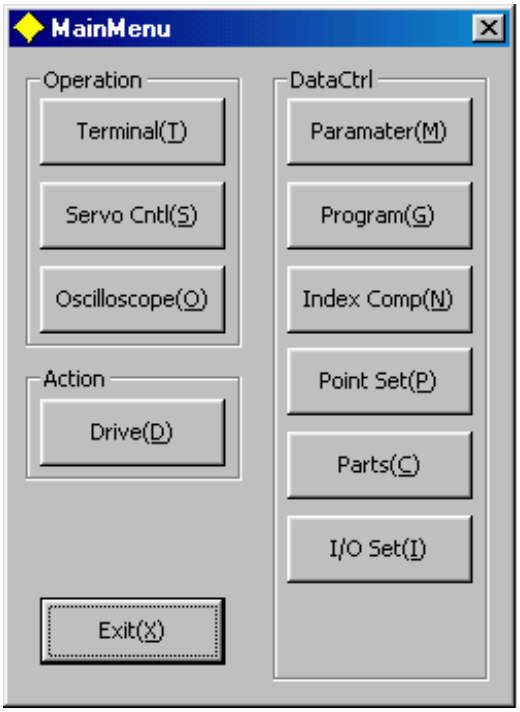
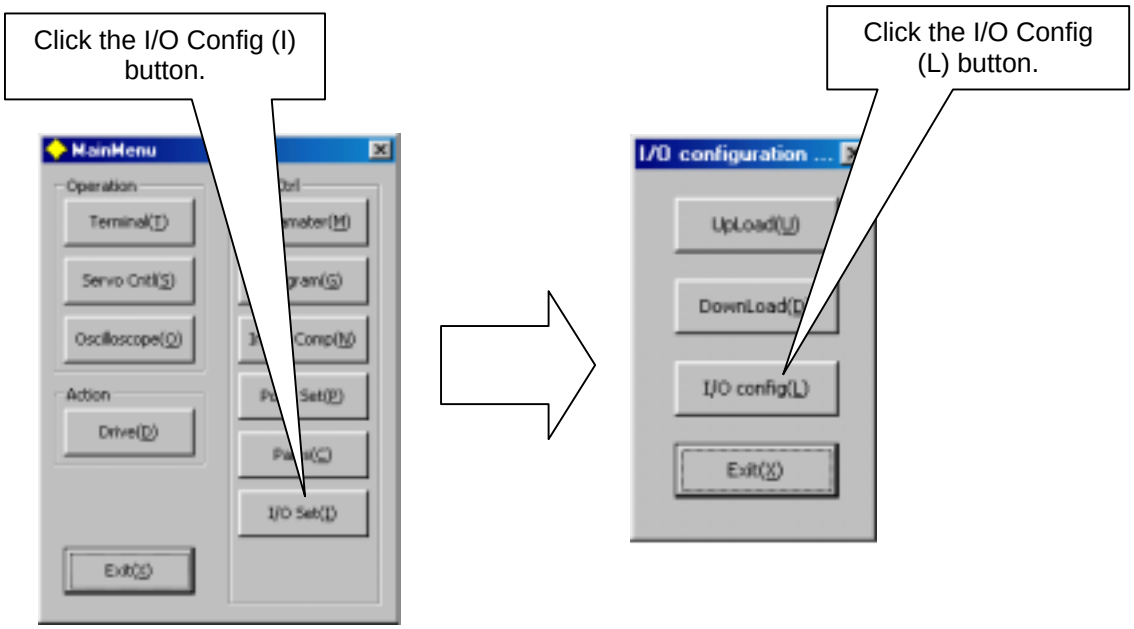


Figure 4.4.4 “MainMenu” dialog box

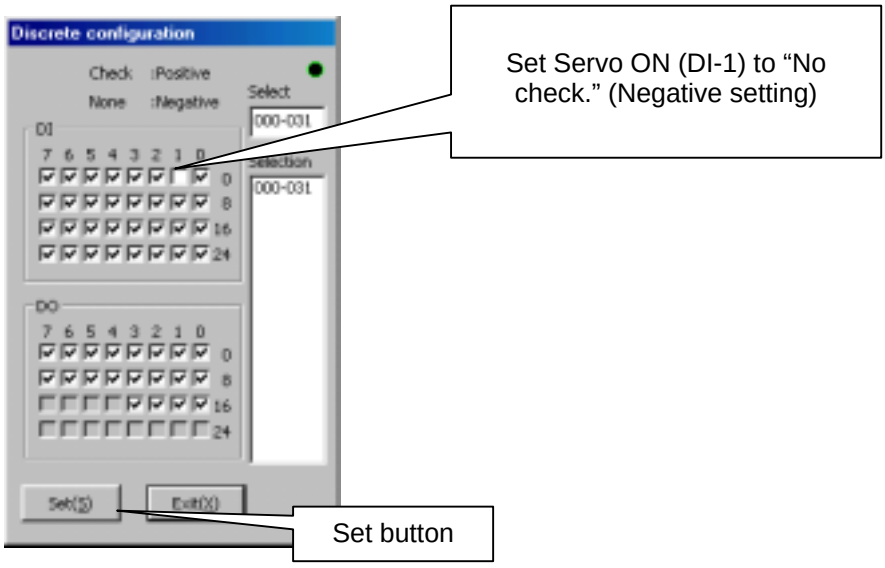
4.5 Setting the Status to Servo ON

The driver can be put into Servo On status through the following operation.

(1) Click the “I/O Config (I)” button in the “MainMenu” and then the “I/O config (L)” button.

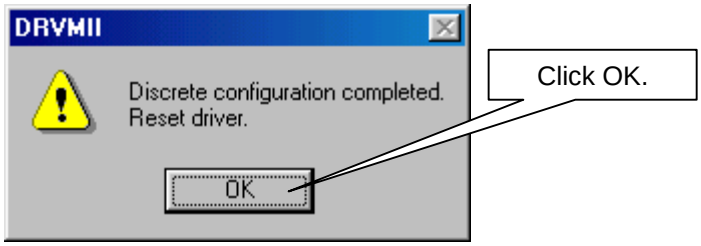


(2) Enable Servo ON. Click the checkmark of “1” under DI and then click the Set (S) button.



△ **Caution**	Make sure to click the “Set” button after finishing the setting (the status will become Servo ON). Verify that the “S-RDY” LED on the front panel is turned on.

(3) Reset the driver according to the message in the dialog box.

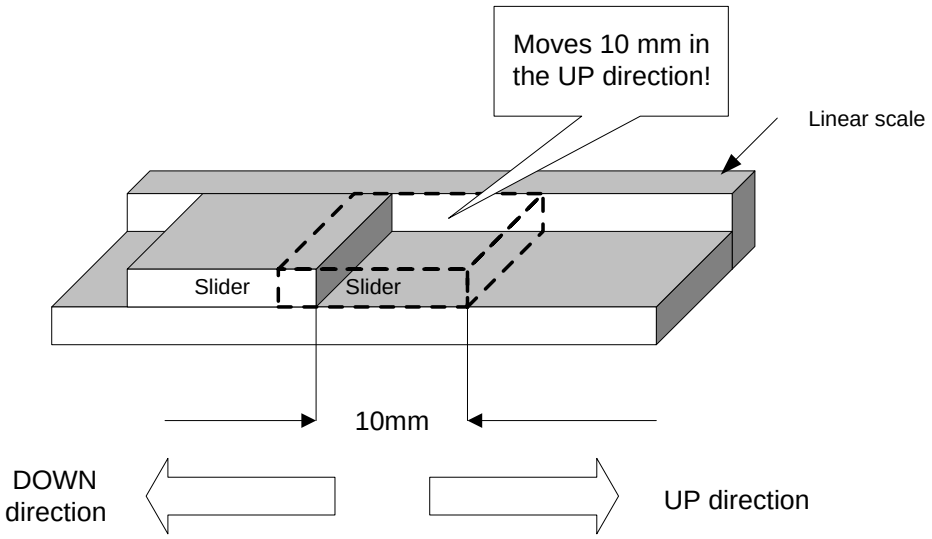


(4) Verify that the driver is reset and the “SRDY” LED on the front panel is turned on.

4.6 Auto-tuning

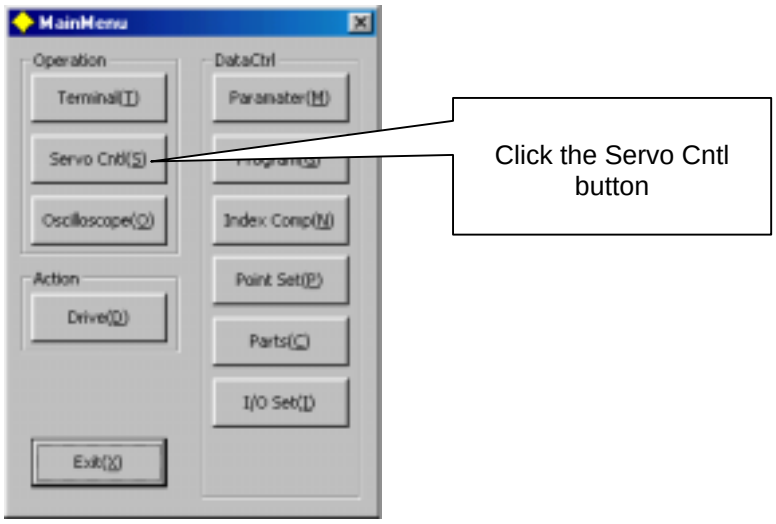
The auto-tuning can be performed according to the following procedure.

(1) Checking the motor
Check the moving direction (UP/DOWN) of the slider



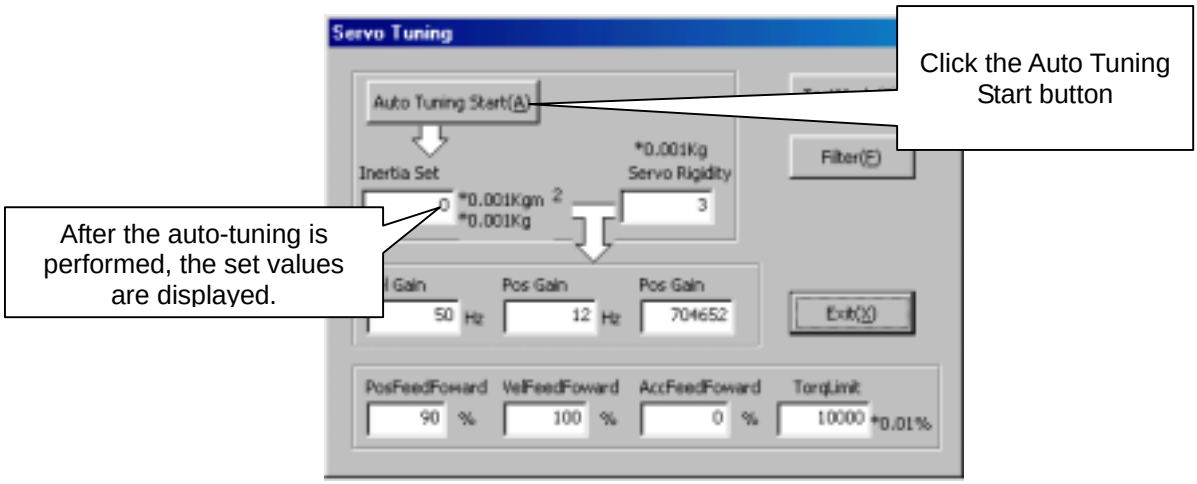
△ **Caution** Check the UP/DOWN directions from the slider side. When the auto-tuning starts, the slider will move 10 mm in the UP direction. Make sure that there is no mechanical interference within 10 mm from the slider's stop position.

(2) Click the "Servo Cntl (S)" button on the "MainMenu."



(3) Click Auto Tuning Start (tuning starts).

(4) Follow the message on the dialog box and click “OK” to start the auto-tuning operation.



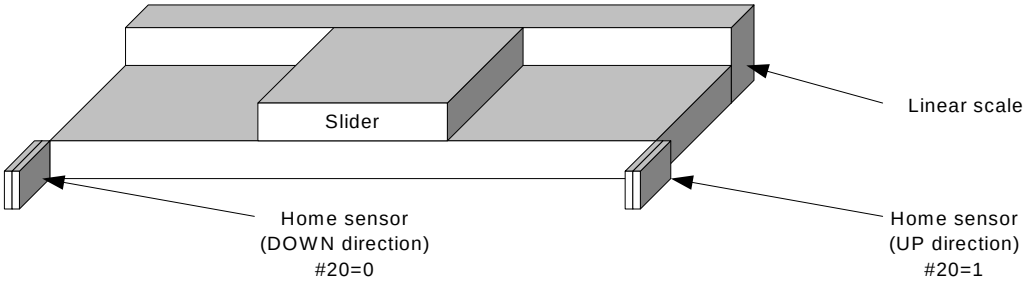
△ **Caution**	The slider will move back and forth 6 times in the UP direction. The length of the movement is 10 mm. Make sure that the slider does not interfere mechanically with other devices in the vicinity.

(5) Each parameter setting value is displayed and the auto-tuning is automatically terminated.

4.7 Performing Homing Operation

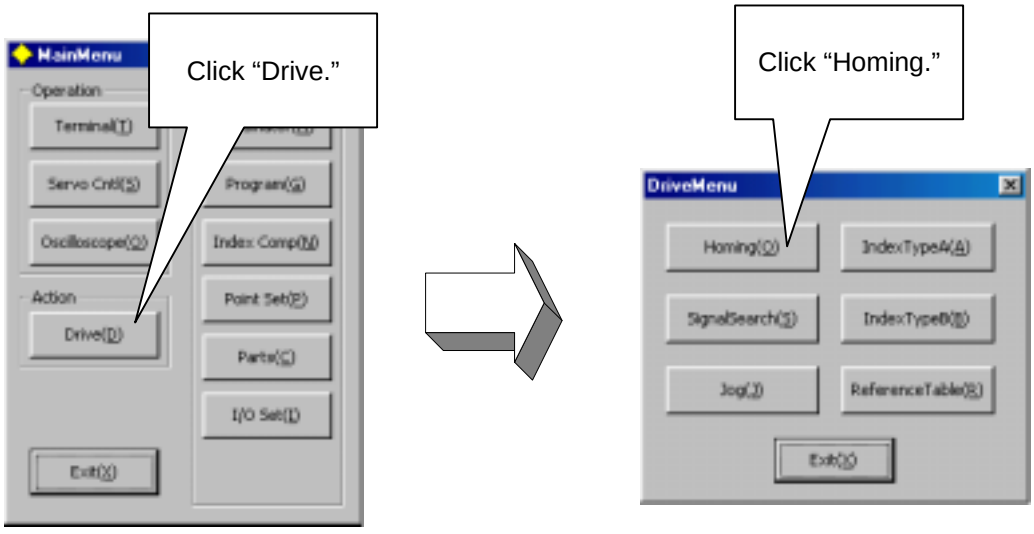
A homing operation can be performed according to the following procedure.

(1) Checking the sensor location
Check the location of the sensor which finds the homing direction.



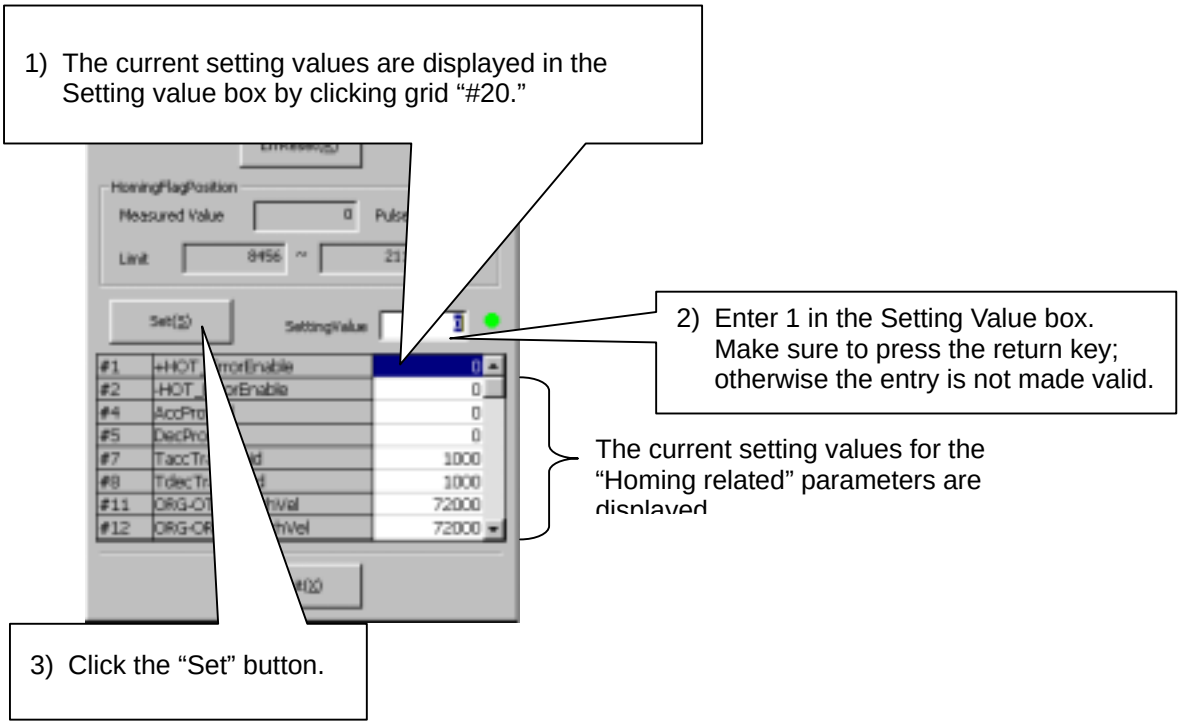
△ Caution Check whether the direction is UP or DOWN when seen from the connector box side of the slider part.

(2) Check the setting of the homing direction through the PC utility.
Click “Drive” in the Main Menu, then “Homing.”



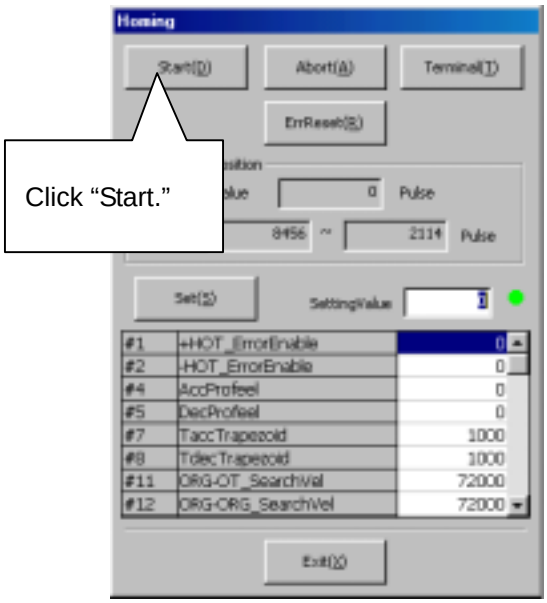
MainMenu DriveMenu

(3) Set the “Homing direction” in the “Homing” dialog box.



△ **Caution** Enter "#20=1" when using the sensor on the UP direction side. When the sensor on the DOWN direction side is to be used, the initial value will be "#20 = 0."

(4) Click the “Start” button to start the homing operation.



△ **Caution**

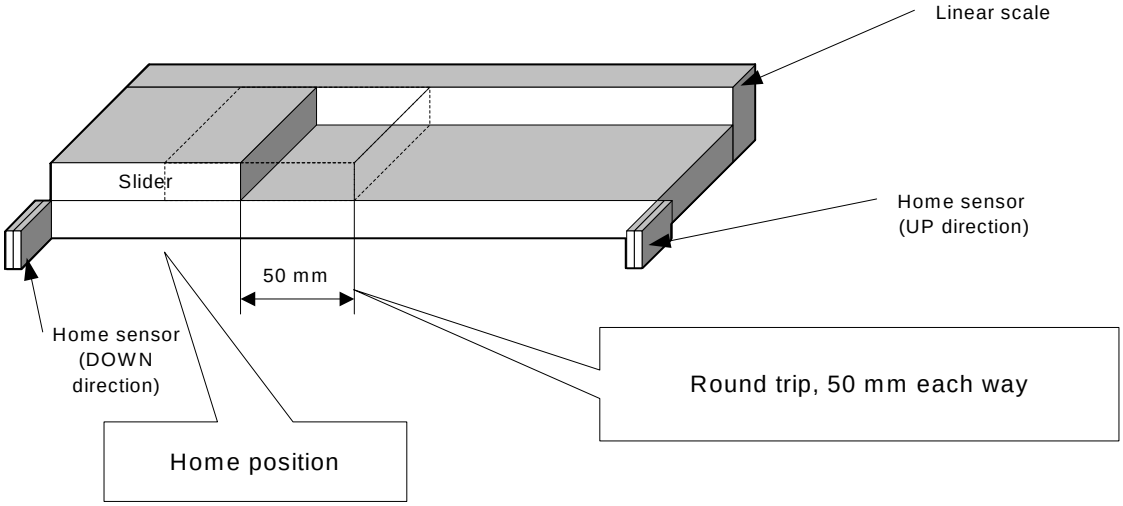
1) The homing operation finishes automatically after the operation is completed. In the event of
   - The homing operation does not finish, and
   - The motor does not stop even when the home position sensor is detected,

   Click “Abort (A)” to stop the motor, then check “Wiring” and “Auto-tuning” again.
2) If a homing abnormality message is displayed, follow the message to adjust the flag position using the limit value as a guideline. If an error occurs, press “ErrReset” as well.

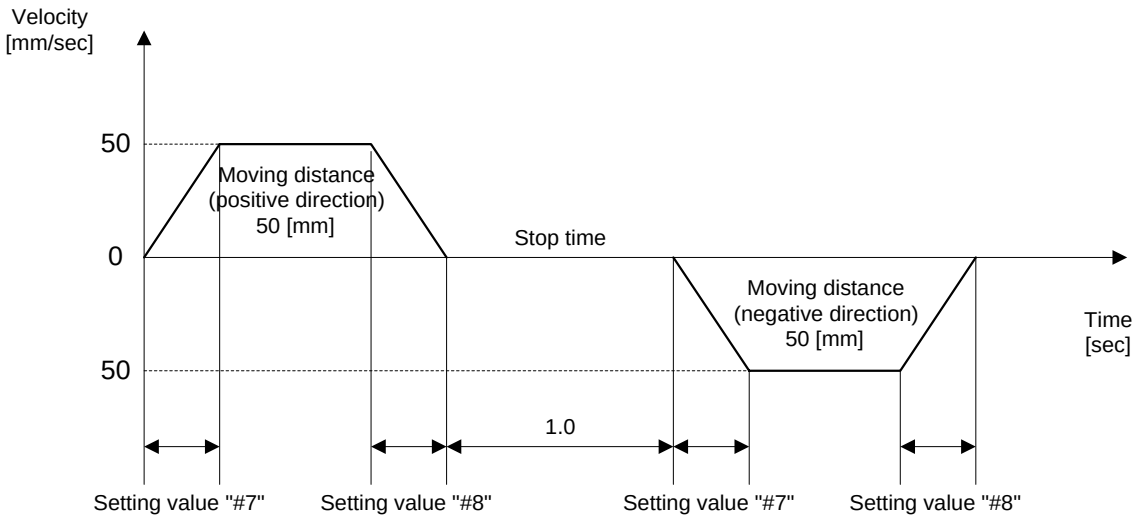
4.8 Performing Program Operation

In this section, you will create a sample program and verify that the motor operates as expected.

4.8.1 Verifying Operation Conditions



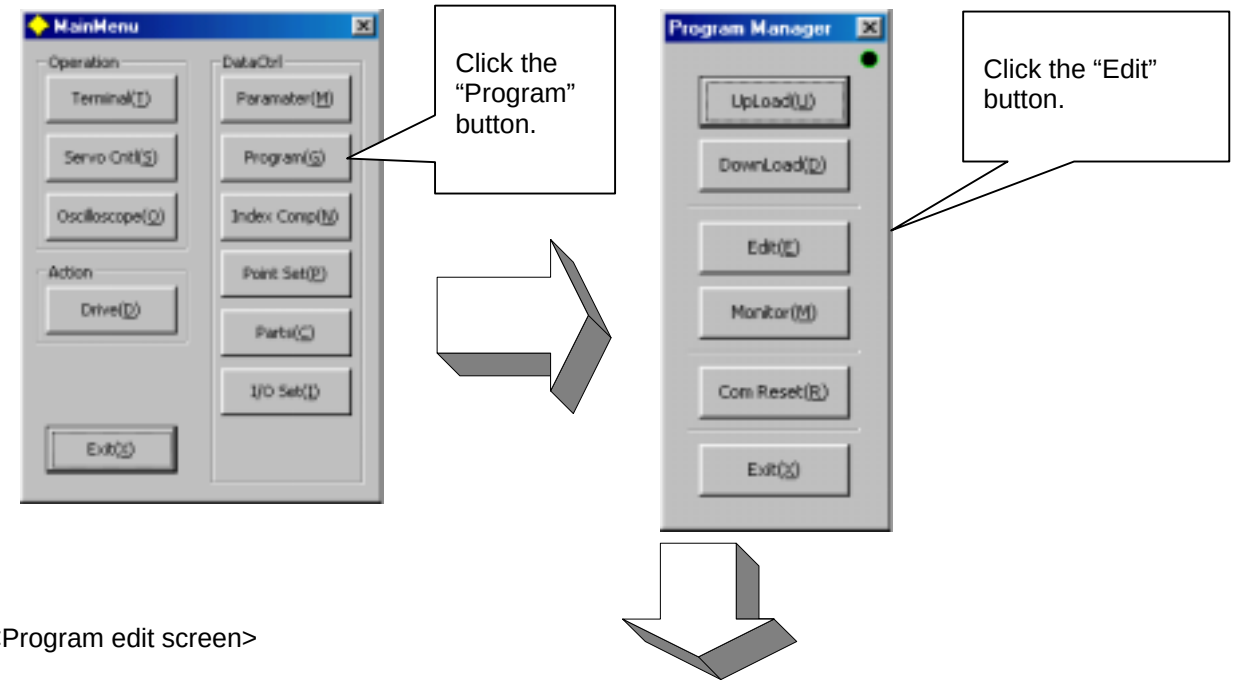
<Velocity pattern>



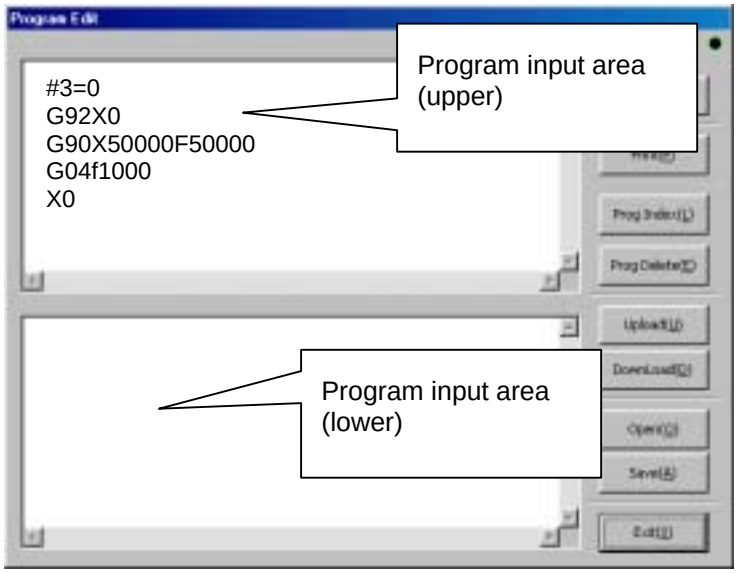
Set the acceleration time (#7) and deceleration time (#8) according to the operation conditions (the initial value is 1000 msec for both #7 and #8). Refer to Section 4.8.3, “Starting the Sample Program, (4) simplified terminal” for how to set these values.

4.8.2 Creating a Sample Program

Select “Program” in the “MainMenu,” then “Edit” in the “Program Manager” menu in order to display the “Program Edit” screen and enter a program.



<Program edit screen>

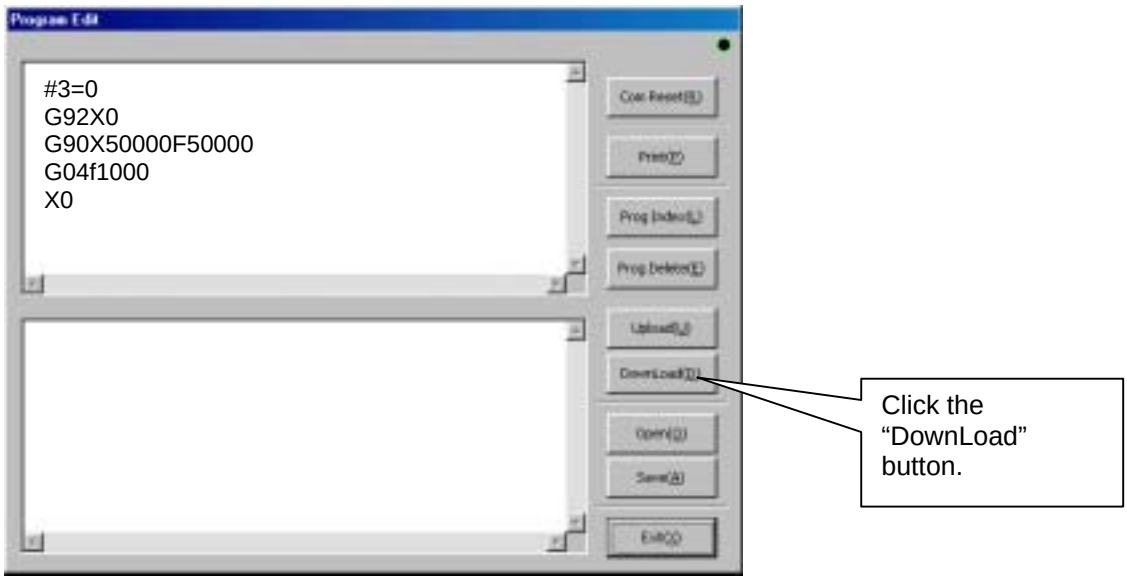


△ **Caution**	The program should be entered in the “Input area (upper)” at this point. Programs should be input in half width, case-sensitive letters.

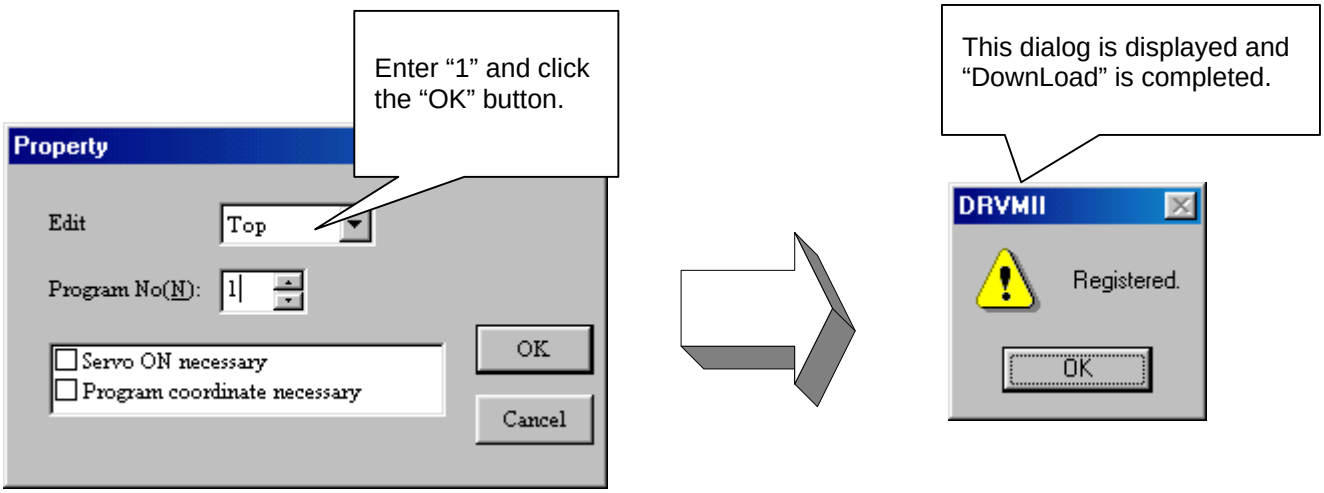
The following is the description of the sample program.

#3=0	Set the cam profile move selection to a trapezoidal move.
G92X0	Coordinate selection (current position = 0)
G90X50000F50000	Move to a position 50 mm from the home position at a velocity of 50 mm/sec. (absolute command)
G04f1000	Stop 1 second.
X0	Move from the 50 mm position to the home position at a velocity of 50 mm/sec. (absolute command)

(2) Press “DownLoad” in order to register the program.



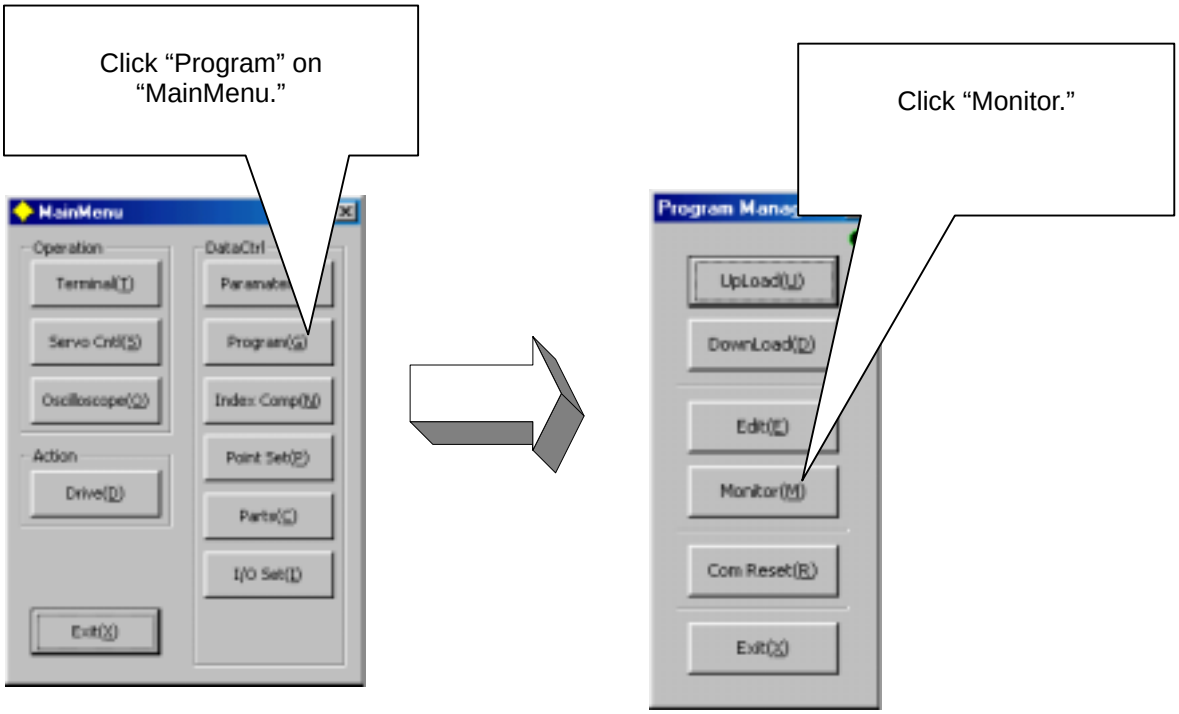
(3) Enter “1” for the program number and click the “OK” button.



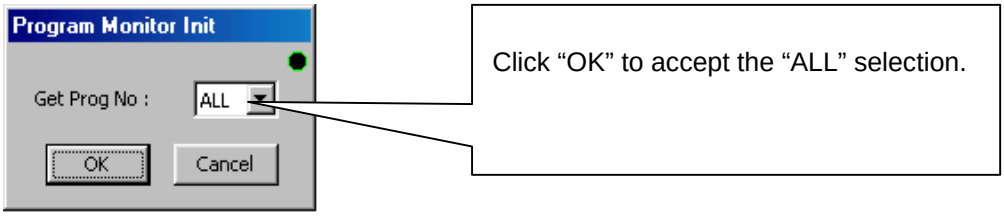
(4) Click the “Exit” button on the “Program Edit” screen to return to the “MainMenu.”

4.8.3 Starting the Sample Program

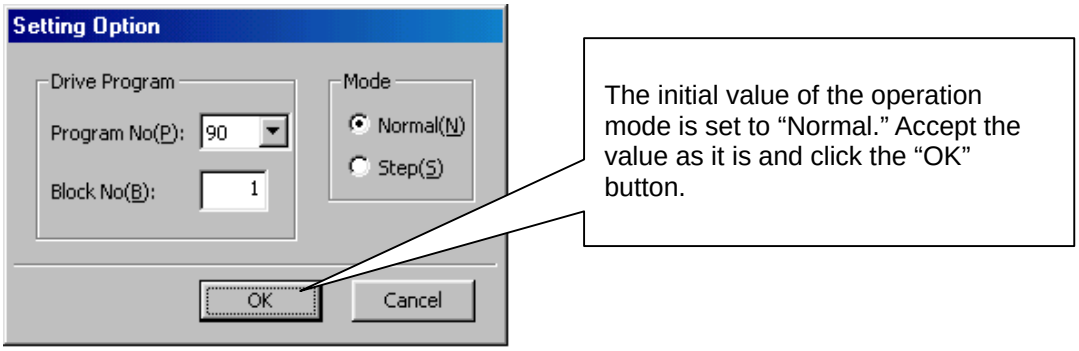
(1) Click “Program” on the “MainMenu,” and then “Monitor.”



(2) Leave the Get Prog No. selection at “ALL” in the “Program Monitor Init” dialog box and click “OK.”

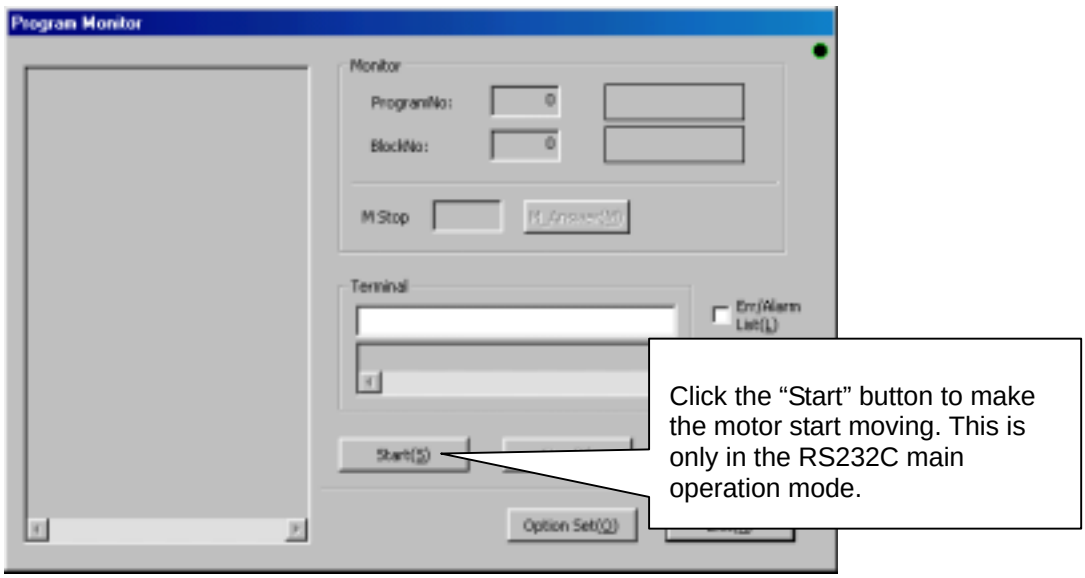


(3) Leave the “Mode” at “Normal” in the “Setting Option” dialog box, and click “OK.”

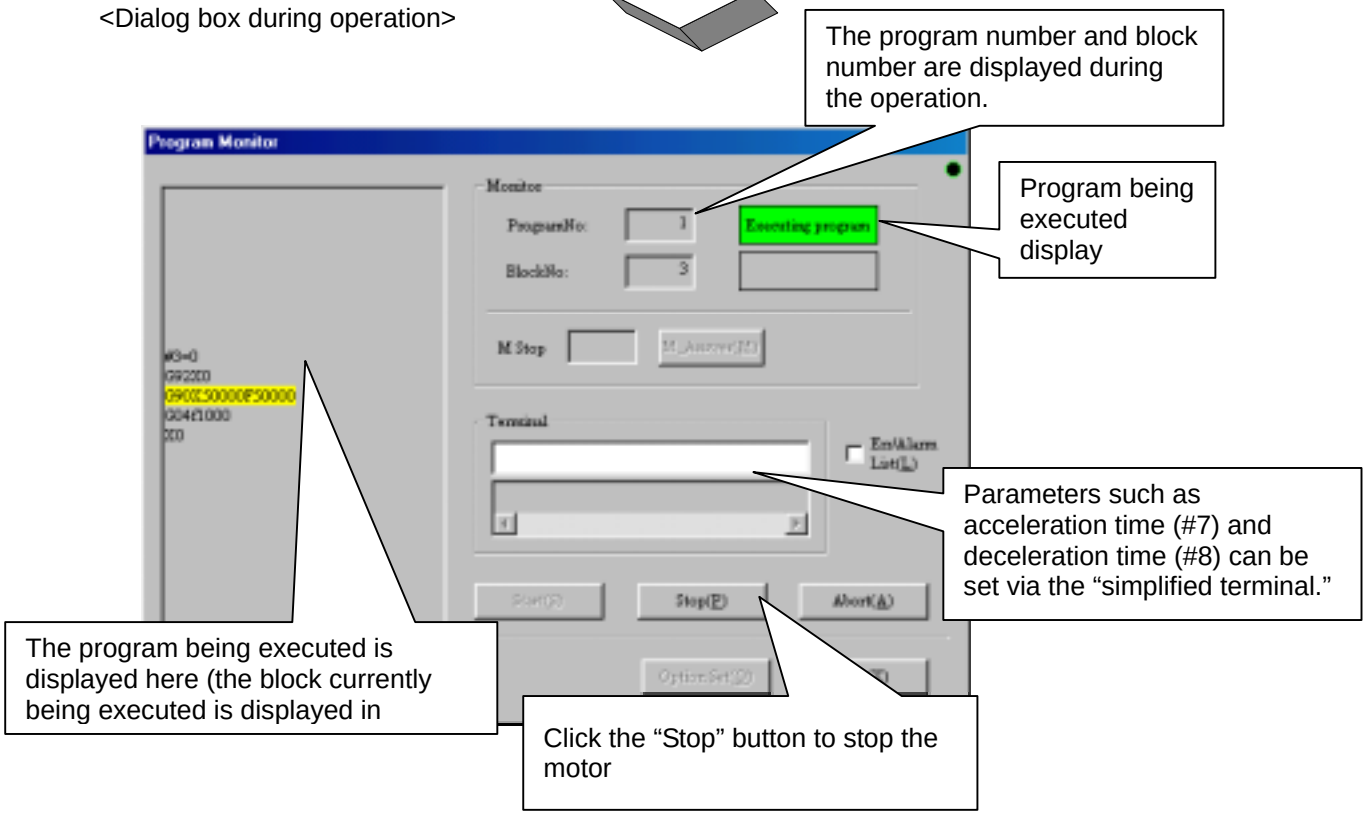


(4) Click the “Start” button in the “Program Monitor” screen to start the operation.
The program starts up once and then stops. If you want to stop the program in the middle of the operation, click the “Stop” button.

<Dialog box before starting>



<Dialog box during operation>



The basic operation is now complete. Refer to the Function Manual for details about the actual operation.

Chapter 5
Functions

5.1 Operation Mode
5.2 Parameters and Monitors
5.3 Operation Functions
5.4 Coordinate System
5.5 Acceleration/Deceleration Function
5.6 Programming Language
5.7 M Function
5.8 Control System
5.9 Other Functions
5.10 Special Parameter Processing

5.1 Operation Mode

The two interfaces, an *RS232C interface* and a *PLC interface*, are available for operation of this driver. The operation items that are available for these interfaces are not identical, but the main operations of the driver can be performed with either of the interfaces, such as operating the motor. However, if an operation, which is incompatible with one of the interfaces, is performed while operating via the other interface, the devices connected to the operating interface are no longer able to manage the conditions. In order to avoid such conditions, the driver controls the interfaces through a concept called *operation mode*, in which the main operations at run time, such as the operation of the motor, can only be performed via one of the interfaces.

The operation mode is set by the status of the bit 2 of the SW1 switch on the front panel when the power is turned on. If the switch is ON, the PLC interface is permitted to perform the main operation. If the switch is OFF, the RS232C interface is permitted to perform the main operation.

The following explains the operation items available in each interface as well as their relationship with the operation mode. The operation display pendant (TBX) conforms to the RS232C interface.

Command name / Operation device	Main operation permission: RS232C			Main operation permission: PLC		
	RS232C	TBX	PLC	RS232C	TBX	PLC
Emergency stop command	-	○	○	-	○	○
Servo ON/OFF	ON/OFF		Allowed/ Prohibited	X		ON/OFF
Start	○		X	X		○
Stop	○		X	X		○
Abort	○		○	○		○
Error reset	○		○	○		○
Interlock	-		○	-		○
Velocity override switching	-		○	-		○
Program auto rewind	-		○	-		○
Integral position control operation switching	○		X	X		○
Homing offset position settings	○		-	○		-
Coordinate system settings	○		-	○		-
M function	*According to #102 Enabling the selection of RS232C for the M function interface parameter settings.*					
Jog move command	According to *#217 Jog move operation: RS232C selection* parameter settings.					
Get error code	According to the monitors *#345* and *#346*		(○)	According to the monitors *#345* and *#346*		(○)
Parameter settings	According to parameter command		(○)	According to parameter command		(○)
Parameter/monitor read	According to parameter command		(○)	According to parameter command		(○)

Note: Items marked with (0) may be available depending on the type of PLC interface.

5.2 Parameters and Monitors

The group of variables expressed by #*** is called *parameters and monitors*. Parameters/monitors are classified according to their numbers as follows.

Parameter No.	Parameter type	Backup	Comments
0 to 199	General parameters	Stored	Always possible to read and write.
200 to 299	Mechanical setting parameters	Stored	Reading is always possible, but writing is possible only when operating in mechanical setting mode.
300 to 399	Monitors	-	Read only. Reading is always possible, but writing is not possible at any time.
400 to 499	Registration parameters	Stored	Always possible to read and write. The user can use them freely.
500 to 599	Volatile parameters	Not stored	Always possible to read and write. The user can use them freely. Set to 0 when the power is turned on.
600 to 699	Point set (value)	Stored	Always possible to read and write. Used during table reference operation.
700 to 799	Point set (option)	Stored	Always possible to read and write. Used during table reference operation.

By referring to any of the parameters/monitors in the form of #***, their values can be referenced (read) by a program or command. It is also possible to change (write) their values by a program or command, as long as the values are within the range allowed for each parameter/monitor.

The following is a list of the setting ranges of the parameters/monitors. General parameters and mechanical setting parameters are explained separately. Moreover, the setting ranges of the monitors are not listed because writing is not allowed.

Parameter No.	Minimum value	Maximum value	Initial value
400 to 499	-999999999	999999999	0
500 to 599	-999999999	999999999	0
600 to 699	-999999999	999999999	0
700 to 799	0	8	0

5.2.1 General Parameters

These parameters can be read and written at all times and an operational meaning is assigned to each. The details of each parameter will be explained separately.
Note that these parameters are stored in the driver; they will not be deleted even if the power is turned off.

5.2.2 Mechanical Setting Parameters

It is always possible to read these parameters, but writing is possible only when operating in mechanical setting mode. These parameters are set only once when the device is started up. An operational meaning is assigned to each. The details of each parameter will be explained separately.
Note that these parameters are stored in the driver; they will not be deleted even if the power is turned off.

Note: If these parameters are changed during execution in mechanical setting mode, the changes made to the values will not be reflected until the power is turned on again. Therefore, please note that if you try to read a parameter value after you change the parameter, the value before the change is read until you turn the power off and on again.

5.2.3 Monitors

These variables are used to display the driver's status. Only reading is allowed at all times and each has an operational meaning. The details of each parameter will be explained separately.

5.2.4 Registration Parameters

These variables can be used freely by the user. Reading and writing are always possible. There is no operational meaning assigned to them.
Note that these parameters are stored in the driver; they will not be deleted even if the power is turned off.

5.2.5 Volatile Parameters

These variables can be used freely by the user. Reading and writing are always possible. There is no operational meaning assigned to them.
Note that these parameters are not stored in the driver; they are all initialized to 0 when the power is turned on.

5.2.6 Points Set (Value and Option)

These values represent data used in table reference operation. Point set (value) and point set (option) are used in pairs. The point set (value) obtained by adding 600 to the specified table entry is used as a target value or relative value for positioning, and the point set (option) obtained by adding 700 to the specified table entry is used as optional data for the moving method at positioning.

The entries in the point set (value) are given in operation units.
The entries in the point set (option) have the following meaning.

0	Moving direction for incremental instruction, rotational coordinates is Type0.
1	Moving direction for incremental instruction, rotational coordinates is Type1.
2	Moving direction for incremental instruction, rotational coordinates is Type2.
3	(Reserved)
4	Moving direction for absolute instruction, rotational coordinates is Type0.
5	Moving direction for absolute instruction, rotational coordinates is Type1.
6	Moving direction for absolute instruction, rotational coordinates is Type2.
7	(Reserved)

Note that these parameters are stored in the driver; they will not be deleted even if the power is turned off.

Note: Regarding the moving direction for rotational coordinates

Type0	Proximity rotational move (multiple rotations are not allowed).
Type1	Does not cross the rotational coordinate's home position (multiple rotations are not allowed).
Type2	Calculates the target position relative to the current rotational coordinate's home position and determines the direction (multiple rotations are allowed).

5.3 Operation Functions

The following operation functions, which can be used as needed, are provided to the driver. Moreover, the status called the *idle status* in which no operation is performed is provided to perform jog moves.

The start, end, and stop actions of these operations can be controlled from either the RS232C side or the PLC interface side, according to the operation mode setting when the power to the driver is turned on. Refer to the related sections (Section 6.1, "RS232C Interface" and Section 6.2, "PLC Interface") for operation methods.

The methods used to end each operation can be classified into three categories: the *self-end type* that ends automatically when the operation is complete, the *non-self-end type* that cannot end the action by itself, and the *non-end type* that cannot be stopped once started. An appropriate end or stop action must be performed for each type.

Operation No.	Name	Comment	Ending type
0	Test operation	Generates a 2.5Hz square wave position command signal for adjustment of the control part.	Non-self-end
1	Auto-tuning operation	Makes the slider oscillate, measures load inertia and load mass, and sets the parameters for the control part automatically.	Self-end
2	(Reserved)		Self-end
3	Homing operation	Finds the home position using the hardware over-travel signal, homing sensor, and motor Zero signal to establish the coordinate system.	Self-end
4	Program operation	Runs a program that was already registered.	Self-end
5	Signal search operation	Moves the slider until it reaches the set signal status and stops it when it reaches the status.	Self-end
6	(Reserved)		
7	MDI operation	Runs NC executable statements and parameter statement input from RC232C as soon as they are input.	Non-self-end
8	Index Type A operation	Performs index sense action based on equal division of the coordinate system. It can be used when the operation unit is set to index Type A.	Self-end
9	Index Type B operation	Performs index sense action based on unequal division of the coordinate system. It can be used when the operation unit is set to index Type B.	Self-end
10	Table reference operation	Refers to points set and performs positioning move.	Self-end
11 to 14	(Reserved)		
15	Mechanical setting mode	Special mode for changing mechanical setting parameters.	Non-end

5.3.1 Jog Operation

The slider can be moved in the positive (+) or negative (-) direction by issuing a jog move command while in idle status.
There are three types of jog move commands: *(+) direction move*, *(-) direction move*, and *stop*. Refer to the sections about the RS232C interface and PLC interface for how to issue a command.
The move can only be trapezoidal but it is possible to set the acceleration and deceleration profile. In addition, the velocity override does function in real time. The velocity in the jog move is determined by the value in the *#10 Jog Velocity* parameter.
If operation is started while the jog move is in progress, the operation is executed as soon as the jog move stops. The settling wait function is not performed at the end of the jog move.

5.3.2 Test Operation

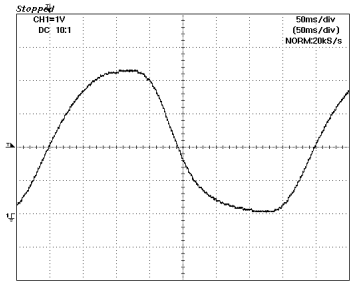
This operation generates a 2.5Hz square wave and uses it as position command signal for adjustment of the control part. Set the analog monitor to *test operation response* to measure the response waveform on the oscilloscope.
The position feed forward, velocity feed forward, and acceleration feed forward are set to 0 internally during the test operation.
The settling wait function is not performed at the end of the test operation.

[Related parameters]

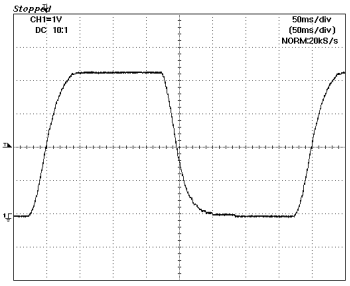
#31	Operation width under testing mode
#50	Position control bandwidth
#51	Velocity control bandwidth
#53	Position integral limiting value
#70	Analog monitor selection
#72	Test operation monitoring gain (analog monitor)

△ **Caution**

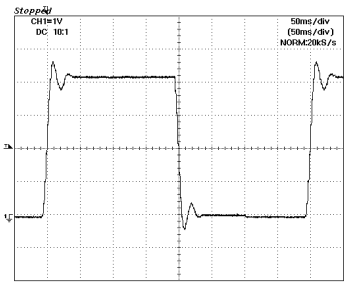
- Obtain the fastest possible rise time of the response waveform and make adjustments so that overshoot will not occur.
- The closer the position control bandwidth and velocity control bandwidth are, the more the waveform will oscillate.
- If the inertia and weight of the load are large, the oscillations may be eliminated by setting the position integral limiting value to a small value.



Increase the position control bandwidth.



Make adjustments until this waveform is reached



If the velocity control bandwidth cannot be increased any further, the position control bandwidth should be decreased.

5.3.3 Auto-Tuning Operation

This operation makes the slider oscillate, measures the inertia and weight of the load, and automatically sets the parameters for the control part.
It accelerates/decelerates with half the rated torque and rated thrust of the motor, and measures the inertia and weight of the load from the velocity changes at that time. The result of the measurement is written to the *#155 Load inertia/load mass* parameter.
From the measured inertia and weight of the load, it adjusts the position control bandwidth, velocity control bandwidth, and position integral limiting value according to the setting value of the *#38 Servo stiffness settings* parameter. It does not adjust position feed forward, velocity feed forward, and acceleration feed forward.
The settling wait function is not performed at the end of the auto-tuning operation.

[Related parameters]

#32	Operation width under Auto-tuning
#33	Maximum deceleration under Auto-tuning
#34	Initializing the deceleration time while under Auto-tuning
#37	Auto-tuning repeat count
#38	Servo stiffness settings

[Auto-set parameters]

#50	Position control bandwidth
#51	Velocity control bandwidth
#53	Position integral limiting value
#155	Load inertia/load mass

5.3.4 Homing Operation

In this operation the slider is moved according to a preset home position search method in order to establish a coordinate system. After first moving to the home position determined by the proximity signal and motor Zero signal, it continues to move an amount further given in the *#29 Offset distance from the Home position* parameter's setting value. It then sets the drive coordinate command value to the value set in the *#30 Homing complete operation command value* parameter.
Software Zero signal is used to generate motor Zero signal.
The homing operation is processed in the following order.
The move can only be trapezoidal but it is possible to set the acceleration and deceleration profile. In addition, the velocity override does function in real time.
The settling wait function is performed at each point during the homing operation.

[Related parameters]

#11	Over-travel search velocity during a homing move
#12	Homing operation: Home sensor proximity signal search velocity
#15	Homing operation: Origin position offset feed velocity
#20	Homing direction
#21	Enable/Disable the over-travel signal under the homing mode
#25	Homing operation: Origin inside selection
#27	Enabling the homing flag position error
#29	Offset distance from the Home position
#30	Homing complete operation command value
#202	Coordinates (+) direction setting

(1) OT search move

The slider moves until it finds an over-travel (OT) signal in the opposite side of the homing direction and in the opposite direction of homing direction. It is executed only when the OT signal search move is enabled in *#21 Enable/Disable the over-travel signal under the homing mode* parameter.
The moving velocity is set to the value in *#11 Over-travel search velocity during a homing move* parameter.

(2) Homing search move

The slider moves until it finds a homing sensor in the homing direction.
The moving velocity is set to the value in *#12 Homing operation: Home sensor proximity signal search velocity* parameter.

(3) Moving to outside of home position proximity area

If #202 = 1, the slider moves in (+) direction until it is out of the home position proximity area. If #202 = 0, the slider moves in (-) direction until it is out of the home position proximity area.
The moving velocity is determined internally by the driver.

(4) First home sensing move

If #202 = 1, the slider searches and moves the home position proximity signal in (-) direction, identifies the first Zero signal edge as the home position upon entering the area, and then stops. If #202 = 0, the slider searches and moves the home position proximity signal in (+) direction, identifies the first Zero signal edge as the home position upon entering the area, and then stops.
The moving velocity is determined internally by the driver.
With *#25 Homing operation: Origin inside selection*, whether the inside Zero signal in the home position proximity signal is set to the home position or the outside Zero signal is set to the home position can be selected. If #25 = 1, it is set to the inside; if #25 = 0, it is set to the outside.

(5) Second home sensing move

The second home sensing move is executed in two steps.

1) Preparation move
   The slider moves as far as diametrically opposite the Zero signal edge in the opposite direction of the Zero signal edge search direction that was used in the first home sensing move.
   The moving velocity is determined internally by the driver.

2) Zero signal edge search move
   The slider moves until it finds the Zero signal edge in the Zero signal edge search direction that was used in the first home sensing move.
   The moving velocity is determined internally by the driver.

(6) Home position move

The home position is moved to the detected Zero signal edge position. Motors that use software Zero signals further perform a second corrective move.
The moving velocity is determined internally by the driver.

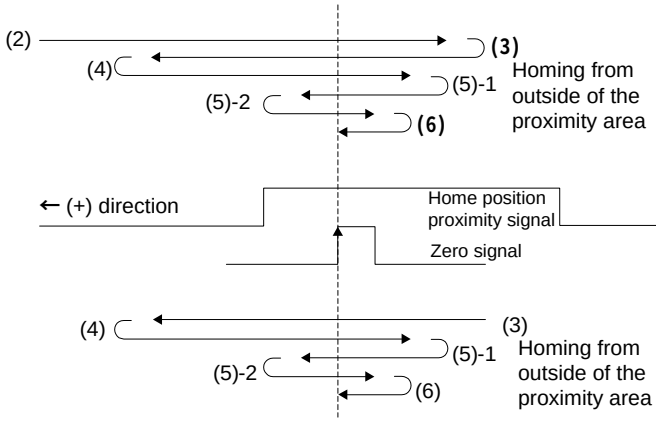
(7) Home position offset move

The home position is moved for the distance given by the value set in *#29 Offset distance from the Home position* parameter, after which the operation coordination command value is set to the value given in *#30 Homing complete operation command value* parameter.
The moving velocity is set to the value in *#15 Homing operation: Origin position offset feed velocity* parameter.

For the homing move to be completed normally, the distance between the home position proximity area and the Zero signal edge must be within a specific range based on the value of *#358 Z-phase signal pulse interval* monitor.
At the first home sensing move, the distance to the first Zero signal edge after passing through the home position proximity area (pulse amount) has been measured and written in *#318 Homing operation: The measured value* monitor. If this value does not satisfy the equations below, an error or warning will occur. In that case, the home position proximity dog should be adjusted and the homing move should be performed again. Repeat these steps until the operation is completed normally.

		#318 value	$<$	0.05*#358 value	error
0.05*#358 value	$\leqq$	#318 value	$<$	0.1 *#358 value	warning
0.1 *#358 value	$\leqq$	#318 value	$\leqq$	0.4 *#358 value	normal
0.4 *#358 value	$<$	#318 value	$\leqq$	0.45*#358 value	warning
0.45*#358 value	$<$	#318 value			error

- Operation example 1 -

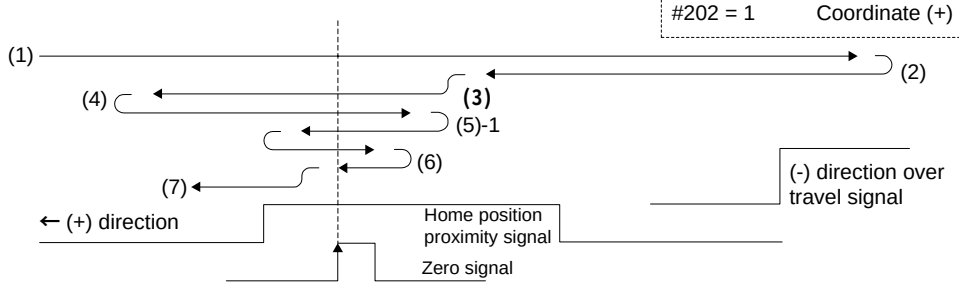


#20 = 0	Homing direction	(-) direction
#21 = 0	Enable/disable the over-travel signal under the homing mode	Not used
#25 = 1	Homing operation: Origin inside selection	Inside
#29 = 0	Offset distance from the Home position	0
#202 = 1	Coordinate (+) direction setting	

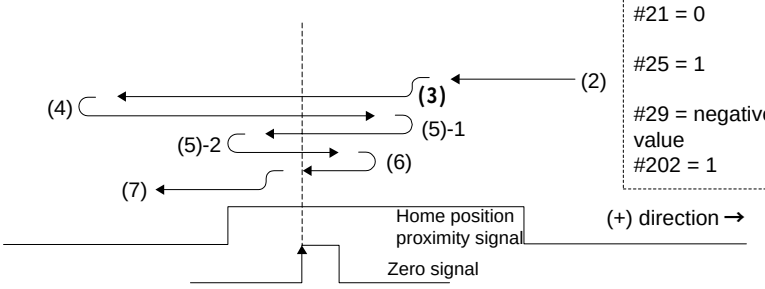
(Initial value setting)

#20 = 0	Homing direction	(+) direction
#21 = 0	Enable/disable the over-travel signal under the homing mode	Used
#25 = 1	Homing operation: Origin inside selection	Inside
#29 = positive value	Offset distance from the Home position	
#202 = 1	Coordinate (+) direction	setting

- Operation example 2 -

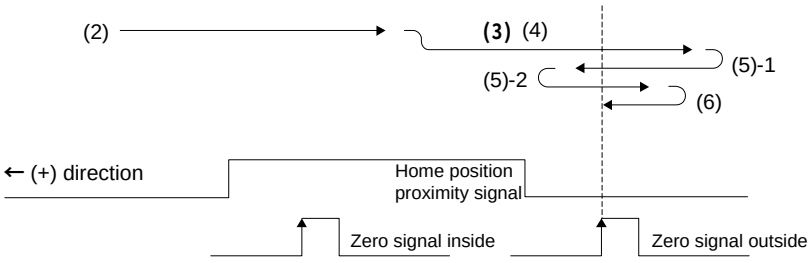


- Operation example 3 -



#20 = 0	Homing direction	(-) direction
#21 = 0	Enable/disable the over-travel signal under the homing mode	Not used
#25 = 1	Homing operation: Origin inside selection	Inside
#29 = negative value	Offset distance from the Home position	
#202 = 1	Coordinate (+) direction setting	

- Operation example 4 -



#20 = 0	Homing direction	(-) direction
#21 = 0	Enable/disable the over-travel signal under the homing mode	Not used
#25 = 0	Homing operation: Origin inside selection	Outside
#29 = 0	Offset distance from the Home position	0
#202 = 1	Coordinate (+) direction setting	

5.3.5 Program Operation

Program that have already been registered can be executed.
In a program, it is possible to use *parameter statements* to change parameters, *control parameters* for expressing branches and loops, and *comment statements* for explaining the program in addition to *NC executable statements* that directly operate the motor. Also, it is possible to call another program from a program as a subroutine.
A program can be registered by the PC utility. Refer to Chapter 7, "M2 PC Utility" for an explanation of how to register a program. Program properties can also be set for the program at registration. In doing so, an error check is performed according to the *program properties* when the program is executed.
Program can be registered as user programs numbered from 0 to 89. The capacity is 1000 blocks in total for user programs. Moreover, 10 programs are available from numbers 90 to 99 as built-in programs. The built-in programs cannot be overwritten.
The program with number 0 is a special program; it is started automatically when the power is turned on.
When the positioning operation is completed during program operation, the settling wait function is performed according to the *#106 Settling wait enable* parameter settings.

Note: The block number is the number of NC executable statements, parameter statements, control parameters, comment statements, etc. in the program. Normally one line makes up one block, but when a comment statement is added to an NC executable statement, parameter statement, or control parameter in one line, it is counted as two blocks.

(1) [Start-up options]

When starting a program operation, the action changes depending on which start-up option is given. Refer to Section 6.1, "RS232C Interface" and Section 6.2, "PLC Interface" for an explanation of the start-up options.

1) Start-up option 1
   Specify a program number. If a program number is not specified, the program whose execution program number is stored will be executed.

2) Start-up option 2
   Specify a block number. If a block number is not specified, the execution starts from the top line if a program number is specified, and the stored execution program line if a program number is not specified.

(2) [Step execution, repeat execution, program stop, and optional stop]

When step execution is enabled by *#82 Enabling the Program step execution* parameter, the operation finishes when one line is executed at each start-up. Thus, if the program is started repeatedly without specifying start-up options 1 and 2, it can be executed line by line.
When the last line of a program has been executed, the program will be executed from the first line again if *program auto-rewind* is enabled by the PLC interface.
The operation is stopped when M code 0 is executed in the program (*program stop*).
If optional stop is enabled with the *#103 Optional stop enable* parameter, the operation is stopped when the execution of M code 1 is completed in the program.

(3) [Program properties]

The following two items can be defined in the program properties. A program can then only be executed under conditions where each check box is enabled.

1) Servo ready necessity

2) Drive coordinate necessity

(4) [Start-up program]

If you register a program to program number 0, that program will be executed automatically when the power of the driver is turned on. This is convenient if you make a program that should always be performed after the power is turned on, such as a homing operation. This function can be switched between enabled and disabled by the #92 *Start-up program enable* parameter. When it is disabled, program number 0 is treated in the same way as programs with other program numbers.

[Related parameters]

#82	Enabling the program step execution
#83	ABS in program/MDI operation
#92	Start-up program enable
#100	M function enable in program/MDI operation
#103	Optional stop enable

5.3.6 Signal Search Operation

This operation moves the slider until the status reaches the preset signal status, and decelerates and stops when the status reaches the preset signal status.
There are three kinds of signals that can be set: (+) direction hardware over-travel signal, (-) direction hardware over-travel, and home position proximity signal. For each of them, it can be set whether to search for detection or for non-detection status. Moreover, if two or more signal detection are specified, the motor decelerates and stops when any one of the signals reaches the status in question.
The move can only be trapezoidal but it is possible to set the acceleration and deceleration profile. In addition, the velocity override does function in real time. The velocity during the signal search move is determined by the value in the *#9 feeding velocity* parameter.
After the signal search move is completed, the settling wait function is performed according to the setting of *#106 Settling wait enable* parameter.

[Related parameters]

#9	Feeding Velocity
#39	Signal search direction
#42	Enabling the proximity signal during homing
#43	Enabling the (+) direction over travel signal during the signal search mode
#44	Enabling the (-) direction over travel signal during the signal search mode
#47	Logic for the proximity signal during homing
#48	Logic for the (+) direction over-travel signal during the signal search mode
#49	Logic for the (-) direction over-travel signal during the signal search mode

5.3.7 MDI Operation

This operation runs *NC executable statement* and *parameter statement* input from RC232C as soon as they are input. Unlike the program operation, control parameters and comment statements cannot be executed.
After the positioning operation is completed by MDI operation, the settling wait function is performed according to the setting of *#106 Settling wait enable* parameter.

[Related parameters]

#83	ABS in program/MDI operation
#100	M function enable in program/MDI operation

5.3.8 Index Type A Operation

This operation performs an index sense action based on equal division of the coordinate system. It can be used only in a rotational coordinate system. It can be used when the operation unit selected by the *#108 Operation unit selection* parameter is set to *index Type A*.
The number of index points that the coordinate system should be divided into is determined by setting the *#109 Index Type A divisions setting* parameter. Refer to Section 5.4, “Coordinate System” for an explanation of the operation units, including how to set them.
In an index Type A operation it is possible to execute the M function after a positioning operation is completed by enabling the M function in the *#101 M function enable during index point operation* parameter. The M function takes the value of the operation command at the positioning location as its M code. Using this function, the host system can confirm the positioning location.
In addition to the trapezoidal move, a cam profile move can be used for the move. The velocity override does also function in real time. Refer to Section 5.5, “Acceleration/Deceleration Function” for details on how to determine the velocity and profile during the move.
After the index Type A operation is completed, the settling wait function is performed according to the setting of *#106 Settling wait enable* parameter.

(1) [Start-up option]

When starting an index Type A operation, the action changes depending on which start-up option is provided. Refer to Section 6.1, “RS232C Interface” and Section 6.2, “PLC Interface” for an explanation of how to give start-up options.

1) Start-up option 1
   Specify an index number. When the incremental instruction is performed, a value relative to the current operation command value is set, and when the absolute instruction is performed, the target operation command value is set. This number must be always set.

2) Start-up option 2
   Specify either an incremental instruction or an absolute instruction. If this is not specified, it is assumed that an incremental instruction is specified. However, if a start-up option dependence is not set in the *#104 ABS/INC setting during table index operation* parameter, the value specified becomes invalid.

3) Start-up option 3
   Specify a moving direction option. If this is not specified, it is assumed that Type 0 is specified as the moving direction type. However, if a start-up option dependence is not set in the *#105 Moving direction option for rotational coordinates* parameter, the value specified becomes invalid.

Note: Regarding the moving direction for rotational coordinates
- Type0 Proximity rotational move (multiple rotations are not allowed).
- Type1 Does not cross the rotational coordinate’s home position (multiple rotations are not allowed).
- Type2 Calculates the target position relative to the current rotational coordinate’s home position and determines the direction (multiple rotations are allowed).

[Related parameters]

#101	M function enable during index point operation
#104	ABS/INC setting during table index operation
#105	Moving direction option for rotational coordinates
#108	Operation unit selection
#109	Index Type A divisions setting

5.3.9 Index Type B Operation

This operation performs an index sense action based on unequal division of the coordinate system. It can be used only in a rotational coordinate system. It can be used when the operation unit selected by the *#108 Operation unit selection* parameter is set to *index Type B*.
The number of index points that the coordinate system should be divided into is determined by setting the *#110 Index Type B divisions setting* parameter. Refer to Section, 5.4 “Coordinate System” for an explanation of the operation units, including how to set them.
In an index Type B operation it is possible to execute the M function after a positioning operation is completed by enabling the M function in *the #101 M function enable during index point operation* parameter. The M function takes the value of the operation command at the positioning location as its M code. Using this function, the host system can confirm the positioning location.
In addition to the trapezoidal move, a cam profile move can be used for the move. The velocity override does also function in real time. Refer to Section 5.5, “Acceleration/Deceleration Function” for details on how to determine the velocity and profile during the move.
After the index Type B operation is completed, the settling wait function is performed according to the setting of *#106 Settling wait enable* parameter.

(1) [Start-up option]

When starting an index Type B operation, the action changes depending on which start-up option is provided. Refer to Section 6.1, “RS232C Interface” and Section 6.2, “PLC Interface” for an explanation of how to give start-up options.

1) Start-up option 1
   Specify an index number. When the incremental instruction is performed, a value relative to the current operation command value is set, and when the absolute instruction is performed, the target operation command value is set. This number must be always set.

2) Start-up option 2
   Specify either an incremental instruction or an absolute instruction. If this is not specified, it is assumed that an incremental instruction is specified. However, if a start-up option dependence is not set in the *#104 ABS/INC setting during table index operation* parameter, the value specified becomes invalid.

3) Start-up option 3
   Specify a moving direction option. If this is not specified, it is assumed that Type 0 is specified as the moving direction type. However, if a start-up option dependence is not set in the *#105 Moving direction option for rotational coordinates* parameter, the value specified becomes invalid.

Note: Regarding the moving direction for rotational coordinates
- Type0 Proximity rotational move (multiple rotations are not allowed).
- Type1 Does not cross the rotational coordinate’s home position (multiple rotations are not allowed).
- Type2 Calculates the target position relative to the current rotational coordinate’s home position and determines the direction (multiple rotations are allowed).

[Related parameters]

#101	M function enable during index point operation
#104	ABS/INC setting during table index operation
#105	Moving direction option for rotational coordinates
#108	Operation unit selection
#110	Index Type B divisions setting

5.3.10 Table Reference Operation

This operation references the points set, and performs the corresponding positioning move.
A point set item consists of a *value* (#600 to #699) and an *option* (#700 to #799). The point set (value) obtained by adding 600 to the specified table number is used as a target value or relative value for positioning and the corresponding point set (option) obtained by adding 700 is used as optional data for the moving method, etc. at positioning.
For details of points set, refer to Section 5.2.6, “Points Set (Value and Option).”
In addition to the trapezoidal move, a cam profile move can be used for the move. The velocity override does also function in real time. Refer to Section 5.5, “Acceleration/Deceleration Function” for details on how to determine the velocity and profile during the move.
After the table reference operation is completed, the settling wait function is performed according to the setting of *#106 Settling wait enable* parameter.

(1) [Start-up option]

When starting table reference operation, the action changes depending on which start-up option is provided. Refer to Section 6.1, “RS232C Interface” and Section 6.2, “PLC Interface” for an explanation of how to give start-up options.

1) Start-up option 1
   Specify a table number. The point set value and option are obtained based on this number. The point set value obtained is set as a value relative to the current operation command value in case of an incremental instruction, and as the target operation command value in case of an absolute instruction. This number must always be specified.

2) Start-up option 2
   Specify either an incremental instruction or an absolute instruction. If this is not specified, it is assumed that an incremental instruction is specified. However, if a start-up option dependence is not set in the *#104 ABS/INC setting during table index* operation parameter, the value specified becomes invalid.

3) Start-up option 3
   Specify a moving direction option. If this is not specified, it is assumed that Type 0 is specified as the moving direction type. However, if a start-up option dependence is not set in the *#105 Moving direction option for rotational coordinates* parameter, the value specified becomes invalid.

Note: Regarding the moving direction for rotational coordinates
- Type0 Proximity rotational move (multiple rotations are not allowed).
- Type1 Does not cross the rotational coordinate’s home position (multiple rotations are not allowed).
- Type2 Calculates the target position relative to the current rotational coordinate’s home position and determines the direction (multiple rotations are allowed).

[Related parameters]

#104	ABS/INC setting during table index operation
#105	Moving direction option for rotational coordinates

5.3.11 Mechanical Setting Mode

This is a special mode for changing parameters related to the mechanical settings. Parameters #200 to #299 can be changed only when this operation is being executed. In addition, once started, this operation cannot be stopped. After you finish changing the necessary parameters, turn the power to the driver off and on again. Note that if you change these parameters, the changed values are not reflected until after the power is turned on again. Therefore, please note that if you try to read a parameter value after you change the parameter, the value before the change is read until you turn the power off and on again.

5.4 Coordinate System

The driver controls the position of the motor by three coordinate systems with different units: an *operation unit coordinate system*, a *command unit coordinate system*, and a *pulse coordinate system*.
The pulse coordinate system manages the coordinate values in units of pulses that can be detected by the encoder resolver. The position control part of the driver controls the motor based on these coordinate values. It has a command value and a current value, and they are always updated and displayed in the *#320 Pulse position command value* monitor and *#321 Pulse position current value* monitor, respectively.
The command unit coordinate system manages the coordinate values in command units specified by the user. The acceleration/deceleration control part of the driver performs computation based on these coordinate values. It has only command values, which are always updated and displayed in the *#323 Command unit command value* monitor.
The operation unit coordinate system is used for the positioning commands by the user. This coordinate system has only command values, which are updated and displayed in the *#326 Operation command value* monitor when the user makes positioning commands and coordinate system settings.
These coordinate systems can be used in two ways: rotational coordinates and linear coordinates. They are switched by the *#212 Straight line coordinate selection* parameter.

5.4.1 Scaling Conversion (Conversion from Command Units to Pulse Units)

The scaling of values can be converted by setting *#208 Command unit selection* parameter. The meaning hereof is not identical in the rotating type motor (DYNASERV) and linear motor (LINEARSERV) contexts. They are hereby explained separately.
In case of the rotating type (DYNASERV), the conversion works as follows with respect to the value set in #208. The value of the *#324 Scaling data (command unit side)* monitor is set as follows according to the setting of #208. The value of the *#325 Scaling data (pulse side)* monitor is set to the value of the *#210 Scaling data (pulse side)* parameter as it is regardless of the setting of #208.

#208	Function	#324	Remark
0	Pulse	#210	The command unit is given in pulse units that can be detected by the encoder/resolver as in the pulse coordinate system.
1	Angle (1/1000 degree unit)	360000	360 degrees correspond to #210. The command unit is 1/1000 degree.
2	Angle (1/100 degree unit)	36000	360 degrees correspond to #210. The command unit is 1/100 degree.
3	Scaling	#209	#209 corresponds to #210. The command unit is the value set in #209.
4	Angle (1/10000 degree unit)	3600000	360 degrees correspond to #210. The command unit is 1/10000 degree.

In case of the linear type (LINEARSERV), the conversion works as follows with respect to the value set in #208. The value in the *#324 Scaling data (command unit side)* monitor is set as follows according to the setting of #208. The value of *#325 Scaling data (pulse side)* monitor is set to the value of the *#210 Scaling data (pulse side)* parameter as it is regardless of the setting of #208.

#208	Function	#324	Remark
0	Pulse	#210	The command unit is given in pulse unit that can be detected by the encoder/resolver as in the pulse coordinate system.
1	Length (1 μm unit)	1000000	1 m corresponds to #210. The command unit is 1 μm.
2	Length (10 μm unit)	100000	1 m corresponds to #210. The command unit is 10 μm.
3	Scaling	#209	#209 corresponds to #210. The command unit is the value set in #209.
4	Length (0.1 μm unit)	10000000	1 m corresponds to #210. The command unit is 0.1 μm.

When converting command values given in command units to pulse units, if the conversion results become smaller than -2147483648 or larger than 2147483647, a coordinate system error occurs.

[Related parameters]

#208	Command unit selection
#209	Scaling data (command unit side)
#210	Scaling data (pulse side)

5.4.2 Conversion from Operation Units to Command Units

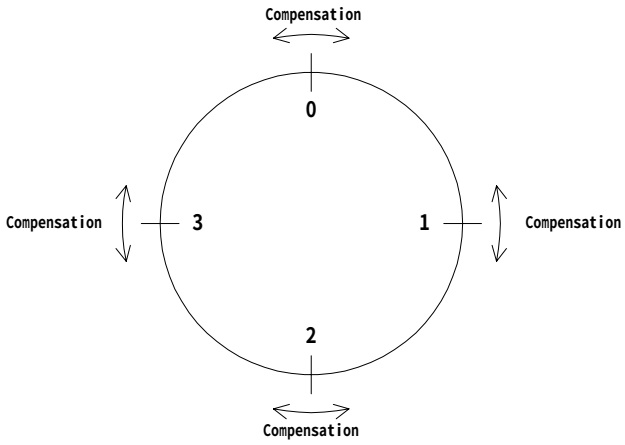
There are three types of operation units: *command unit*, *index Type A*, and *index Type B*. Switching among them can be done by setting the *#108 Operation unit selection* parameter. In a linear coordinate system, only command units can be used.
If a command unit is selected as operation unit, the unit of a positioning command becomes the command unit as it is.
If index Type A is selected as operation unit, the unit of the positioning command becomes the index point number obtained by equally dividing one rotation of rotational coordinates defined by *#324 Scaling data (command unit side)*. The number of divisions is set by the *#109 Index Type A divisions setting* parameter. The setting can be done in two ways: the number of divisions can be set directly (1 to 100) or indirectly by selecting an index compensation file that defines a compensation value for each index point (-1: index compensation file A, 0: index compensation file B). The range of divisions is 1 to 100 for both methods. Note that when index compensation file A or B is selected, the file must be registered to the driver, otherwise the positioning command cannot be made.
If index Type B is selected as operation unit, the unit of the positioning command becomes the index point number obtained by equally dividing one rotation of rotational coordinates defined by *#324 Scaling data (command unit side)*. The number of divisions is set by the *#110 Index Type B divisions setting* parameter. The setting is made indirectly by selecting an index Type B file that defines command unit command values for each index point (-1: index Type B file A, 0: index Type B file B). The range of divisions is 1 to 100 for both methods. Note also that the selected index Type B file must be registered to the driver, otherwise the positioning command cannot be made.

(1) [Index compensation file]

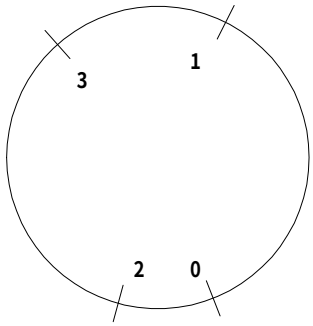
In an index compensation file, the number of index divisions and compensation data for each index point are registered. The unit of compensation data is command unit. Refer to Chapter 7, “M2 PC Utility” for how to register a file.

(2) [Index Type B file]

In an index Type B file, the number of index divisions and command unit command value data for each index point are registered. The command unit command value data should be given in command units. Refer to Chapter 7, “M2 PC Utility” for how to register a file.



[Index compensation file image]



[Index Type B file image]

[Related parameters]

#108	Operation unit selection
#109	Index Type A divisions setting
#110	Index Type B divisions setting

5.4.3 Rotational Coordinates

In a rotational coordinate system, the command unit coordinate system and the pulse coordinate system are each normalized using the values found in the *#324 Scaling data (command unit side)* and *#325 Scaling data (pulse side)* monitors respectively as one full rotation. If the value in #325 is the same as the value found in the *#355 Motor resolution* monitor, the coordinate value after one rotor rotation is the same as the coordinate value before the rotation. In rotational coordinates, the coordination system does not overflow even if the rotor is kept rotating in the same direction. In addition, the software limit function available in linear coordinates is not provided.
In rotational coordinates, the following three *moving direction options for rotational coordinates* are available for determining the moving direction for a positioning move. The options are furthermore set as follows.

(1) [Moving direction for rotational coordinates]

Type0 Proximity rotational move (multiple rotations are not allowed).
Type1 Does not cross the rotational coordinate's home position (multiple rotations are not allowed).
Type2 Calculates the target position relative to the current rotational coordinate's home position and determines the direction (multiple rotations are allowed).

(2) [Operation method and moving direction options for rotational coordinates]

Program operation, MDI operation
The moving direction is determined by the *#105 Moving direction options for rotational coordinates* parameter.
Table reference operation
The moving direction is determined by the *#105 Moving direction options for rotational coordinates* parameter.
If the setting value is "3," the moving direction is determined by a start-up option.
If the setting value is "4," the moving direction is determined by the point set.
Index Type A operation, index Type B operation
The moving direction is determined by the *#105 Moving direction options for rotational coordinates* parameter.
If the setting value is "3," the moving direction is determined by a start-up option.

[Related parameters]

#105	Moving direction options for rotational coordinates

5.4.4 Linear Coordinates

In linear coordinates, the range that commands can be given in is –999999999 to +999999999 command units. When a command value in command units exceeds this range during the operation, either a + or – direction soft over-travel error occurs regardless of the settings of the *#85 Enable the (+) direction soft limit error* and *#86 Enable the (-) direction soft limit error* parameters.
In linear coordinates, two software limit functions are available. Set the limit amounts in the *#87 (+) direction soft limit settings* and *#88 (-) direction soft limit settings* parameters, then enable the error for each direction in #85 and #86. By doing so, a soft over-travel error occurs when a command unit command value exceeds the specified range.

[Related parameters]

#85	Enable the (+) direction soft limit error
#86	Enable the (-) direction soft limit error
#87	(+) direction soft limit settings
#88	(-) direction soft limit settings

5.4.5 Operation Coordinates Unsettled Status

The operation coordinates unsettled status refers to a condition where the match between an operation command value and the command unit command value is lost. When the power is turned on, the status is unsettled. Incremental instruction is prohibited in unsettled status.
To recover from unsettled status to settled status, either of the following operations is required.

- Normal completion of homing move
- Coordination system settings
- Absolute instruction

Furthermore, when one of the following operations is performed the status becomes unsettled.

- Servo OFF
- Starting a jog move, homing move, and signal search move
- Stopping positioning move (including test operation, auto-tuning operation)
- Changing operation units
- Changing *#109 Index Type A divisions setting* parameter when operation unit is index Type A.
- Changing *#110 Index Type B divisions setting* parameter when operation unit is index Type B.

5.5 Acceleration/Deceleration Function

The velocity profile of the driver can largely be classified into *trapezoidal moves* and *cam profile moves*. In addition, it has a velocity override function for switching velocity during a move, which works in real time even during the move *(real time velocity override function).*
The maximum velocity of the motor is defined in *#213 Maximum velocity*, but is limited by the maximum velocity defined within the driver. The limited value is displayed in the *#357 Maximum velocity* monitor.

5.5.1 Velocity Override Function

Three values, the values in the *#16 Velocity override percentage 1* and *#17 Velocity override percentage 2* parameters, and 0, are selected for the velocity override value by the PLC interface. Refer to Section 6.2, "PLC Interface" for how to select them. Of course, if the values of #16 and #17 are directly changed, the velocity override value is changed when the values are selected.
The velocity override value can be set in increments of 0.01% from 0 to 200%. Please note that if a value greater than 100% is set, the velocity during the move may exceed the maximum velocity, which may cause an error. How the velocity override is applied differs for the trapezoidal move and cam profile move.

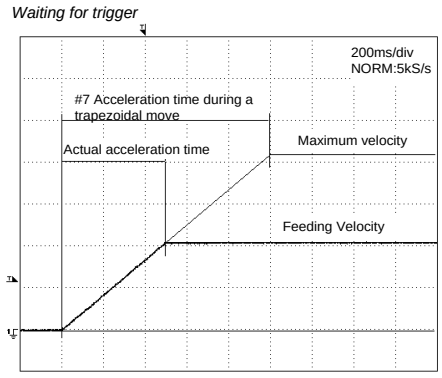
[Related parameters]

#16	Velocity override percentage 1
#17	Velocity override percentage 2

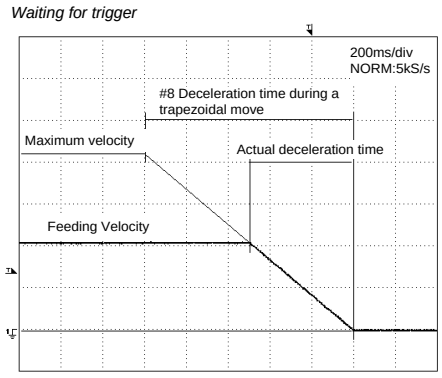
5.5.2 Trapezoidal Motion Profile

In a trapezoidal motion, the acceleration of the moving part follows the acceleration type set by the *#4 Selecting acceleration type* parameter until it reaches the feed velocity. Hereafter, the move is continued with the feed velocity, then decelerates and stops according to the deceleration type set by the *#5 Selecting deceleration type* parameter. By default, the feed velocity is equal to the velocity set by the *#9 Feeding Velocity* parameter.
The acceleration and deceleration types can be selected separately. There are two options for acceleration and deceleration: a *constant acceleration type* and an *S-shaped type* (where the acceleration/deceleration follows a second order spline). Generally, the S-shaped type can limit vibrations in the machine better, but the peak torque or peak thrust at acceleration/deceleration become greater and a correspondingly larger motor torque or motor thrust will be required.
The acceleration/deceleration time can also be selected separately. The values of the *#7 Acceleration time during a trapezoidal move* and *#8 Deceleration time during a trapezoidal move* parameters are set to values equivalent to the maximum velocity shown in the *#357 Maximum velocity* monitor. The actual acceleration/deceleration time becomes the value obtained by multiplying the maximum velocity by the velocity ratio during a trapezoidal move. By doing so, the same acceleration can be maintained without changing #7 and #8 even when the velocity setting is changed at a positioning command.

[Constant acceleration]

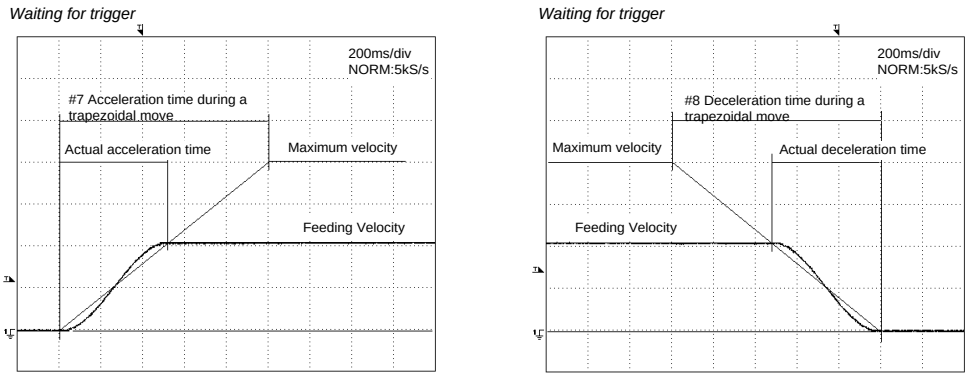


[Acceleration]



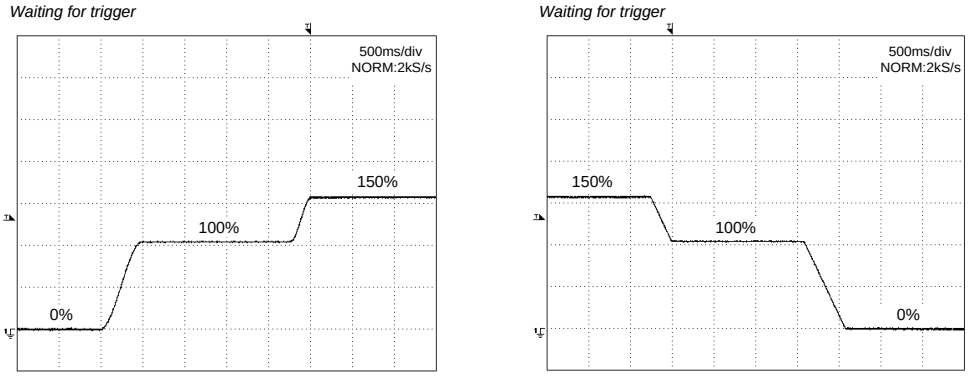
[Deceleration]

[S-shaped]



[Acceleration] [Deceleration]

The feed velocity during a move becomes the commanded velocity multiplied by the velocity override value. If the velocity override value is changed during a move, the moving part is accelerated at the same acceleration profile and acceleration as the normal acceleration time when the velocity override value is increased. In the same way, the moving part is decelerated at the same deceleration profile and deceleration as the normal deceleration time when the velocity override value is decreased.



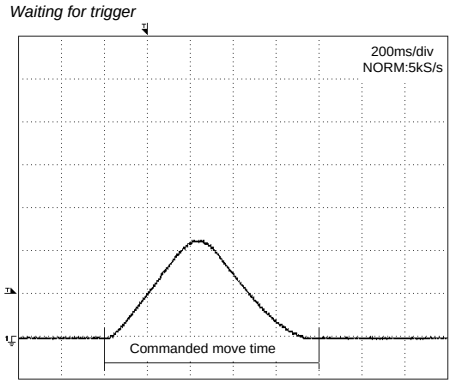
[Acceleration: s-shaped, deceleration: constant acceleration]

[Related parameters]

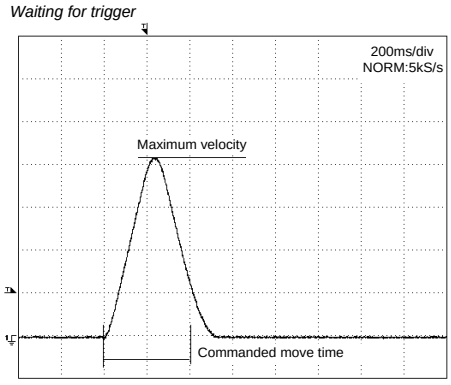
#3	Selecting the type of cam profile move
#4	Selecting the acceleration type
#5	Selecting the deceleration type
#7	Acceleration time during a trapezoidal move
#8	Deceleration time during a trapezoidal move

5.5.3 Cam Profile Move

In a cam profile move, the acceleration/deceleration is performed according to the cam profile set *by #6 Selection of the cam profile* parameter.
If the *#84 Enabling the peak velocity during a cam profile move* parameter is enabled during a cam profile move, the moving part is accelerated/decelerated so that the maximum velocity during the move becomes the commanded feed velocity. If it is disabled, the moving part is accelerated/decelerated so that the move time becomes the value of the *#79 Move time during a cam profile selection* parameter. At this point, if the move time setting is short enough to make the maximum velocity during the move exceeds the maximum velocity, the move time is extended so that the maximum velocity during the move becomes equal to the maximum velocity.
The cam profile can be chosen from eight standard built-in cam profiles, as well as from eight user registered cam profiles. Refer to Chapter 7, “M2 PC Utility” for how to create and register a user registered cam profile.

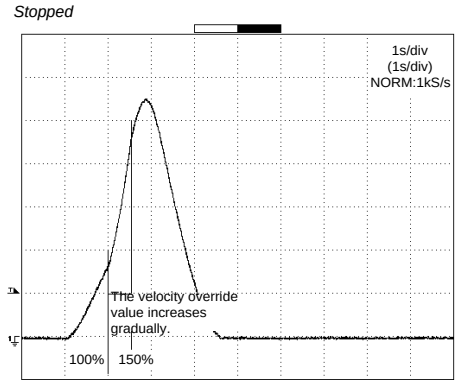


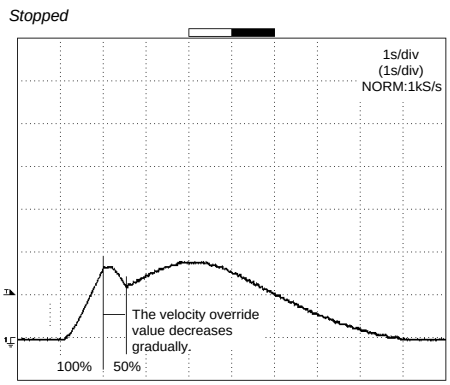
[Cam profile specified by move time]

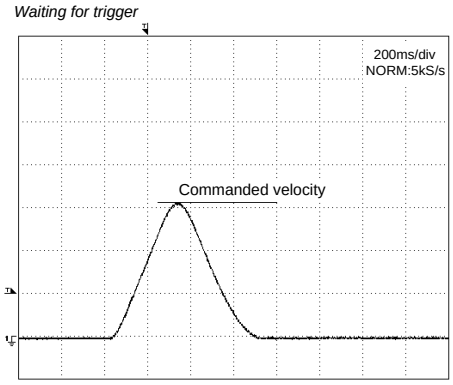


[Cam profile specified by move time]
Limited by the maximum velocity

The velocity override value acts as a ratio by which the elapsed time in the acceleration/deceleration control is multiplied. In other words, if the commanded move time is 1000msec and the velocity override value is 200% the move is finished at 500msec, and if the velocity override value is 50% the move is finished at 2000msec.
If the velocity override value is changed during a move, the velocity override value within the driver to which the acceleration/deceleration control refers is updated gradually. If the velocity override value is changed to a bigger value, the value of the *#80 Acceleration time during a cam profile selection* parameter is used to update the velocity override value within the driver. If it is changed to a smaller value, the *#81 Deceleration time during a cam profile selection* parameter is used. Since both values are defined as a time equivalent by which to change the velocity override value at 100%, the actual change time changes according to the amount of velocity override change.



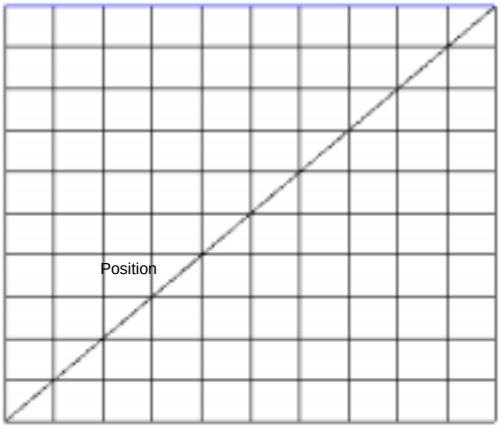




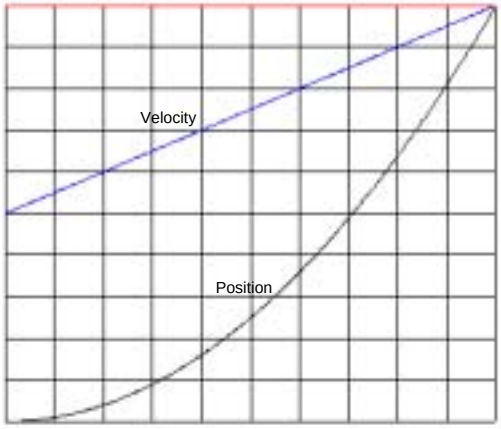
[Cam profile specified by peak velocity]

[Standard built-in cam profile]

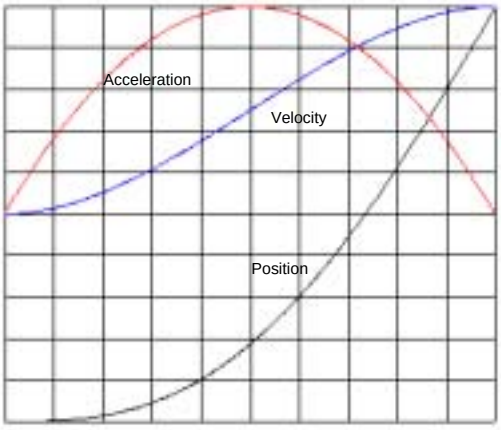
Number	Name	Non-dimensional maximum velocity Vmax	Non-dimensional minimum velocity Vmin	Non-dimensional starting velocity Vs	Non-dimensional terminal velocity Ve	Non-dimensional maximum acceleration Amax	Non-dimensional minimum acceleration Amin
1	Constant velocity	1.000000	1.000000	1.000000	1.000000	0.000000	0.000000
2	Constant acceleration	2.000000	0.000000	0.000000	2.000000	2.000000	2.000000
3	Second order spline	2.000000	0.000000	0.000000	2.000000	3.000000	0.000000
4	Sinusoidal curve	1.570796	0.000000	0.000000	0.000000	4.934803	-4.934803
5	Cycloid	2.000000	0.000000	0.000000	0.000000	6.283184	-6.283184
6	Trapecroid curve	2.182160	0.000000	0.000000	0.000000	6.170441	-6.170441
7	Modified sinusoidal curve	1.759603	0.000000	0.000000	0.000000	5.527958	-5.527958
8	Modified trapezoid	2.000000	0.000000	0.000000	0.000000	4.888123	-4.888123



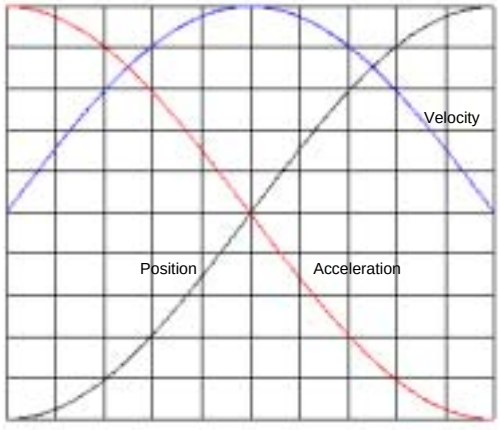
[1 Constant velocity]



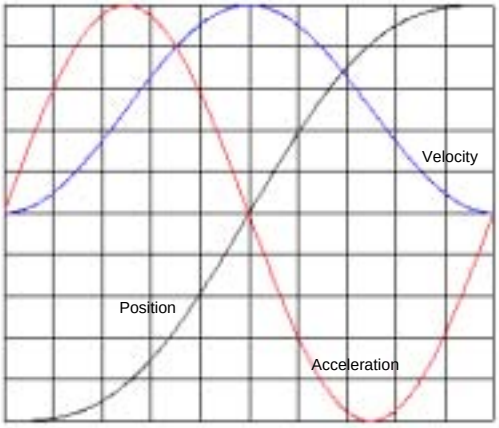
[2 Constant acceleration]



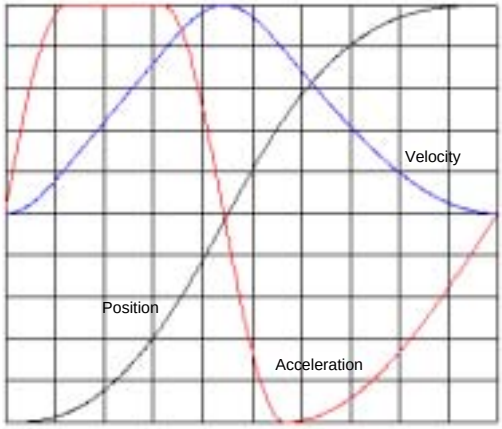
[3 Second order spline]



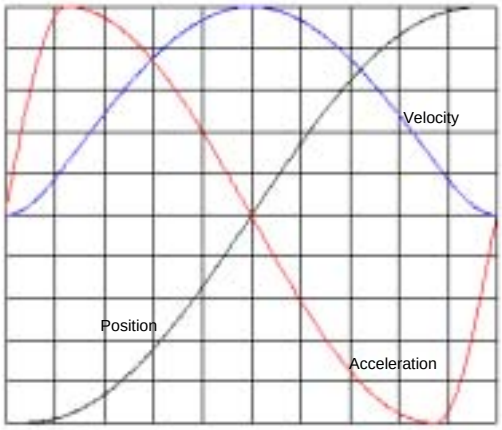
[4 Sinusoidal curve]



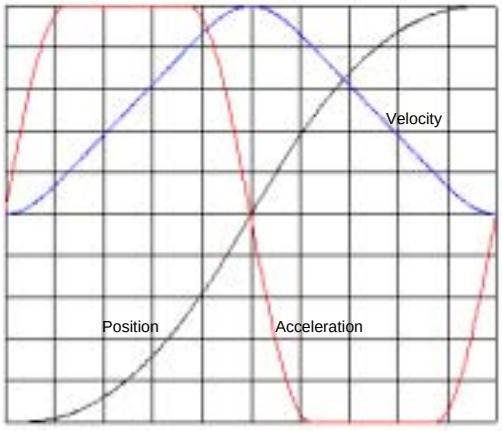
[5 Cycloid]



[6 Trapecroid curve]



[7 Modified sinusoidal curve]



[8 Modified trapezoid]

[Related parameters]

#3	Selecting the type of cam profile move
#6	Selection of the cam profile
#79	Move time during a cam profile selection
#80	Acceleration time during a cam profile selection
#81	Deceleration time during a cam profile selection
#84	Enabling the peak velocity during a cam profile move

5.6 Programming Language

The programming language is a language used by program operation and MDI operation. In this section, the language specification is explained.
The programming language statements are classified into four types: *NC executable statements*, *parameter statements*, *control parameters*, and *comment statements*. Among them, the control parameters and comment statements cannot be used in MDI operation.

5.6.1 NC Executable Statements

An NC executable statement expresses an action with a combination of one English character and one or more numerical values. In addition to direct numerical values, the numerical value part of the statement may contain parameter/monitor variables referred to by #***.
All the English characters may only appear once in one line. Furthermore, only one of the characters "*F*," "*A*," "*f*," or "*a*" may appear in one line.

(1) [G] preparation function

Group 0 (not stored)

- G04 dwell
  Waits for the amount of time specified.
  Usage → G04f*** (*** is the dwell time [msec])

- G27 hardware homing
  Performs a homing operation.
  Usage → G27X

- G28 software homing
  Moves to the home position of the operation unit coordinate system.
  Usage → G28X

- G92 coordinate system setting
  Sets the operation unit coordinate value to a specified value.
  Usage → G92X*** (*** is the operation unit coordinate value)

Group 1 (stored)

- G00 positioning
  Sets the operation unit coordinate value to a specified value.
  Usage → G00X*** (*** is the positioning value, the target value in an absolute instruction and the relative value in an incremental instruction)

Group 2 (stored → saved in *#83 ABS in program/MDI operation*)

- G90 absolute instruction
  Sets the operation to absolute instruction mode
  Usage → G90

- G91 incremental instruction
  Sets the operation to incremental instruction mode
  Usage → G91

(2) [X] axis specification

Specifies an axis. The expression method changes depending on the preparation function.
Usage

→	Positioning	X*** (*** is the positioning value [operation unit])
	Coordinate system setting	X*** (*** is the operation unit coordinate value [operation unit])
	Hardware homing, software homing	X (simply an axis specification)

(3) [F, A] velocity specification

Specifies the feeding velocity. The *#9 feeding velocity* parameter is changed. Also, when a cam profile move is selected in *the #3 selecting the type of cam profile move* parameter, *#84 enabling the peak velocity during a cam profile* parameter is changed to enabled.
Usage → F*** (*** is the feeding velocity [command unit/sec])
A*** (*** is the feeding velocity [command unit/sec])

(4) [f, a] time specification

Specifies time. In a positioning move, the *#79 move time during a cam profile selection* parameter is changed. Also, when a cam profile move is selected in the *#3 selecting the type of cam profile move* parameter, *the #84 enabling the peak velocity during a cam profile* parameter is changed to disabled. Moreover, only f can be used in the dwell status.
Usage → f*** (*** is time [msec]), a*** (*** is time [msec])

(5) [M] signal code for external devices

Specifies the value of the M function, which is used as a signal code for external devices. M code 0 and 1 have special meaning.
M code 0 Program stop
M code 1 Optional stop
Usage → M*** (*** is M code 0 to 99)

(6) [P] program call

This statement calls other programs. The nesting depth (i.e., the number of other programs called from a program) can be up to 15.
Usage → P*** (*** is program number between 0 and 99)

The following shows a program example using only NC executable statements.

```
G27X            Hardware homing
G04f1000        1000msec wait
G92X90000       Sets the current position to operation unit coordinate value 90000.
G90             Sets the operation to absolute instruction mode.
X180000F90000   Positioning move at a velocity of 90000 to operation unit coordinate value 180000
G91X-180000     Switches to incremental instruction mode, and make a positioning move to a position -180000 from the current
                position.
F#400           Sets the feeding velocity to the value set in #400.
G90X#401        Switches to absolute instruction mode, and make a positioning move to the position given by the value in #401.
G04f#400        Waits for a time equal to the value in #401.
G28X            Software homing
```

5.6.2 Parameter Statements

In a parameter statement, it is possible to assign numerical values and variables (*simple assignment statement*) and assign results of arithmetic operations on numerical values and variables (*calculation result assignment statement*) to parameters. The left side must be a variable. The right side can contain numerical values or variables such as values in parameters/monitors referred to by #***.

[Simple assignment statement]

#400 = 100	Assigns 100 to variable #400.
#400 = #401	Assigns the value stored in #401 to variable #400.

[Calculation result assignment statement]

The following operands can be used:

+	Addition
-	Subtraction
*	Multiplication
/	Division
%	Remainder at integer division

#400 = 100 + 20	Assigns the result of adding 100 and 20 to variable #400.
#400 = #401 - 20	Assigns the value obtained by subtracting 20 from the value stored in #401 to variable #400.
#400 = 100 * #401	Assigns the result of multiplication of 100 and the value stored in #401 to variable #400.
#400 = #400/#401	Assigns the value obtained by dividing the value stored in #400 by the value stored in #401 to variable #400.
#400 = 100%30	Assigns the remainder of 100 divided by 30 to variable #400.

5.6.3 Control Parameters

Control parameters control the entire flow of a program. By using these statements, branches (IF statements) and loops (WHILE and FOR statements) can be built into the program.
The following relational operators can be used.

<	The left-side value is smaller than the right-side value.
<=	The left-side value is equal to or smaller than the right-side value.
>	The left-side value is greater than the right-side value.
>=	The left-side value is equal to or greater than the right-side value.
==	The left-side value is equal to the right-side value.
!=	The left-side value is not equal to the right-side value.

The maximum nesting depth that can be achieved with the control parameters is 15.

(1) [IF statements] IF, ELSEIF, ELSE, END

The format is as shown below. The variable part must be a variable referred to by #***. The relational operator part must be one of the relational operators described above. The comparison data part can be numerical values or variables referred to by #***.
Note that the ELSEIF and the ELSE statements may be omitted.

IF:variable:relational operator:comparison data
ELSEIF:variable:relational operator:comparison data
ELSE
END

[Example]

```
IF: #400: > : 5
    G90
    X1000
ELSEIF: #400: > = : #401
    G91
    X2000
ELSE
    G28X
END
```

Execute this part If the value of #400 is greater than 5.

Execute this part If the previous IF statement was not satisfied and the value of #400 is greater than or equal to the value of #401.

Execute this part if neither the previous IF statement nor the ELSEIF statement was satisfied.

(2) [WHILE statements] WHILE, END

The format is as shown below. The variable part must be a variable referred to by #***. The relational operator part must be one of the relational operators described above. The comparison data part can be numerical values or variables referred to by #***.
The WHILE statement performs a condition check and executes the block if the condition is true. After the block is executed, it checks the condition again. If the condition remains true, it executes the block again. If the condition is not true, it jumps to the next line after the END statement.

WHILE:variable:relational operator:comparison data
END

[Example]

```
#400 = 0
WHILE: #400: < : 7
    X1000
    M10
    #400 = #400 + 1
END
```

Executes this part if the value of #400 is smaller than 7 (if it is not smaller, jump to the next line after END).

Checks the condition again (jump to the WHILE statement).

(3) [FOR statements] FOR, END

The format is as shown below. The variable part must be a variable referred to by #***. The relational operator part must be one of the relational operators described above. The initial value, comparison data, and step number parts can be numerical values or variables referred to by #***.
In a FOR statement, the initial value is assigned to the variable first. Secondly, a condition check is performed and the block is executed if the condition is true. After executing the block, the step number is added to the variable and the condition is checked again. If the condition remains true, the block is executed again. If the condition is not true, the program jumps to the next line after the END statement.

FOR:variable:initial value:relational operator:comparison data:number of steps
END

[Example]

```
FOR: #400: 0: < : 7:1
    X1000
    M10
END
```

This example expresses the WHILE statement example shown in the previous example using a FOR statement instead.

5.6.4 Comment Statements

Comment statements are used to explain a program. Note that only half-width characters can be used. Up to 40 characters can be registered.
One comment statement can form one line by itself, or it can be attached after an NC executable statement, parameter statement, or control parameter.

[’]

All characters after a single quotation mark [’] are treated as comments.

[Example]

```
'This is comment.
G90                    'This is comment.
#400=100   'This is comment.
```

An independent comment statement.
A comment statement combined with an NC executable statement.
A comment statement combined with a parameter statement.

5.6.5 Explanation of Built-in Programs

The driver contains the 10 built-in programs numbered 90 to 99. This part is fixed and cannot be overwritten. In this section, the built-in programs are introduced and explained so that you may use to reference them as program examples.

90	In case of a rotating motor	Index Type A demo.
	In case of a linear motor	Reciprocating operation demo.
91	In case of a rotating motor	Reserved
	In case of a linear motor	Reciprocating operation demo.
92 to 94	(Reserved)	
95	Coordinate system clear	
96	Store the current position	
97	Move to a stored position	
98 to 99	(Reserved)	

(1) No. 90 (for a rotating motor) index Type A demo.

This program is executed when #210 (scaling data pulse count) remains as the initial value and a rotational coordinate system is used.
The operation unit is set to index Type A and divided into eight (equal divisions).
It moves the slider step by step in the positive direction, eight times in total, and then two steps at a time in the negative direction four times in total. This process is repeated five times.
The positioning moves are performed as cam profile (trapecroid) moves. After each positioning, the slider dwells for 500ms or 200ms.

```
                          'Rotating motor: index Type A demo.
IF:#210:==:#355           'Only when the scaling pulse value is the same as the resolution
  IF:#212:==:0            'Only in rotational coordinates
    #109=8                'Set to Index A 8 divisions
    #108=1                'Set the operation unit to index A
    #3=1                  'Cam profile move
    #6=6                  'Trapecroid
    G92X0                 'Clear the coordinate system
    FOR:#549:0:<:5:1      'Repeat 5 times
      FOR:#548:1:<=:#109:1 'Step by step in + direction
        G90X#548f500      'Move for 500ms
        G04f500           'Dwell for 500ms
      END
      #547=#109-2         'Store 6 in #547
      FOR:#548:#547:>=:0:-2 'Two steps at a time in - direction
        G90X#548f1000     'Move for 1000ms
        G04f200           'Dwell for 200ms
      END
    END
  END
END
```

(2) No. 90 (for a linear motor) reciprocating operation demo.

This program is executed when #210 (scaling data pulse count) remains as the initial value. Make sure not to execute the program until after setting the slider at the center of the stroke.
It moves the motor 10cm in the positive direction twice and returns to the starting position, then moves 10 cm in the negative direction twice and returns to the starting position. This process is repeated five times.
The positioning moves are performed as trapezoid moves. After each positioning, the motor dwells for 200 ms, 500 ms or 1s.

```
                    'Linear motor: demo
 IF:#210:==:#355    'Only when the scaling pulse value is same as the resolution
   #108=0           'Set the operation unit to index A
   #3=0             'Trapezoidal move
   #9=#213/10       'The moving velocity is 10% of the maximum velocity
   #546=#324/10     'Value equivalent to 10cm
   #547=0-#546      'Value equivalent to -10cm
   G92X0            'Clear coordinate system
   FOR:#549:0:<:5:1 'Repeat 5 times
     G91X#546       '10cm INC move
     G04f200        'Dwell for 200ms
     G91X#546       '10cm INC move
     G04f200        'Dwell for 200ms
     G90X0          'To the original position
     G04f500        'Dwell for 500ms
     G91X#547       '-10cm INC move
     G04f200        'Dwell for 200ms
     G91X#547       '-10cm INC move
     G04f200        'Dwell for 200ms
     G90X0          'To the original position
     G04f1000       'Dwell for 1s
   END
 END
```

(3) No. 91 (for a linear motor) reciprocating operation demo.

This program is executed when #210 (scaling data pulse count) remains as the initial value. Make sure not to execute the program until after setting the slider at the center of the stroke.
It moves the motor 10 mm in the positive direction twice and returns to the starting position, then moves 10 mm in the negative direction twice and returns to the starting position. This process is repeated five times.
The positioning moves are performed as trapezoid moves. After each positioning, the motor dwells for 200 ms, 500 ms or 1s.

```
                    'Linear motor: demo
 IF:#210:==:#355    'Only when the scaling pulse value is same as the
resolution
   #108=0           'Set the operation unit to command unit
   #3=0             'Trapezoidal move
   #9=#213/10       'The moving velocity is 10% of the maximum velocity
   #546=#324/10     'Value equivalent to 10cm
   #547=0-#546      'Value equivalent to -10cm
   G92X0            'Clear coordinate system
   FOR:#549:0:<:5:1 'Repeat 5 times
     G91X#546       '10cm INC move
     G04f200        'Dwell for 200ms
     G91X#546       '10cm INC move
     G04f20         'Dwell for 200ms
     G90X0          'To the original position
     G04f50         'Dwell for 500ms
     G91X#547       '-10cm INC move
     G04f200        'Dwell for 200ms
     G91X#547       '-10cm INC move
     G04f20         'Dwell for 200ms
     G90X0          'To the original position
     G04f1000       'Dwell for 1s
   END
 END
```

(4) No. 95 coordinate system clear

This program sets the current position to the home position of the operation unit coordinate system.
It can be operated regardless of the value of the operation unit selection (#108).

```
                    'Clear coordinate system
G92X0
```

(5) No. 96 store current position

This program stores the current command unit command value to the storage variable #449.

```
                    'Store command unit command value
#449=#323
```

(6) No. 97 move to a stored position

This program moves to the position stored in program No. 96.
It can be executed only when the operation unit selection (#108) is set to command units.

```
                    'Restore stored position
IF:#108:==:0        'Only when the operation unit is 0
  G90X#449          'Move to the position stored in #449
END
```

5.7 M Function

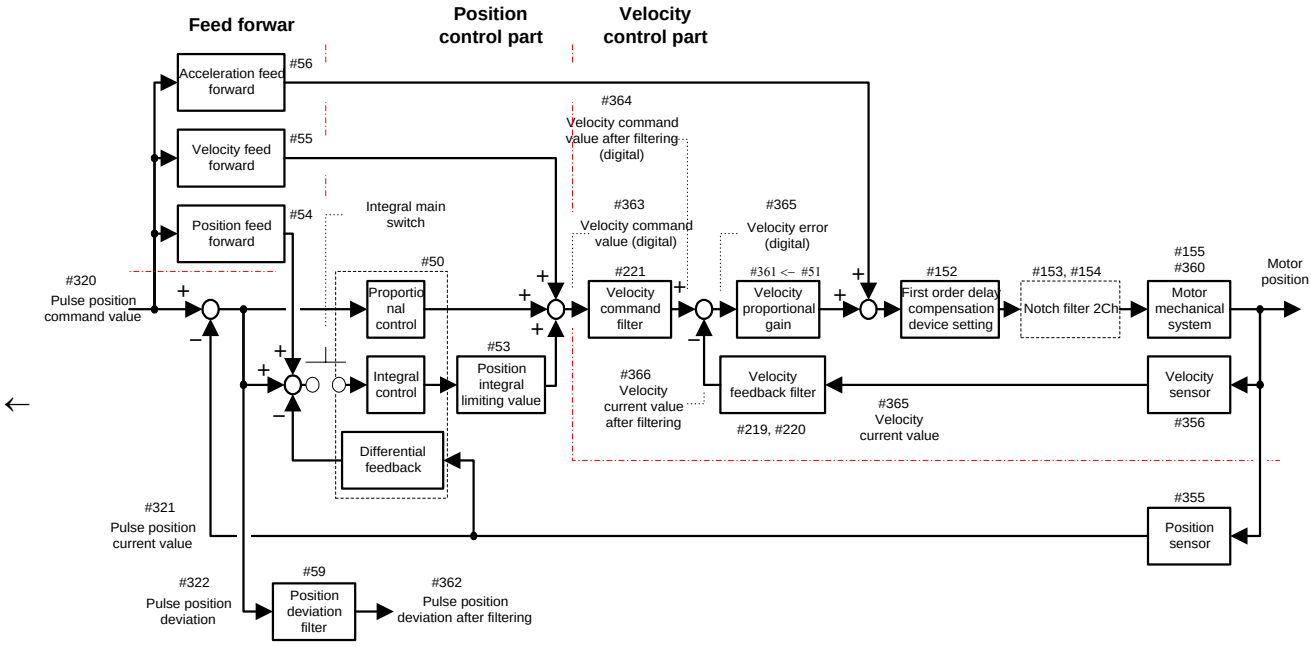
The M function expresses the execution status of the driver in numerical values called *M codes*, and notifies them to the above layer (RS232C interface and PLC interface). The M function can be used in *program operation*, *MDI operation*, *index Type A operation*, and *index Type B operation*.
In program operation and MDI operation, it is possible by giving a command of the form “M**” (where ** is a numerical value or variable) in an NC executable statement, to notify the numerical value or variable value expressed as ** to the above layer as an M code. Values from 0 to 99 can be notified. M codes 0 and 1 have special meanings in program operation. M code 0 has a *program stop* function and M code 1 has an *optional stop* function. For the detail of each, refer to Section 5.3.5, “Program Operation.”
In index Type A and B operations, the operation command value at the positioning location is notified to the top as an M code.

[Related parameters]

#100	M function enable in program/MDI operation
#101	M function enable during index operation
#102	Enabling the selection of RS232C for the M function interface
#103	Optional stop enable

5.8 Control System

In this section, the position control part, velocity control part, and feed forward of the driver are explained. The block diagram of the control system is shown below.

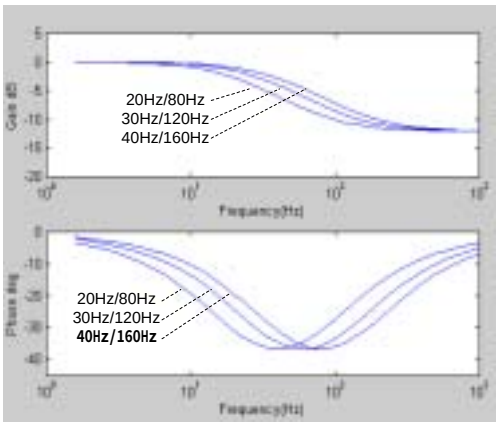


5.8.1 Velocity Control Part

The velocity control part calculates the *#361 Velocity proportional gain* monitor value from the *#155 Load inertia/load mass* parameter value, which is measured and set by the auto-tuning operation or set directly as a numerical value, according to the *#51 Velocity control bandwidth* parameter value. At this point, the notch filter frequency characteristics are not considered but the frequency characteristics of the velocity feedback filter and first order delay compensation device are considered in the calculation of the velocity proportional gain.
The first order compensation device can be set in four ways via the *#152 First order delay compensation device setting* parameter. The first order delay compensation device has the effect that it improves the gain characteristics of the velocity control part. The frequency characteristics for each setting of the first order compensation device is shown below. Note that there is no frequency dependency when the first order compensation is not set.
There are two channels available for the notch filter (optional). The central frequency value can be set independently for each channel via the *#153 Notch filter: Frequency selection 1* and *#154 Notch filter: Frequency selection 2* parameters. The notch filter has the effect that it improves the gain characteristics of mechanical systems that tend to resonate.
The velocity feedback filter can be enabled or disabled by setting the *#219 Enable/Disable velocity feedback filter* parameter. When enabled, the filter bandwidth can be set in the *#220 Velocity feedback filter bandwidth* parameter. The velocity feedback filter is effective in eliminating noise at motor operation, but with a small bandwidth the velocity control part tends to oscillate.
The filter bandwidth of the velocity command filter can be set in the *#221 Velocity command filter bandwidth* parameter. The velocity command filter helps achieving smoother acceleration/deceleration, but with a small bandwidth the position control part tends to oscillate.

[Related parameters]

#51	Velocity control bandwidth
#152	First order delay compensation device setting
#153	Notch filter: Frequency selection 1
#154	Notch filter: Frequency selection 2
#155	Load inertia/load mass
#219	Enable/Disable velocity feedback filter
#220	Velocity feedback filter bandwidth
#221	Velocity command filter bandwidth



[Frequency characteristics of the first order delay compensation device]

5.8.2 Position Control Part

The position control part calculates the proportional control gain, integral control gain, and differential feedback gain based on the *#50 Position control bandwidth* parameter value.
The position integral limiter set by the *#53 Position integral limiting value* parameter is effective in suppressing integrator windup due to motor torque and thrust saturation. Decrease the value in order to suppress windup further. On the other hand, if you set the value too small, the motor torque and thrust are limited; set a value as large as possible in the range where windup does not occur.
The integral operation of the position control part can also be set either to be active (allow) or inactive (prohibit). If you clamp the motor with an external device when the motor is stopped, the integral operation should be prohibited after clamping in order to prevent overloading the controller. Refer to Section 6.1, “RS232C Interface” and Section 6.2, “PLC Interface” for details of the operation.
Though not related directly to the position control part, the position deviation filter outputs the result of filtering the position deviation to the *#362 Pulse position deviation after filtering* monitor. The filter bandwidth can be set in the *#59 Position deviation filter frequency* parameter. The filter output can be used as a signal source when generating a position settling status signal. For details about this, refer to Section 5.9.1, “Settling Wait, Position Settling Status, and Positioning Status.”

[Related parameters]

#50	Position control bandwidth
#53	Position integral limiting value
#59	Position deviation filter frequency

5.8.3 Feed Forward

Three types of feed forward functions are available: position feed forward, velocity feed forward, and acceleration feed forward. Feed forward function is effective for quick positioning.
The position feed forward can be set by percentage in the *#54 Position feed forward percentage* parameter. The position feed forward makes the position deviation at equal velocity move smaller and helps to achieve a smooth settling at acceleration/deceleration.
The velocity feed forward can be set by percentage in the *#55 Velocity feed forward percentage* parameter.
The acceleration feed forward calculates the acceleration feed forward gain from the *#155 Load inertia/load mass* parameter values, which are measured and set by the auto-tuning operation or set directly by numerical values, based on the *#56 Acceleration feed forward gain* parameter. #56 parameter is given as a percentage.

[Related parameters]

#54	Position feed forward percentage
#55	Velocity feed forward percentage
#56	Acceleration feed forward percentage
#155	Load inertia/load mass

5.8.4 Servo Stiffness Parameter

The *#38 Servo stiffness settings* parameter is for general settings for the control system. The control parameters are set based on this parameter after measuring the load inertia/load mass in the auto-tuning operation.
If #38 is changed, the position control bandwidth (#50) and the velocity control bandwidth (#51) are set, so the gain of the control system is automatically set. The position integral limiting value (#53) is also automatically set to the minimum position integral limiting value that can generate the maximum torque and maximum thrust in motor lock status. It is not necessary to set parameters related to feed forward and filter again.

[Related parameters]

#38	Servo stiffness settings
#50	Position control bandwidth
#51	Velocity control bandwidth
#53	Position integral limiting value

5.9 Other Functions

5.9.1 Settling Wait, Position Settling Status, and Positioning Status

Position settling status refers to the status where the position deviation (in pulse coordinates) is within the prescribed range in the pulse coordinate system. The prescribed range is set by *#58 Position settling width* in command units. The *#362 Pulse position deviation after filtering* monitor value or the *#322 Pulse position deviation* monitor value are used for the evaluated position deviation depending on whether the *#60 Enable/Disable filter at position settling signal generation* parameter is set to enable or disable, respectively. Position settling status is displayed in the *#328 Position settling status* monitor. The "COIN": position settling status LED on the front panel also turns on and indicates the position settling status. In addition, it is output to "COIN" of the CN3 analog monitor interface as a digital signal.
Positioning status is the status where an acceleration/deceleration command for a move is completed and in position settling status. The positioning settling status is displayed in the *#329 Positioning status* monitor.
Settling wait is a function invoked to keep on waiting until the positioning status is reached at the end of a move. Settling wait can be made in three ways depending on the operation: never perform settling wait, always perform settling wait, or follow the setting of the *#106 Settling wait enable* parameter.

	Operation
Never perform	Jog move, test operation, auto-tuning operation
Always perform	Homing move
Follow #106 setting	Program operation, signal search move, MDI operation, index Type A operation, index Type B operation, table reference operation

[Related parameters]

#58	Positioning settling width
#60	Enable/disable filter at position settling signal generation
#106	Settling wait enable

5.9.2 Velocity Monitor and Analog Monitor

The current velocity value of the motor is output to "VEL" of the CN3 analog monitor interface as the *velocity monitor* signal.
The signal sensitivity of the velocity monitor ([V/rps] in case of a rotating DYNASERV motor and [V/mps] in case of a linear LINEARSERV motor) can be obtained by multiplying the *#69 Velocity monitor gain* parameter setting by the *#356 Digital velocity sensitivity* monitor value. For example, if the setting of #69 is 6.55V/8192 digits and the value of #356 is 4800 [digit/rps] in a rotating motor, the signal sensitivity of the velocity monitor is 3.84 [V/rps]. The velocity monitor signal is output in the range of ± 6.55V.
Moreover, it is possible to output only the AC element of the current velocity value of the motor via the setting of the *#75 Velocity monitor selection* parameter.
One of the data items below is output to "AMON" of the CN3 analog monitor interface as an *analog monitor* signal.
The content of the analog monitor can be selected by the *#70 Analog monitor selection* parameter. Depending on the selected content, the signal sensitivity is adjusted using the corresponding monitor gains #71 to 74. The analog monitor signals are output in the range of ± 6.55V.

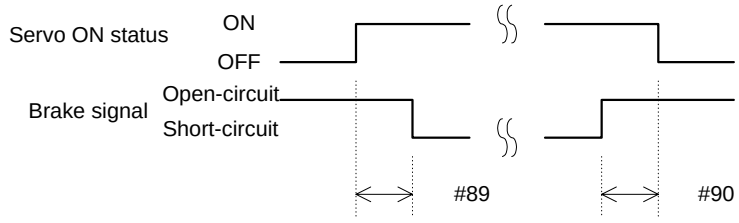
Analog monitor selection	Unit	Monitor gain
Position deviation	Pulse	#71 Positioning error monitoring gain (Analog monitor)
Test operation response	Pulse	#72 Test operation monitoring gain (Analog monitor)
Position command value	Pulse	#73 Position monitoring gain (Analog monitor)
Position current value	Pulse	
Position command differential value	pps	#74 Position differential value monitor gain (Analog monitor)
Position current differential value	pps	

[Related parameters]

#69	Velocity monitoring gain
#70	Analog monitor selection
#71	Positioning error monitoring gain (Analog monitor)
#72	Test operation monitoring gain (Analog monitor)
#73	Position monitoring gain (Analog monitor)
#74	Position difference value monitor gain (Analog monitor) #74
#75	Velocity monitor selection

5.9.3 Brake Signal

BRK+ and BRK- are output to the TB2 external sensor interface as brake signal outputs that operate linked to the Servo ON status. The brake signal is a relay contact output. The contact opens when the brake should be applied such as when the power is disconnected or the driver is in Servo OFF status, and the contact short circuits when the brake should be released.
The Servo ON status and the brake signal operate according to the timing diagram shown below via the *#89 Brake turn OFF delay time upon Servo ON* and #90 *Advanced brake turn ON before Servo OFF* parameters.

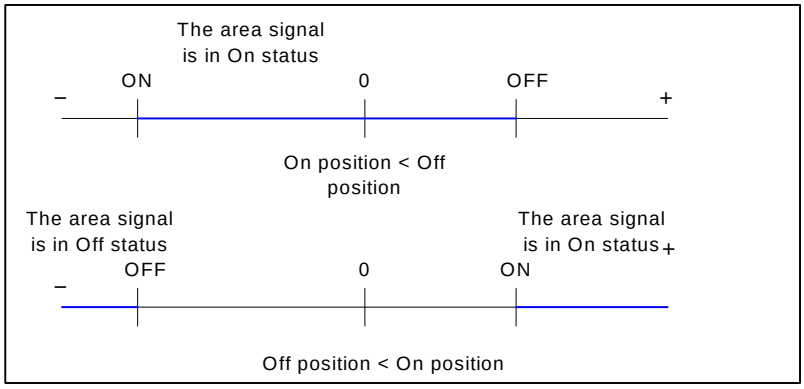


[Related parameters]

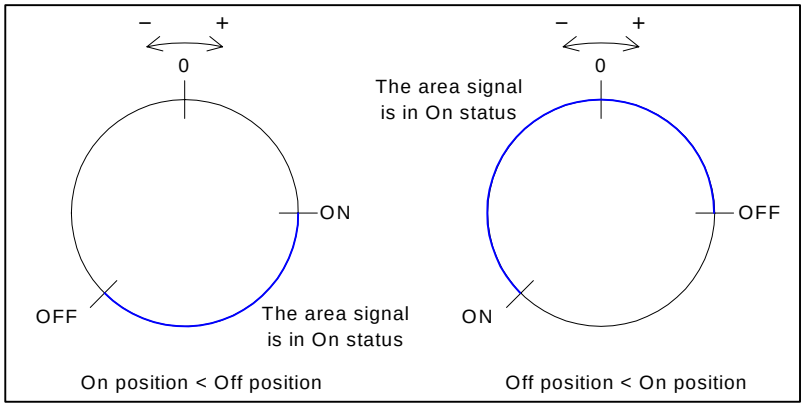
#89	Brake turn OFF delay time upon Servo ON
#90	Advanced Brake turn ON before Servo OFF

5.9.4 Area Signals

The area signals display whether or not the motor is within the pre-set range. Two channels are available.
The range is set by setting the coordinate value at which to turn on the signal in the *#161 Area signal 0 On* and *#163 Area signal 1 On* parameters in command units. In addition the coordinate value at which to turn off the signal in the *#162 Area signal 0 Off* and *#164 Area signal 1 Off* parameters in command units. How the signal is output with respect to the motor position depends on whether the coordinate value to turn on is greater or smaller than the coordinate value to turn off. The figure below shows each case in linear coordinate and rotational coordinate systems.
The status of the area signal is displayed in *#331 Area signal 0 status* and *#332 Area signal 1 status* monitors. It is also output to the PLC interface. Refer to Section 6.2, “PLC Interface.”



[Linear coordinate system]



[Rotational coordinate system]

[Related parameters]

#161	Area signal 0 On
#162	Area signal 0 Off
#163	Area signal 1 On
#164	Area signal 1 Off

5.10 Special Parameter Processing

The setting values of the parameters listed below are automatically changed inside the driver by the operation of the driver.

5.10.1 Internal Generation of Parameter Initial Values

The initial values of all of the following parameters when they are reset will be generated internally by the motor:

[Related parameters]

#9	Feeding velocity	Value obtained by converting the motor rating velocity into command units.
#10	Jog velocity	Value obtained by converting the motor rating velocity into command units.
#11	Over-travel search velocity during a homing move	Value obtained by converting the motor rating velocity∗0.1 into command units.
#12	Homing operation: Home sensor proximity signal search velocity	Value obtained by converting the motor rating velocity∗0.1 into command units.
#15	Homing operation: Origin position offset move feed velocity	Value obtained by converting the motor rating velocity∗0.1 into command units.
#31	Operation width under testing mode	Value obtained by converting the motor rating velocity∗0.002 into command units.
#32	Operation width under Auto-tuning	Value obtained by converting the motor rating velocity∗0.02 into command units.
#209	Scaling data (command unit side)	Motor resolution
#210	Scaling data (pulse side)	Motor resolution
#212	Straight line coordinate selection	Select rotation coordinates for rotation type, and straight line coordinates for linear type.
#213	Maximum velocity	Value obtained by converting the motor rating velocity into command units.

5.10.2 Limiting and Checking the Maximum Velocity When Changing Scaling Data and Maximum Velocity Parameter

The following maximum velocity limiting and checking are performed during processing when the power is turned ON after data related to scaling is changed:

Limit item 1) A limit is set when the maximum velocity in command units [unit/s] exceeds 9999999.

2) A data checksum error is generated when the maximum velocity in pulse units [pls/s] exceeds 8000000.

5.10.3 Auto Conversion and Clear Functions When Changing Scaling Data

When scaling data is changed, the following parameters are automatically converted:

[Related parameters]

#9	Feeding velocity
#10	Jog velocity
#11	Over-travel search velocity during a homing move
#12	Homing operation: Home sensor proximity signal search velocity
#15	Homing operation: Origin position offset move feed velocity
#29	Offset distance from the Home position
#31	Operation width under testing mode
#32	Operation width under Auto-tuning
#58	Positioning settling width
#87	(+) direction soft limit settings
#88	(-) direction soft limit settings
#161	Area signal 0 ON
#162	Area signal 0 OFF
#163	Area signal 1 ON
#164	Area signal 1 OFF
#213	Maximum velocity

When coordinates (rotation/straight line) scaling data is changed, the following parameters will be cleared:

[Related parameters]

#161	Area signal 0 ON
#162	Area signal 0 OFF
#163	Area signal 1 ON
#164	Area signal 1 OFF

5.10.4 Limiting the Maximum Parameter Values

Limit processing is performed for the following parameters related to velocity at the time of parameter entry and when the power is turned ON.

[Related parameters]

#9	Feeding velocity
#10	Jog velocity
#11	Over-travel search velocity during a homing move
#12	Homing operation: Home sensor proximity signal search velocity
#15	Homing operation: Origin position offset move feed velocity

[At the time of parameter entry]

If a parameter value exceeds #357 *Maximum velocity monitor value* when it is set by the user, it will be processed as out of range data. However, limit processing is not performed in the hold-on state (while downloading parameters), when changing data related to scaling, or when changing the maximum velocity data.

[When the power is turned ON]

A limit is applied using the value equivalent to *#357 Maximum velocity monitor value* during processing when the power is turned ON after data related to scaling or maximum velocity data is changed.

Chapter 6
Control Interfaces

6.1 RS232C Interface

6.1.1 Overview
6.1.2 Connection and Setting
6.1.3 Communication Specifications
6.1.4 @ Commands
6.1.5 Parameter Commands

6.2 PLC Interface

6.2.1 Overview
6.2.2 Connection, Setting, and I/O Mapping
6.2.3 Operation 1
6.2.4 Operation 2

6.1 RS232C Interface

6.1.1 Overview

The CN1 RS232C communication connector is provided in order to make connection with host devices such as PCs and PLCs via the RS232C. The *operation display pendant* (abbreviated as TBX, optional device) can also be connected to this connector. Refer to Chapter 8, “Operation Display Pendant” for a description of how to use the operation display pendant. This chapter explains how to connect the CN1 RS232C communication connector to devices other than the operation display pendant.
In the RS232C interface, two communication modes are available. The first is a *single channel communication* where the connection is made 1:1 with the host device, and the other is a *multi-channel communication* where 1:N communication can be performed by connecting one host device with several of these drivers (a maximum of nine). Please note that the connection and operation methods are different for each communication mode.
In addition, the *PC utility* (optional) running under Windows can also be connected to the drivers via the RS232C interface in order to support setting, operation, and maintenance work on the drivers. Refer to Chapter 7, “DrvMII PC Utility” for a description of how to use the PC utility.

6.1.2 Connection and Setting

(1) [Connectors and terminal assignment]

Made by Japan Aviation Electronics
DELC-J9SAF13L6 (9 pins)

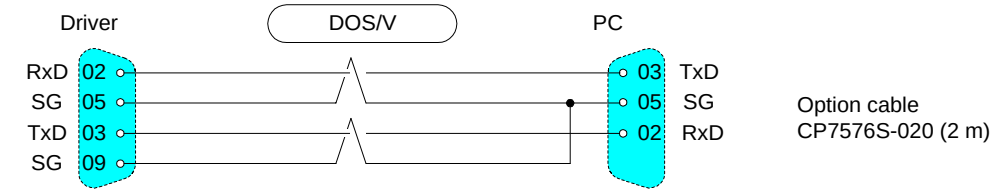
01	FG
02	RxD
03	TxD
04	(NC)
05	SG

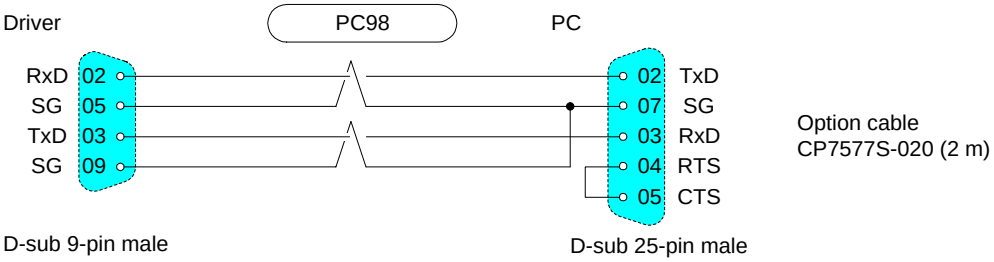
06	XTBXON
07	XTBXEMG
08	+5V
09	SG

Do not use these connections, since they are used for the operation display pendant.

(2) [Single channel]

The connection cables (optional) are available according to the PC to be connected (DOS/V, NEC PC98).

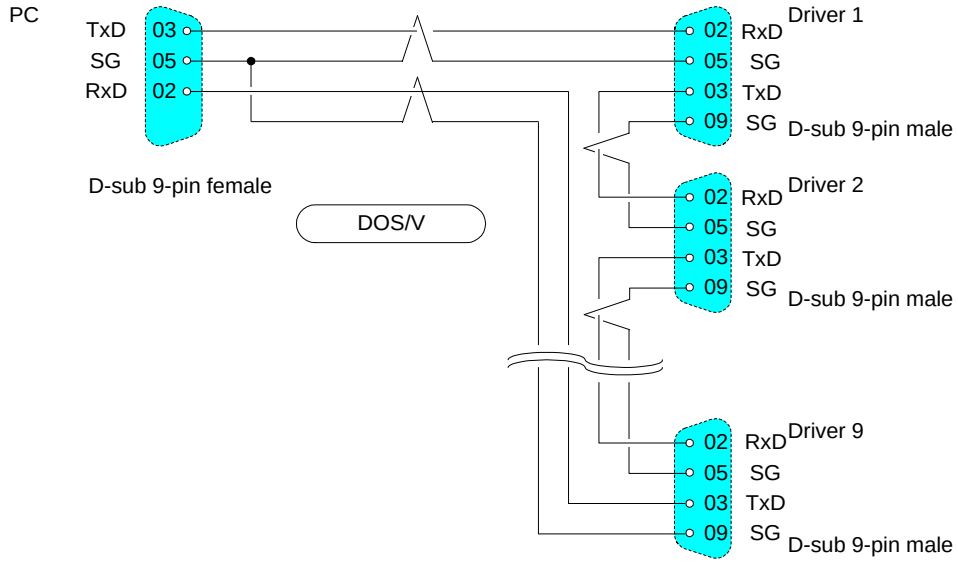


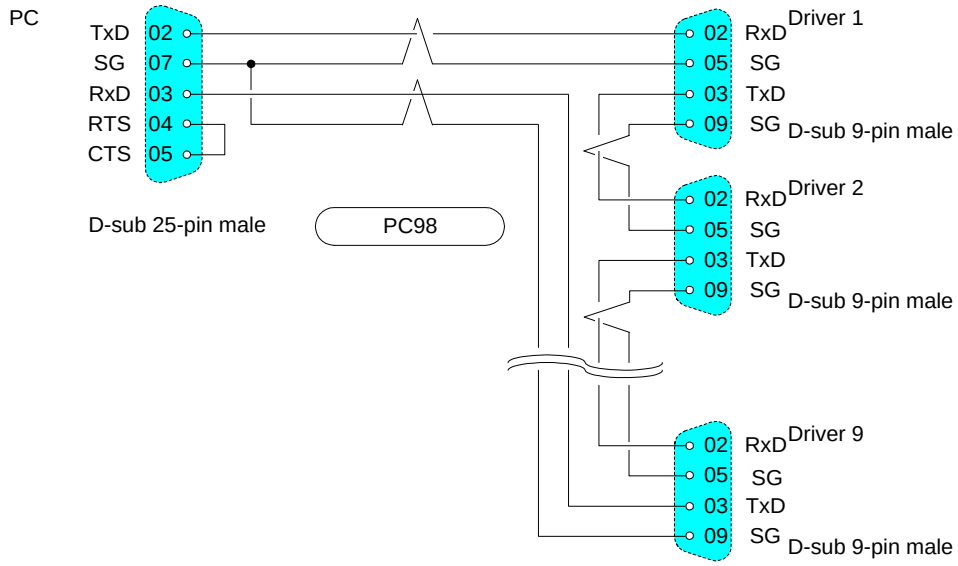


In order to set the communication mode to the single channel mode, the rotary switch RS-ID on the front panel should be turned to “0.” This setting should be made before turning the power on.

(3) [Multi-channel]

When preparing for multi-channel communication, connect the host device and a maximum of nine drives in a loop shape as shown in the figure below.





In multi-channel communication, the host device is assigned to *host ID* "0." The drivers should be set as *slave stations* and the *ID*s should be set for each driver from "1" to "9" in such a way that the station numbers do not overlap, using the rotary switch RS-ID on each driver's front panel. The order of the IDs does not need to be the same as the order of connection. This setting should also be made before turning the power on.

6.1.3 Communication Specifications

(1) [Communication parameters]

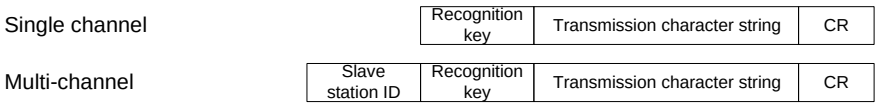
Communication method	Start-stop system, text communication
Communication speed	9600 bps
Stop bit	1 bit
Data length	8 bits
Parity	None
Terminate	CR (both transmission and reception)
Flow control	None

(2) [Single channel and multi-channel]

	Single channel	Multi-channel
Topology	Cross	Ring
ID	Unnecessary	Host device 0 Drivers 1 to 9
Destination specification	Unnecessary	Add the destination ID at the beginning of the packets From the host device to drivers: n****CR (n: slave station ID) From drivers to the host device: 0n****CR (n: slave station ID)

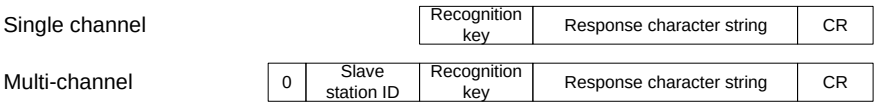
(3) [Transmission from the host device to drivers]

The number of characters to be transmitted should be 128 letters or less, including the slave station ID, recognition key, transmission character string, and CR.



(4) [Response from drivers to the host device]

The number of response characters should be 128 letters or less, including 0, slave station ID, recognition key, transmission character string, and CR.



(5) [Recognition key]

The recognition key is a function provided so that the host device can recognize that a response is a reply to a specific transmission by the host device. A maximum of 15 “!” characters can be included in the recognition key part. If more than 15 are added, the remainder of the number divided by 16 is processed as the actual recognition key number.
When the host device transmits a message to a driver and attaches N recognition keys to the transmission character string, the driver will send a response message back in which it attaches N recognition keys to that transmission character string. If, for instance, the host device issues a transmission character string to a driver that does not generate an immediate response, the host device may issue the next transmission character string before the response is returned. In such cases, it becomes difficult for the host device to recognize to which transmission character string the response character string returned afterward is issued. In this case, by issuing transmission character strings with different recognition key numbers, it becomes possible to judge to which transmission character string a particular response corresponds, simply by obtaining the recognition key number as the response is received.

(6) [Transmission character string]

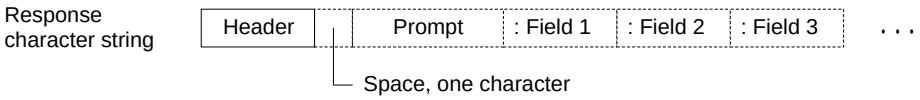
Transmission character strings are classified as follows. The details about the @ commands and parameter commands will be explained separately in Section 6.1.4, "@ Commands" and Section 6.1.5, "Parameter Commands."

	Explanation	Receivable status
@ commands	Commands for operating the driver	Receivable status changes depending on the command.
Parameter commands	Commands for setting parameters and reading parameter/monitor values.	Receivable status changes depending on the parameter.
Parameter statements	Statements for setting parameters in program operation and MDI operation.	Receivable at program registration and during MDI operation.
NC executable statements	Statements for operating the motor in program operation and MDI operation.	
Control parameters	Statements for controlling the flow of a program during program operation.	Receivable only at program registration.
Comment statements	Statements for making the content of a program easy to understand, not related to operation.	

Refer to Section 5.6, "Programming Language."

(7) [Response character string]

A response character string is structured as follows.



Response character strings are classified as follows.

	Header structure	□ part	■ part	Explanation
General	R□■	Number of fields	For the field expression method; see the note.	Normal response character strings to a transmission character string. The number of fields changes depending on the content of the response.
Error	ERR□□.■	Error/alarm code (main)	Error/alarm code (sub)	Response character string to a transmission character string at error and alarm. Fields never exists.
Alarm	ALM□□.■			

Note: In case of a general response the ■ part of the field is expressed in one of the following ways:
- 0: In case there is no field
- D: Character string expressed in decimal
- B: Binary expression (8, 16, 32 digits)
- H: Hexadecimal expression (2, 4, 8 digits)
- S: Character string
- Z: Other than above (character string, etc.)

The following shows some examples of response character strings.

R00
R1D Position control bandwidth: 12
R1B Sensor group signal status: 00010000
ERR30.0 Servo not ready
ALM60.0 Cannot interpret command

6.1.4 @ Commands

Some of the @ commands may not be controlled because of the *operation mode* setting. Refer to Section 5.1, "Operation Mode" for an explanation of the operation mode.

Command format

@	Command number	: Field 0	: Field 1	: Field 2	. . .

Command name	Command No.	No. of fields	Response at normal operation
Abort	1	0	R00
Stop	2	0	
Start	3	Maximum of 4	
Error reset	4	0	
Servo ON/OFF	8	1	
M answer	9	0	
Homing offset position setting	10	1	
Jog move command	11	1	
Integral position control operation switching	12	1	
Coordinate system settings	13	1	

(1) Servo ON/OFF @8: Field 0

The Servo ON/OFF command can be issued via the RS232C interface when the setting of the operation mode has given the main operation authority to the RS232C interface rather than the PLC interface.
The servo goes into the ON status when field 0 contains "1," and OFF when it contains "0." In addition to this command, the actual Servo ON/OFF status is affected by the Servo ON enable/disable command of the PLC interface and the Servo ON disable setting of the SRV DS switch on the front panel. Refer to the table below. Moreover, the initial Servo ON status is determined by the setting of the *#216 Servo ON status upon power up during RS232C operations* parameter.
A response is issued immediately regardless of whether or not the motor actually went into the servo ready status. If the next command requires that the motor is in the servo ready status, make sure to check the *#305 Servo ready* monitor before sending that command.

Servo ON/OFF @8: □	PLC interface Servo ON enable/disable	Front panel SRV DS Servo ON disable	Actual servo status
OFF @8:0	Disabled	Disabled	Servo OFF
		Enabled	
	Enabled	Disabled	
		Enabled	
ON @8:1	Disabled	Disabled	
		Enabled	
	Enabled	Disabled	
		Enabled	Servo ON

[Related parameters]

#216	Servo ON status upon power up during RS232C operations

(2) Start @3: Field 0: Field 1: Field 2: Field 3

Start commands begin operating actions other than jog moves. They can be issued via the RS232C interface when the setting of the operation mode has given the main operation authority to the RS232C interface. They are only executable while in the *idle status*.
The number and content of fields vary depending on the operations to run. They are summarized in the table below. The program operations, index Type A operations, index Type B operations, and table reference operations are explained in detail because it is different how the fields are assigned in these cases.
A response is issued when the corresponding operation is completed. In case of test operations, MDI operations, etc. that do not end by themselves, the next operation can be performed without waiting for the response.

Operating action name	Command	No. of fields
Test operation	@3:0	1
Auto-tuning operation	@3:1	1
Homing operation	@3:3	1
Program operation	@3:4:start-up option 1:start-up option 2	1 to 3
Signal search operation	@3:6	1
MDI operation	@3:7	1
Index Type A operation	@3:8:start-up option 1:start-up option 2: start-up option 3	2 to 4
Index Type B operation	@3:9:start-up option 1:start-up option 2: start-up option 3	2 to 4
Table reference operation	@3:10:start-up option 1:start-up option 2: start-up option 3	2 to 4
Mechanical setting mode	@3:15	1

[Program operation]

Start-up options 1 and 2 provide the program number and block number, respectively. There are three ways to provide a command: *provide neither start-up option 1 nor 2*, *provide only start-up option 1*, and *provide both start-up options 1 and 2*.
The commands that provide only start-up option 1 are the most commonly used commands. The execution begins from the top of the program with the number specified by start-up option 1 and runs from there [@3:4:program number].
In case of a command that does not provide neither start-up option 1 nor 2, the program with the execution program number stored is run from the block with the stored block number [@3:4].
In case of a command that provides both start-up options 1 and 2, the program with the number specified by start-up option 1 is run from the block with the block number specified by start-up option 2 [@3:4:program number: block number].

[Index Type A Operation, Index Type B Operation]

Start-up option 1 provides the index number, start-up option 2 provides the selection of absolute or incremental move, and start-up option 3 provides the moving direction option. There are three ways to provide a command: *provide only start-up option 1*, *provide start-up options 1 and 2*, and *provide start-up options 1, 2 and 3*. Start-up option 1 must always be provided.
Start-up option 1 (index number) becomes the relative value to the current operation command value in case of an incremental move, and the target operation command value in case of an absolute move.
Start-up option 2 (selection of absolute move/incremental move) selects an incremental move when the field contains 0 and an absolute move when the field contains 1. If the *#104 ABS/INC setting during table index operation* parameter is set to anything other than *start-up option dependence*, the specification of start-up option 2 is invalid. When #104 is set to start-up option dependence, start-up option 2 must be set. If start-up option 2 is not set at this point, it is assumed that 0 is set and an incremental move is performed.
Start-up option 3 (moving direction option) selects Type 0 if the field contains 0, Type 1 if the field contains 1, and Type 2 if the field contains 2. If the *#105 Moving direction option for rotational coordinates* parameter is set to anything other than *start-up option dependence*, the specification of start-up option 3 is invalid. When #105 is set to start-up option dependence, start-up option 3 must be set. If start-up option 3 is not set at this point, it is assumed that 0 is set and Type 0 is selected.

Note: Regarding the moving direction in rotational coordinates
Type0 Proximity rotational move (multiple rotations are not allowed).
Type1 Does not cross the rotational coordinate's home position (multiple rotations are not allowed).
Type2 Calculates the target position relative to the current rotational coordinate's home position and determines the direction (multiple rotations are allowed).

[Related parameters]

#104	ABS/INC setting during table index operation
#105	Moving direction option for rotational coordinates

[Table reference operation]

Start-up option 1 provides the table number, start-up option 2 provides the selection of absolute or incremental move, and start-up option 3 provides the moving direction option. There are three ways to provide a command: *provide only start-up option 1*, *provide start-up options 1 and 2*, and *provide start-up options 1, 2 and 3*. Start-up option 1 must always be provided.
The table data value and option are obtained based on the number specified in start-up option 1 (table number). The table data value obtained is set as a value relative to the current operation command value in case of an incremental move, and as the target operation command value in case of an absolute move.
Start-up option 2 (selection of absolute move/incremental move) selects an incremental move when the field contains 0 and an absolute move when the field contains 1. If the *#104 ABS/INC setting during table index operation* parameter is set to anything other than *start-up option dependence*, the specification of start-up option 2 is invalid. When #104 is set to start-up option dependence, start-up option 2 must be set. If start-up option 2 is not set at this point, it is assumed that 0 is set and an incremental move is performed.
Start-up option 3 (moving direction option) selects Type 0 if the field contains 0, Type 1 if the field contains 1, and Type 2 if the field contains 2. If the *#105 Moving direction option for rotational coordinates* parameter is set to anything other than *start-up option dependence*, the specification of start-up option 3 is invalid. When #105 is set to start-up option dependence, start-up option 3 must be set. If start-up option 3 is not set at this point, it is assumed that 0 is set and Type 0 is selected.

Note: Regarding the moving direction in rotational coordinates
Type0 Proximity rotational move (multiple rotations are not allowed).
Type1 Does not cross the rotational coordinate's home position (multiple rotations are not allowed).
Type2 Calculates the target position relative to the current rotational coordinate's home position and determines the direction (multiple rotations are allowed).

[Related parameters]

#104	ABS/INC setting during table index operation
#105	Moving direction option for rotational coordinates

(3) Stop @2

Stop commands are used to end operating actions other than jog moves. They can be issued via the RS232C interface when the setting of the operation mode has given the main operation authority to the RS232C interface. The driver's response to stop commands varies depending on the current operating action. Refer to the table below.
The response is issued immediately.

Operating action name	Driver response
Test operation	Ends the operation when the motor returns to the start position.
Auto-tuning operation	Ends the operation when the oscillation command to the motor is completed.
Homing operation	Immediately decelerate and stop the move, and ends the operation.
Program operation	Ends the operation when a block is executed and completed.
Signal search operation	Immediately decelerate and stop the move, and end the operation.
MDI operation	Ends the operation when execution of the current NC executable statement or parameter statement input via RS232C is completed.
Index Type A operation	Invalid (ignored)
Index Type B operation	Invalid (ignored)
Table reference operation	Invalid (ignored)
Mechanical setting mode	Invalid because this operation cannot be completed (ignored).

(4) Abort @1

The abort command stops operating actions other than jog moves. It functions irrespectively of the operation mode.
Unlike with the stop commands, the motor immediately decelerates and stops, and the operating action is ended even during an operation that involves movement. When the M function is being executed, the abort command stops the execution and ends the operating action.
The response is issued immediately.

(5) Error reset @4

The error reset command cancels error statuses of the driver. It functions irrespectively of the operation mode. It can only be run while in the *idle status*.
Depending on the error content, there are errors that cannot be canceled or errors that cause the same errors again immediately after being canceled.
The response is issued immediately.

(6) Integral position control operation switching @12:field 0

The integral position control operation switching command switches the operation status of integral operation in the position control part. It can be issued via the RS232C interface when the setting of the operation mode has given the main operation authority to the RS232C interface.
When field 0 is "1," integral operation is executed, and when it is "0," integral operation is not executed. When the power is turned on, the integral operation is in the executable status by default.
The response is issued immediately.

(7) Homing offset position setting @10: Field 0

The homing offset position setting command instructs the *#29 Offset distance from the home position* parameter to auto-set so that the current motor position will become the position after homing is completed from the next time. It functions irrespectively of the operation mode. It can only be run while in the idle status.
Field 0 provides the method by which to determine the #29 parameter in a rotational coordinate system. In linear coordinates, the method to determine the parameter is unique, so the setting of field 0 is ignored.
When the command is issued, the current command unit command value and the value of the #29 parameter at that point are added. This value is temporality stored in the #29 parameter. In a linear coordinate system, this temporary value is used as the setting value as it is. In a rotational coordinate system, the temporary set value is processed by the setting of field 0, then stored in the #29 parameter.
In rotational coordinates, the temporary set value is standardized so that it becomes 0 or more and less than the *#324 Scaling data (command unit side)* monitor value. The set value of the #29 parameter is generated from the standardized value and the setting of field 0.

- If field 0 is “0”
  If the standardized value is 0 or more and less than half of the value of #324:
  #29 is set to the standardized value.
  If the standardized value is more than half of the value of #324:
  #29 is set to the value of #324 minus the standardized value.
- If field 0 is “1”: #29 is set to the standardized value.
- If field 0 is “-1”: #29 is set to the value of #324 minus the standardized value.

The response is issued immediately.

(8) M answer @9

The M answer command is a response command to an M code notification from the driver. This command can be issued when the *#102 Enabling the selection of RS232C for the M function interface* parameter is set in such a way that the M function communicates with the host device via the RS232C interface and a response character string of the form “M**” is notified as M code to the host device from the driver. Issuing this command to the driver completes the M function.
There is no response to this command.

(9) Jog move command @11: Field 0

The jog move command is for performing jog operations. This command can be executed in the idle status when the *#217 Jog move operation: RS232C selection* parameter is set so that operations are performed via the RS232C interface.
When field 0 contains “1” a move in the + direction is commanded, when it contains “-1” a move in the - direction is commanded, and when it contains “0,” a stop command is issued.
In the idle status, a jog move is performed as commanded by this command. If a start command is issued during a jog move, the move is immediately decelerated and stopped, after which the operation is started. If the operation is ended after that, the motor remains stopped regardless of the jog move status before starting the operation.
The response is issued immediately.

(10) Coordinate system setting @13: Field 0

The coordinate system setting command is for setting the current operation command value to the value specified by field 0. It functions irrespectively of the operation mode. It can only be run in the idle status.
The response is issued immediately.

(11) Other convenient commands

Command name	Command No.	Number of fields	Response at normal operation
Status request	0	1	R3H
Software driver reset	96	0	None

[Status request] @0:0

This command notifies the status of the driver. It functions irrespectively of the operation mode. The response is issued immediately.

Response character string: R3H:driver status:execution program number:execution block number

Driver status

Bit No.	Content	Value	
		0	1
0	Operation mode	PLC	RS232C
1	Servo ready	Not ready	Ready
2	Operation is being executed	Not being executed	Being executed
3	Axis operation is being executed	Not being executed	Being executed
8	Program is being executed	Not being executed	Being executed
9	Mechanical setting mode is being executed	Not being executed	Being executed
16	Error status	Not in error status	Error status
17	Alarm status	Not in alarm status	Alarm status
Others	(Reserved)		

[Software driver reset] @96

This command is software equivalent to turning the power to the driver off and on.
There is no response.

6.1.5 Parameter Commands

Through the use of parameter commands, it is possible to refer to values of parameters and monitor (*reference commands*), assign numerical values and variables to parameters (*simple setting commands*), and assigning results of arithmetic operations on numerical values and variables to parameters (*calculation result setting commands*). The response is issued immediately.
A reference command issues a transmission character string simply containing a variable given directly by #***. At normal operation, if a response is generated, a response character string that begins from "R1□" and one data is returned.
In simple setting commands and calculation result setting commands, the left-hand side must be variables expressed by #***. The right-hand side can be direct numerical values, or it can contain variables such as a parameter/monitor referred to by #***. The response at normal operation is "R00."

[Reference commands]

#50	Reads the value of parameter #50	Response character string: R1D position control bandwidth:12
#337	Reads the value of monitor #337	Response character string: R1B sensor group signal status:00010000

[Simple setting commands]

#400=100	Sets 100 to variable #400.
#400=#401	Sets the value stored in #401 to variable #400.

[Calculation result setting commands]

The following operands can be used:

+	Addition
-	Subtraction
*	Multiplication
/	Division
%	Remainder at integer division

#400=100 + 20	Sets the result of adding 100 and 20 to variable #400.
#400=#401 - 20	Sets the value obtained by subtracting 20 from the value stored in #401 to variable #400.
#400=100 * #401	Sets the result of multiplication of 100 and the value stored in #401 to variable #400.
#400=#400 / #401	Sets the value obtained by dividing the value stored in #400 by the value stored in #401 to variable #400.
#400=100%30	Sets the remainder of 100 divided by 30 to variable #400.

6.2 PLC Interface

6.2.1 Overview

The PLC interface for this driver can be chosen from a list of selectable options. It is possible to select a *contact I/O interface*, a *CC-Link interface* (under development), a *PROFIBUS interface*, and a *DeviceNet interface* (under development).
The contact I/O interface is an I/O interface with 32 input points (sink type output support photo coupler inputs) and 20 output points (sink type open collector output).
In this technical manual, only the contact I/O interface is explained.

6.2.2 Connection, Setting, and I/O Mapping

[Connectors and terminal assignment]

CN4
Made by Honda Tsushin Kogyo
Connector PCR-S50FS
Cover PCR-LS50LA

26	DI_24	01	COMP1
27	DI_25	02	DI_0
28	DI_26	03	DI_1
29	DI_27	04	DI_2
30	DI_28	05	DI_3
31	DI_29	06	DI_4
32	DI_30	07	DI_5
33	DI_31	08	DI_6
34	DO_0	09	DI_7
35	DO_1	10	DI_8
36	DO_2	11	DI_9
37	DO_3	12	DI_10
38	DO_4	13	DI_11
39	DO_5	14	DI_12
40	DO_6	15	DI_13
41	DO_7	16	DI_14
42	DO_8	17	DI_15
43	DO_9	18	DI_16
44	DO_10	19	DI_17
45	DO_11	20	DI_18
46	DO_12	21	DI_19
47	DO_13	22	DI_20
48	DO_14	23	DI_21
49	DO_15	24	DI_22
50	COMN1	25	DI_23

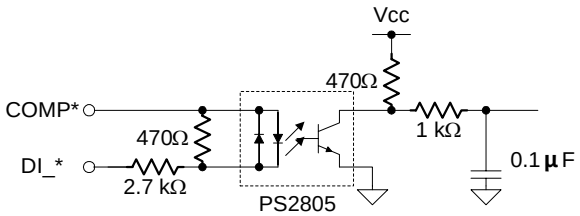
CN5
Made by Phoenix Contact
Plug MC1,5/6-ST-3,81

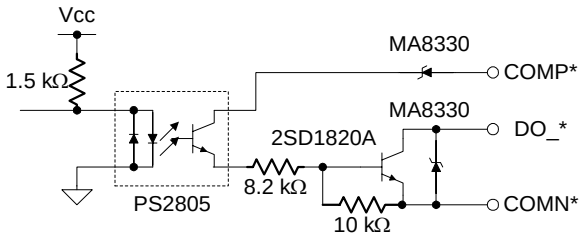
01	COMP2
02	DO_16
03	DO_17
04	DO_18
05	DO_19
06	COMN2

[Electrical specifications]

Input specifications	
Rated voltage	12 to 24 VDC (±10%)
Rated input current	4.1 mA/point (at 12 VDC) 8.5 mA/point (at 24 VDC)
Input impedance	3.0 kΩ
Operation voltage (relative to COMP*)	At OFF 3.0VDC or less At ON 9.0VDC or more
Allowable leak current	OFF is guaranteed at 1.0 mA or less

Output specifications	
Rated voltage	12 to 24 VDC (±10%)
Maximum load current	0.1A/point, 0.5 A/common
ON voltage	0.5VDC or less
Leak current at OFF	0.1mA or less





Note: COMP1 and COMP2 are independent, as are COMN1 and COMN2; they are not short circuited within the driver.

[I/O map]

DI

	7	6	5	4	3	2	1	0	
Block0	IN_MODE .3	IN_MODE .2	IN_MODE .1	IN_MODE .0	IN_MODE _STOP	IN_MODE _START	IN_SERV O	IN_EMG	DI_7 to DI_0
Block1	IN_I_CODE .7	IN_I_CODE .6	IN_I_CODE .5	IN_I_CODE .4	IN_I_CODE .3	IN_I_CODE .2	IN_I_CODE .1	IN_I_CODE .0	DI_15 to DI_8
Block2	IN_POS _INH	reserve	IN_ ERRCODE _REQ	IN_M_AN S	IN_ERR _RESET	IN_ABOR T	IN_INTER LOCK	IN_PRG _REWIN D	DI_23 to DI_16
Block3	reserve	IN_ABS _STR_OP T	IN_ROT DIR_STR_ OPT.1	IN_ROT DIR_STR _OPT.0	IN_SIGN _INDEX	IN_OVER RIDE_SE L	IN_JOG _DN	IN_JOG _UP	DI_31 to DI_24

DO

	7	6	5	4	3	2	1	0	
Block0	reserve	OUT_ ERRCODE _OUT	OUT_M_ EN	OUT_ ALARM	OUT_ER R	OUT_ MODE _EXE	OUT_ SRDY	OUT_ CPURDY	DO_7 to DO_0
Block1	OUT 0_CODE.7	OUT 0_CODE.6	OUT 0_CODE.5	OUT 0_CODE.4	OUT 0_CODE.3	OUT 0_CODE.2	OUT 0_CODE.1	OUT 0_CODE.0	DO_15 to DO_8
Block2	-	-	-	-	OUT_ AREA1	OUT_ AREA0	reserve	OUT_COI N	DO_23 to DO_16

6.2.3 Operation 1

Some of the operations commanded by input signals via the PLC interface may not function depending on the *operation mode* setting. Refer to Section 5.1, “Operation Mode” for an explanation of the operation mode, In this section, the operations that are common regardless of the type of the PLC interface will be explained.

(1) Explanation of signals

[Input signals]

Abbreviated name	Signal name	Contact I/O
IN_EMG	Emergency stop input	DI_0
IN_SERVO	Servo command input	DI_1
IN_MODE_START	Operating action start command input (start)	DI_2
IN_MODE_STOP	Operating action end command input (stop)	DI_3
IN_MODE [3..0]	Operation mode number input (binary)	DI_7 to 4
IN_I_CODE [7..0]	Code input (BCD)	DI_15 to 8
IN_PRG_REWIND	Program auto-rewind input	DI_16
IN_INTERROCK	Interlock command input	DI_17
IN_ABORT	Operating action abort command input (abort)	DI_18
IN_ERR_RESET	Error reset command input	DI_19
IN_M_ANS	M answer input	DI_20
IN_ERRCODE_REQ	Error code request input	DI_21
IN_POS_INH	Integral position control operation disable input	DI_23
IN_JOG_UP	Jog + command input	DI_24
IN_JOG_DN	Jog – command input	DI_25
IN_OVERRIDE_SEL	Velocity override selection input	DI_26
IN_SIGN_INDEX	Index sign input during index operation	DI_27
IN_ROTDIR_STR_OPT [1..0]	Moving direction at rotational coordinate start-up option input (binary)	DI_29 to 28
IN_ABS_STR_OPT	ABS/INC start-up option input	DI_30

[Output signals]

Abbreviated name	Signal name	Contact I/O
OUT_CPURDY	CPU ready output	DO_0
OUT_SRDY	Servo ready output	DO_1
OUT_MODE_EXE	Operation under execution output	DO_2
OUT_ERR	Error status output	DO_3
OUT_ALARM	Alarm status output	DO_4
OUT_M_EN	M code enable output	DO_5
OUT_ERRCODE_OUT	Output during error/alarm code output	DO_6
OUT_O_CODE [7..]	Code output (BCD)	DO_15 to 8
OUT_COIN	Position settling status output	DO_16
OUT_AREA0	Area signal 0 output	DO_18
OUT_AREA1	Area signal 0 output	DO_19

(2) I/O logic setting

It is possible to set the logical relationship between the contact status and the driver signal status for all input and output signals separately in contact units.
The input signals IN*** are processed via the I/O logical conversion and are then expressed as a logical input signal ***. If the signal status is reached, it is expressed as 1 and if the status is not reached, expressed as 0.
A logical output signal *** is expressed as 1 if the status is reached and 0 if the status is not reached. After conversion via the I/O logical setting, it becomes an output signal OUT_***.
The I/O logical settings at the time of shipment from the factory are set to *positive logic* for all input and output signals. In other words, the internal input and output signals are set to 1 when the corresponding contact points are turned ON. By setting the I/O logical setting for IN_SERVO to negative logic using the PC utility, it is possible to connect a PLC interface in the same state as it was shipped from the factory. As a result, an RS232C interface can be used to confirm basic operations.
Refer to Chapter 7 “DrvMII PC Utility” for a description of how to set the I/O logic.

(3) Servo ON/OFF (SERVO)

The function of SERVO, the servo command input, changes depending on whether the main operation authority by the operation mode is given to the RS232C interface or the PLC interface.
If the operation mode is set to the RS232C interface, the servo command input functions as the Servo ON enable/disable function. Servo ON is enabled in the status 1. Servo ON is enabled at contact ON by positive logic setting for the input signal IN_SERVO (the setting at shipment). By negative logic setting, Servo ON is enabled at contact OFF. Refer to Section 6.1.4, “@ Commands” for an explanation of how to use the Servo ON/OFF command in the RS232C interface operation mode.
If the operation mode is set to the PLC interface, the servo command input functions as the Servo ON/OFF command function. It becomes a Servo ON command when in the status 1. As for the input signal IN_SERVO, the Servo ON command will be enabled at contact ON by positive logic setting (factory setting). By negative logic setting, the Servo ON command is enabled at contact OFF. Other than by this command, the actual Servo ON/OFF status is also affected by the setting of the Servo ON disable switch SRV DS on the front panel. See the table below.

Servo ON/OFF SERVO	SRV DS Servo ON disable on the front panel	Actual servo status
Status 0	Disabled	Servo OFF
	Enabled	
Status 1	Disabled	
	Enabled	Servo ON

(4) Start and stop (MODE_START MODE_STOP MODE [3..0])

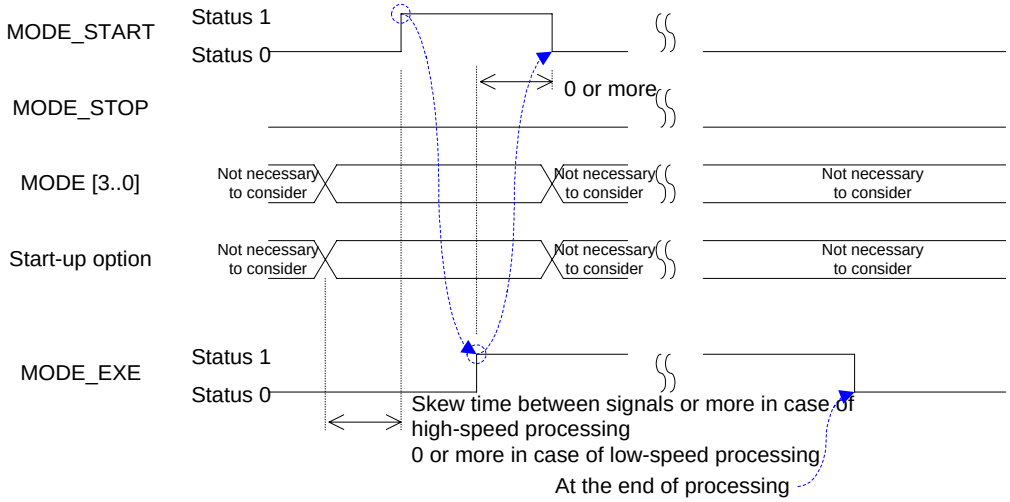
The MODE_START and MODE_STOP commands start and stop operating actions other than job moves. They can be issued via the PLC interface when the setting of the operation mode has given the main operation authority to the PLC interface rather than the RS232C interface.
The operating action start and stop commands are issued in the status 1.
When a start command is issued by MODE_START it depends on the operating action which signals (start-up options) that must be preset. When issuing a MODE [3..0] command, the number of the operating action you wish to perform must be always set. See the table below.
Moreover, if high-speed processing is not selected in the setting of the *#215 PLC operation: Start signal processing speed selection* parameter, the processing of start commands by MODE [3..0] and start-up options takes place at the next 10ms scan. In this case, the content of each signal is read and processed after 10ms has passed after the start command was issued. If the time delay between outputs by the PLC is in the order of several ms, the commands will function properly provided that they are set to the same time or before the start command is issued by MODE_START. However, the time wasted with respect to issuing the start command should be added (10ms) instead. The total amount of wasted time until the motor start is thus 10ms (scan time) + 10ms (read delay) + internal delay time.
When high-speed processing is selected in the #215 parameter setting, the processing of the start command takes place at the next 2ms scan. In this case, the content of each signal when the start command is issued is read and processed. The time lag between outputs must be taken into consideration on the PLC side and set that much earlier. On the other hand, the time wasted with respect to the start command itself can be eliminated. The total amount of wasted time until the motor start is the 2ms (scan time) + internal delay time.
In MODE_STOP, the driver response to a stop command varies depending on the operating action. See the table below. Note furthermore that operations of the *self-end type*, which end automatically when the operation is completed, do not require a stop command. Operations of the *non-self-end type*, which cannot complete the operation by itself, must be ended by a stop command by MODE_STOP. In case of operations of the *non-end type*, which cannot be stopped once started, the stop command issued by MODE_STOP is invalid. Refer to Section 5.3, “Operation Functions” for a listing of the end types of the different operations.
Outputs during MODE_EXE operation execution maintain their under-execution status during MODE_START command status (during status 1) after the operation is started, even after the operation is over.

<table>
<tr><th>No.</th><th>Name</th><th>Start-up option</th><th>End type</th><th>Response to stop command</th></tr>
<tr><td>1</td><td>Test operation</td><td rowspan="3">None</td><td>Non-self-end</td><td>End the operation upon returning to the start position.</td></tr>
<tr><td>2</td><td>Auto-tuning operation</td><td rowspan="4">Self-end</td><td>End the operation when the oscillation command to the motor is completed.</td></tr>
<tr><td>3</td><td>Homing operation</td><td>Immediately decelerate and stop the move, and end the operation.</td></tr>
<tr><td>4</td><td>Program operations</td><td>I_CODE [7..0]</td><td>End the operation when execution of the current block is completed.</td></tr>
<tr><td>5</td><td>Signal search operation</td><td rowspan="2">None</td><td>Immediately decelerate and stop the move, and end the operation.</td></tr>
<tr><td>7</td><td>MDI operation</td><td>Non-self-end</td><td>End the operation when execution of the current NC executable statement or parameter statement input via RS232C is completed.</td></tr>
<tr><td>8</td><td>Index Type A operation</td><td rowspan="2">I_CODE [7..0], SIGN_INDEX, ROTDIR_STR_OPT [1..0] (when necessary), ABS_STR_OPT (when necessary)</td><td rowspan="3">Self-end</td><td rowspan="3">Invalid (ignored)</td></tr>
<tr><td>9</td><td>Index Type B operation</td></tr>
<tr><td>10</td><td>Table reference operation</td><td>I_CODE [7..0], ROTDIR_STR_OPT[1..0] (when necessary), ABS_STR_OPT (when necessary)</td></tr>
<tr><td>15</td><td>Mechanical setting mode</td><td>None</td><td>Non-end</td><td>Invalid because the operation cannot be ended (ignored).</td></tr>
</table>

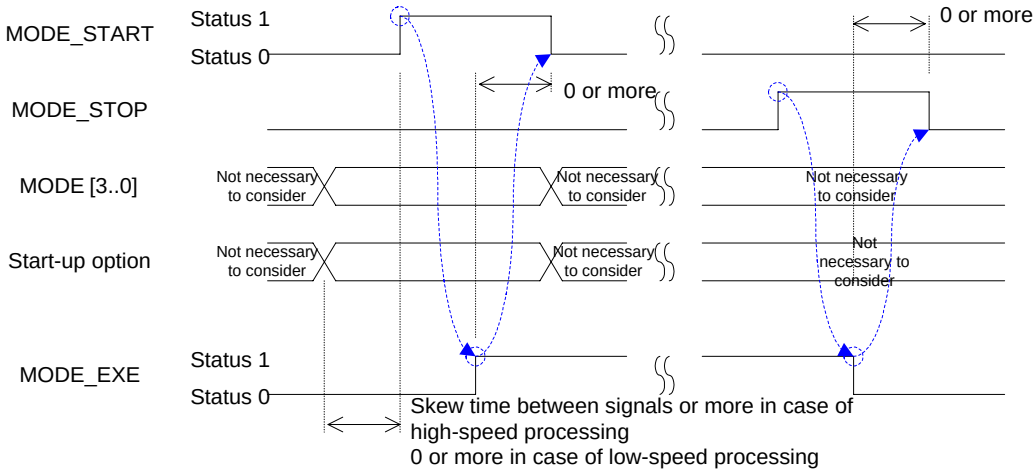
[Related parameters]

#215	PLC operation: Start signal processing speed selection

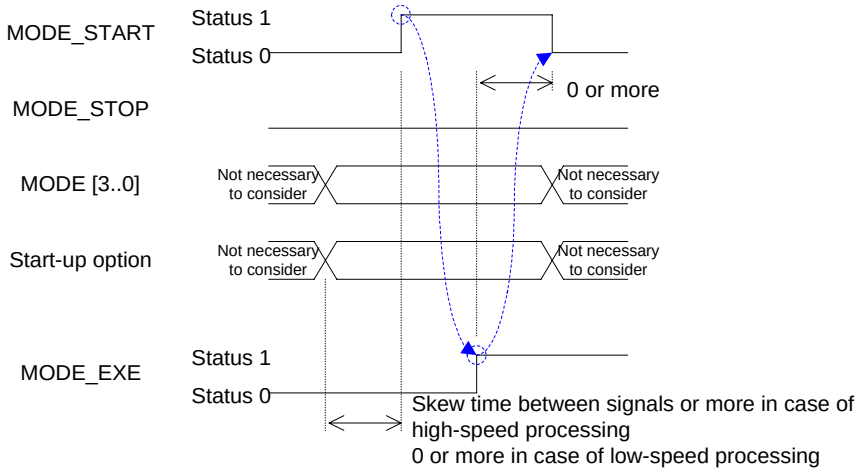
[Self-end type]



[Non-self-end type]



[Non-end type]



[Start-up option at program operation]

The program number to be started at program operation is provided by the I-CODE [7..0] code entry.
In a setting where it is properly read as a BCD code, the program with the set number is executed from the top.
In a setting where it is not properly read as a BCD code, the program with the stored execution program number is executed from the block with the stored block number.

[Start-up options at index Type A operation and index Type B operation]

In index Type A and B operations, the index number is provided by the I_CODE [7..0] code entry and the SIGN_INDEX sign entry at index operation, the choice between absolute or incremental move is provided by the ABS_STR_OPT ABS/INC start-up option entry, and the moving direction option is provided by ROTDIR_STR_OPT [1..0].
I_CODE [7..0] must be set so that it can be read properly as a BCD code. When SIGN_INDEX is the in status 1, the value read by I_CODE [7..0] is treated as a negative value, and when in the status 0, treated as a positive value. In case of an incremental move the value is treated as a value relative to the current operation command value and in case of an absolute move, it is treated as the target operation command value.
ABS_STR_OPT indicates an absolute move in the status 1 and an incremental move in the status 0. When the *#104 ABS/INC setting during table index operation* parameter is set to anything other than *start-up option dependence*, the option specification given by ABS_STR_OPT is invalid.
ROTDIR_STR_OPT [1..0] indicates Type 0 when 0 or 3 is provided by binary code, Type 1 when 1 is provided, and Type 2 when 2 is provided. When the *#105 moving direction option for rotational coordinates* parameter is set to anything other than *start-up option dependence*, the option specification given by ROTDIR_STR_OPT [1..0] is invalid.

[Related parameters]

#104	ABS/INC setting during table index operation
#105	Moving direction option for rotational coordinates

[Start-up option at table reference operation]

In table parameter operations, the table entry is provided by the I_CODE [7..0] code entry, the choice between an absolute and incremental move is provided by the ABS_STR_OPT ABS/INC start-up option entry, and the moving direction option is provided by ROTDIR_STR_OPT [1..0].
I_CODE [7..0] must be set so that it can be read properly as a BCD code. The table data value and option are obtained based on the specified number. The value of the obtained table data is treated as a value relative to the current operation command value in case of an incremental move and in case of an absolute move it is treated as the target operation command value.
ABS_STR_OPT indicates an absolute move in the status 1 and an incremental move in the status 0. When the *#104 ABS/INC setting during table index operation* parameter is set to anything other than *start-up option dependence*, the option specification given by ABS_STR_OPT is invalid.
ROTDIR_STR_OPT [1..0] indicates Type 0 when 0 or 3 is provided by binary code, Type 1 when 1 is provided, and Type 2 when 2 is provided. When the *#105 Moving direction option for rotational coordinates* parameter is set to anything other than *start-up option dependence*, the option specification given by ROTDIR_STR_OPT [1..0] is invalid.

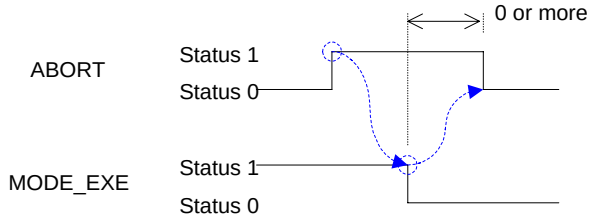
[Related parameters]

#104	ABS/INC setting during table index operation
#105	Moving direction option for rotational coordinates

Note: Regarding the moving direction in rotational coordinates
- Type0 Proximity rotational move (multiple rotations are not allowed).
- Type1 Does not cross the rotational coordinate's home position (multiple rotations are not allowed).
- Type2 Calculates the target position relative to the current rotational coordinate's home position and determines the direction (multiple rotations are allowed).

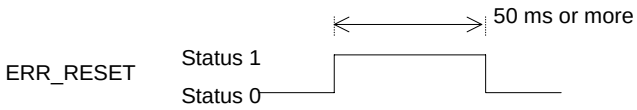
(5) Abort (ABORT)

An operation abort command, ABORT, stops any operation other than a jog move. It functions irrespectively of the operation mode.
The operation abort command is issued as the status 1.
Unlike MODE_STOP, the operating action stop command, the motor immediately decelerates and stops and ends the operation even during an operation involving movement. In case the M function is being executed, the execution is aborted and the operation is stopped.



(6) Error reset (ERR_RESET)

The error reset command, ERR_RESET, cancels an error status in the driver. It functions irrespectively of the operation mode. It can only be executed while in the idle status.
The error reset command is issued as the status 1.
Depending on the error content, there are errors that cannot be canceled or errors that cause identical errors immediately after canceling. Avoid creating a program that maintains ERR_RESET and waits until ERR, the error status output, is canceled.



(7) Interlock and velocity override selection (INTERLOCK OVERRIDE_SEL)

Both the interlock command INTERLOCK and the velocity override selection OVERRIDE_SEL select a velocity override value of the driver. They function irrespectively of the operation mode.
The relationship between the INTERLOCK and OVERRIDE_SEL status and the velocity override value can be seen in the table below.

Interlock INTERLOCK	Velocity override selection OVERRIDE_SEL	Selected velocity override value
Status 0	Status 0	#16 Velocity override percentage 1
	Status 1	#17 Velocity override percentage 2
Status 1	Status 0	0
	Status 1	

[Related parameters]

#16 Velocity override percentage 1
#17 Velocity override percentage 2

(8) Program auto-rewind (PRG_REWIND)

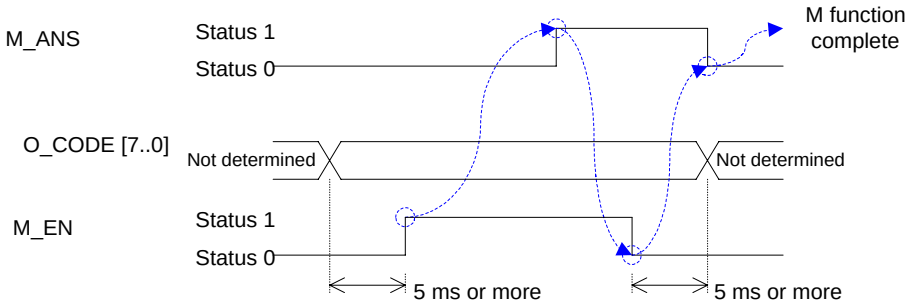
The program auto-rewind PRG_REWIND is a signal that specifies whether or not to repeat the program from the top block when the execution of the last block of the program is completed in program operation. If the program is repeated, it also monitors the signal status each time execution of the last block is completed and judges whether or not to repeat again. It functions irrespectively of the operation mode.
The program is repeated if the status of the signal is 1.

(9) Integral position control operation inhibition (POS_INH)

The integral position control operation inhibition POS_INH is a signal that prohibits the integral operation at the position control part. It can be issued via the PLC interface when the setting of the operation mode has given the main operation authority to the PLC interface.
Integral operation is inhibited in the status 1 and allowed in the status 0.

(10) Function (M_ANS, M_EN, O_CODE [7..0])

The *M function* operates under the PLC interface when the *#102 Enabling the selection of RS232C for the M function interface* parameter is set in such a way that the M function communicates with the host device via the PLC interface. See Section 5.7, “M Function” for details about the M function.
The notification from the driver takes place through the use of M_EN and O_CODE [7..0]. In O_CODE [7..0], the M function is expressed by a two-digit BCD code. It notifies that an M code is issued by setting M_EN to the status 1. The host device connected via the PLC interface, upon detecting the notification of the M code from the driver, performs the necessary processing on its own side, then responds to the driver by setting M_ANS to the status 1. The driver sets M_EN to 0 when receiving this response. Hereafter, the driver detects when M_ANS is set to the status 0 and completes the rest of the M function interface.
O_CODE [7..0] is a shared property of the M function and error code request function; when the error code request function is being executed, the M function is in the wait status.



(11) Jog move command (JOG_UP and JOG_DN)

A jog move command via the PLC interface functions in the idle status when the *#217 Jog move operation: RS232C selection* parameter is set so that operations are performed via the PLC interface.
A jog move is executed by JOG_UP and JOG_DN as explained in the table below.
In the idle status, a jog move is executed according to the command as it is. If a start command is issued during the jog move, the jog move is immediately decelerated and stopped, after which the operation is started. After that, when the operation ends, the jog move is performed according to the jog move command after stopping.

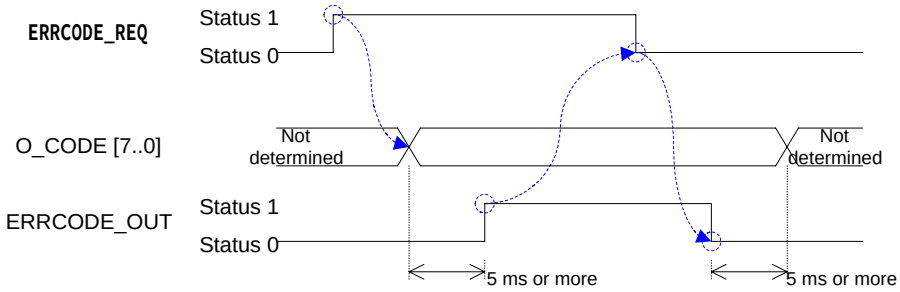
Jog (–) command JOG_DN	Jog (+) command JOG_UP	Jog move command
Status 0	Status 0	Stop command
	Status 1	(+) direction move command
Status 1	Status 0	(-) direction move command
	Status 1	Stop command

6.2.4 Operation 2

This section explains operations unique to the contact I/O interface.

(1) Error code request function (ERRCODE_REQ, ERRCODE_OUT, O_CODE [7..0])

The error code request function operates irrespectively of the operation mode.
The error code request function starts when the host device connected to the PLC interface sets ERRCODE_REQ to the status 1. The driver notifies about the presence of the error code using ERRCODE_OUT and O_CODE [7..0]. In O_CODE [7..0], an error code is expressed as a two-digit BCD code. It notifies that an error code is issued by setting ERRCODE_OUT to the status 1. If no error has occurred, the error code is output as 0. The host device sets ERRCODE_REQ to the status 0 upon detecting the notification of an error code from the driver. The driver detects that ERRCODE_REQ is set to the status 0, sets ERRCODE_OUT to the status 0, and completes the rest of the error code request function.
O_CODE [7..0] is a shared property of the M function and the error code request function; when the M function is being executed, the error code request function is in the wait status.



Chapter 7
DrvMII PC Utility

7.1 Overview

7.1.1 Overview of the Operation Menu
7.1.2 Overview of the Action Menu
7.1.3 Overview of the Data Management Menu

7.2 Installation

7.2.1 Installation under Windows 95/98/98SE/Me/NT4.0/2000
7.2.2 Starting the PC Utility

7.3 Preparation

7.3.1 Selecting a Communication Port
7.3.2 Selecting Channels
7.3.3 Displaying Communication Strings
7.3.4 Main Menu

7.4 Operation Menu

7.4.1 Terminal
7.4.2 Servo Tuning
7.4.3 Oscilloscope

7.5 Action Menu

7.5.1 Homing
7.5.2 Signal Search Move
7.5.3 Index Type A Operation
7.5.4 Index Type B Operation
7.5.5 Table Reference Operation
7.5.6 Jog Move

7.6 Data Management Menus

7.6.1 Parameter Manager
7.6.2 Program Manager
7.6.3 Index Menu
7.6.4 Point Set
7.6.5 Parts
7.6.6 I/O Set

7.1 Overview

The DrvMII PC Utility consists of three components that are accessed from the following menus: *"operation menu," "action menu,"* and *"data management menu."*

7.1.1 Overview of the Operation Menu

The operation menu contains the following three functions：

- ***Terminal***:
  This menu allows you to send and receive character strings to/from the M2 driver (hereinafter referred to as the "driver"), monitor parameters/monitors as well as errors/alarms, and use parameter/command help.
- ***Servo control***:
  This menu allows you to adjust the servo parameters of the motor through auto-tuning and manual tuning. It also allows you to adjust various compensation filters.
- ***Oscilloscope***:
  This function displays graphs of time-series of parameter/monitor values.

7.1.2 Overview of the Action Menu

In the action menu, you can set and display parameters, display monitors, and start or stop actions related to the operations listed below.
"Homing move," "signal search move," "index Type A operation," "index Type B operation," "table reference operation," and *"jog move."*

7.1.3 Overview of the Data Management Menu

The data management menu contains the following seven functions:

- ***Parameter***:
  This function allows you to save all the parameters to files and register them from files. It also allows you to edit the machine setting parameters.
- ***Program***:
  This function allows you to edit, register, and save the programs necessary for "program operation." In addition, you can start or stop a program and monitor the program block currently being executed.
- ***Index compensation***:
  This function allows you to edit, register, and save data necessary for "index Type A operation" and "index Type B operation."
- ***Point set***:
  This function allows you to edit, register, and save data necessary for "table reference operation."
- ***Parts***:
  This function allows you to edit (universal cam), register, and save parts necessary for cam profile moves.
- ***I/O set***:
  This function allows you to set the logical setting of I/O points.
- ***Absolute precision compensation***:
  This function allows you to edit, register, and save absolute precision compensation data.
  (This setting is valid if the absolute precision option is set to "yes," with which the absolute precision of the motor is compensated for the entire circumference.)

7.2 Installation

7.2.1 Installation under Windows 95/98/98SE/Me/NT4.0/2000

The DrvMII utility (hereinafter referred to as the "PC utility") runs on Windows 95, 98, 98SE, Me, NT4.0 and 2000. It can be installed via "Add/Remove Programs" under the "Control Panel" in Windows. If an older version of the PC utility is present, delete it first and then install the new version.
Display the "Properties of Adding/Removing Programs" dialog box and click "*Set Up (1)*." Then proceed according to the instructions displayed on the screen. The PC utility setup program starts up.
Proceed with the setup according to the instructions on the screen. A dialog box for determining the directory in which to install the PC utility appears (see Figure 7.1).

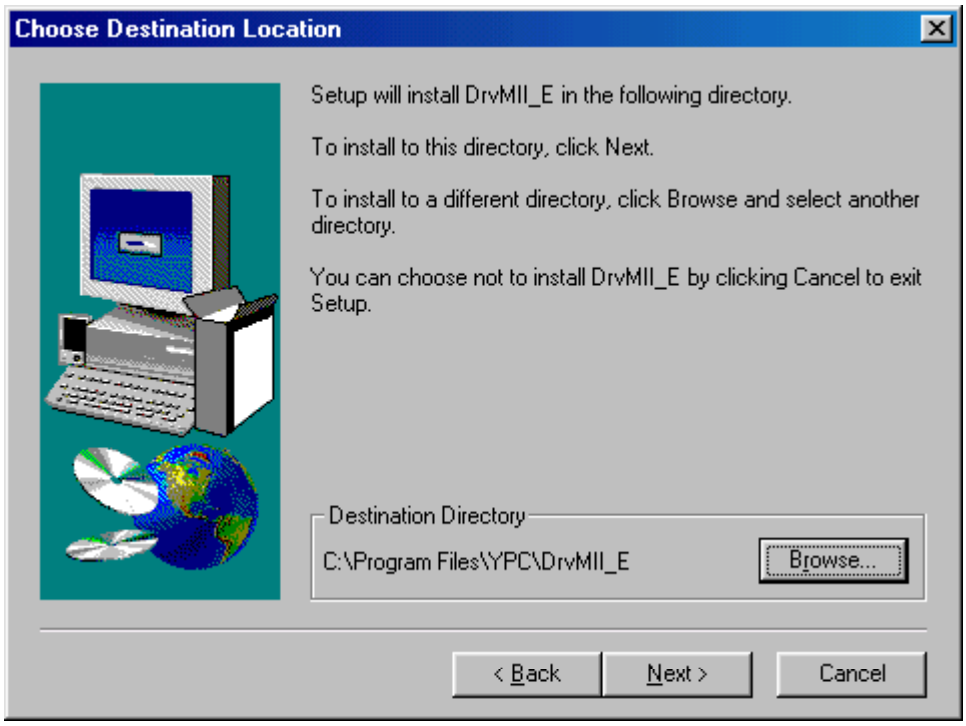


Figure 7.1 "Choose Destination Location" dialog box

Click "*Browse*" to display the "Select Directory" dialog box and select the desired drive and directory. Click "*Next*" to display "Select Program Folder" (see Figure 7.2).

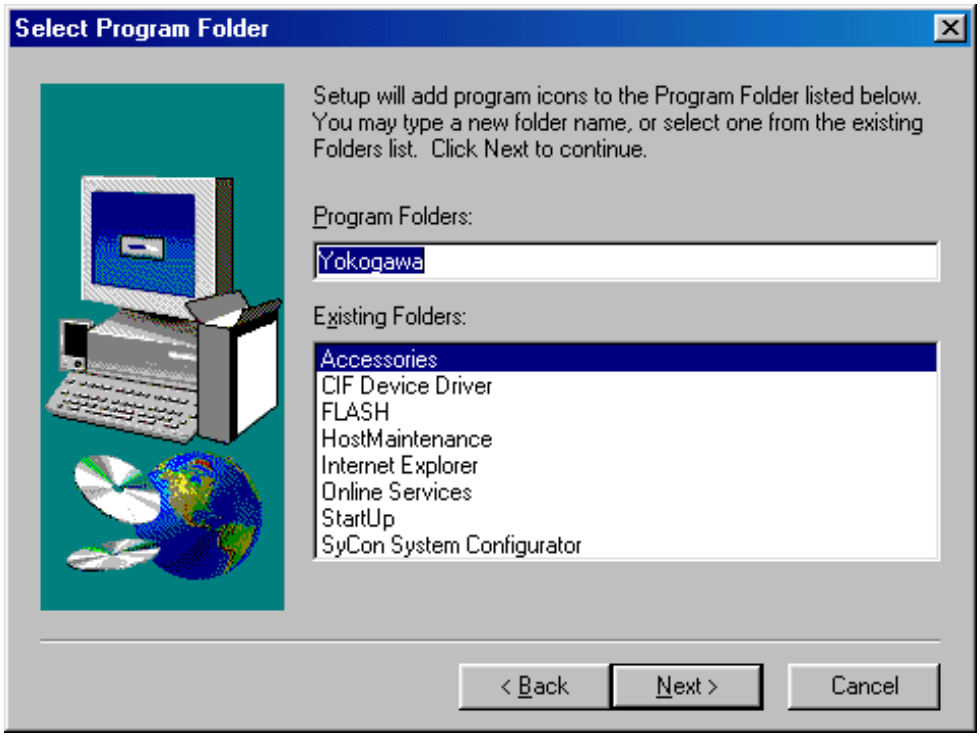


Figure 7.2 "Select Program Folder" dialog box

Select a program folder and click "*Next*." The installation begins. Follow the instructions on the screen and change disks. When the setup is completed, the "Setup Complete" dialog box appear (see Figure 7.3).

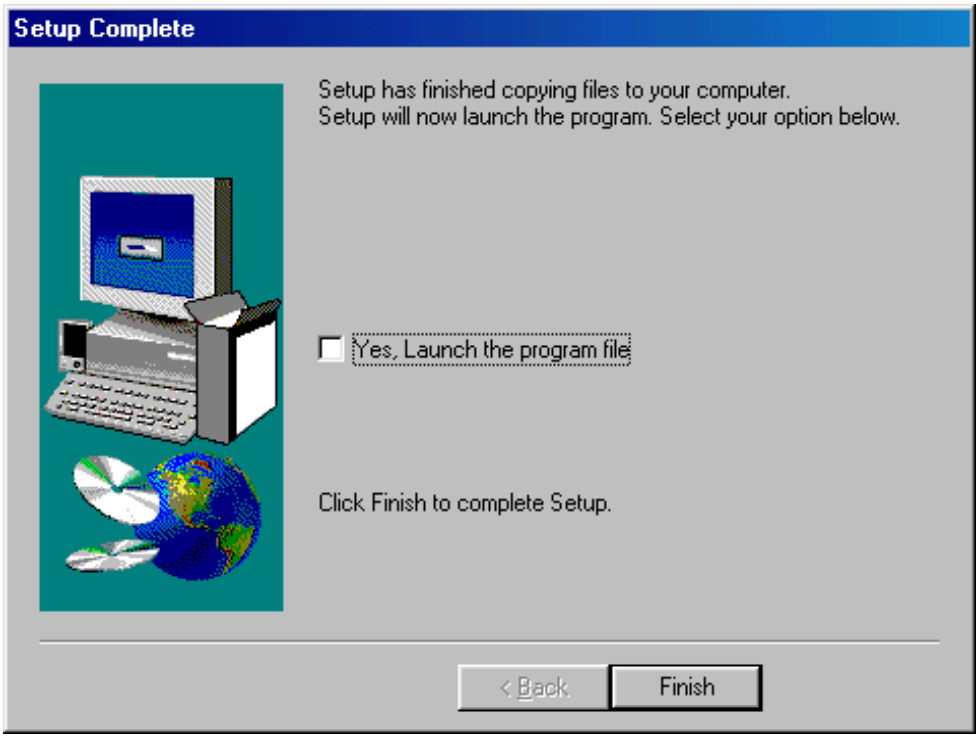


Figure 7.3 "Setup Complete" dialog box

To start the program, select "Launch program file" and click "*Finish*." If you do not want to start the program, just click "*Finish*." If you are prompted to restart the computer, simply follow the message and restart it.

7.2.2 Starting the PC Utility

In order to start the PC utility under Windows, click the “Start” button, “Program,” “Specified program folder,” and then “YOKOGAWA_E” The “Opening” dialog box (see Figure 7.4) is displayed for several seconds, and the PC utility starts up. (By default, the specified program folder is “YOKOGAWA_E”)

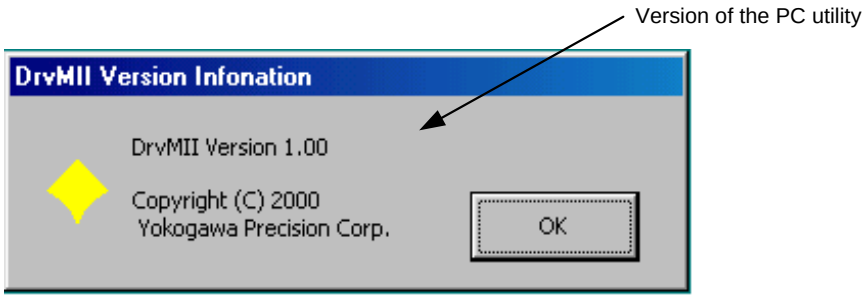


Figure 7.4 “Opening” dialog box

7.3 Preparation

Connect the serial port of the PC with the serial port of the driver with a dedicated cable.
(Do not use commercial off-the-shelf cables. There is a terminal for which 5V is output from the driver for connecting with the teaching box.)

7.3.1 Selecting a Communication Port

When you start the PC utility, the “ComPortSelect” dialog box appears in the left side of the screen (see Figure 7.5). Change the setting according to the communication port of the connected PC.

Figure 7.5 “ComPortSelect”

Note: Settings made in the “ComPortSelect” dialog box are stored in a file. It is not necessary to make settings from the next time you start the PC utility. Change the setting as necessary.

7.3.2 Selecting Channels

When you start the PC utility, the “Communication mode” dialog box appears in the upper left corner of the screen (see Figure 7.6). If you are using one driver, select a single channel, and if you are using multiple drivers, select multi-channel addresses. (See Chapter 5 for how to make setting on the driver side.)

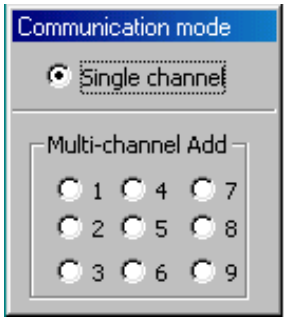


Figure 7.6 “Communication mode”

Note: The settings made in the “Communication mode” dialog box are not stored. When the PC utility is started up, a single channel is always set.

7.3.3 Displaying Communication Strings

When you start the PC utility, the “Communication string” dialog box appears in the upper right corner of the screen. (See Figure 7.7.) Any strings that the PC utility sends to the driver as well as any strings received from the driver are displayed regardless of the menu.

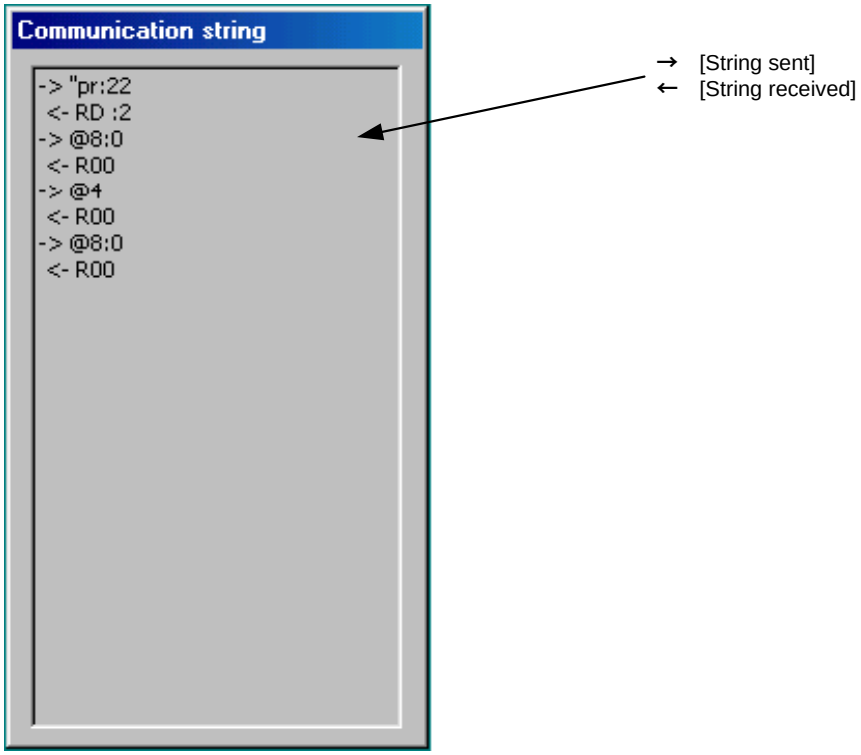


Figure 7.7 “Communication string”

7.3.4 Main Menu

When you start the PC utility, the “MainMenu” dialog box appears (see Figure 7.8). See the following chapters for how to start the actual operation.

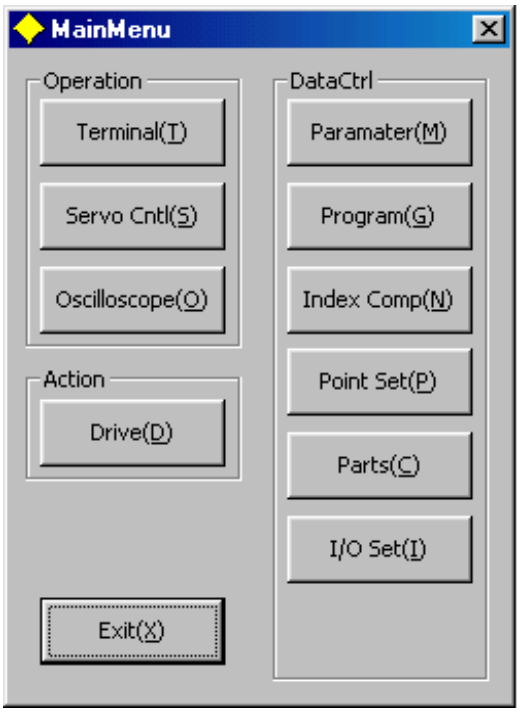


Figure 7.8 “MainMenu” dialog box

7.4 Operation Menu

7.4.1 Terminal

Using this menu, you can send and receive character strings to/from the driver, monitor parameters/monitors as well as errors/alarms, and use parameter/command help.

Click *"Terminal (T)"* under "MainMenu" to display the "Terminal" dialog box (see Figure 7.9).

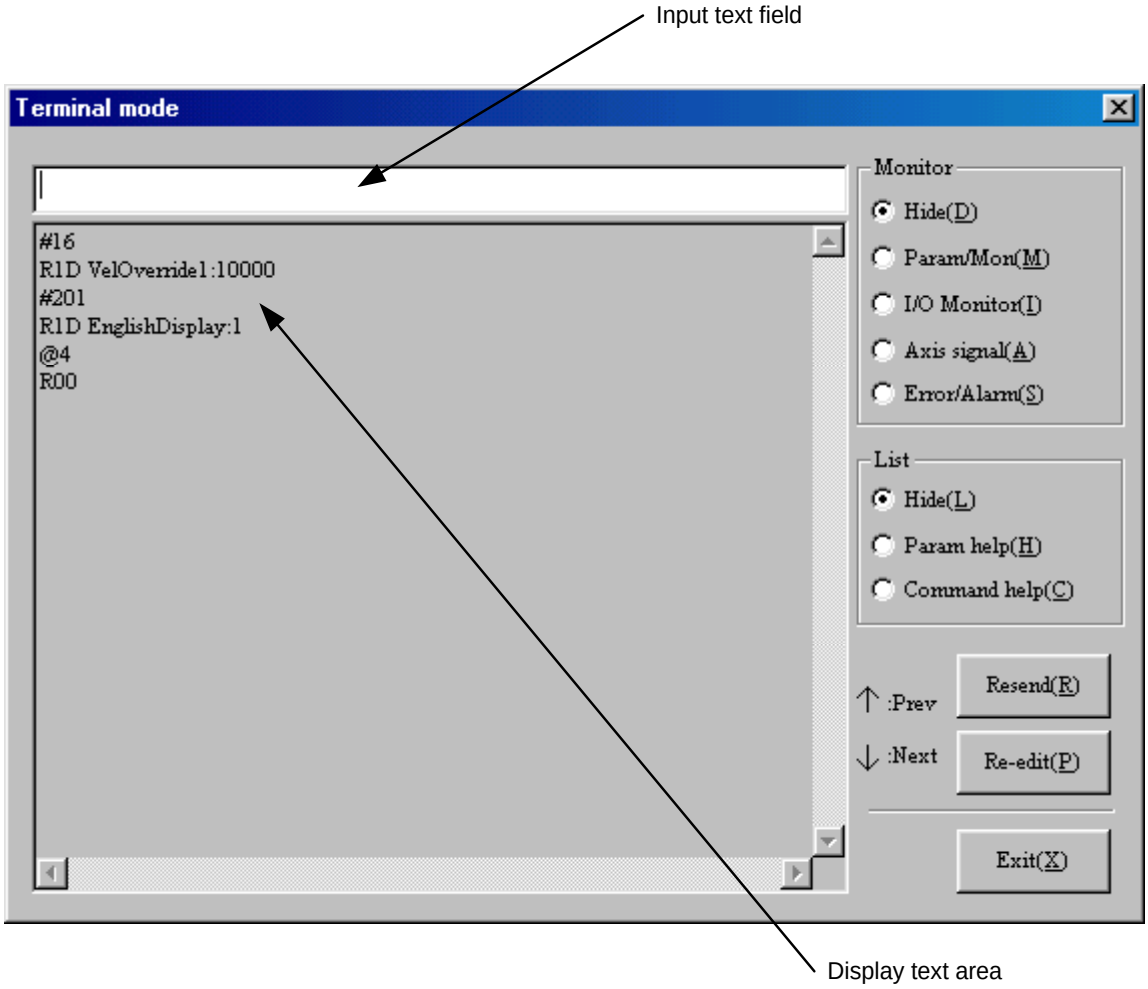


Figure 7.9 "Terminal" dialog box

[Sending/receiving character strings]

1) Enter a character string in the input text field and press the Enter (Return) key. The character string is sent to the driver and, at the same time, displayed in the display text area.
2) When a character string is received from the driver, it is displayed in the display text area.
3) If you click *"Resend,"* the character string transmitted last time is sent again.
4) If you click *"Re-edit,"* the character string transmitted last time is displayed in the input text field.
5) It is possible to display a maximum of ten transmitted character strings in the order of transmission in the input text field by pressing the ↑ arrow key on the PC keyboard. By pressing the ↓ arrow key, the character strings displayed by pressing the ↑ arrow key can be displayed in the reverse order.

(1) Parameter/monitor

In the “Terminal” menu, click *“Parameter/Monitor (M)”* under “Monitor” to display the “Parameter/Monitor” dialog box (see Figure 7.10).

If the number of a parameter/monitor you want to monitor is entered in the parameter/monitor number text field, the contents and values of the corresponding parameters are displayed. Up to five parameters/monitors can be monitored, and they can be switched on and off by clicking their respective switch check boxes (the update cycle of the parameter/monitor values can be shortened by decreasing the number of parameters to be monitored).

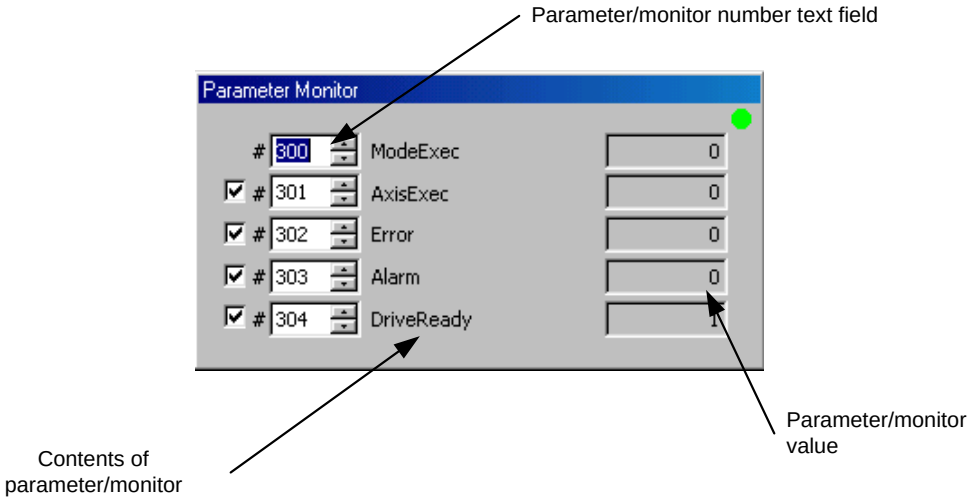


Figure 7.10 “Parameter Monitor” dialog box

(2) I/O monitor

In the “Terminal” menu, click *“I/O Monitor (I)”* under “Monitor” to display the “I/O Monitor” dialog box (see Figure 7.11).

Via the “I/O monitor” dialog box, it is possible to monitor the on/off status of DI and DO points. The number of DI/DO points that can be monitored at the same time is set to 32/32. If you are using an interface where the number of DI/DO points is greater than 32/32, the ones you wish to monitor should be chosen from among the available options.

If the driver is started up in the DI emulation status, the DI points can be turned on/off directly using the buttons in the lower part of the dialog box. (Even if the displayed DI/DO points are changed by choosing from among the available options, the status of the DI point, once set, is maintained.)

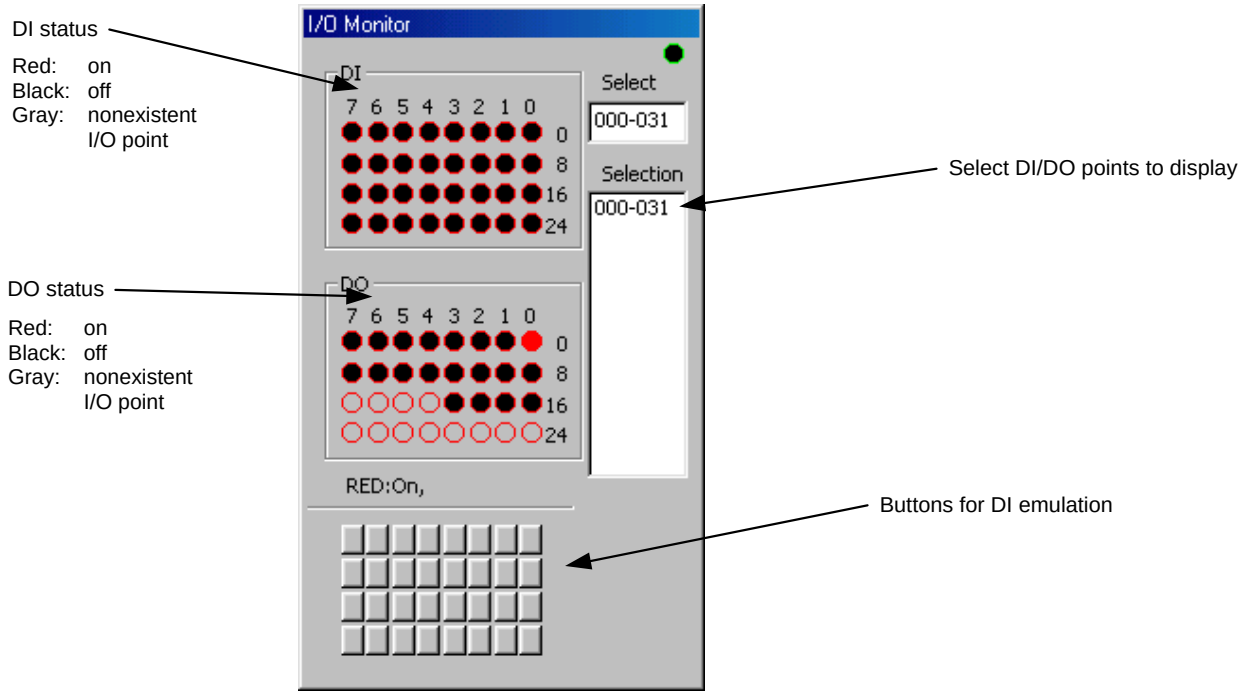


Figure 7.11 “I/O Monitor” dialog box

(3) Axis signal status display

In the “Terminal” menu, click *“Axis signal status (A)”* under “Monitor” to display the “Axis signal status” dialog box (see Figure 7.12).

Via the “Axis signal status” dialog box, it is possible to monitor the axis status, etc. of the driver.

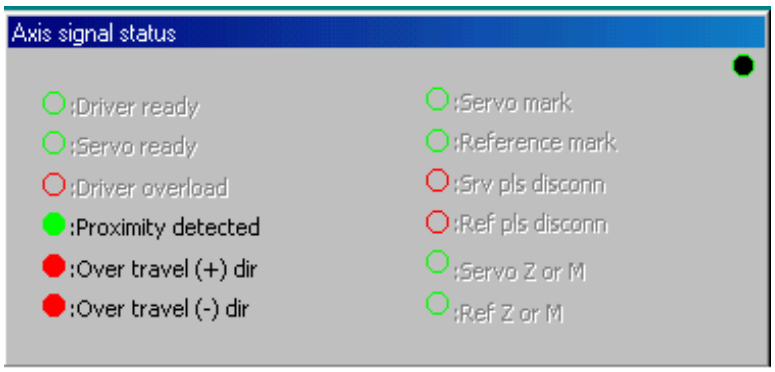


Figure 7.12 “Axis signal status” dialog box

(4) Error or alarm monitor

In the “Terminal,” click *“Error or Alarm (S)”* under “Monitor” to display the “Error or Alarm” dialog box (see figure 7.13).

When an error occurs, this dialog box displays “error message” and shows the error history in the display text area. When an alarm occurs, it displays “alarm message” and shows the alarm history in the display text area. In the display text area, the error history is displayed first, and a maximum of 16 errors/alarms is displayed.

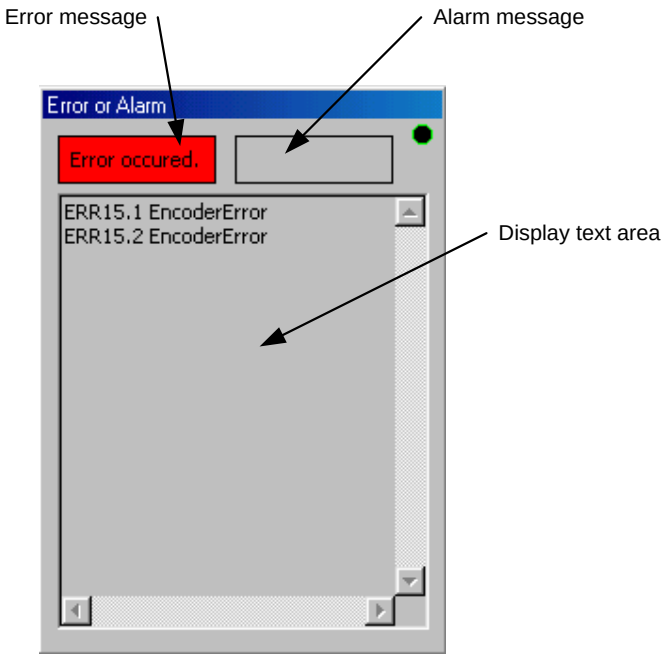


Figure 7.13 “Error or Alarm” dialog box

(5) Parameter/monitor help

In the “Terminal” menu, click *“Parameter/Monitor help (H)”* under “List” to display the “Parameter/Monitor help” dialog box (see Figure 7.14).

The Parameter/Monitor help can display the contents of a maximum of ten parameters/monitors. If you click *“Prev (P),”* parameters/monitors with smaller numbers than the currently displayed parameters/monitors are displayed. If you click *“Next (N),”* parameters/monitors with larger numbers than the currently displayed parameters/monitors are displayed.

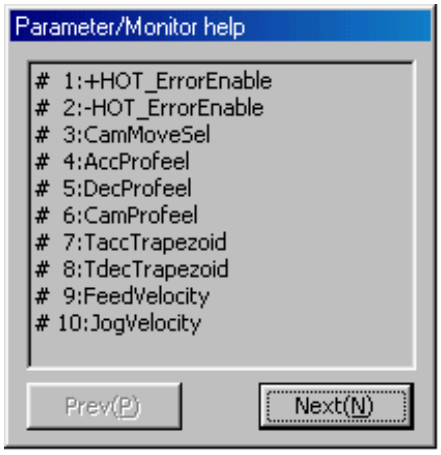


Figure 7.14 “Parameter/Monitor help” dialog box

(6) Command help

In the “Terminal” menu, click *“Command Help (C)”* under “List” to display the “Command Help” dialog box (see Figure 7.15).

The Command Help can display the contents of a maximum of ten commands. If you click *“Prev (P),”* commands with smaller numbers than the currently displayed commands are displayed. If you click *“Next (N),”* commands with larger numbers than the currently displayed commands are displayed.

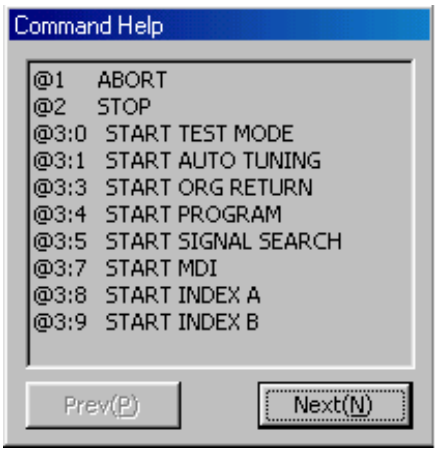


Figure 7.15 “Command Help” dialog box

7.4.2 Servo Tuning

This menu allows you to adjust the servo parameters of the motor through auto-tuning and manual tuning in addition to adjust various compensation filters

Click *"Servo Cntl (S)"* on "MainMenu" to display the "Servo Tuning" dialog box.

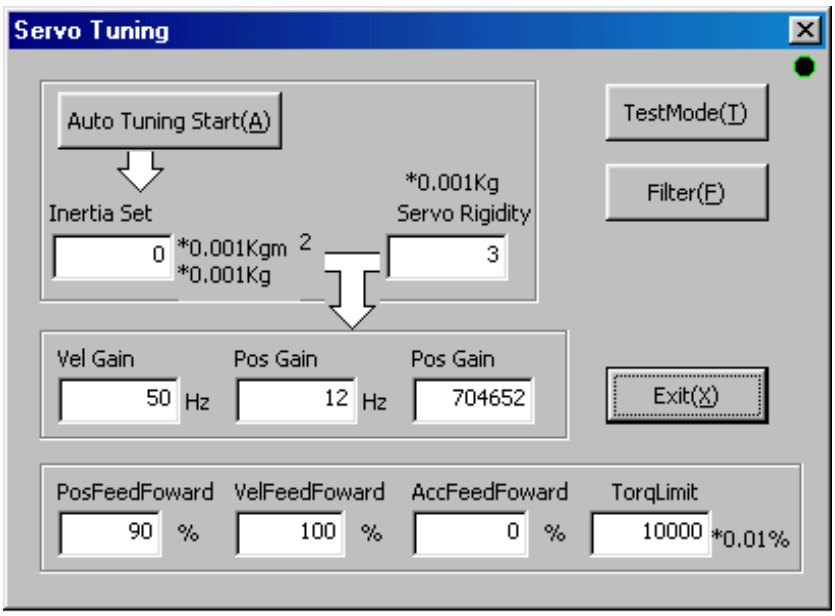


Figure 7.16 "Servo Tuning" dialog box

(1) Auto-tuning

1) Change the main operation mode to the RS232C main operation mode (see Chapter 5).
2) Set the motor to Servo ON (the operations until this point must be made before opening the "Servo Tuning" dialog box.)
3) Click *"Auto Tuning (A)"* to start the auto-tuning.

Note: The motor performs reciprocating movements in order to estimate the inertia and weight of the load (the operation width of the reciprocating movement can be changed in parameter <#32>). Please make sure that there are no interfering objects within the range of the operation width.

4) After performing reciprocating movement for several times, the estimation of the inertia and weight is completed and reflected in the parameter for inertia/weight.

* By changing the servo stiffness setting parameter, three parameters – the velocity control bandwidth, position control bandwidth, and position control integral limiter – are changed and thus the servo stiffness changes.

(2) Manual tuning

1) Change the main operation mode to the RS232C main operation mode (see Chapter 5).
2) Set the motor to Servo ON (the operations until this point must be made before opening the "Servo Tuning" dialog box.)
3) Click *"Test Mode (T)"* to enter the test mode.

Note: The motor performs small width reciprocating movements (the operation width of the reciprocating movement can be changed in parameter <#32>). Please make sure that there are no interfering objects in the range of the operation width.

4) In the test mode, the driver monitors the waveform of the motor position by oscilloscope, etc., and manipulates three parameters – the velocity control bandwidth, position control bandwidth, and position control integral limiter – to tune the servo (see Chapter 5 for more details).

[Other parameters]

* The three parameters – position feed forward, velocity feed forward, and acceleration feed forward – have no relation with servo stiffness. They are parameters for adjusting the settling time decrease.
* The torque limiter parameter should be changed when limiting the motor torque.

(3) Filter setting

Click *"Filter (F)"* under "Servo Tuning" to display the "Filter" dialog box (see Figure 7.17).
The filter setting is divided into two sections, a first order delay filter setting and a notch filter setting.

1) To set the first order delay filter: Select one from None, 20/80, 30/120, and 40/160.
2) To set the notch filter: Manipulate the frequency setting scroll bar and set.
   - Clicking the arrows at either end: The frequency changes in steps of one.
   - Clicking between the slider bar and an arrow: The frequency changes in steps of ten.
   - Dragging the slider bar: The frequency is set to the value at the position to which the slider bar is moved.

* Refer to Chapter 5 for how to use the filters.

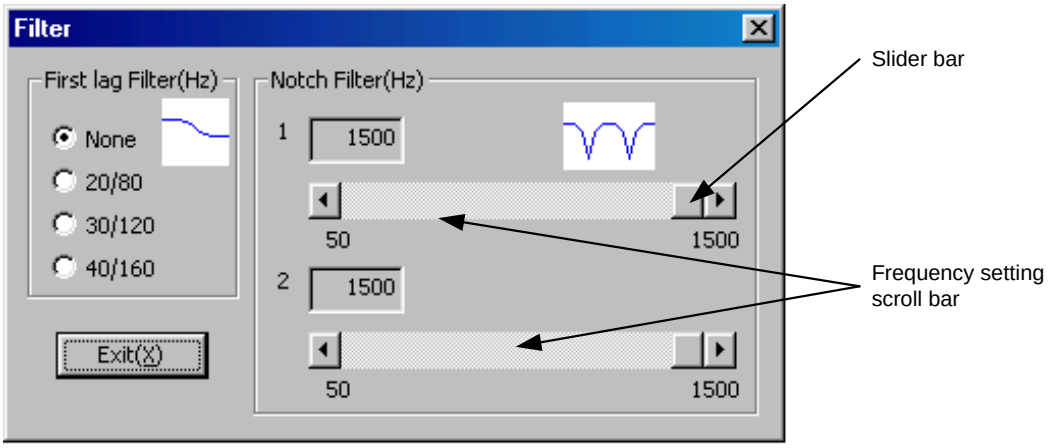


Figure 7.17 "Filter" dialog box

7.4.3 Oscilloscope

The oscilloscope displays time-series of parameter/monitor values.

Click *"Oscilloscope (O)"* under "MainMenu" to display the "Oscilloscope" dialog box.

Note: The parameter/monitor information is obtained automatically from the driver when the PC utility is started. Please wait for a while until it can be used. (This operation is required only once for the initial use.)

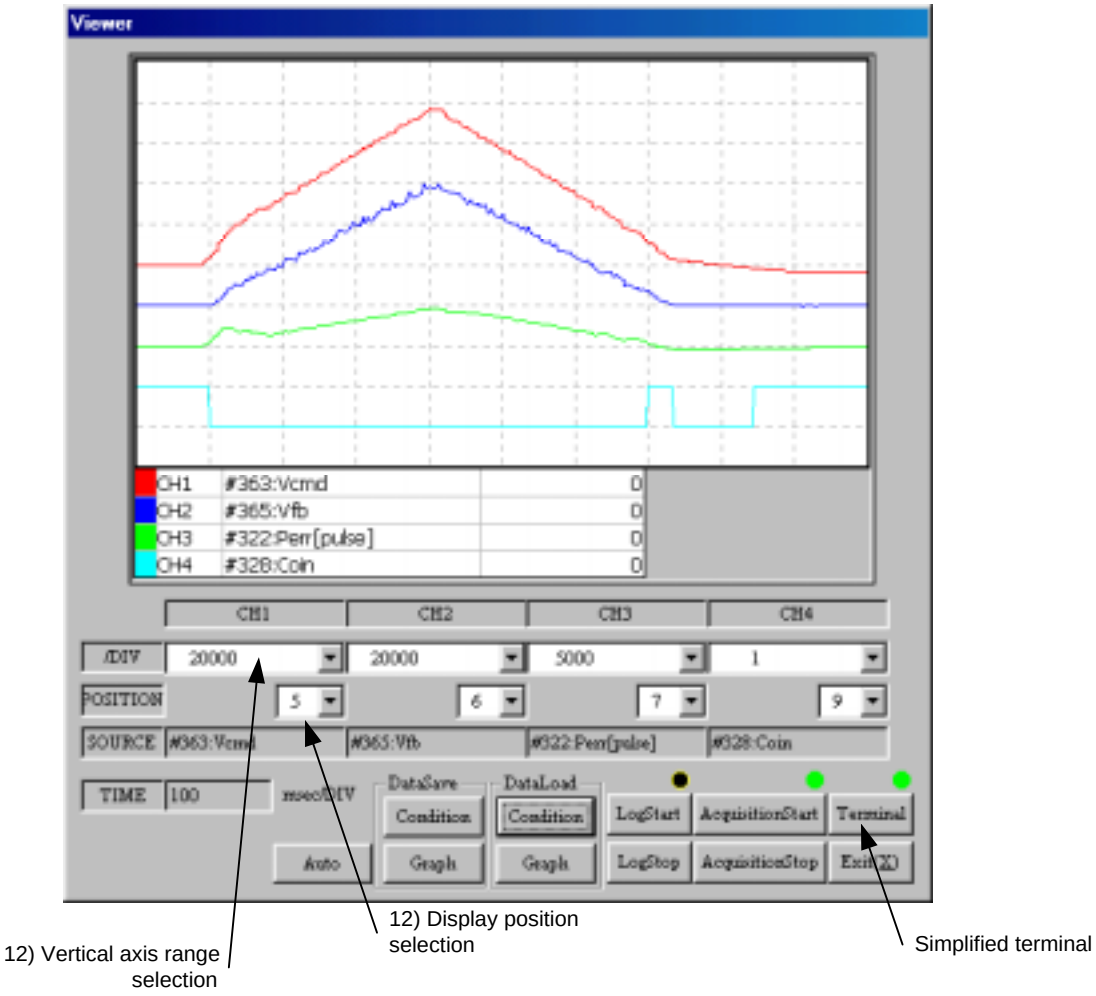


Figure 7.18 "Oscilloscope" dialog box

[How to use the oscilloscope]

(1) Click *"Log Start"* on the "Oscilloscope" dialog box to display the "SetCondition/ELogStart" dialog box (see Figure 7.19).

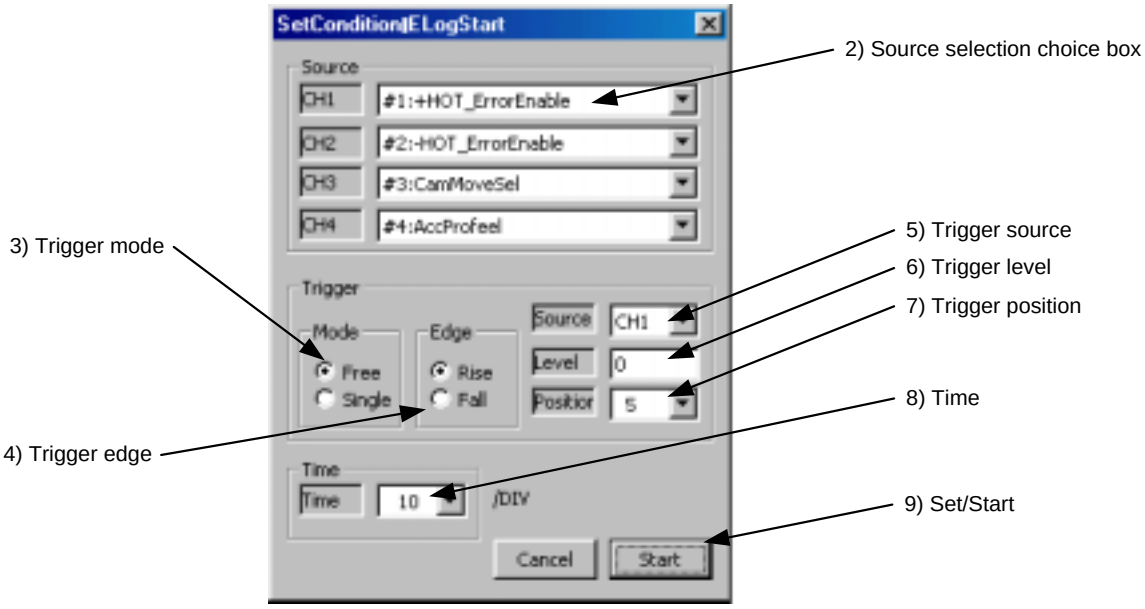


Figure 7.19 "SetCondition/ELogStart" dialog box

(2) Click the ↓ to select the parameters/monitors you want to display in the "source selection choice box." (CH1 to CH4)
(3) Select a trigger mode.
Free: Obtains data immediately without using the trigger.
Single: Obtains data when the trigger conditions are met.
(4) Select a trigger edge. (Valid when the trigger mode is Single.)
(5) Click the ↓ to select a trigger source. (Valid when the trigger mode is Single.)
(6) Enter a trigger level. (Valid when the trigger mode is Single.)
(7) Click the ↓ to select a trigger position. (Valid when the trigger mode is Single.)
(8) Click the ↓ to select a time (horizontal axis). (The unit is msec.)
(9) Click *"Set/Start"* to return to the "Oscilloscope" dialog box and wait for the completion of data acquisition.
(10) When the data is obtained, *"Start Acquisition"* becomes active.
(11) Click *"Start Acquisition"* to extract data from the driver and display it in the "Oscilloscope" dialog box.

Note: If the set trigger conditions are not satisfied and *"Start Acquisition"* does not become active, click *"Log Stop"* and set the trigger conditions again.

(12) The displayed waveform can be reshaped using “Display position selection” and “Vertical axis range selection” in the "Oscilloscope" dialog box.
When *"Auto"* is clicked at this time, the "Display position selection" value is changed to "5," and the "Vertical axis range selection" value is changed automatically to a value that enables to display the entire waveform as much as possible without cutting any part of the waveform.

(13) When *"Condition and display"* under "Data save" is clicked, the status set in the "SetCondition/ELogStart" dialog box as well as the values set by "Display position selection" and "Vertical axis range selection" can be saved by assigning a file name.
In addition, when *"Graph"* under "Data save" is clicked, the waveform currently being displayed, the status set in the "SetCondition/ELogStart" dialog box, and the values set by "Display position selection" and "Vertical axis range selection" can be saved by assigning a file name.

* The status set in the "SetCondition/ELogStart" dialog box as well as the values set by "Display position selection" and "Vertical axis range selection" are automatically saved in the "oscscope.cnd" file when the "Oscilloscope" dialog box is closed.

(14) When *"Condition and display"* under "Data load" is clicked, the setting status of the "SetCondition/ELogStart" dialog box as well as the values set by "Display position selection" and "Vertical axis range selection" are loaded from files, and then displayed. The waveform is cleared at this time and all values are set to 0.
In addition, when *"Graph"* under "Data save" is clicked, the waveform data, the setting status in the "SetCondition/ELogStart" dialog box, and the values set by "Display position selection" and "Vertical axis range selection" are loaded from files, and then a waveform is displayed.

Note: If the trigger condition that has been set cannot be established and *"Start acquisition"* cannot be activated, click *"Log stop"* and set a new trigger condition again.

* The simplified terminal function is provided in the "Oscilloscope" dialog box. Use this function in order to change parameter values or to start an operation.

7.5 Action Menu

In the operation menu, you can set and display parameters, display monitors, and start or stop actions related to the operations listed below.
"Homing move," "signal search move," "index Type A operation," "index Type B operation," "table reference operation," and *"jog move."*

Click *"Drive (D)"* under "MainMenu" to display the "DriveMenu" dialog box (see Figure 7.20).

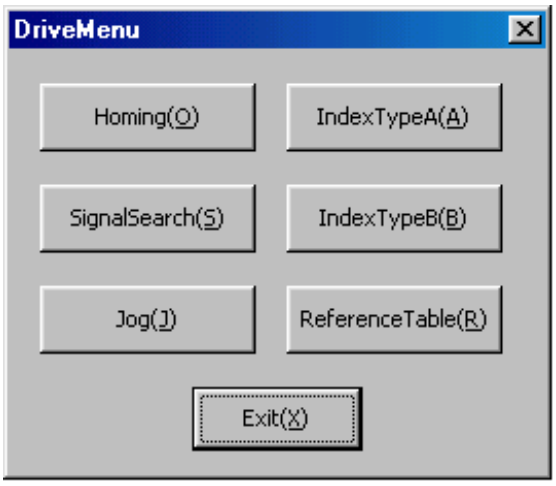


Figure 7.20 "DriveMenu" dialog box

7.5.1 Homing

Click *"Homing (O)"* in the "DriveMenu" dialog box to display the "Homing" dialog box (see Figure 7.21). If the connection with the driver is established, the current values of the related parameters are read and can be edited.

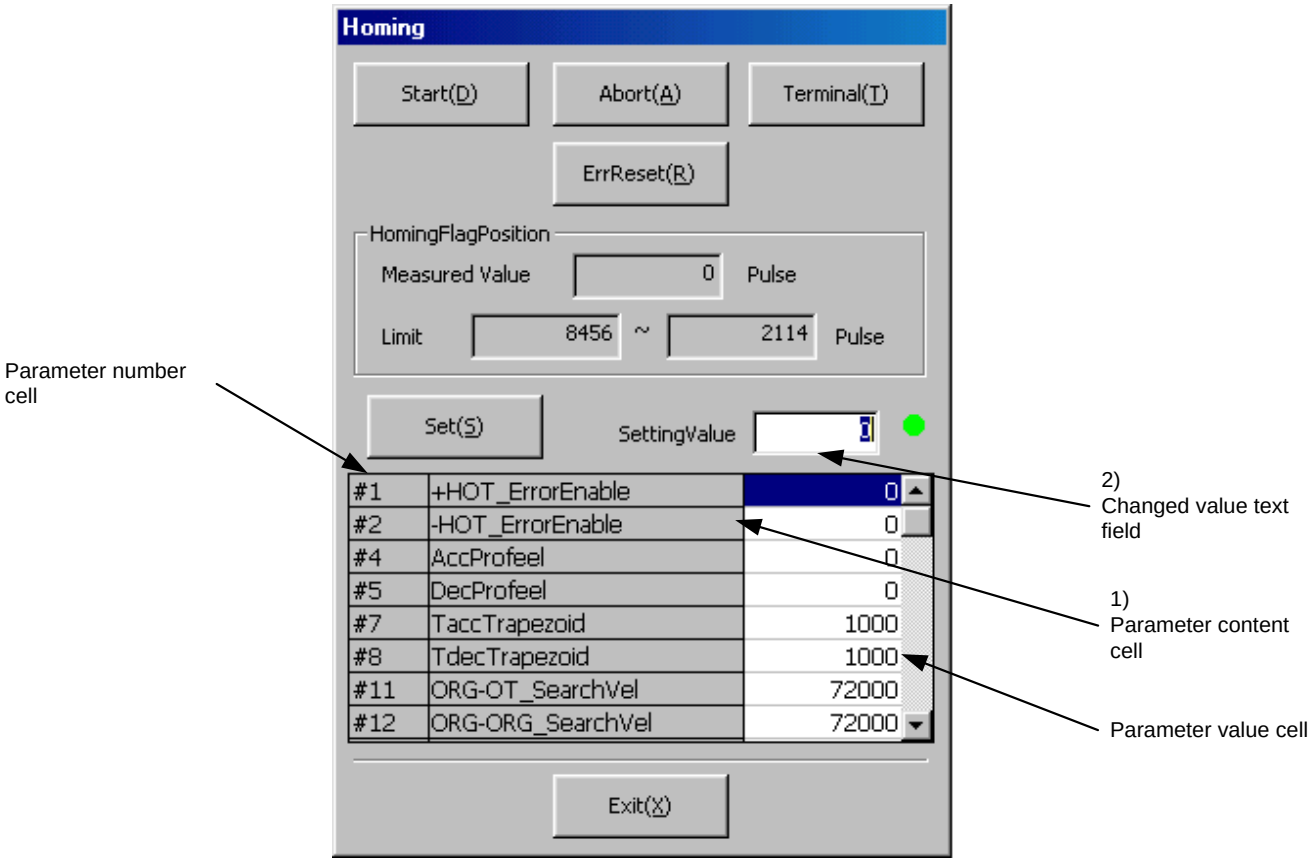


Figure 7.21 "Homing" dialog box

(1) Editing parameters

1) Click the parameter content cell of the parameter to be changed. The current value is displayed in the changed value text field.
2) Enter a value in the changed value text field and click the Enter (Return) key or click another cell to make the new parameter value valid.
3) Repeat steps 1) and 2) as necessary and click *"Set (S)"* to set the changed values in the driver. If there is an error in the set value, a warning message is displayed and the parameter is read again. Parameters whose setting values are erroneous will not be changed; therefore the changed parameters should be verified.

(2) Operation

Operation of the motor is possible when the main operation mode is set to the RS232C main operation mode (see Chapter 5). (In the PLC main operation mode, several buttons are disabled.)

[Start]

1) Set the motor to Servo ON (see Chapter 5).
2) Click *"Start (D)."*
   → The measured value of the homing is displayed after the homing operation is finished.

[Abort]

1) Click *"Abort (A)."*
   → The motor decelerates and stops.

[Error Reset]

1) Click *"ErrReset (R)."*
   → Errors that can be recovered are canceled.

[Simplified terminal]

1) Click *"Terminal (T)."*
2) Send or receive character strings.

7.5.2 Signal Search Move

Click *"SignalSearch (S)"* in the "DriveMenu" dialog box to display the "Signal Search" dialog box (see Figure 7.22). If the connection with the driver is established, the current values of the related parameters are read and can be edited.

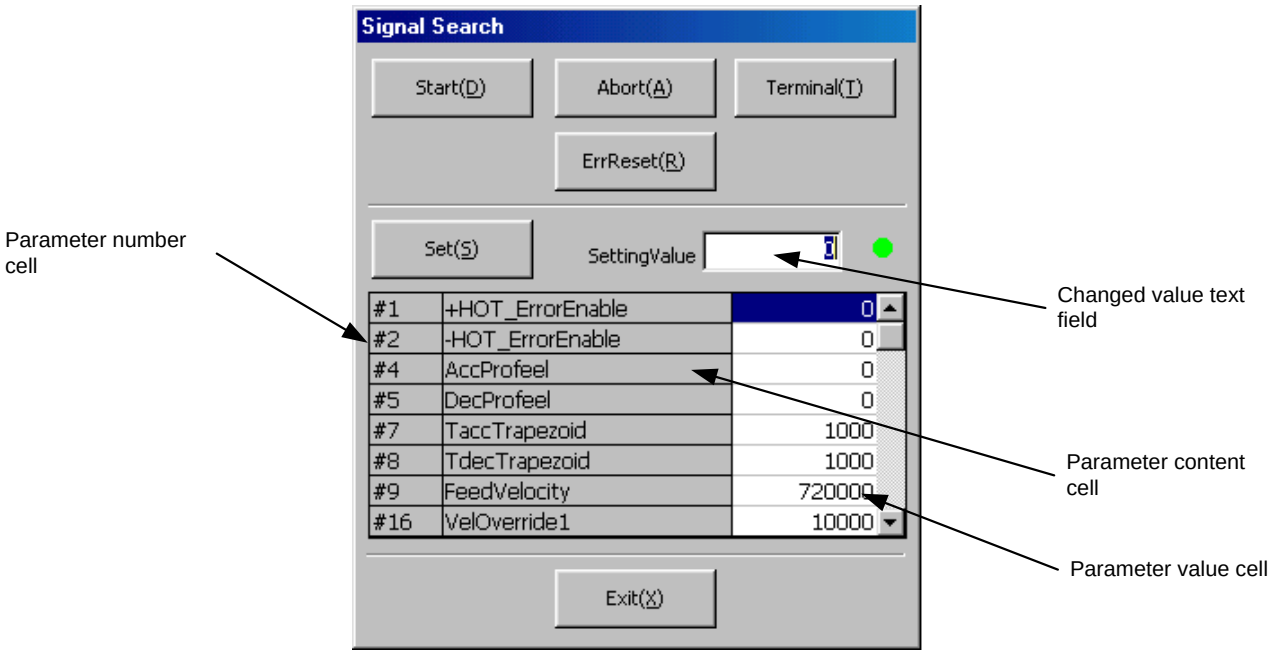


Figure 7.22 "Signal Search" dialog box

(1) Editing parameters

1) Click the parameter value cell of the parameter to be changed. The current value is displayed in the changed value text field.
2) Enter a value in the changed value text field and click the Enter (Return) key or click another cell to make the new parameter value valid.
3) Repeat steps 1) and 2) as necessary and click *"Set (S)"* to set the changed values in the driver. If there is an error in the set value, a warning message is displayed and the parameter is read again. Parameters whose setting values are erroneous will not be changed; therefore the changed parameters should be verified.

(2) Operation

Operation of the motor is possible when the main operation mode is set to the RS232C main operation mode (see Chapter 5). (In the PLC main operation mode, several buttons are disabled.)

[Start]
1) Set the motor to Servo ON (see Chapter 5).
2) Click *"Start (D)."*

[Abort]
1) Click *"Abort (A)."*
   → The motor decelerates and stops.

[Error Reset]
1) Click *"ErrReset (R)."*
   → Errors that can be recovered are canceled.

[Simplified terminal]
1) Click *"Terminal (T)."*
2) Send or receive character strings.

7.5.3 Index Type A Operation

Click *"IndexTypeA (A)"* in the "DriveMenu" dialog box to display the "Index TypeA" dialog box (see Figure 7.23). If the connection with the driver is established, the current values of the related parameters are read and can be edited.

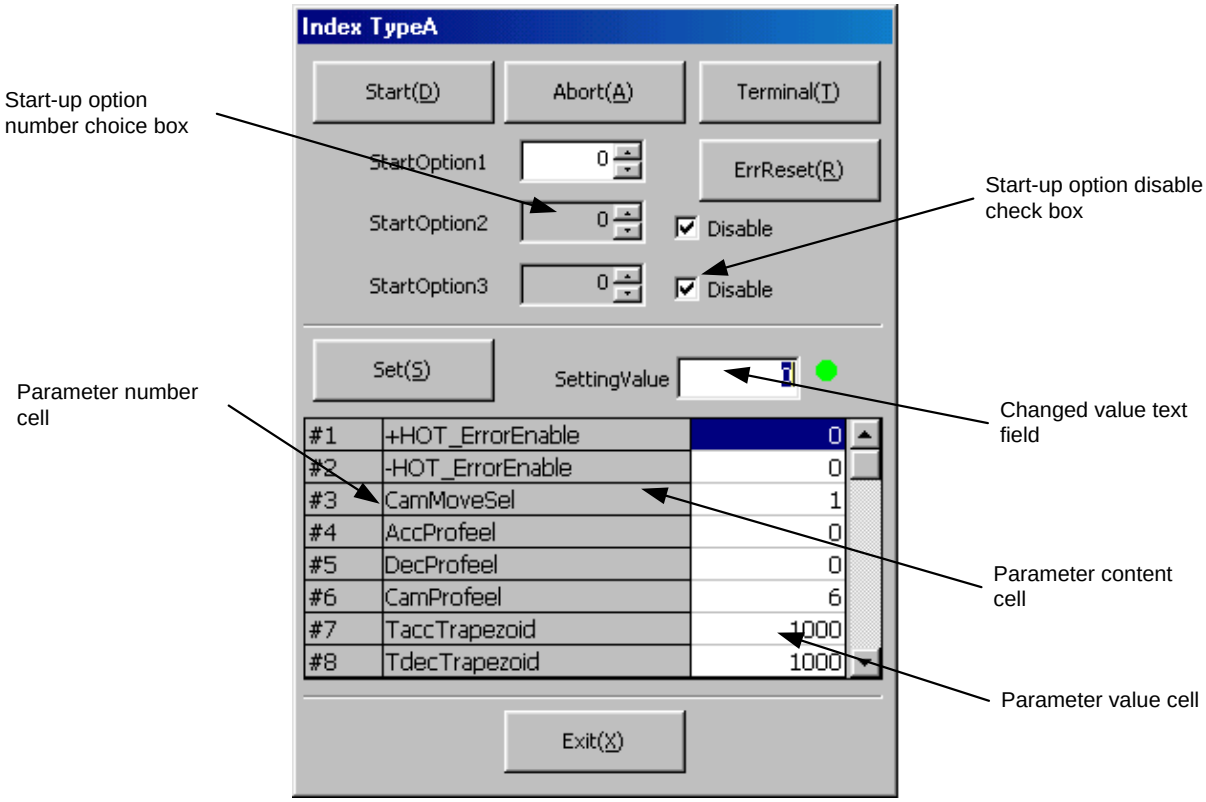


Figure 7.23 "Index TypeA" dialog box

(1) Editing parameters

1) Click the parameter value cell of the parameter to be changed. The current value is displayed in the changed value text field.
2) Enter a value in the changed value text field and click the Enter (Return) key or click another cell to make the new parameter value valid.
3) Repeat steps 1) and 2) as necessary and click *"Set (S)"* to set the changed values in the driver. If there is an error in the set value, a warning message is displayed and the parameter is read again. Parameters whose setting values are erroneous will not be changed; therefore the changed parameters should be verified.

(2) Editing start-up options

1) Select the number in the start-up option number choice box for the start-up option to be changed. However, it is not possible to edit when the start-up option disable check box is set to disabled.
2) If start-up options are not necessary, the start-up option disable check box should be clicked to disable the start-up option number choice box. In this case, the disabled start-up options will not be set.

(3) Operation

Operation of the motor is possible when the main operation mode is set to the RS232C main operation mode (see Chapter 5). (In the PLC main operation mode, several buttons are disabled.)

[Start]
1) Set the motor to Servo ON (see Chapter 5).
2) Click *"Start (D)."*

[Abort]
1) Click *"Abort (A)."*
   → The motor decelerates and stops.

[Error Reset]
1) Click *"ErrReset (R)."*
   → Errors that can be recovered are canceled.

[Simplified terminal]
1) Click *"Terminal (T)."*
2) Send or receive character strings.

7.5.4 Index Type B Operation

Click *"IndexTypeB (B)"* in the "DriveMenu" dialog box to display the "Index TypeB" dialog box (see Figure 7.24). If the connection with the driver is established, the current values of the related parameters are read and can be edited.

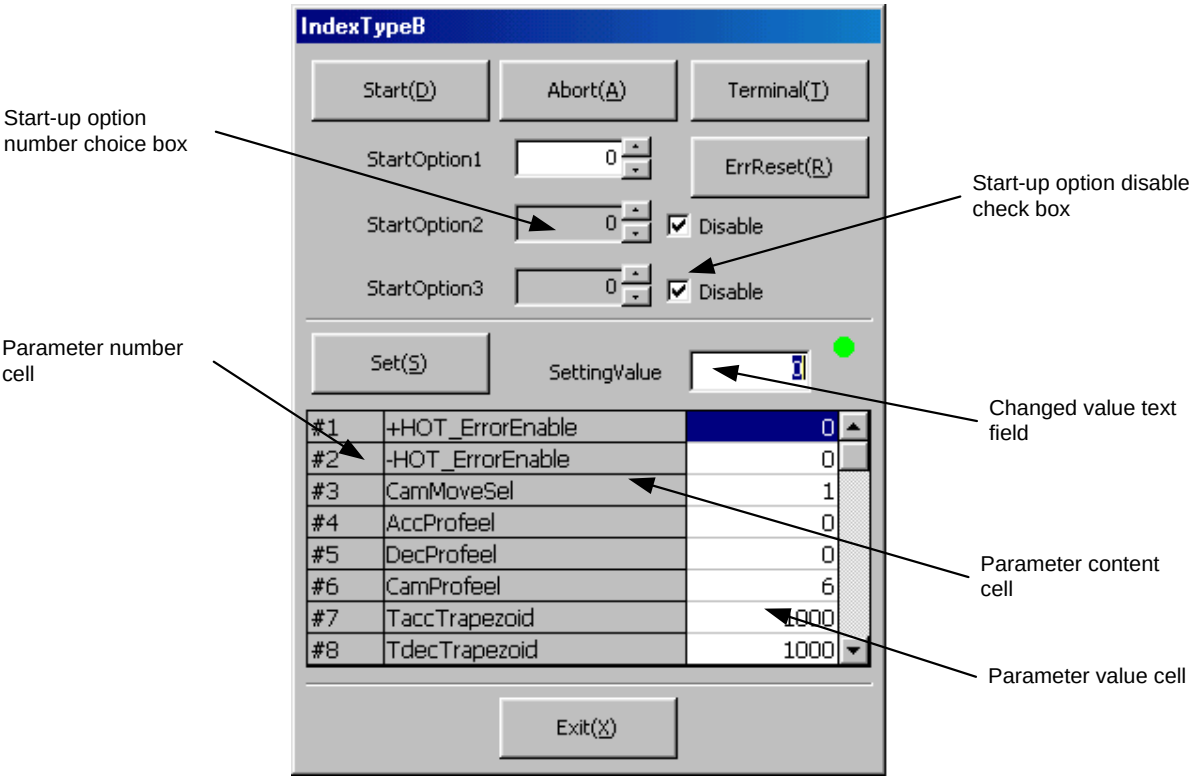


Figure 7.24 "Index TypeB" dialog box

(1) Editing parameters

1) Click the parameter value cell of the parameter to be changed. The current value is displayed in the changed value text field.
2) Enter a value in the changed value text field and click the Enter (Return) key or click another cell to make the new parameter value valid.
3) Repeat steps 1) and 2) as necessary and click *"Set (S)"* to set the changed values in the driver. If there is an error in the set value, a warning message is displayed and the parameter is read again. Parameters whose setting values are erroneous will not be changed; therefore the changed parameters should be verified.

(2) Editing start-up options

1) Select the number in the start-up option number choice box for the start-up option to be changed. However, it is not possible to edit when the start-up option disable check box is set to disabled.
2) If start-up options are not necessary, the start-up option disable check box should be clicked to disable the start-up option number choice box. In this case, the disabled start-up options will not be set.

(3) Operation

Operation of the motor is possible when the main operation mode is set to the RS232C main operation mode (see Chapter 5). (In the PLC main operation mode, several buttons are disabled.)

[Start]
1) Set the motor to Servo ON (see Chapter 5).
2) Click *"Start (D)."*

[Abort]
1) Click *"Abort (A)."*
   → The motor decelerates and stops.

[Error Reset]
1) Click *"ErrReset (R)."*
   → Errors that can be recovered are canceled.

[Simplified terminal]
1) Click *"Terminal (T)."*
2) Send or receive character strings.

7.5.5 Table Reference Operation

Click "*ReferenceTable (R)*" in the "DriveMenu" dialog box to display the "Reference Compensation Table" dialog box (see Figure 7.25). If the connection with the driver is established, the current values of the related parameters are read and can be edited.

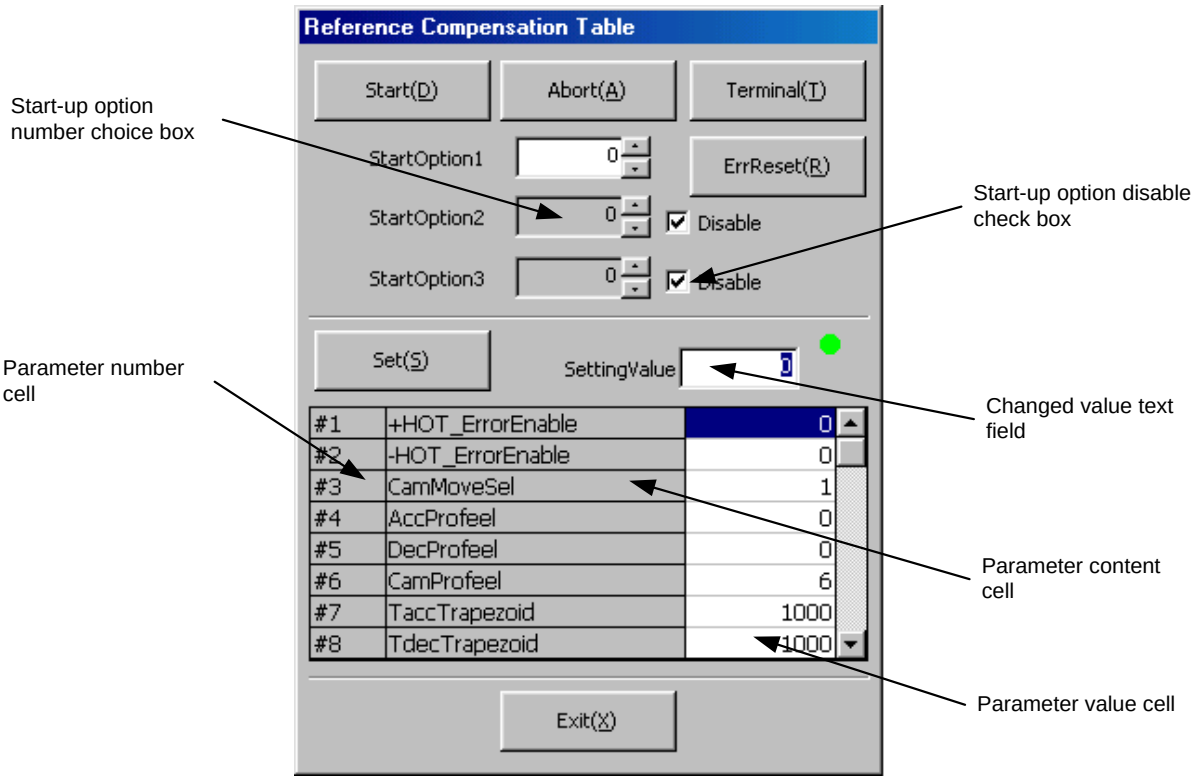


Figure 7.25 "Reference Compensation Table" dialog box

(1) Editing parameters

1) Click the parameter value cell of the parameter to be changed. The current value is displayed in the changed value text field.
2) Enter a value in the changed value text field and click the Enter (Return) key or click another cell to make the new parameter value valid.
3) Repeat steps 1) and 2) as necessary and click *"Set (S)"* to set the changed values in the driver. If there is an error in the set value, a warning message is displayed and the parameter is read again. Parameters whose setting values are erroneous will not be changed; therefore the changed parameters should be verified.

(2) Editing start-up options

1) Select the number in the start-up option number choice box for the start-up option to be changed. However, it is not possible to edit when the start-up option disable check box is set to disabled.
2) If start-up options are not necessary, the start-up option disable check box should be clicked to disable the start-up option number choice box. In this case, the disabled start-up options will not be set.

(3) Operation

Operation of the motor is possible when the main operation mode is set to the RS232C main operation mode (see Chapter 5). (In the PLC main operation mode, several buttons are disabled.)

[Start]
1) Set the motor to Servo ON (see Chapter 5).
2) Click *"Start (D)."*

[Abort]
1) Click *"Abort (A)."*
   → The motor decelerates and stops.

[Error Reset]
1) Click *"ErrReset (R)."*
   → Errors that can be recovered are canceled.

[Simplified terminal]
1) Click *"Terminal (T)."*
2) Send or receive character strings.

7.5.6 Jog Move

Click *"Jog (J)"* in the "DriveMenu" dialog box to display the "JogMove" dialog box (see Figure 7.26). If the connection with the driver is established, the current values of the related parameters are read and can be edited.

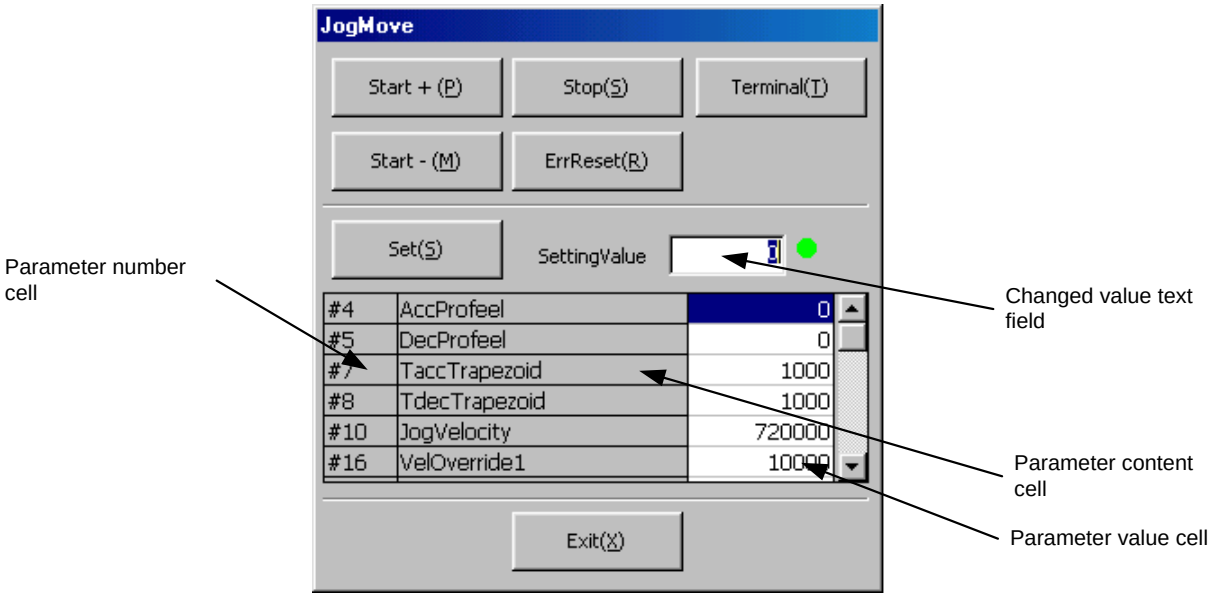


Figure 7.26 "JogMove" dialog box

(1) Editing parameters

1) Click the parameter value cell of the parameter to be changed. The current value is displayed in the changed value text field.
2) Enter a value in the changed value text field and click the Enter (Return) key or click another cell to make the new parameter value valid.
3) Repeat steps 1) and 2) as necessary and click *"Set (S)"* to set the changed values in the driver. If there is an error in the set value, a warning message is displayed and the parameter is read again. Parameters whose setting values are erroneous will not be changed; therefore the changed parameters should be verified.

(2) Operation

The following operation can be performed when jog move is selected via the RS232C interface (see Chapter 5). (When the main operation mode is set to the PLC main operation mode, some buttons are disabled.)

[Jog move in positive direction]
1) Set the motor to Servo ON (see Chapter 5).
2) Click *"Start + (P)."*

[Jog move in negative direction]
1) Set the motor to Servo ON (see Chapter 5).
2) Click *"Start – (M)."*

[Jog stop]
1) Click *"Stop (S)."*

[Error Reset]
1) Click *"ErrReset (R)."*
   → Recoverable errors are canceled.

[Simplified terminal]
1) Click *"Terminal (T)."*
2) Send or receive character strings.

7.6 Data Management Menus

7.6.1 Parameter Manager

This menu allows you to save all the parameters to files and register them from files in addition to edit the machine setting parameters.

Click *"Parameter (M)"* under "MainMenu" to display the "Parameter Manager" dialog box (see Figure 7.27).

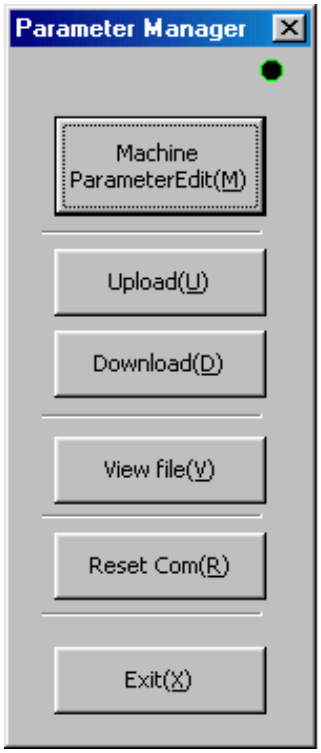


Figure 7.27 "Parameter Manager" dialog box

(1) Editing machine parameters

Click *"Machine Parameter Edit (M)"* in the "Parameter Manager" dialog box to display the "MachineParameterEdit" dialog box (see Figure 7.28). If the connection with the driver is established, the current values of the machine parameters are read and can be edited.

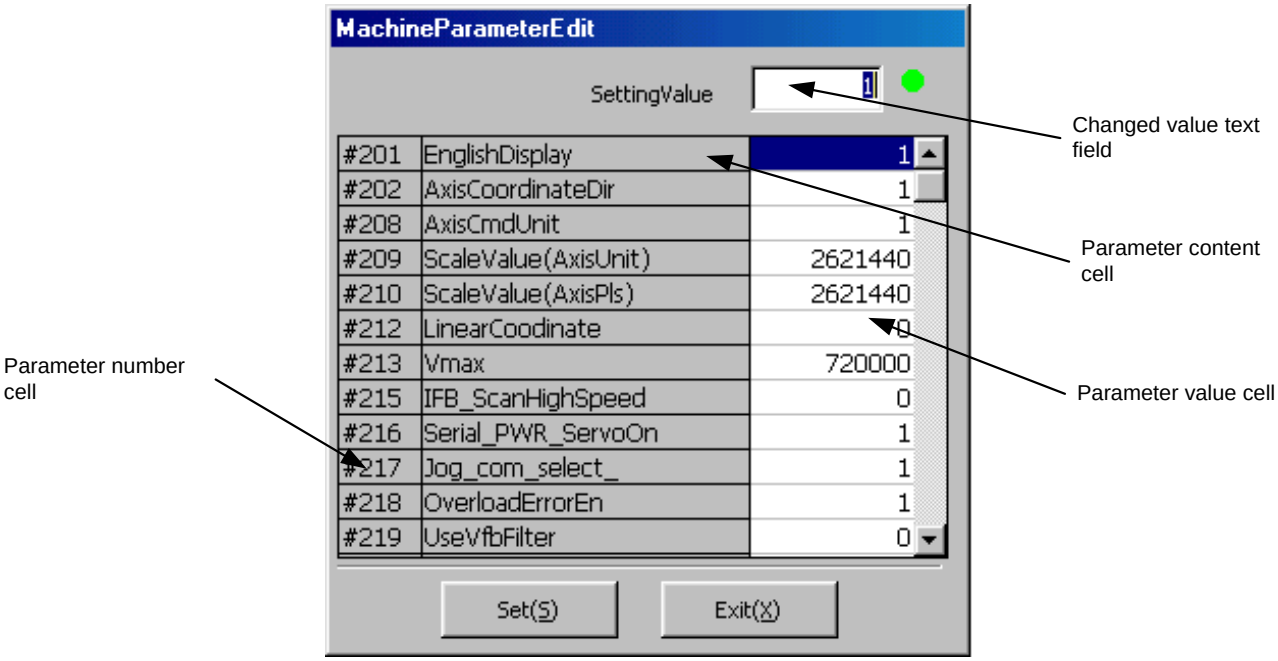


Figure 7.28 "MachineParameterEdit" dialog box

[Editing]

1) Click the parameter value cell of the parameter to be changed. The current value is displayed in the changed value text field.
2) Enter a value in the changed value text field and click the Enter (Return) key or click another cell to make the new parameter value valid.
3) Repeat steps 1) and 2) as necessary and click *"Set (S)"* to start downloading to the driver. (If you click *"Exit (X),"* the parameter values are not changed.)
4) When the downloading is finished, follow the message and reset the driver.

Note: If any erroneous data are set, the parameters that could not be downloaded are displayed. In this case, reset the driver once and set the parameters again.

Note: In the case of the machine setting parameters, the changed values cannot be updated until you reset the driver.

(2) Uploading (from the driver to a file)

1) Click *"Upload (U)"* in the "Parameter Manager" dialog box.
2) Enter the name of the file to which parameters are to be saved. Do not enter a file extension (*.prm); it is added automatically.
3) Click *"Save (S)"* to start uploading. If you wish to stop the uploading, click *"Cancel."*

(3) Downloading (from a file to the driver)

1) Click *"Download (D)"* in the "Parameter Manager" dialog box.
2) Enter the name of the file to be downloaded to the driver. Do not enter a file extension (*.prm); it is added automatically.
3) Click *"Open (O)"* to start downloading. If you wish to stop the downloading, click *"Cancel."*
4) When the downloading is finished, follow the message and reset the driver.

(4) Displaying the file contents

1) Click *"View file (V)"* in the "Parameter Manager" dialog box.
2) Enter the name of the file you want to display. Do not enter a file extension (*.prm); it is added automatically.
3) Click *"Open (O)"* to begin displaying the file contents in the "Parameter file display" dialog box (see Figure 7.29).
4) If you want to print the file, click *"Print (P)."*
5) Click *"Exit (X)"* and return to the "Parameter Manager" dialog box.

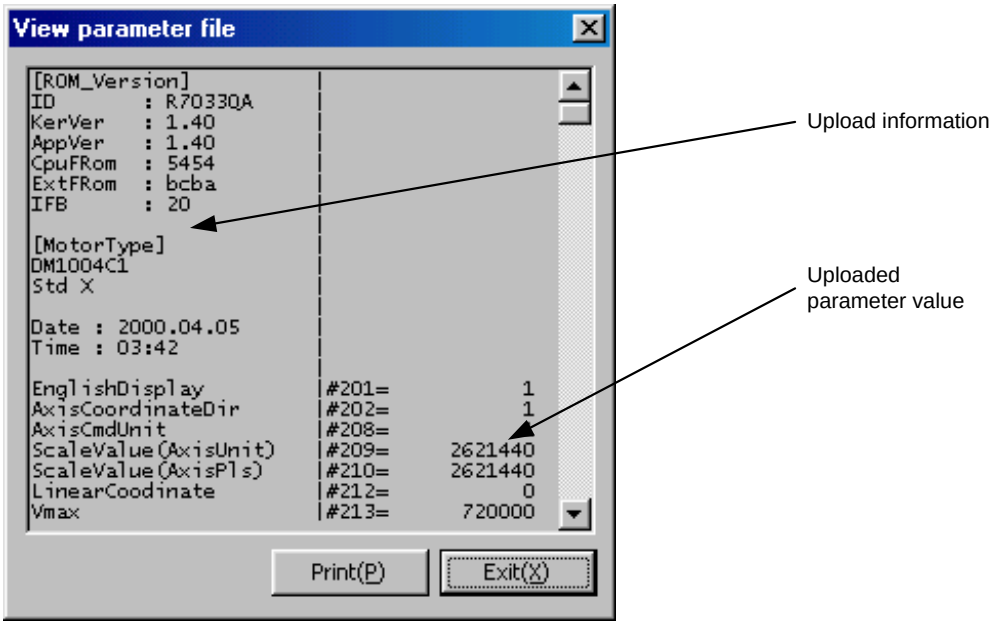


Figure 7.29 "Parameter file display" dialog box

(5) Resetting the communication

If the communication with the driver finishes abnormally, click *"Reset Com (R)"* in the "Parameter Manager" dialog box to return to the normal status.

7.6.2 Program Manager

This menu allows you to edit, register, and save the programs necessary for program operation. In addition, you can start or stop a program and monitor the program block currently being executed.

Click *"Program (G)"* under "MainMenu" to display the "Program Manager" dialog box (see Figure 7.30).

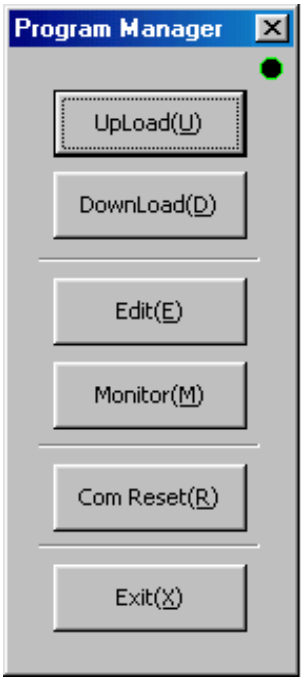


Figure 7.30 "Program Manager" dialog box

(1) Uploading (from the driver to a file)

Programs can be uploaded by the following two methods:

- "Normal upload": Upload a single program.
- "Upload all": Upload all the user programs registered in the driver.

Click "*Upload (U)"* in the "Program Manager" menu to display the "Select type" dialog box (see Figure 7.31).

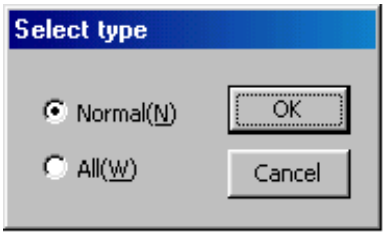


Figure 7.31 "Select type" dialog box

[Normal upload]

1) Click *"Normal (N)"* in the "Select type" dialog box.
2) The "Properties" dialog box is displayed where you can enter a program number. Enter a program number between 0 and 89 (user programs).
   (90 to 99 are system programs.)
3) Enter the name of the file in which the program is to be saved. Do not enter a file extension (*.prg); it is added automatically.
4) Click *"Save (S)"* to start uploading. If you wish to stop uploading, click *"Cancel."*

[Upload all]

1) Click "*All (W)"* in the "Select type" dialog box.
2) Enter the name of the file in which the programs are to be saved. Do not enter a file extension (*.whp); it is added automatically.
3) Click *"Save (S)"* to start uploading. If you wish to stop uploading, click *"Cancel."*

Note: When you use the upload all method, a batch program file is created together with each program (file extension .prg). A program number registered in the batch program file name is attached to each program file name.

(2) Downloading (from a file to the driver)

Programs can be downloaded by the following two methods:

- "Normal download": Download a single program.
- "Download all": Download all the user programs saved in the batch download file.

Click "*Download (D)"* in the "Program Manager" menu to display the "Select type" dialog box (see Figure 7.32).

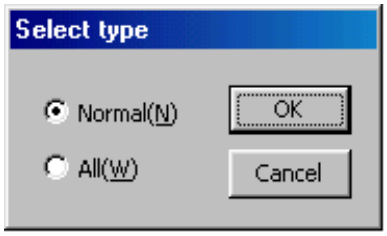


Figure 7.32 "Select type" dialog box

[Normal download]

1) Click *"Normal (N)"* in the "Select type" dialog box.
2) The "Properties" dialog box is displayed where you can enter a program number. Enter a program number between 0 and 89 (user programs). (90 to 99 are system programs.)
3) Enter the name of the file to be downloaded. Do not enter a file extension (*.prg); it is added automatically.
4) Click *"Open (O)"* to start downloading. If you wish to stop downloading, click *"Cancel."*

[Download all]

1) Click *"All (W)"* in the "Select type" dialog box.
2) Enter the name of the batch file to be downloaded. Do not enter a file extension (*.whp); it is added automatically.
3) Click *"Open (O)"* to start downloading. If you wish to stop downloading, click *"Cancel."*

(3) Editing programs

Via the Program Edit menu, you can create new programs and edit programs registered in the driver and files. In addition, edited files can be registered in the driver or saved in a file.

Click *"Edit (E)"* in the "Program Manager" menu to display the "Program Edit" dialog box (see Figure 7.33).

The editing screen is divided into two windows, an upper and a lower editor window, and either of them may be used for editing. It is effective to use the upper editor window for editing work and load an existing program into the lower editor window to cut and paste only the necessary parts. (Cut and paste can be performed through the basic Windows operations.)
Select the text with the mouse and then cut (Ctrl + X), copy (Ctrl + C), and paste (Ctrl + V).

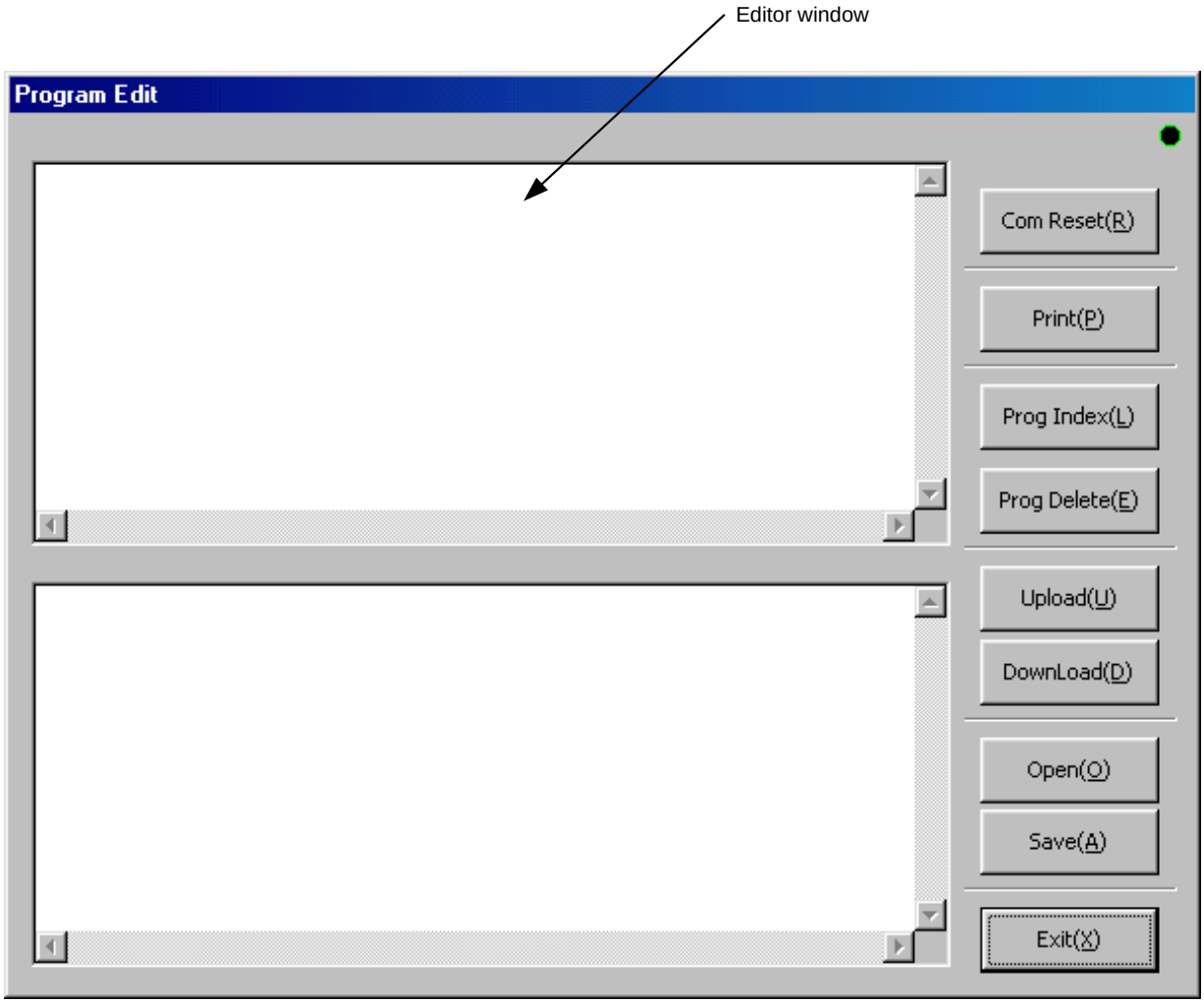


Figure 7.33 "Program Edit" dialog box

[Creating a new program]

1) Write a program on the upper or lower editor window following the program syntax (see Chapter 5).

[Editing a program registered in the driver]

1) Click *"Upload (U)"* in the "Program Edit" dialog box.
2) The "Properties" dialog box is displayed. In the "Editor window," select the window (upper/lower) in which to display the program and enter the program number.
3) Click *"OK."* The program is read from the driver and displayed in the editor window.
4) Edit or add to the program as necessary, following the program syntax (see Chapter 5).

[Editing a program saved in a file]

1) Click *"Open (O)"* in the "Program Edit" dialog box.
2) The "Properties" dialog box is displayed. In the "Editor window," select the window (upper/lower) in which to display the program and click *"OK."*
3) Enter the program file name in "File name" in the "Open file" dialog box, then click *"Open (O)."* The program is read from the file and displayed in the editor window.
4) Edit or add to the program as necessary, following the program syntax (see Chapter 5).

[Registering a program in the driver]

1) Click *"Download (D)"* in the "Program Edit" dialog box.
2) The "Properties" dialog box is displayed. In the "Editor window," select the window (upper/lower) in which the file you want to register is displayed and enter the program number to which the program is to be registered, then select any necessary registration options (see below) and click *"OK."*

[Registration option]

Required for Servo ON:

At program execution, the Servo ON status of the driver is checked. If the Servo is not in ON status, none of the program lines are executed and an error occurs. (If you do not check this option, the program is executed until it reaches a line that operates the motor, then if the motor is not in the Servo ON status at this point, an error occurs.)

Required for operation coordinate settling:

At program execution, the operation coordinate settling status is checked. If the operation coordinate system is not settled, none of the program lines are executed and an error occurs. (If you do not check this option, the program is executed until it reaches a line that makes the motor move incrementally, then if the operation coordinate system is not settled at this point, an error occurs.)

3) When a program error occurs, the registration stops and lines with errors are displayed in red.

[Saving a program in a file]

1) Click *"Save (A)"* in the "Program Edit" dialog box.
2) The "Properties" dialog box is displayed. In the "Editor window," select the window (upper/lower) in which the file you want to save is displayed and enter the program number to which the program is to be registered, then select any necessary registration options (see above) and click *"OK."*
3) Enter the program file name and click *"Save (S)."* Do not enter a file extension (*.prg); it is added automatically.

[Deleting a program]

1) Click *"Prog Delete (E)"* in the "Program Edit" dialog box.
2) The "Properties" dialog box is displayed. Enter the number of the program to be deleted and click *"OK."*

[Index of programs]

1) Click *"Prog Index (L)"* in the "Program Edit" dialog box. The program numbers registered in the driver are displayed along with the numbers of blocks used and the number of blocks remaining.

[Printing a program]

1) Click *"Print (P)"* in the "Program Edit" dialog box.
2) The "Properties" dialog box is displayed. In the "Editor window," select the window in which the file you want to print is displayed and click *"OK."*

[Resetting the communication]

1) If the communication with the driver was terminated abnormally, click *"Reset Com (R)"* in the "Program Edit" dialog box to return to the normal status.

(4) Program monitor

This menu allows you to monitor the program number and block number currently being executed. In addition, if you set the main operation mode to the RS232C main operation mode (see Chapter 5), programs can be started up.

Click *"Monitor (M)"* in the "Program Manager" dialog box to display the "Program Monitor Init" dialog box (see Figure 7.34).

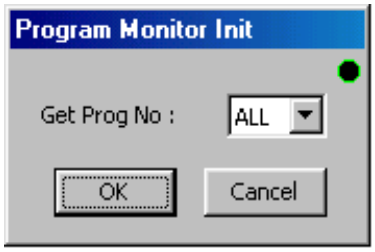


Figure 7.34 "Program Monitor Init" dialog box

In the "Program Monitor Init" dialog box, you can select the program number whose program contents you wish to acquire. If you select [All], the contents of all the programs registered to the driver are obtained. The contents of programs not acquired cannot be displayed at monitoring. Click *"OK"* to start acquiring the contents of the programs. Click *"Cancel"* to return to the "Program Manager" dialog box.

When the acquisition of the program contents is finished, the "Program Monitor" dialog box is displayed and the "Setting Option" dialog box is displayed on top of it (see Figure 7.35).

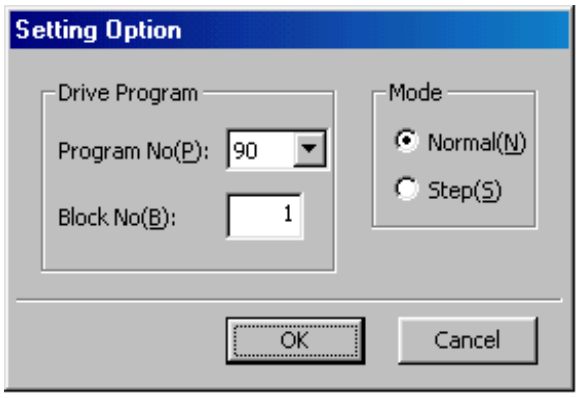


Figure 7.35 "Setting Option" dialog box

[Setting an option]

1) Enter the numbers of the program and block to be executed.
2) Select the operation mode (if you select step execution, the program stops after each execution of a block).
3) Click *"OK"* to make the setting status valid and return to the "Program Monitor" dialog box. If you click *"Cancel,"* the PC utility returns to the "Program Monitor" dialog box without making the setting status valid (see Figure 7.36).

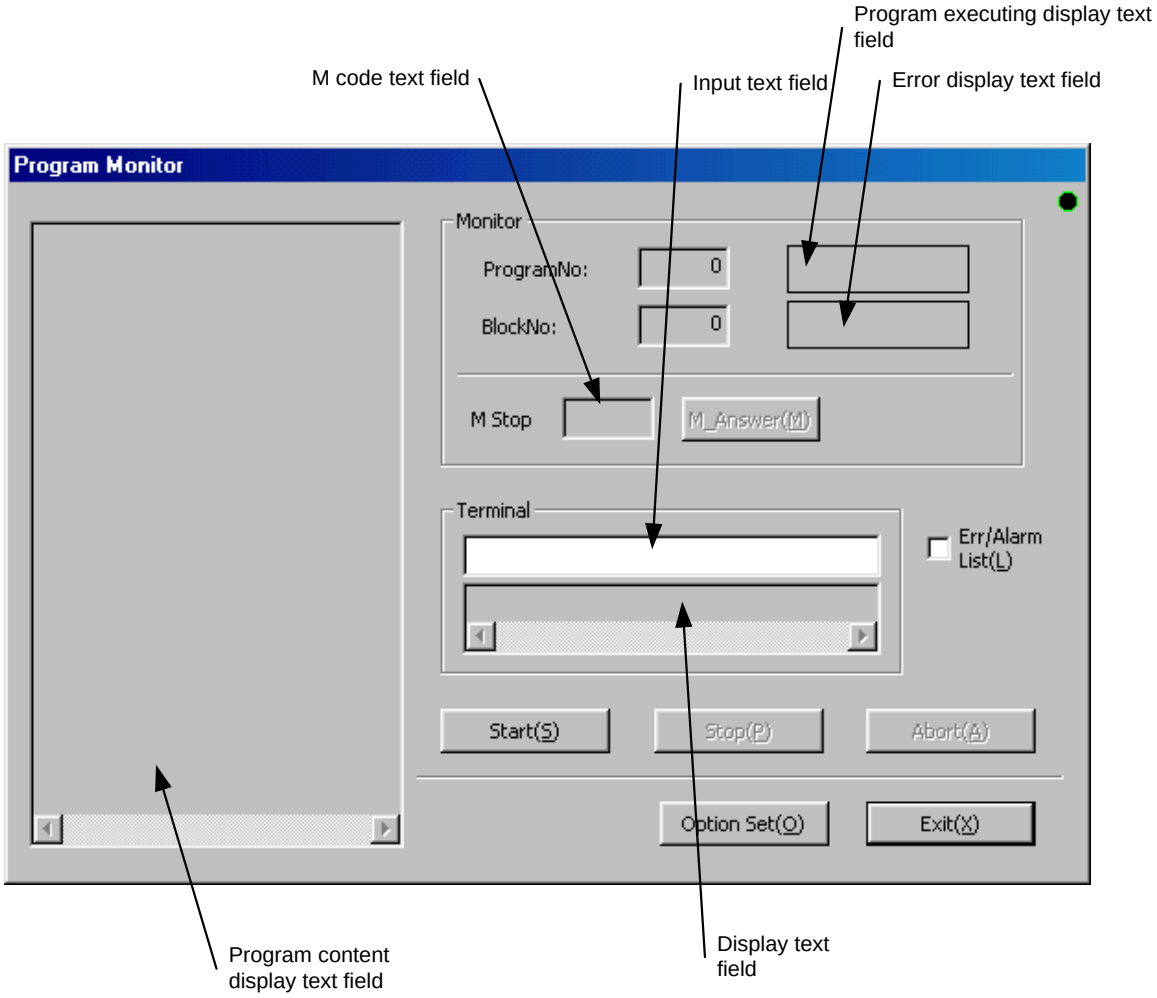


Figure 7.36 "Program Monitor" dialog box.

[Starting a program]

1) When you click *"Start (S),"* the message "Program being executed" is displayed in the "Monitor" box. During the execution of a program, the program number and block number are displayed and, in the program content display text field, the block that is being executed is displayed in yellow.

[Stopping a program]

1) Click *"Stop (P)"* during execution of a program to finish the execution of the current block and stop the program. If you click *"Abort (A),"* the program that is being executed is immediately terminated.

[Simplified terminal]

1) Enter a character string in the input text field and press the Enter (Return) key to send the character string to the driver.
2) The character string is displayed in the display text field when it is received from the driver.

[M answer]

1) If serial communication is selected for the M function interface, click *"M Answer (M)"* to send an M answer in response to the M code output in a program.

[Processing at error/alarm occurrence]

1) If an error occurs, "Error" is displayed in the error display text field. Click *"Error or Alarm (L)"* to display the "Error or Alarm" dialog box (see Figure 7.37). To reset an error status, click *"ErrReset (R)."*

Note: If you leave the "Error or Alarm" dialog box displayed, errors and alarms are not updated. Close the "Error or Alarm" dialog box once and display it again. Or, click *"Update (F)."*

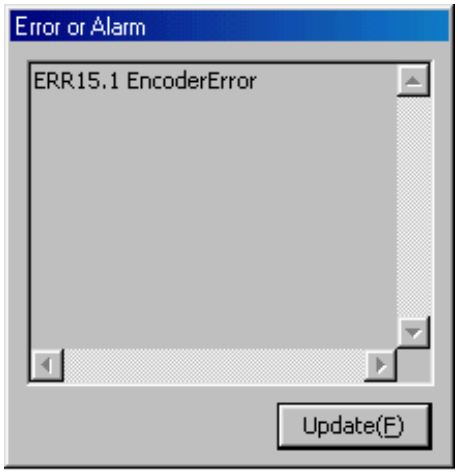


Figure 7.37 "Error or Alarm" dialog box

7.6.3 Index Menu

This menu allows you to edit, register, and save data necessary for “index Type A operation (equal division compensation)” and “index Type B operation (unequal division compensation).”

Click *“Index Comp (N)”* under “MainMenu” to display the “Index Menu” dialog box (Figure 7.38).

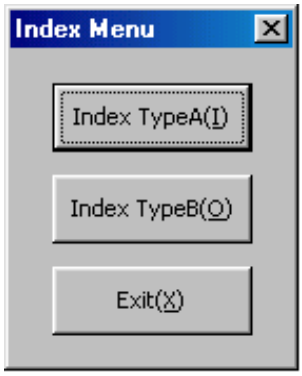


Figure 7.38 “Index Menu” dialog box

(1) Equal division compensation

Click *“IndexTypeA (I)”* on “Index Menu” to display the “Index compensation” dialog box (see Figure 7.39).

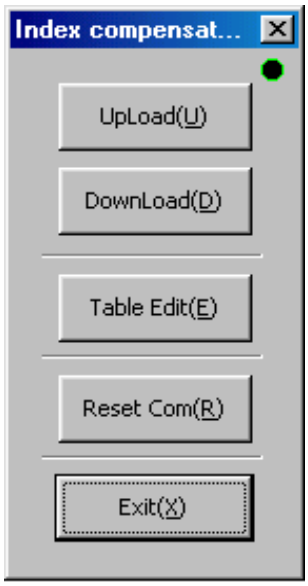


Figure 7.39 “Index compensation” dialog box

(2) Uploading (from the driver to a file)

Index compensation data can be uploaded by the following two methods:

- "Normal upload": Upload single index compensation data.
- "Upload all": Upload all the index compensation data registered in the driver.

Click *"Upload (U)"* in the "Index Comp" menu to display the "Select type" dialog box (see Figure 7.40).

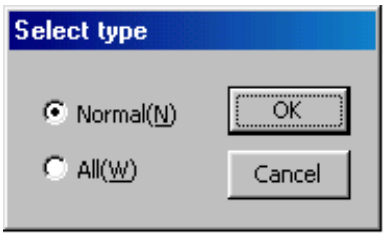


Figure 7.40 "Select type" dialog box

[Normal upload]

1) In the "Select type" dialog box, click "Normal (N)" and then *"OK."*
2) Select the file to upload using the ↓ and click *"OK."*
   Two types of index compensation files (files A and B) are available for the driver.
3) Enter the name of the file in which to save the compensation file. Do not enter a file extension (*.idx); it is added automatically.
4) Click *"Save (S)"* to start uploading. If you wish to stop uploading, click *"Cancel."*

[Upload all]

1) In the "Select type" dialog box, click "All (W)" and then *"OK."*
2) Enter the name of the batch index compensation file. Do not enter a file extension (*.whi); it is added automatically.
3) Click *"Save (S)"* to start uploading. If you wish to stop uploading, click *"Cancel."*

Note: When you use the upload all method, a batch index compensation file is created together with each index compensation file (file extension .idx). A file number registered in the batch index compensation file name is attached to each index compensation file name.

(3) Downloading (from a file to the driver)

Index compensation data can be uploaded by the following two methods:

- "Normal download": Download single index compensation data.
- "Download all": Download all the index compensation data registered in the driver.

Click *"Download (D)"* in the "Index Comp" menu to display the "Select type" dialog box (see Figure 7.41).

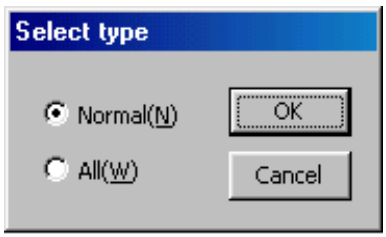


Figure 7.41 "Select type" dialog box

[Normal download]

1) Click *"Normal (N)"* in the "Select type" dialog box.
2) Select a file download using the ↓ and click *"OK."*
   Two types of index compensation file (files A and B) are available for the driver.
3) Enter the name of the file to register to the driver. Do not enter a file extension (*.idx); it is added automatically.
4) Click *"Open (O)"* to start downloading. If you wish to stop downloading, click *"Cancel."*

[Download all]

1) In the "Select type" dialog box, click "All (W)" and then *"OK."*
2) Enter the batch index compensation file name to register to the driver. Do not enter a file extension (*.whi); it is added automatically.
3) Click *"Open (O)"* to start downloading. If you wish to stop downloading, click *"Cancel."*

(4) Editing unequal division compensation data

You can edit compensation files registered in the driver or saved in files.

Click *"Table Edit (E)"* in the "Index compensation" dialog box to display the "Index Compensation Table" dialog box (see Figure 7.42).

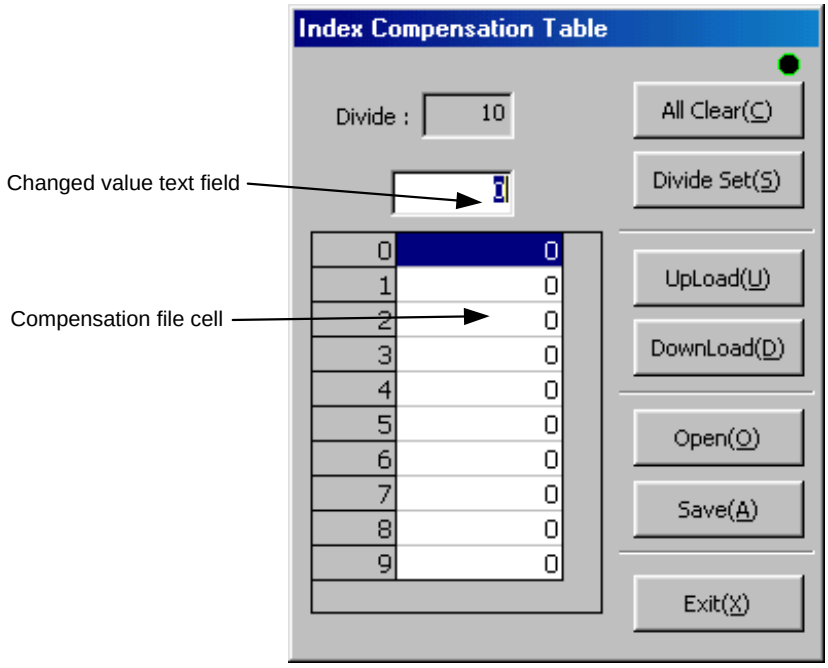


Figure 7.42 "Index Compensation Table" dialog box

[Creating a new compensation file]

1) Click *"Divide Set (S)"* and set the number of divisions.
2) Click the compensation file cell to be changed in order to display the current value in the changed value text field.
3) Enter a value in the changed value text field, then press the Enter (Return) key or click another compensation file cell. The value of the compensation file cell you want to change will then be updated.
4) Repeat steps 1) and 2) as necessary.

[Registering a compensation file (from the edit screen to the driver)]

1) Click *"Download (D)"* to display the "Select compensation file" dialog box.
2) Select the file number and click *"OK."*

[Saving a compensation file (from the edit screen to a file)]

1) Click *"Save (A),"* enter a file name, and click *"Save (S)."* (Do not enter a file extension (*.idx); it is added automatically.)

[Displaying a compensation file (from the driver to the edit screen)]

1) Click *"Upload (U)"* to display the "Select compensation file" dialog box.
2) Select the file number and click *"OK."*

[Displaying a compensation file (from a file to the edit screen)]

1) Click *"Open (O),"* enter the file name, and click *"Open (O)."* (Do not enter a file extension (*.idx); it is added automatically.)

[Clearing all]

1) Click *"Clear All (C)"* to clear all the values of the compensation file cell array to 0.

(5) Resetting communication

If the communication with the driver finishes abnormally, click *"Reset Com (R)"* in the "Index compensation" dialog box to return to the normal status.

(6) Unequal division

Click *"Index TypeB (O)"* in the "Index compensation" dialog box to display the "Index TypeB Menu" dialog box (see Figure 7.43).

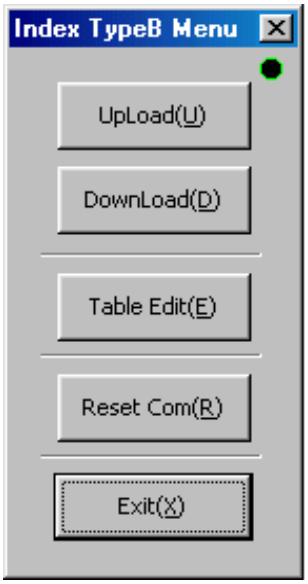


Figure 7.43 "Index TypeB Menu" dialog box

(7) Uploading (from the driver to a file)

Index type B data can be uploaded by the following two methods:

- "Normal upload": Upload single index type B data.
- "Upload all": Upload all the index type B data registered in the driver.

[Normal upload]

Click *"Upload (U)"* in the "Index TypeB Menu" to display the "Select type" dialog box (see Figure 7.44).

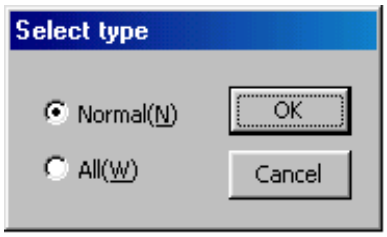


Figure 7.44 "Select type" dialog box

1) In the "Select type" dialog box, click "Normal (N)" and then *"OK."*
2) Select the file to upload using the ↓ and click *"OK."*
   Two types of index type B files (files A and B) are available for the driver.
3) Enter the name of the file in which to save the index type B file. Do not enter a file extension (*.idp); it is added automatically.
4) Click *"Save (S)"* to start uploading. If you wish to stop uploading, click *"Cancel."*

[Upload all]

1) In the "Select type" dialog box, click "All (W)" and then *"OK."*
2) Enter the name of the batch index type B file. Do not enter a file extension (*.whp); it is added automatically.
3) Click *"Save (S)"* to start uploading. If you wish to stop uploading, click *"Cancel."*

Note: When you use the upload all method, a batch index type B file is created together with each index type B file (file extension .idp). A file number registered in the batch index type B file name is attached to each index type B file name.

(8) Downloading (from a file to the driver)

Index type B data can be uploaded by the following two methods:

- "Normal download": Download single index type B data.
- "Download all": Download all the index type B data registered in the driver.

Click *"Download (D)"* in the "Index TypeB Menu" to display the "Select type" dialog box (see Figure 7.45).

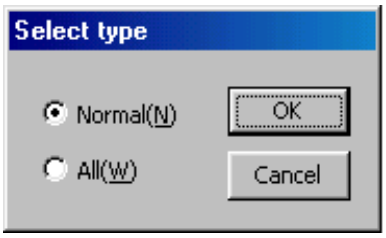


Figure 7.45 "Select type" dialog box

[Normal download]

1) In the "Select type" dialog box, click *"Normal (N)"* and then *"OK."*
2) Select a file download using the ↓ and click *"OK."*

Two types of index type B file (files A and B) are available for the driver.

3) Enter the name of the file to register to the driver. Do not enter a file extension (*.idp); it is added automatically.
4) Click *"Open (O)"* to start downloading. If you wish to stop downloading, click *"Cancel."*

[Download all]

1) In the "Select type" dialog box, click "All (W)" and then *"OK."*
2) Enter the batch index type B file name to register to the driver. Do not enter a file extension (*.whp); it is added automatically.
3) Click *"Open (O)"* to start downloading. If you wish to stop downloading, click *"Cancel."*

(9) Editing unequal division data

You can edit type B files registered in the driver or saved in files.

Click *"Table Edit (E)"* in the "Index Type B" dialog box to display the "Index TypeB Compensation Table" dialog box (see Figure 7.46).

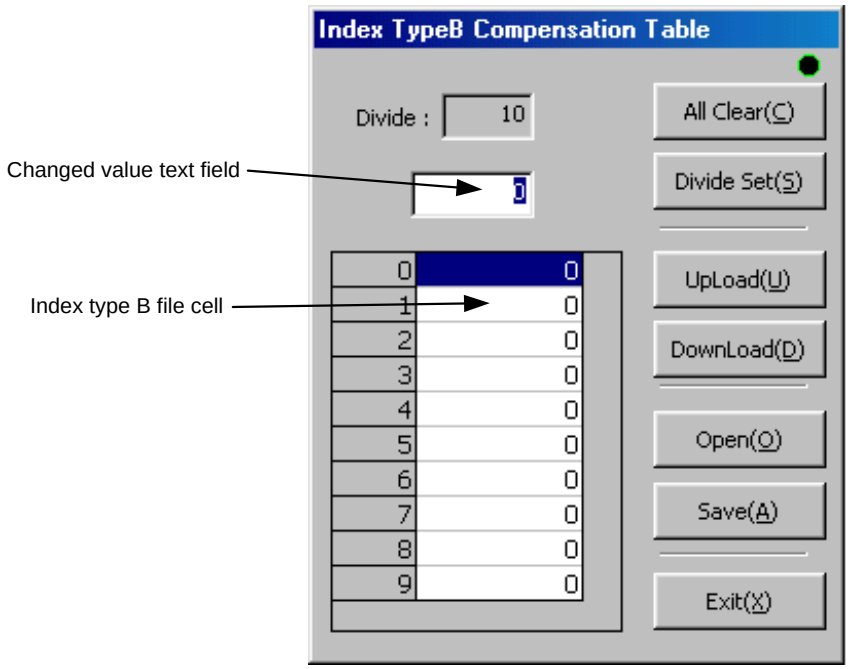


Figure 7.46 "Index TypeB Compensation Table" dialog box

[Creating a new compensation file]

1) Click *"Divide Set (S)"* and set the number of divisions.
2) Click the index type B file cell to be changed in order to display the current value in the changed value text field.
3) Enter a value in the changed value text field, then press the Enter (Return) key or click another index type B file cell. The value of the index type B file cell you want to change will then be updated.
4) Repeat steps 1) and 2) as necessary.

[Registering an index type B file (from the edit screen to the driver)]

1) Click *"Download (D)"* to display the "Select index typeB file" dialog box.
2) Select the file number and click *"OK."*

[Saving an index type B file (from the edit screen to a file)]

1) Click *"Save (A),"* enter a file name, and click *"Save (S)."* (Do not enter a file extension (*.idp); it is added automatically.)

[Displaying an index type B file (from the driver to the edit screen)]

1) Click *"Upload (U)"* to display the "Select index typeB file" dialog box.
2) Select the file number and click *"OK."*

[Displaying an index type B file (from a file to the edit screen)]

1) Click *"Open (O),"* enter the file name, and click *"Open (O)."* (Do not enter a file extension (*.idp); it is added automatically.)

[Clearing all]

1) Click *"Clear All (C)"* to clear all the values of the index type B file cell array to 0.

(10) Resetting the communication

If the communication with the driver finishes abnormally, click *"Reset Com (R)"* in the "Index TypeB Menu" to return to the normal status.

7.6.4 Point Set

This menu allows you to edit, and register data necessary for table reference operation.

Click *"Point Set (P)"* under "MainMenu" to display the "Point Set" dialog box (see Figure 7.47.)"

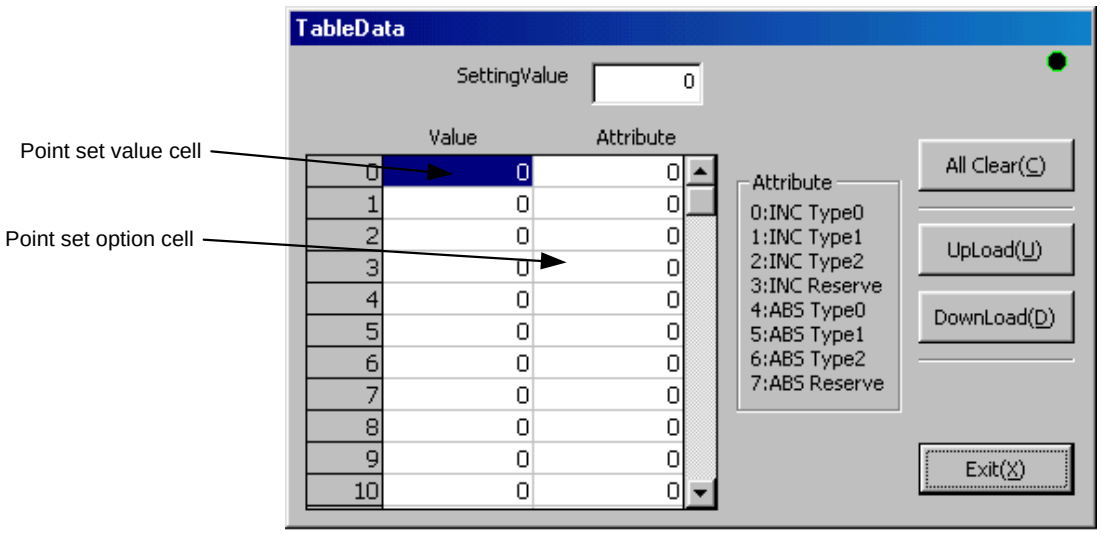


Figure 7.47 "Point Set" dialog box

(1) Editing points set

1) Click the value cell you want to set in order to display the current value in the changed value text field.
2) Enter a value in the value cell, then press the Enter (Return) key or click another value cell or option cell. The value of the value cell you want to change will then be updated.
3) Click the option cell you want to set in order to display the current value in the changed value text field.
4) Enter a value in the option cell, then press the Enter (Return) key or click another option cell or value cell. The value of the option cell you want to change will then be updated.
5) Repeat steps 1) to 4) as necessary.

(2) Displaying points set (from the driver to the edit screen)

1) Click *"Upload (U)"* in the "Point Set" dialog box to display the setting values on the edit screen.

(3) Registering points set (from the edit screen to the driver)

1) Click *"Download (D)"* in the "Point Set" dialog box to register the setting values displayed on the edit screen to the driver.

Note: Point set values (#600 to 699) and point set options (#700 to 799) are treated as parameters; they are called from the driver to the screen by "Upload" and registered from the screen to the driver by "Download."

7.6.5 Parts

This menu allows you to edit (universal cam), register, and save parts necessary for cam profile moves.

Click *"Parts (C)"* under "MainMenu" to display the "Cam Profile Menu" dialog box (see Figure 7.48).

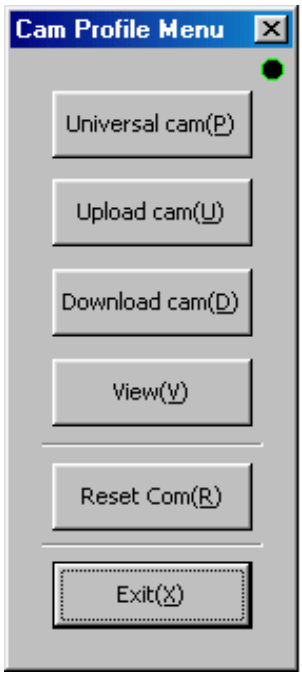


Figure 7.48 "Cam Profile Menu" dialog box

(1) Creating a universal cam

Click *"Universal cam (P)"* in the "Cam Profile Menu" dialog box to display the Universal cam profile dialog box (see Figure 7.49).

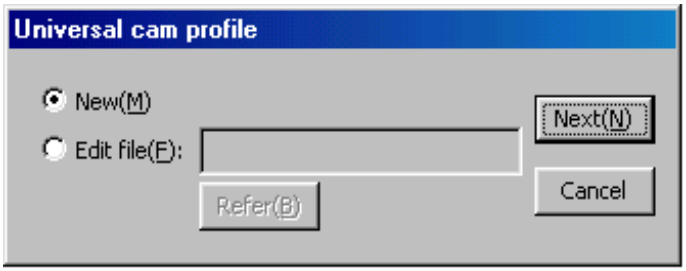


Figure 7.49 "Universal cam profile" dialog box

[Creating a new cam profile]

1) Select “New (M)” in the “Universal cam profile” dialog box and click *“Next (N)”* to display the next “Universal cam profile” dialog box.

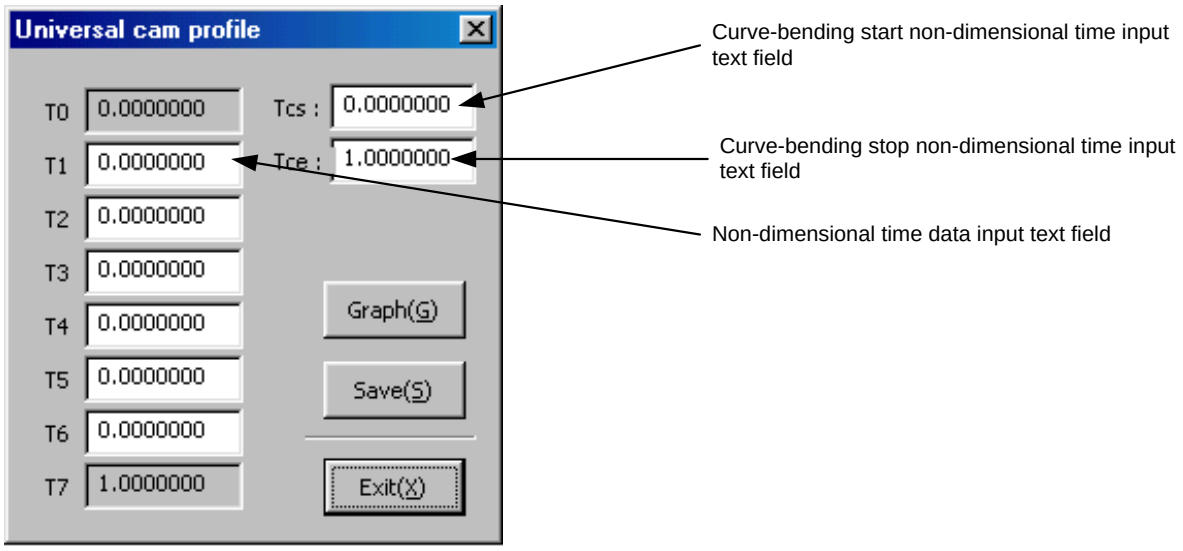


Figure 7.50 “Universal cam profile” dialog box

2) Enter data in the non-dimensional time data input text fields “T1” to “T6.” The data should be in the range from 0 to 1. Press the Enter (Return) key to make the entry valid.

Note: When you enter data in “T1” to “T6,” you cannot enter a value smaller than the data already entered in the items above the one in which you are entering data (“T0” to “T3” if you are entering data in “T4,” for instance). Moreover, if the data you enter is larger than the data in items below the current entry (“T5” and “T6” if you are making an entry in “T4,” for instance), the data below is limited to this value.

3) If you want to bend only the necessary part of a universal cam profile, enter the curve-bending start non-dimensional time (Tcs) and curve-bending stop non-dimensional time (Tce) in the range from 0 to 1. Press the Enter (Return) key to make the entry valid. If you do not intend to bend the curve, enter 0 for Tcs and 1 for Tce.

4) Click *“Graph (G)”* in the “Universal cam profile” dialog box to display the graph of a universal cam profile (see figure 7.51).

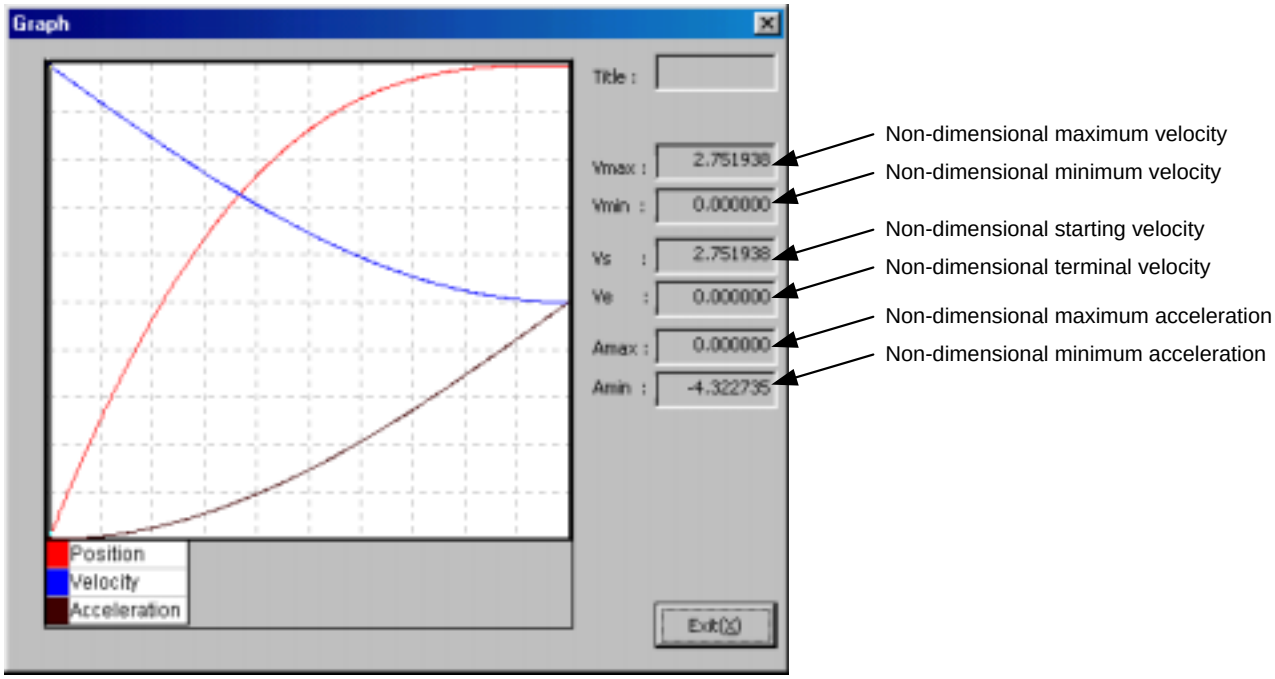


Figure 7.51 "Universal cam profile" graph

- Along the horizontal axis, the far left is non-dimensional time 0 and the far right is non-dimensional time 1.
- Along the vertical axis, the bottom is non-dimensional position 0 and the top is non-dimensional position 1. Also along the vertical axis, the center is non-dimensional velocity and acceleration 0, the top is maximum non-dimensional velocity and acceleration, and the bottom is minimum non-dimensional velocity and acceleration.

[Saving a universal cam profile to a file]

1) Click *"Save (S)"* in the "Universal cam profile" dialog box.
2) Enter the name of the file to which the universal cam profile is saved. Do not enter a file extension (*.unv); it is added automatically.
3) Click *"Save (S)"* to save to the file. If you wish to abort, click *"Cancel."*

Note: When you save a universal cam profile to a file, a cam profile data file with the same file name (extension .cdt) is also created.

[Saving to a file and closing "Universal cam profile"]

1) Click *"Exit (X)"* in the "Universal cam profile" dialog box.
2) A message box is displayed which asks whether or not to save to a file. If you want to save, click *"Yes (Y)."* The procedure for saving is the same as for [Saving a universal cam profile to a file].
3) If you want to exit without saving, click *"No (N)"* and if you want to return to the "Universal cam profile," click *"Cancel."*

[Modifying an existing universal cam profile file]

1) Select *"Edit file"* in the "Universal cam profile" dialog box shown in Figure 7.49.
2) Enter the file name of a universal cam profile (extension .unv) in the file name input text field. If you are not sure about the accurate file path and name, click *"Browse (R)."*
3) Select a file and click *"Open (O)."*
4) Click *"Next (N)"* to return to the "Universal cam profile" dialog box.
5) The "Universal cam profile" dialog box appears.
   (The procedure from here is the same as [Creating a new cam profile].)

(2) Uploading (from the driver to a file)

Parts data can be uploaded by the following two methods:

- "Normal upload": Upload single cam profile data.
- "Upload all": Upload all the cam profile data registered in the driver.

Click *"Upload cam (U)"* in the "Cam Profile Menu" to display the "Select type" dialog box.

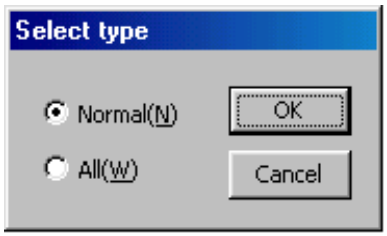


Figure 7.52 "Select type" dialog box

[Normal upload]

1) In the "Select type" dialog box, click *"Normal (N)"* and then *"OK."*
2) Enter a cam profile number to upload and click *"OK."*

Note: Eight standard cam profiles (cam profile numbers 1 to 8) and eight user cam profiles (cam profile numbers 9 to 16) are available for the driver.

3) Enter the name of file to which the cam profile is saved. Do not enter file extension (*.cdt); it is added automatically.
4) Click *"Save (S)"* to start uploading. If you wish to stop uploading, click *"Cancel."*

[Upload all]

1) In the "Select type" dialog box, click *"All (N)"* and then *"OK."*
2) Enter a batch cam profile data file name. Do not enter a file extension (*.whc); it is added automatically.
3) Click *"Save (S)"* to start uploading. If you wish to stop uploading, click *"Cancel."*

Note: When you use the upload all method, a batch cam profile data file is created together with each cam profile data file (file extension .cdt). A cam profile numbers registered in the batch cam profile data file is attached to each cam profile data file name.

(3) Downloading (from a file to the driver)

Parts data can be downloaded by the following two methods:

- "Normal download": Download a single cam profile parts file.
- "Download all": Download all the cam profile files saved in the batch download file.

Click *"Download cam (D)"* in the "Cam Profile Menu" to display the "Select type" dialog box.

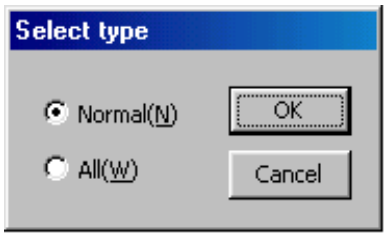


Figure 7.53 "Select type" dialog box

[Normal download]

1) Click "*Normal (N)*" in the "Select type" dialog box and click *"OK."*
2) Enter the download destination cam profile number and click *"OK."*

Note: You can download only user cam profiles (cam profile numbers 9 to 16).

3) Enter the cam profile file to download. Do not enter a file extension (*.cdt); it is added automatically.
4) Click *"Open (O)"* to start downloading. If you wish to stop downloading, click *"Cancel."*

[Download all]

1) In the "Select type" dialog box, click *"All (N)"* and then *"OK."*
2) Enter the batch cam profile data file name. Do not enter a file extension (*.cdt); it is added automatically.
3) Click *"Open (O)"* to start downloading. If you wish to stop downloading, click *"Cancel."*

(4) Viewing the cam profile

Click *"View (V)"* in the "Cam Profile Menu" dialog box to display the "View cam profile" dialog box.

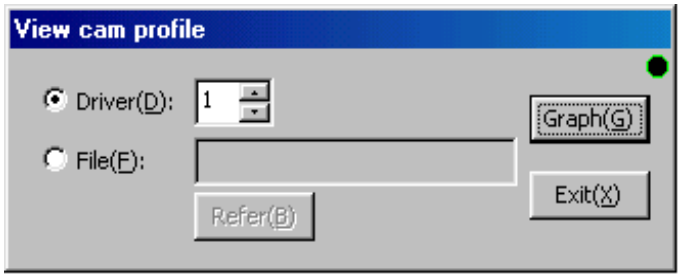


Figure 7.54 "View cam profile" dialog box

[Cam profiles registered in the driver]

1) Select "Driver" in the "View cam profile" dialog box.
2) Enter the cam profile number to display as a graph.
3) Click *"Graph (G)"* to display the graph of the cam profile.

[Cam profiles saved in a file]

1) Select "File" in the "View cam profile" dialog box.
2) Enter the cam profile data file name (extension .cdl) in the file name input text field. If you are not sure about the accurate file path and name, click *"Browse (R)."* Select the cam profile data file you want to display as a graph.
3) Click *"Open (O)"* to return to the "View cam profile" dialog box.
4) Click *"Graph (G)"* to display the graph of the cam profile.

(5) Resetting the communication

1) If the communication with the driver finishes abnormally, click *"Reset Com (R)"* in the "Cam Profile Menu" to return to the normal status.

7.6.6 I/O Set

In this menu, you can set the logical setting of DI/DO points (connector number varies depending on the interface used).

Click *"I/O set (I)"* under "MainMenu" to display the "I/O configuration" dialog box (see Figure 7.55).

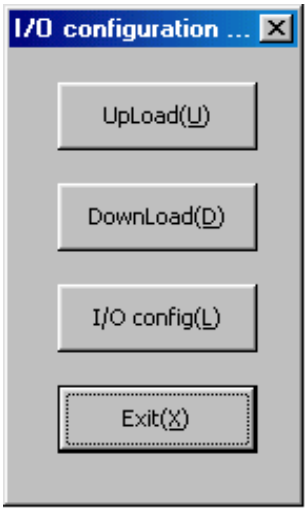


Figure 7.55 "I/O configuration" dialog box

(1) Logical setting

1) Click *"I/O config (L)"* in the "I/O configuration" dialog box.
2) The "Discrete configuration" dialog box is displayed and the current setting status is read. (The maximum number of setting statuses that can be displayed at once is 32 points for both DI and DO.)

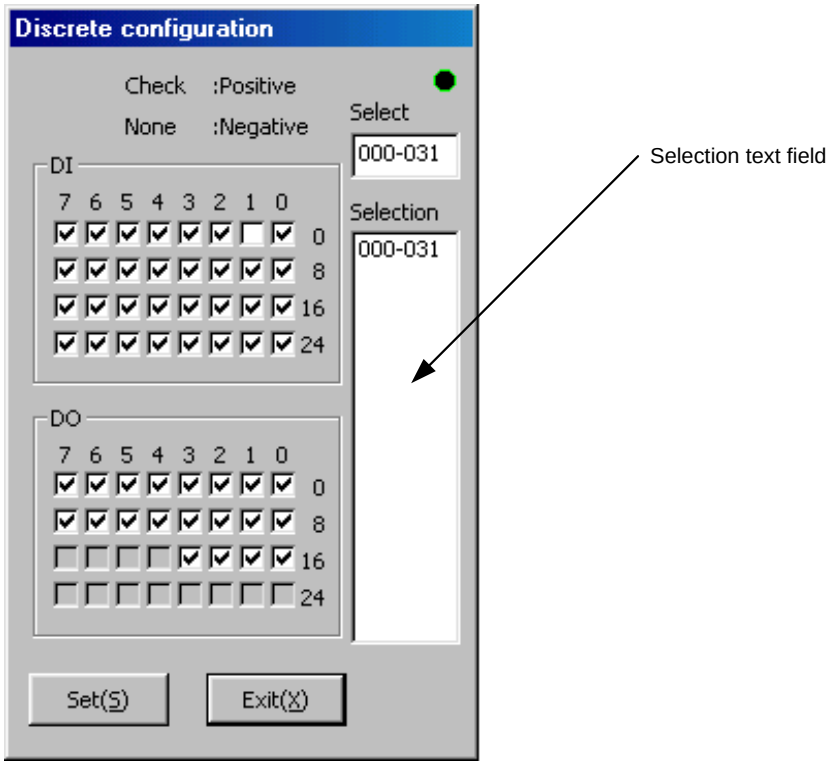


Figure 7.56 "Discrete configuration" dialog box

3) Click a number you want to display from the selection text field in order to display the logical setting of that number.
4) Click the check boxes of the I/O points to change the logical setting.

Note: An I/O point with a check mark is set to positive logic (A contact) and an I/O without a check mark is set to negative logic (B contact).

5) Click *"Set (S)."*
6) When the setting is completed, follow the message and reset the driver.

(2) Uploading (from the driver to a file)

1) Click *"Upload (U)"* in the "I/O configuration" dialog box.
2) Enter the name of the file to which I/O configuration is to be saved. Do not enter a file extension (*.ioc); it is added automatically.
3) Click *"Save (S)"* to start uploading. If you wish to stop the uploading, click *"Cancel."*

(3) Downloading (from a file to the driver)

1) Click *"Download (D)"* in the "I/O configuration" dialog box.
2) Enter the name of the file to be downloaded to the driver. Do not enter a file extension (*.ioc); it is added automatically.
3) Click *"Open (O)"* to start downloading. If you wish to stop the downloading, click *"Cancel."*
4) When the downloading is finished, follow the message and reset the driver.

Chapter 8
Operation Display Pendant

8.1 Overview

8.2 Features and Part Names

8.3 Switching Displays

8.4 Terminal Mode Display

8.5 Parameter Monitor Display

8.6 Parameter Settings Display

8.7 I/O Monitor Display

8.8 Special Command Display

8.9 Program Menu Display

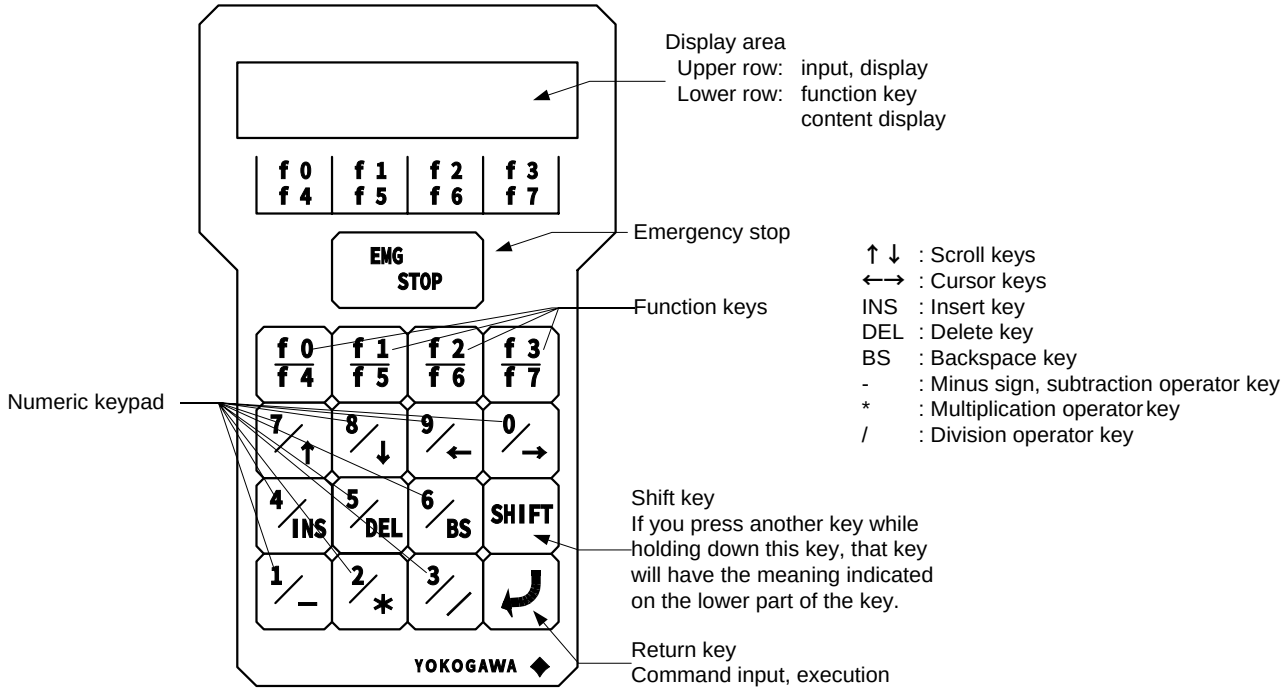
8.9.1 Program Edit Displays
8.9.2 Copying a Program
8.9.3 Deleting a Program

8.1 Overview

The operation display pendant (abbreviated as TBX, optional device) should be connected to the CN1 RS232C communication connector. It is set to the same status as the single channel communication in the RS232C interface, regardless of the setting status of the rotary switch RS-ID on the front panel.
The following functions are provided for the operation display pendant.

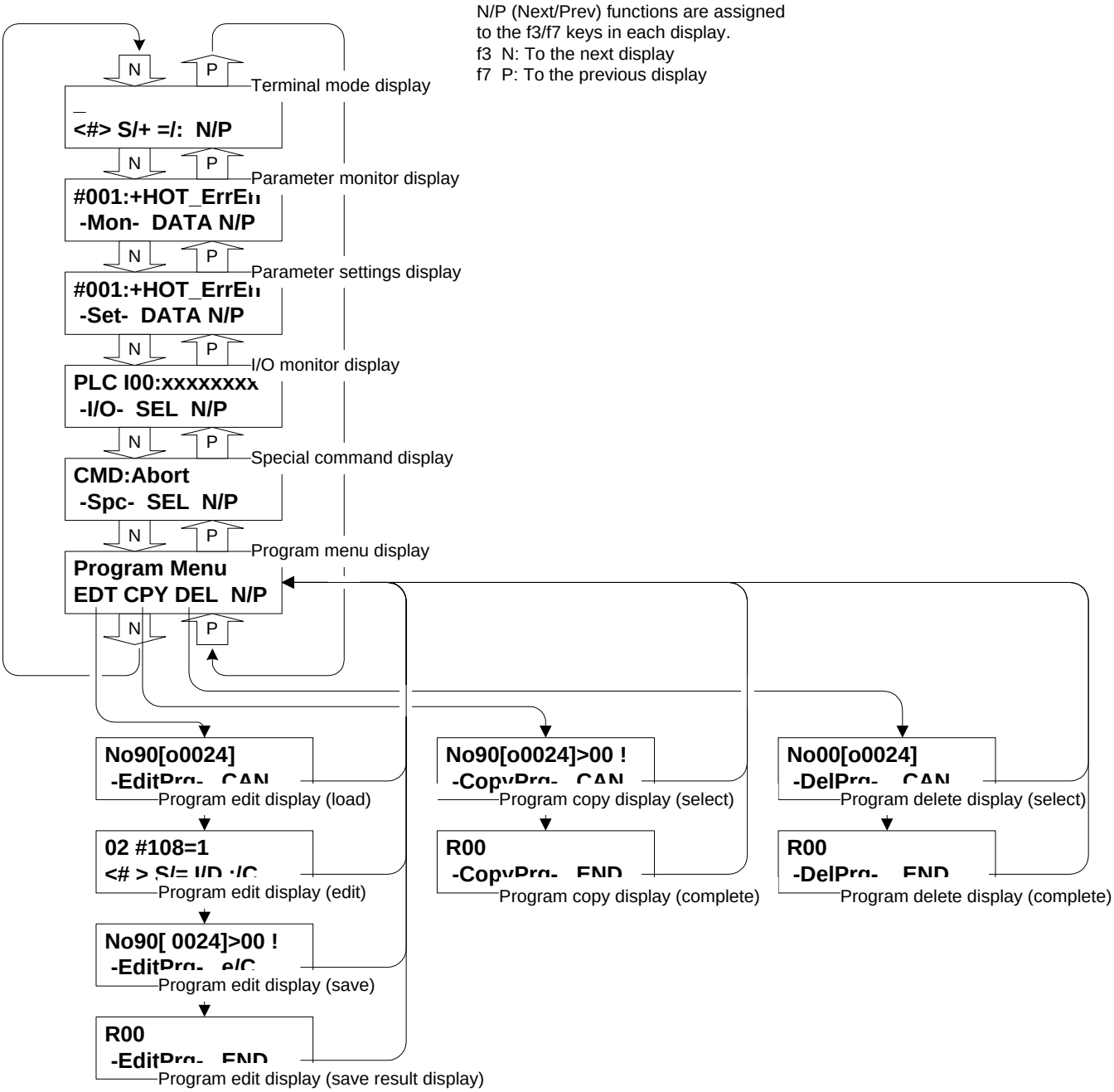
- ***Terminal mode display:***
  In this display, you can send @ commands and parameter commands in the same way as with the RS232C interface and display response character strings.
- ***Parameter monitor display:***
  In this display the updated contents of parameters/monitors referred to by #*** can be displayed repeatedly.
- ***Parameter settings display:***
  In this display, you can make changes to parameters referred to by #***, if they can be written to.
- ***I/O monitor display:***
  In this display the updated I/O status of the PLC interface and the signal status of the TB2 sensor/break can be displayed repeatedly.
- ***Special command display:***
  In this display, you can issue common commands in a simple manner.
- ***Program menu display:***
  In this display, you can edit, copy, and delete programs.

8.2 Features and Part Names



8.3 Switching Displays

Each display shifts in the order shown in the figure below. When the power is turned on and the operation display pendant is connected, the initial screen shows the terminal mode display.
Each display of the program menu display can be accessed by selecting edit (EDT), copy (CPY), or delete (DEL).



8.4 Terminal Mode Display

The terminal mode display allows you to send a character string entered from the keypad to the driver and display the response character string in the display.
In the example below, "#50" is input in display 2) and the response character string "R1D position bandwidth:12" is shown in display 3).
In the response character string display, the header part of a response character string (e.g., R00, ALM**, *) is not displayed. Even though the cursor is not displayed, hidden parts can be horizontally scrolled through and displayed by pressing the ← and → keys.
The cursor is displayed by pressing the input key (a key that allows character input when pressed), or the ↑ and ↓ keys. Once it is displayed, you can enter character strings.

1) Initial display

```
_
<#> S/+ =/:  N/P
```

2) During character string input

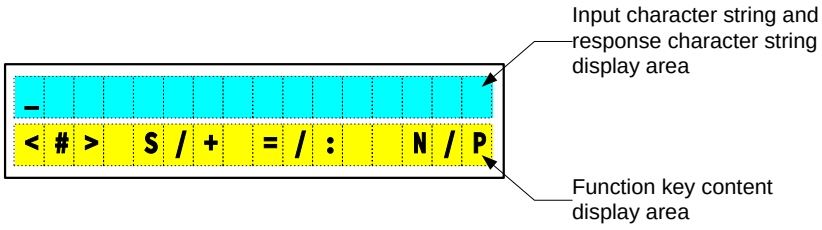
```
#50_
<#> S/+ =/:  N/P
```

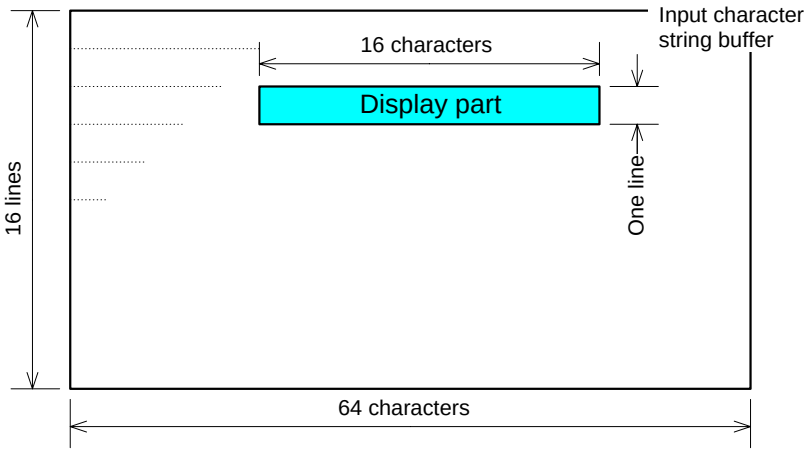
3) Displaying a response character string

```
Position bandwidth: 12
<#> S/+ =/:  N/P
```

Characters in < >: #, @, %, G, X, F, f, A, a, M, P, +

f0 < >:	Character selection (positive direction) The character in < > changes at every key press.
f1 S:	Inputs the selected character. The character in < > is input by pressing this key.
f2 =:	Inputs =.
f3 N:	To the next display
f4 < >:	Character selection (opposite direction) The character in < > changes at every key press.
f5 +:	Inputs +.
f6:	Inputs :.
f7 P:	To the previous display
0 to 9: -, *, /	Each character is input.
INS key:	Shifts the character string one character after the cursor position and insert a space at the cursor position.
DEL key:	Deletes the character immediately before the cursor. The cursor does not move the position.
BS key:	Deletes the character immediately before the cursor and move the cursor to the position one character before.
Return key:	Sends the entered character string to the controller.
← and → keys:	Moves the cursor on the input character string to the left or right.
↑ and ↓ keys:	Moves up and down in the input character string buffer.





8.5 Parameter Monitor Display

The current values of parameters/monitors with the numbers input from the keypad are displayed periodically. It is not necessary to press the Return in order to set a number.

In the previous example of display 2), parameter number “50” is entered to display the prompt character string of that parameter/monitor. When the f2 DATA key is pressed here, display 3) appears and displays the current value of the parameter/monitor.

If a nonexistent number is entered, both the data and comment displays show the comment in display 4).

1) Initial display

```
#001:+HOT_ErrEn
 -Mon-  DATA N/P
```

Cursor

2) Displaying comment

```
#050: Position bandwidth
 -Mon-  DATA N/P
```

3) Displaying data

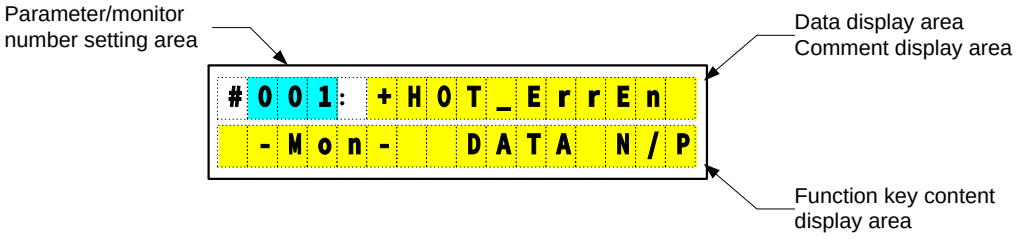
```
#050:        12
 -Mon-  CMNT N/P
```

4) Displaying comment (when a number does not exist)

```
#000:----------
 -Mon-  DATA N/P
```

(CMNT)

f2 DATA:	Switches display; to the data display
CMNT:	To the comment (prompt) display
f3 N:	To the next display
f7 P:	To the previous display
0 to 9:	Each character is entered.
← and → keys:	Moves the cursor on the parameter/monitor number to the left or right.
↑ and ↓ keys:	Increases/decreases the numeric value of the digit indicated with the cursor.



8.6 Parameter Settings Display

This display is for changing the values of parameters.
When you input a parameter number from the keypad, the display shows a prompt for the parameter when comments are being displayed, and the current value of the parameter when data is being displayed.
When data is being displayed, it is possible to move the cursor to the data setting area using the ← and → keys. In the data setting area (sign part), it is possible to reverse the sigh by pressing the – key. In the data setting area (absolute value part), the setting value can be set using the numeric keys as well as the ↑ and ↓ keys. When you press the Return key, the entered data is set as the parameter value of the parameter number (the Return key is accepted only when data is being displayed).
When the data is set, the setting result is displayed. If it is set normally, the display shows “OK!” If it failed to set, the display shows “NG! [ERROR ALARM CODE]”. When you press the f2 DATA key, the display switches to the data display; when you press the numeric keys or the ↑ and ↓ keys, it switches to the comment display.
In the previous example of display 2), the parameter number “50” is entered to display the prompt character string of that parameter/monitor. If the f2 DATA key is pressed here, display 3) appears and displays the current value of the parameter/monitor. Display 4) shows the status in which the cursor is moved and data is input from the keypad. When the Return key is pressed here, the setting result is displayed as in display 5).
If a nonexistent number is entered, both the data and comment displays show the comment in display 6). In addition, if you input a parameter that exists but cannot be written to, the data display (but not the comment display) changes similarly to display 6).

1) Initial display

```
#001:+HOT_ErrEn
 -Set-  DATA N/P
```

Cursor

2) Displaying comment

```
#050: Position bandwidth
 -Set-  DATA N/P
```

3) Displaying data

```
#050: 0000000012
 -Set-  CMNT N/P
```

4) Setting data

```
#050:-0000000018
 -Set-  CMNT N/P
```

5) Displaying setting result

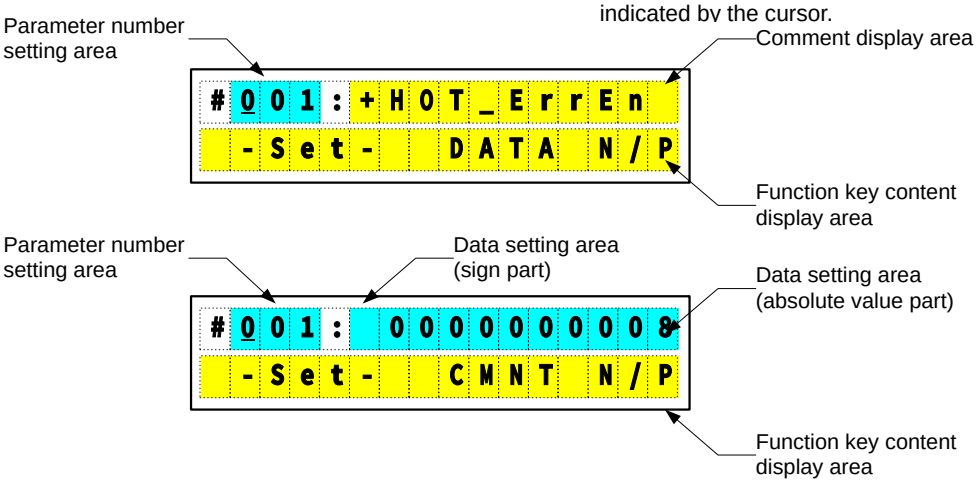
```
#050:NG![ALM62.0
 -Set-  DATA N/P
```

6) Displaying comment (when a number does not exist)

```
#000:----------
 -Set-  DATA N/P
```

(CMNT)

f2 DATA:	Switches display; to the data display
CMNT:	To the comment (prompt) display
f3 N:	To the next display
f7 P:	To the previous display
0 to 9:	Each character is entered.
-:	The sign of the data value is reversed if pressed while the cursor is at the data setting area (sign part).
Return key:	Sets the input data to the driver.
← and → key:	Moves the cursor in the parameter number setting area and data setting area (absolute value part) to the left or right.
↑ and ↓ key:	Increases/decreases the numeric value of the digit indicated by the cursor.



8.7 I/O Monitor Display

The I/O signal status of the blocks with numbers input from the keypad for the selected I/O type is displayed periodically. It is not necessary to press the Return key in order to set a number.
There are three types of I/O that can be selected: “PLC I,” “PLC 0,” and “Drv I.” You can switch among them by pressing the f2/f6 SEL key.
In the previous example of display 2), block number “1” is entered to display the signal status of PLC DI block 1. If the f2/f6 SEL key is pressed here, the display changes similarly to display 3) and switches to the PLC DO display.

1) Initial display

```
PLC I00:xxxxxxxx
 -I/O-  SEL  N/P
```

Cursor

2) Displaying PLC DI

```
PLC I01:xxxxxxxx
 -I/O-  SEL  N/P
```

3) Displaying PLC DO

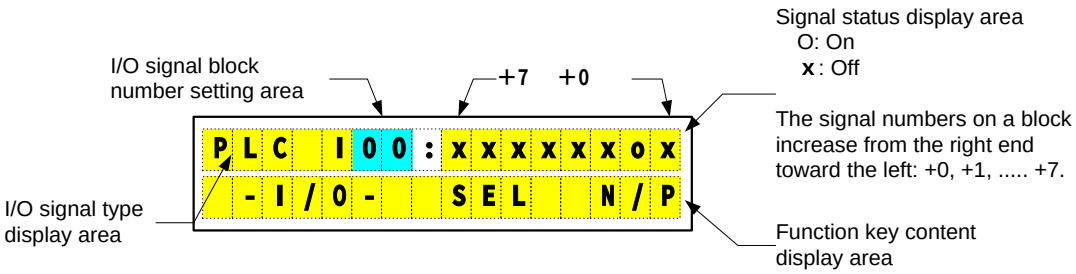
```
PLC OO0:xxxxxxxx
 -I/O-  SEL  N/P
```

f2 SEL:	Switches I/O signal type (positive direction).
f3 N:	To the next display
f6 SEL:	Switches I/O signal type (opposite direction).
f7 P:	To the previous display
0 to 9:	Each character is entered.
← and → key:	Moves the cursor in the I/O signal block number setting area to the left or right.
↑ and ↓ key:	Increases/decreases the numeric value of the digit indicated by the cursor.

PLC I
Block number 0: DI07 to DI00
1: DI15 to DI08

PLC 0
Block number 0: DO07 to DO00
1: DO15 to DO08

Drv I (Driver input signal status)
Block number 0: DI07 to DI00
1: DI15 to DI08



Driver input signal status

DI00: (Reserved)	DI08: (Reserved)
DI01: (Reserved)	DI09: (Reserved)
DI02: (Reserved)	DI10: (Reserved)
DI03: (Reserved)	DI11: (Reserved)
DI04: Homing detection	DI12: (Reserved)
DI05: (+) direction over-travel detection	DI13: (Reserved)
DI06: (-) direction over-travel detection	DI14: (Reserved)
DI07: (Reserved)	DI15: (Reserved)

8.8 Special Command Display

In this display, you can transmit selected commands to the controller.
The selection of the commands you can choose from depends on the main operation mode. They can be switched by pressing the f2/f6 SEL key.
When you select a command from the list and press the Return key, it is transmitted to the controller and the result of the command is shown in the display. By pressing the f2/f6 SEL key, it is switched to the command content display.
In the previous example of display 2), command “toMode15” is selected. When you press the Return key, this command is transmitted to the controller and the result of the command is displayed in display 3).

Command	Display	Main operation mode	
		RS232C main operation mode	PLC main operation mode
Abort	CMD:Abort	○	○
Error reset	CMD:RstErr	○	○
Servo ON	CMD:SrvOn	○	X
Servo OFF	CMD:SrvOff	○	X
Mechanical setting operation start	CMD:toMode15	○	○
Software driver reset	CMD:RstDrive	○	○
Motor type request	CMD:MotorType	○	○
ROM version request	CMD:ROM Version	○	○
Error status request/refresh	CMD:RefErrSts	○	○

1) Initial display

```
CMD:Abort
 -Spc- SEL N/P
```

2) Displaying command

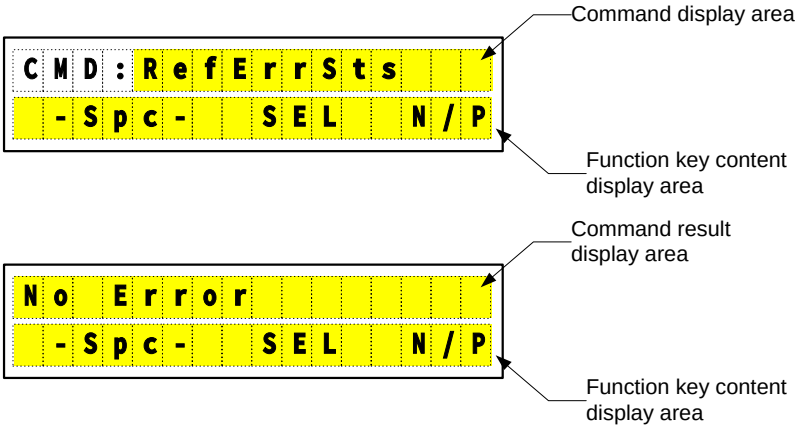
```
CMD:toMode15
 -Spc- SEL N/P
```

3) Displaying command result

```
Mode15 Exec!
 -Spc- SEL N/P
```

f2 SEL: Switches command (positive).
f3 N: To the next display
f6 SEL: Switches command (opposite).
f7 P: To the previous display

Return key: Transmits the selected command to the driver.



8.9 Program Menu Display

In the program menu display, you can edit, copy, and delete programs.
By pressing the f1, f2, or f3 key in the initial display 1), you can jump to each operation.

Note: A special display for showing the registration status of a program file is not available. If you wish to verify the registration status, use the program load display in the program edit displays. Once you have verified, press the f3 CAN key to return to the program menu display.

1) Initial display

```
Program Menu
EDT CPY DEL  N/P
```

f0 EDT:	To the program edit displays
f1 CPY:	To the program copy displays
f2 DEL:	To the program delete display
f3 N:	To the next display
f7 P:	To the previous display

8.9.1 Program Edit Displays

In the program edit displays, you operate in the order of program load, program edit, and then program save.

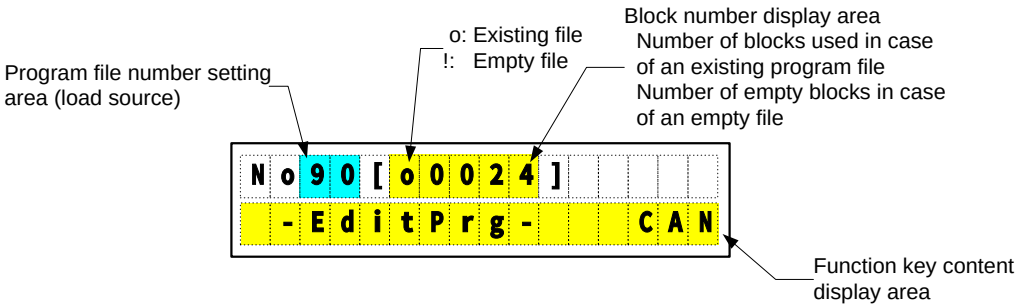
(1) Loading a program

You can select a program file number and load it into the editor.
When choosing the program file number of the source file, the stored program file number with the smallest number is displayed initially. If all the user area files are empty, a built-in program file is displayed initially.

1) Program file number selection

```
No90[o0024]
 -EditPrg-   CAN
```

f3 CAN:	Cancels To the program menu display
0 to 9:	Each character is entered.
Return key:	Loads the selected program file.
← and → key:	Moves the cursor in the program file number setting area to the left or right.
↑ and ↓ key:	Increases/decreases the numeric value of the digit indicated by the cursor.



(2) Editing a program

You can edit the content of a program loaded to the editor.
A block is treated as one unit in the editing. If a comment statement is attached to an NC executable statement, parameter statement, or control parameter in one line, that line occupies two blocks. Other lines occupy one block. The program edit function in the operation display pendant can handle up to 100 blocks. If you wish to edit more than that, use the PC utility.
Comment statements cannot be edited by the program edit function in the operation display pendant. However, it is possible to delete the block.
Moreover, a block attached to an NC executable statement, parameter statement, or control parameter in one line is treated specially. The NC executable statement, parameter statement, and control parameters are treated as the main blocks and their block numbers are displayed. Main blocks can be edited. Comment statements are treated as sub-blocks and their block numbers are not displayed. Since these blocks are comment statements, they cannot be edited. It is possible to delete a sub-block, but it is not possible to insert a block in front of a sub-block.

1) Initial display in case of an existing file (in case of a block that cannot be edited)

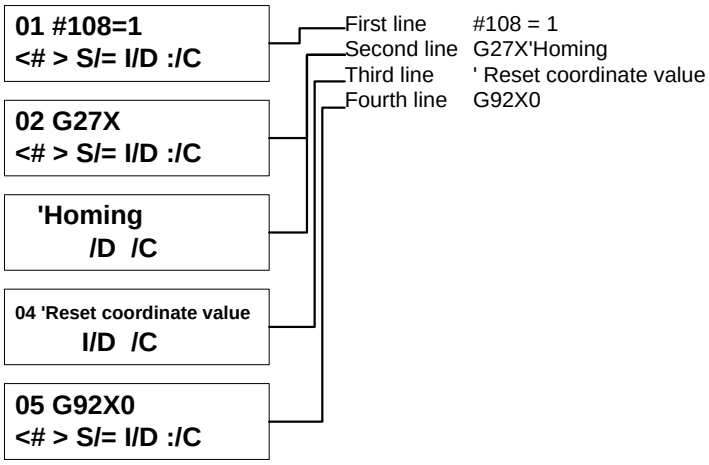
```
01 'Index operation
      I/D  /C
```

2) Initial display in case of an existing file (in case of a block that can be edited)

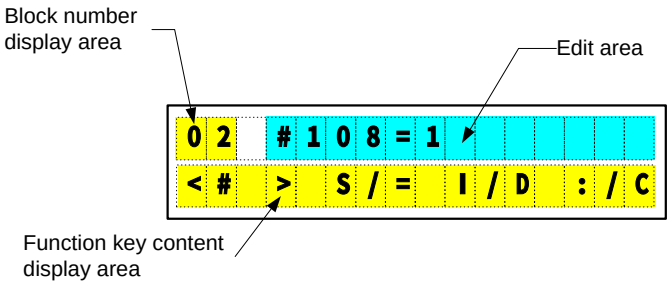
```
01 #108=1
<# > S/= I/D :/C
```

3) Initial display in case of an empty file

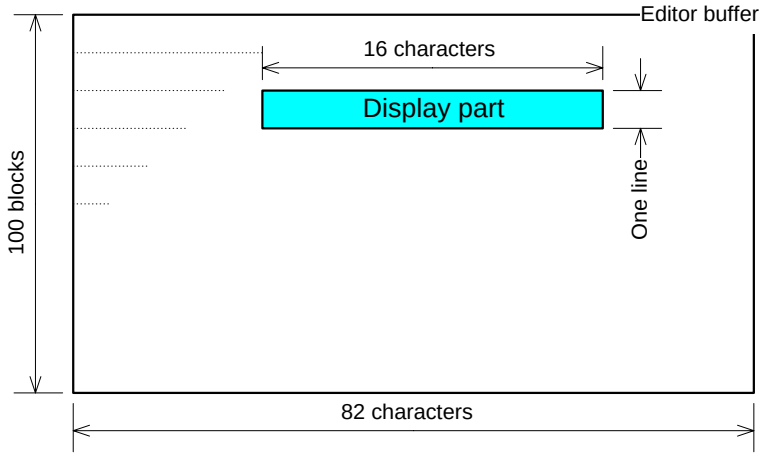
```
01 _
<# > S/= I/D :/C
```



Characters in < >: #, G, X, F, f, A, a, M, P,
WH (WHILE), FR (FOR),
IF (IF), ED (END),
EI (ELSEIF), EL (ELSE),
>, <, !, +, %



f0 < >:	Character selection (positive direction) The character in < > changes at every key press.
f1 S:	Inputs the selected character. The character in < > is input by pressing this key.
f2 I:	Inserts a block. Inserts an empty block before the current line.
f3 :	Inputs
f4 < >:	Selects character (opposite direction). The character in < > changes at each key press.
f5 =	Inputs =.
f6 D:	Deletes a block.
f7 C:	Cancels To the program menu display
0 to 9: -, *, /	Each character is input.
INS key:	Shifts the character string one character after the cursor position and insert a space at the cursor position.
DEL key:	Deletes the character immediately before the cursor. The cursor does not move.
BS key:	Deletes the character immediately before the cursor and move the cursor to the position one character before.
Return key:	To the program save function with the edited content
← and → key:	Moves the cursor on a block to the left or right.
↑ and ↓ key:	Moves up or down within the editor buffer. If the ↓ key is pressed at the last non-empty line, an empty line is inserted after that line



(3) Saving a program

You can select a program file number to save the program edited in the editor.
If you are editing a file in the user area, the number during loading is initially displayed as the program file number at the save destination.
If you are editing a built-in program, the empty file with the smallest number is displayed as the initial program file number at the save destination.

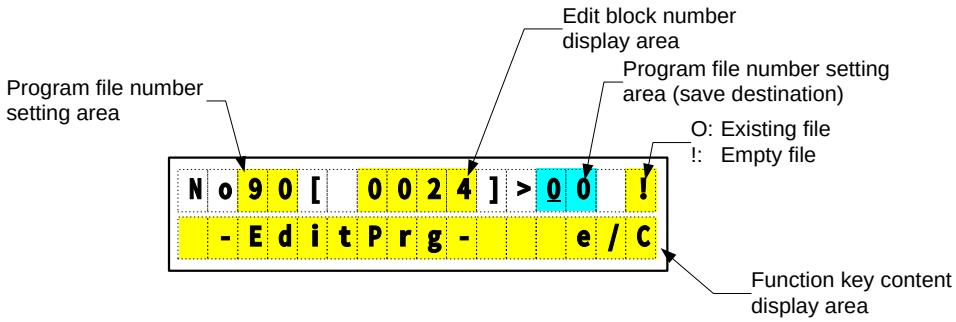
1) When editing a file in the user area

```
No01[ 0024]>01 o
 -EditPrg-   e/C
```

2) When editing a built-in file

```
No90[ 0024]>00 !
 -EditPrg-   e/C
```

f3 e:	Edits To the program edit display
f1 c:	Cancels To the program menu display
0 to 9:	Each character is input.
Return key:	Loads the selected program file.
← and → key:	Moves the cursor in the program file number setting area to the left or right.
↑ and ↓ key:	Increases/decreases the numeric value of the digit indicated by the cursor.



(4) Displaying program save result

This display shows the result of the program save operation.
If the program is properly saved, the display shows similar to display 1). After confirming that the file is saved, press the f3 END key to return to the program menu display.
If the program is not properly saved, the display shows similar to display 2). Press the f3 e key to return to the program editor where blocks that were not properly processed are displayed. Make corrections and save the program again. Press the f7 C key to return to the program menu display.

1) When a file is properly saved

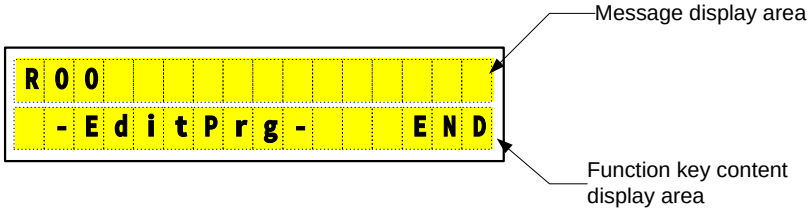
```
R00
 -EditPrg-   END
```

f3 END:	Ends To the program menu display

2) When a file is not properly saved

```
Cannot interpret command
 -EditPrg-   e/C
```

f3 e:	Edits To the program edit display
f7 c:	Cancels To the program menu display
← and → key:	Scrolls the message display area horizontally.



8.9.2 Copying a Program

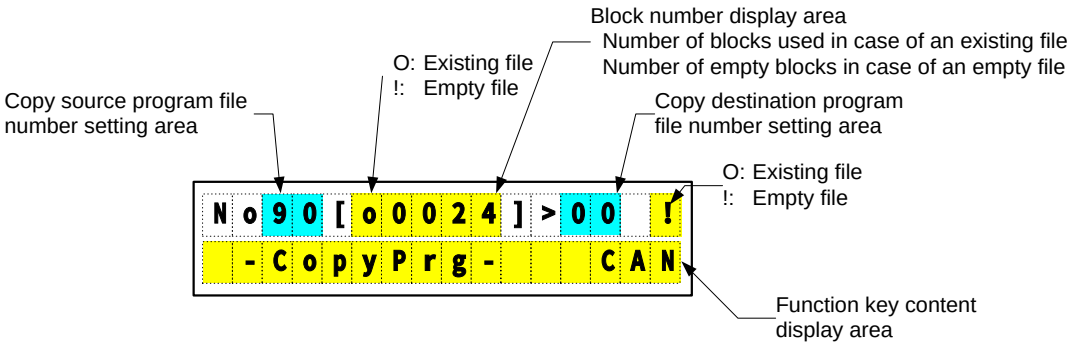
(1) Selecting a program

You can copy a program by selecting the program file numbers of the copy source and destination.
When choosing the program file number of the copy source, the stored program file number with the smallest number is displayed initially. If all the user area files are empty, a built-in program file is displayed initially.
When choosing the program file number of the copy destination, the empty file with the smallest number is displayed initially.

1) Program file number selection

```
No90[o0024]>00 !
 -CopyPrg-   CAN
```

f3 CAN:	Cancels To the program menu display
0 to 9:	Each character is input.
Return key:	Copies the selected copy source program to the copy destination program file.
← and → key:	Moves the cursor in the copy source and destination program file number setting areas to the left or right.
↑ and ↓ key:	Increases/decreases the numeric value of the digit indicated by the cursor.



(2) Displaying program copy result

This display shows the result of the program copy.
If the program is properly copied, the display shows similar to display 1). After confirming that the file is saved, press the f3 END key to return to the program menu display.
If the program is not properly copied, the display shows similar to display 2). After confirming this, press the f3 END key to return to the program menu display.

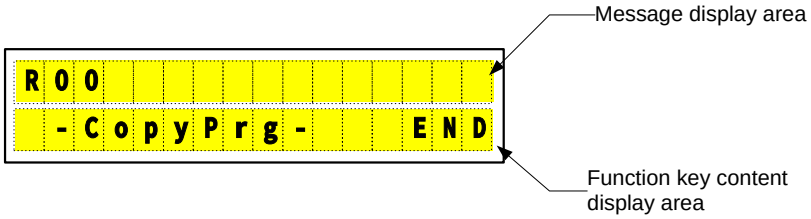
1) When the file is properly copied

```
R00
 -CopyPrg-   END
```

f3 END:	Ends To the program menu display
← and → key:	Scrolls the message display area horizontally.

2) When the file is not properly copied

```
Cannot interpret command
 -CopyPrg-   END
```



8.9.3 Deleting a Program

(1) Selecting a program

You can select a program file number to delete that program.
When choosing the program file number to be deleted, the stored program file with the smallest number is displayed initially similar to display 1). If all the user area files are empty, the display shows similar to display 2).

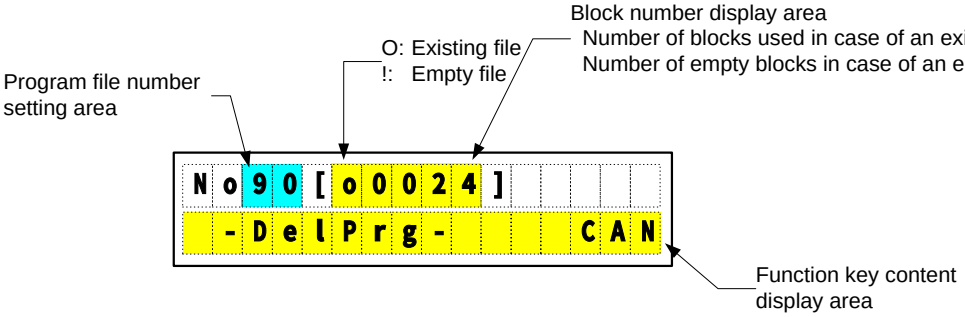
1) Program file number selection

```
No00[o0024]
 -DelPrg-    CAN
```

2) When the user area contains only empty files

```
None Prg
 -DelPrg-    CAN
```

f3 CAN:	Cancels To the program menu display
0 to 9:	Each character is input.
Return key:	Deletes the selected program file.
← and → key:	Moves the cursor in the program file number setting area to the left or right.
↑ and ↓ key:	Increases/decreases the numeric value of the digit indicated by the cursor.



(2) Displaying program delete result

This display shows the result of the program delete.
If the program is properly deleted, the display shows similar to display 1). After confirming that the file is deleted, press the f3 END key to return to the program menu display.
If the program is not properly deleted, the display shows similar to display 2). After confirming this, press the f3 END key to return to the program menu display.

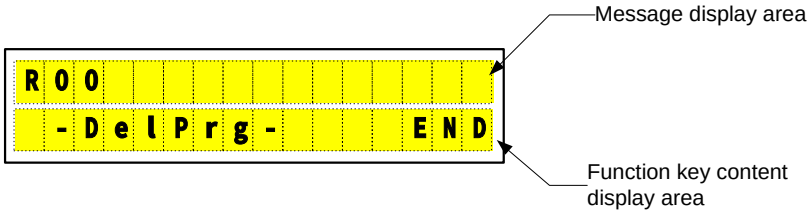
1) When the program is properly deleted

```
R00
 -DelPrg-    END
```

f3 END:	Ends To the program menu display
← and → key:	Scrolls the message display area horizontally.

2) When the program is not properly deleted

```
Cannot interpret command
 -DelPrg-    END
```



Chapter 9
Maintenance and Inspection

9.1 Motor Part

9.2 Driver Part

9.3 Replacement of Battery for Memory Back-Up

9.4 Backup and Restore Operations of Driver Memory Contents

9.4.1 Backup Operation
9.4.2 Restore Operation

9.5 Motor Problems and Corrective Actions

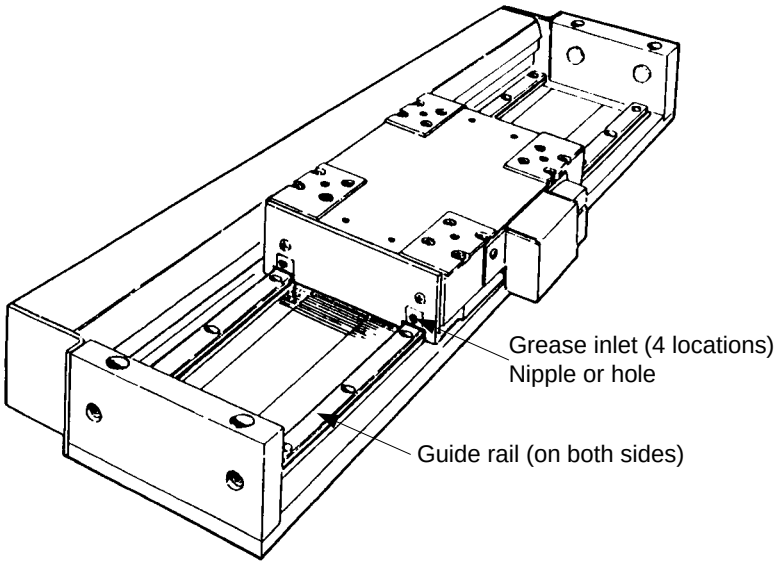
9.1 Motor Part

Simple daily checks need be performed on the motor part. Check the motor for excessive noise or abnormal vibration.
Do not dismount the motor.
If the motor operates abnormally 20,000 hours of operation or five years since installation, depending on the environment and conditions used, replace the motor, and also the driver if necessary.

⚠ **Important**

Add grease to the motor part from the grease inlets (see the figure below) every 100km or three months.
If you cannot use the inlet to add grease, apply grease on the guide rail
The grease in the motor at shipping varies depending on the model as shown below. Please add grease of the same brand.

Model	Grease used
LM105/LM110/ LM505/LM510	Multemp PS No.2 (Kyodo Yushi)
LM205/LM210/ LM305/LM310	AFB Grease (THK)/for THK guide Alvania EP2 (Showa Shell Sekiyu)/for IK0 guide
LM130/LM530/ LM230/LM330	AFB Grease (THK)/for THK guide Alvania EP2 (Showa Shell Sekiyu)/for IK0 guide



9.2 Driver Part

There is no need for a daily maintenance and inspection of the driver part. However, it is prudent to clean the driver unit periodically to protect it from dust or particles since they may damage insulation.

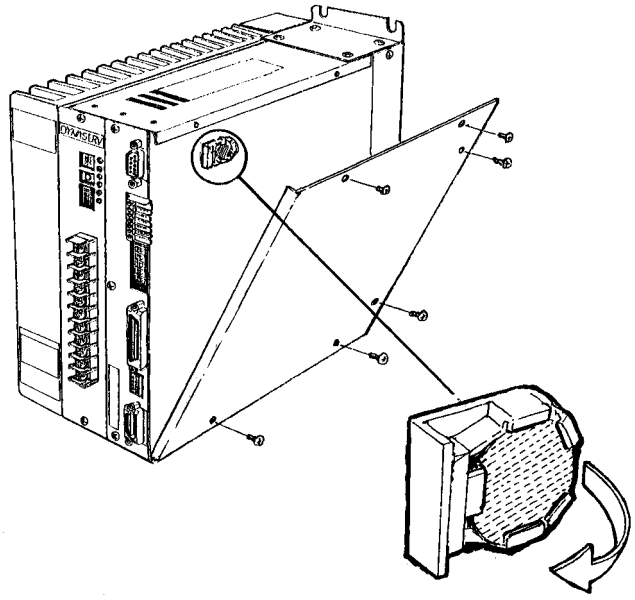
9.3 Replacement of Battery for Memory Back-Up

A lithium battery is installed in the driver for parameter data memory.
The average service life of a lithium battery is 20,000 hours.
When the battery runs out, an alarm signal will be displayed. Replace the battery as soon as possible.
Battery replacement should be done in the following steps:

1) Before changing the battery, backup the parameters, programs, cam data, etc. stored in the memory to flash ROM.

2) Turn off the power.

3) Take off the driver side panel by removing the six screws. (See figure below.)



4) Change the battery quickly (within 10 seconds).

5) Check the connection and turn the power on.
If no error is displayed, battery replacement is completed successfully.

Note 1: If a battery error is displayed, you must reset the system after changing the battery. In such a case, all parameters must be set again.

Note 2: Be sure to use <CR2032> 3 V battery or an equivalent. Be careful about battery orientation (front and back).

9.4 Backup and Restore Operations of Driver Memory Contents

Be sure to back up the driver memory contents in case of the occurrence of problems. If a problem occurs in the driver memory contents, it may become necessary to initialize (all reset) the driver to the same settings at the time of shipment from the factory. If the driver memory contents have been backed up in such a case, the driver memory contents can easily be restored to the factory settings.

9.4.1 Backup Operation

It is recommended to back up the driver memory contents to a file using the PC utility as well as to back up the driver memory contents to the driver's built-in flash ROM.

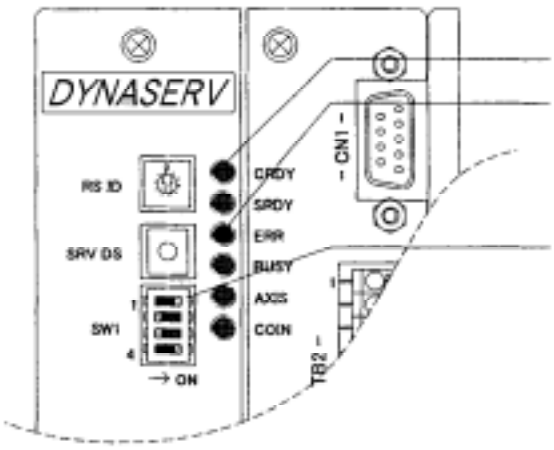
[Backup to a file using the PC utility]

Using the PC utility, back up the driver memory contents to an electronic file in the personal computer. Perform the following backup operation using the PC utility:

- Parameters upload
- Programs upload (batch)
- Index equal division compensation upload (batch)
- Index unequal division upload (batch)
- Parts upload (batch)
- I/O settings upload

[Backup to the driver's built-in flash ROM]

Back up the driver memory contents to the on-board flash ROM by copying the driver memory contents to the driver's built-in flash ROM in a batch operation.
Set the slide switch on the front panel as shown in the figure below, and turn ON the power. When the CRDY lamp flashes after several seconds, the backup operation is completed.



CRDY: LED that flashes when backup is properly completed.

ERR: LED that flashes when backup is not properly completed.

SW1 slide switch
Only bit 1 and bit 4 are ON; bit 2 and bit 3 are OFF.

9.4.2 Restore Operation

The restore operation uses either of the backup data that was copied to a file via the PC utility or that was copied to the driver's built-in flash ROM. Perform either of the following restore operations:

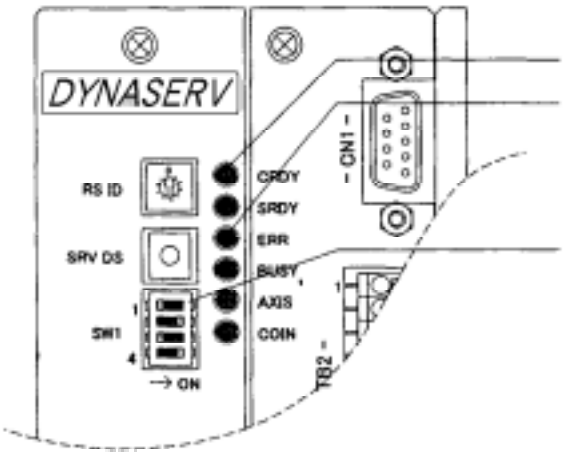
[Restore using the PC utility]

Using the PC utility, restore the backed up driver memory contents to an electronic file in the personal computer to the driver. Perform the following restore operation using the PC utility:

- Parameters download
- Programs download (batch)
- Index equal division compensation download (batch)
- Index unequal division download (batch)
- Parts download (batch)
- I/O settings download

[Restore from the driver's built-in flash ROM]

Restore the driver memory contents from the on-board flash ROM by copying the driver memory contents to the driver's built-in flash ROM in a batch operation.
Set the slide switch on the front panel as shown in the figure below, and turn ON the power. When the CRDY lamp flashes after several seconds, the restore operation is completed.



CRDY: LED that flashes when backup is properly completed.

ERR: LED that flashes when backup is not properly completed.

SW1 slide switch
Only bit 2 and bit 4 are ON; bit 1 and bit 3 are OFF.

9.5 Motor Problems and Corrective Actions

When an abnormality occurs during motor operation, first connect the PC or the operation display pendant to check the content of the problem.
If you turn off the power to the driver, the content of the problem will be erased.
If the driver is still not be able to return to a normal operating condition despite corrective actions taken, stop operating the drive and contact us.

Problem	Possible cause	Item(s) to be inspected	Corrective action
The motor does not servo-lock.	◆ No AC power is being supplied.	Check the wiring.	Turn on the power.
	◆ The servo ON terminal is set to H.	Inspect.	Set to L.
	◆ The Servo ON disable (SRVDS) button is being pressed.	Inspect.	Release the button.
	◆ Position control bandwidth, velocity control bandwidth, and/or position integral limiting value are too small.	Inspect.	Adjust to the proper value(s). Perform auto-tuning.
The motor does not start.	◆ Motor is overloaded.	Check to see if the motor operates without any load.	Reduce the load or replace a motor with higher torque if the motor starts.
	◆ Incorrect external wiring	Inspect the wiring.	Refer to the connection diagram and connect correctly.
	◆ Position control bandwidth, velocity control bandwidth, and/or position integral limiting value are too small.	Inspect.	Adjust to the proper value(s).
The motor rotation is unstable.	◆ Improper connections	Check the motor connections in phases A, B, C, and GND.	Refer to the connection diagram and connect correctly.
	◆ Incorrect motor/driver model combination	Check the model numbers on the rating nameplates.	If the combination is incorrect, change to the correct combination.
The motor overheats.	◆ Ambient temperature is too high.	Check if the ambient temperature is above 45 °C.	Lower the ambient temperature to 45 °C or less.
	◆ Motor is overloaded.	Check to see if the motor operates without any load.	Reduce the load or replace a motor with higher torque if the motor starts.
Abnormal sounds are generated.	◆ Improper mounting	Mounting screws are loosened.	Tighten the screws.
	◆ Linear guide problem	Check for abnormal sound and vibration from the Linear guide.	Motor replacement is necessary. (Contact us.)
	◆ Mounting base vibration	Check the mounting base.	Reinforce the mounting base.
Motor torque is too small.	◆ Incorrect motor/head amp model combination	Check the model numbers on rating nameplates.	If the combination is incorrect, change it to the correct combination.
	◆ Motor is overloaded.	Check the OVL error signal.	Review the operation. Reduce the load.
	◆ Position control bandwidth, velocity control bandwidth, and/or position integral limiting value are too small.	Inspect.	Adjust to the proper value(s).
Motor runs out of control.	◆ Incorrect motor/head amp model combination	Check the model numbers on rating nameplates.	If the combination is incorrect, change it to the correct combination.
	◆ Improper connections	Check the motor/encoder connections	Refer to the connection diagram and connect correctly.

Chapter 10
Specifications

10.1 Standard Specifications
10.2 Speed - Force Characteristics
10.3 External Dimensions (Unit: mm)

10.1 Standard Specifications

(1) Motor part

Note: A: standard, B: high-rigidity, C: high-speed, D: high-rigidity + high-speed

		50N series				100N series				300N series				400N series
Motor classification		A	B	C	D	A	B	C	D	A	B	C	D	B
Maximum force (N) 7)		50		40		100		90		300		270		400
Maximum velocity (m/s) 7)		0.83 (0.16)		2.0		0.83 (0.16)		2.0		0.83 (0.16)		2.0		0.83(0.16)
Encoder resolution (μm/pulse)		0.25		0.5		0.25		0.5		0.25		0.5		0.5
Positioning repeatability accuracy (μm)		± 0.25		± 0.5		± 0.25		± 0.5		± 0.25		± 0.5		± 0.5
Stroke length (Ls) (50 to 1800 mm) dependent absolute positioning accuracy (μm) 6)	Glass scale	$5+\frac{5}{1000}$Ls												
	Metal scale	$5+\frac{5}{1000}$Ls												
Maximum load (N)		200	600	200	600	200	600	200	600	1000	2000	1000	2000	2000
Slider mass (kg)		1.0	1.4	1.0	1.4	1.4	1.8	1.4	1.8	5.0	8.5	5.0	8.5	10.0
Rail mass (kg/m)		12	15	12	15	15	18	15	18	25	4.5	25	4.5	50
Home position signal (pulse/mm)		1/2.048		1/4.096		1/2.048		1/4.096		1/2.048		1/4.096		1/2.048
Stroke length (Ls) (mm)		50/100/150/200/300/400/500/600/700/800/900 1000/1100/1200/1300/1400/1500/1600/1700/1800												
Life span (km)		1)	2)	1)	2)	1)	2)	1)	2)	3)	4)	3)	4)	5)

1) $(2000/(200+W/4))^3\times 50$ 2) $(4700/(200+W/4))^3\times 50$ 3) $(8330/(600+W/4))^3\times 50$ 4) $(2000/(600+W/4))^3\times 50$ W
5) (20000/(800+W/4)3×50 W: Load (N)
6) Scale accuracy specification at 23°C (linear expansion coefficient: $8x10^{-6}$/°C)
7) Both maximum thrust and maximum speed are for a driver input voltage of 200 V. See the characteristics graph for 100 V.

(2) Motor Environment Specification

		Motor	Comment
Ambient operating conditions	Temperature	0 to 45°C	
	Humidity	20 to 85% R.H	Should have no condensation.
Ambient storage conditions	Temperature	-20 to 85°C	
	Humidity	20 to 85% R.H	Should have no condensation.
Operating environment		No corrosive gases and dust should be present.	

(3) Driver Part (General Specifications)

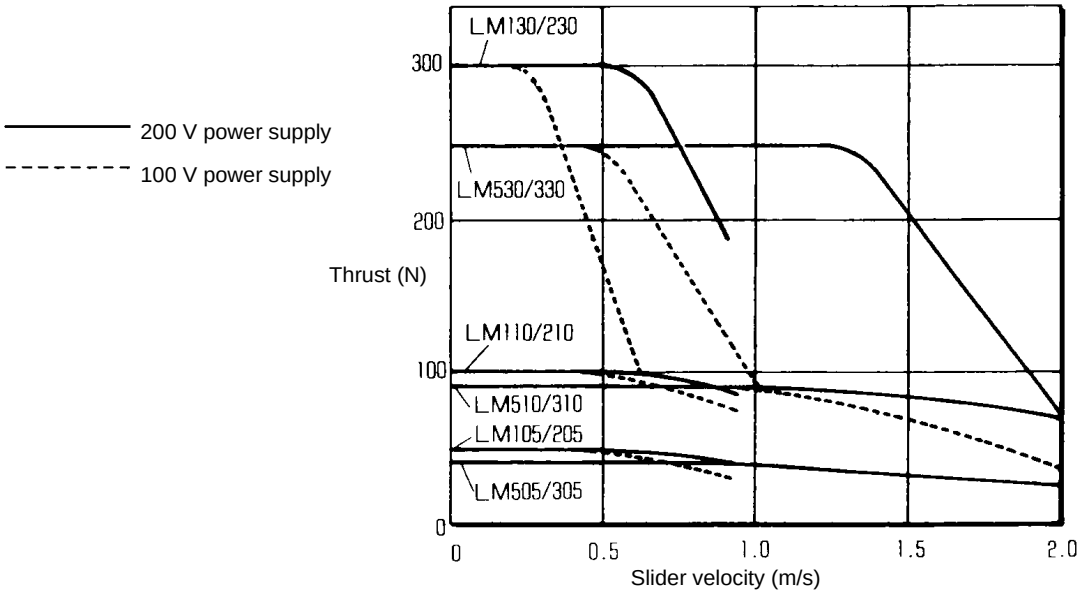
Type	500W type		
Model name	U□□□□□□□A□-*□□_ U□□□□□□□B□-*□□_		
Input power supply voltage	100 to 115V AC + 10%, -15% 50Hz/60Hz	200 to 230V AC + 10%, -15% 50Hz/60Hz	
Maximum current consumption (KVA)	0.8		
Ambient operating air temperature and humidity	0 to 50°C, 20 to 90% RH, without condensation		
Ambient storage air temperature and humidity	-20 to 85°C, 20 to 90% RH, without condensation		
Operating environment	No corrosive gases and dust should be present.		
Mass (kg)	1.7	3.6	3.2

* Input voltage 100 to 115V AC: 1, 200 to 230V AC: 2

(4) Driver Function Specifications

Item		Specifications
Higher interface		RS232C interface (single channel communication, multi-channel communication) PLC interface (can be selected from the following four types at ordering: contact I/O, PROFIBUS-DP, CC-LINK, DeviceNet)
Input signal	Control input signal	Emergency stop, servo command, start, stop, operation number, code input (BCD 2-digit), error reset, integral position control operation disable, interlock, velocity override selection, jog (+) command, job (–) command, M answer input etc.
	Mechanical input signal	Homing signal, (+) direction hardware over-travel signal, (-) direction hardware over-travel signal
Output signal	Control input signal	CPU ready, servo ready, operation under execution, error status, alarm status, position settling status, area signal 0, area signal 1, M code enable, code output (BCD 2-digit) etc.
	Mechanical input signal	Brake signal
Position detector resolution		Rotating type Standard DMA series: 4096000 pls/rev, standard DMB series: 2621440 pls/rev, small-diameter/flat: 2621440 pls/rev Standard DRA series: 1638400 pls/rev, standard DRB series: 1015808 pls/rev, standard DRE series: 1228800 pls/rev High-speed DRB series: 557056 pls/rev, high-speed DRE series: 638976 pls/rev Linear Standard, high-stiffness series: 0.25 μm, high-speed, high-speed/high-stiffness series: 0.5 μm,
Coordinate system		Either rotational coordinates or linear coordinates can be selected. Command unit coordinate (rotating type): pulse, angle (1/100, 1/1000, 1/10000 degree), user setting unit (linear): pulse, angle (10μm, 1μm, 0.1μm), user setting units Operation unit: command unit, index Type A, index Type B
Control part	Method	I-PD position control (position: integral proportional control, velocity: proportional control) Various feed forward functions (position, velocity, acceleration) Various standard filters (velocity command filter, velocity feedback filter, first order delay filter) Optional filter (notch filter 2 channels)
	Adjustment	Position control bandwidth:1 Hz to 32Hz, velocity control loop width: 5Hz to 200Hz Position integral limiter setting Various feed forward percentages (position, velocity, acceleration) Various standard filter settings (velocity command filter bandwidth, velocity feedback filter enable/disable, bandwidth, first order delay filter setting) Optional filter setting (notch filter bandwidth) *1) Calculates proportional gain and acceleration feed forward gain of the velocity control part automatically based on measurement by the auto-tuning operation or manual setting of the load inertia/weight with respect to the settings of velocity control bandwidth and acceleration feed forward percentage. *2) Calculates position control bandwidth, velocity control loop bandwidth, and position integral limiting value automatically during execution of the auto-tuning operation or by manual setting of the servo stiffness.
Acceleration/deceleration control		Trapezoidal move: Acceleration curve and deceleration curve can be selected individually. Acceleration time or deceleration time can be selected individually (with respect to the maximum velocity). Cam profile move: Cam profile selection (8 standard parts and 8 user parts) Acceleration curve or deceleration curve at velocity override change can be selected individually. Acceleration time or deceleration time at velocity override change can be selected individually (ratio to override is 100%). Feed time setting or maximum velocity setting *3) Real time override possible, interlock possible
Operation function		Jog move operation, test operation, auto-tuning operation, homing operation, signal search operation, index operation A, index operation B, table reference operation, program operation, MDI operation
	Program operation	No. of program blocks: 1000, No. of programs:100 (including 10 fixed programs) Possible to change parameters and reference monitors in a program. IF statements, FOR statements, WHILE statements, and subprogram calls are possible. Possible to write programs that use variables (50 registration variables and 50 volatile variables). Step execution and repetitive execution are possible.

10.2 Speed - Force Characteristics

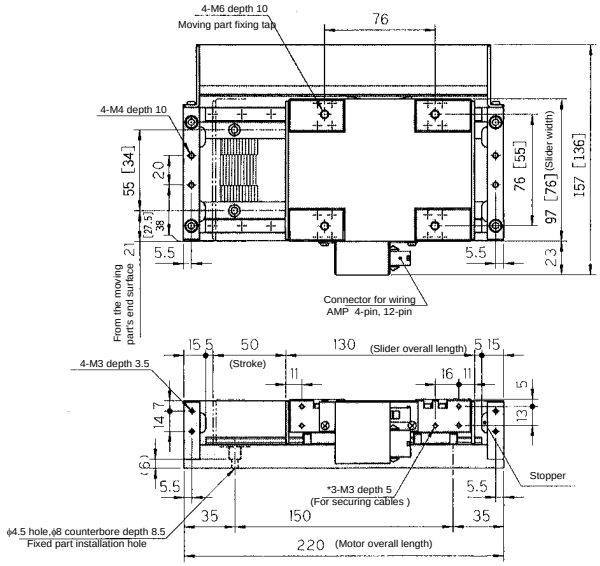


10.3 External Dimensions (Unit: mm)

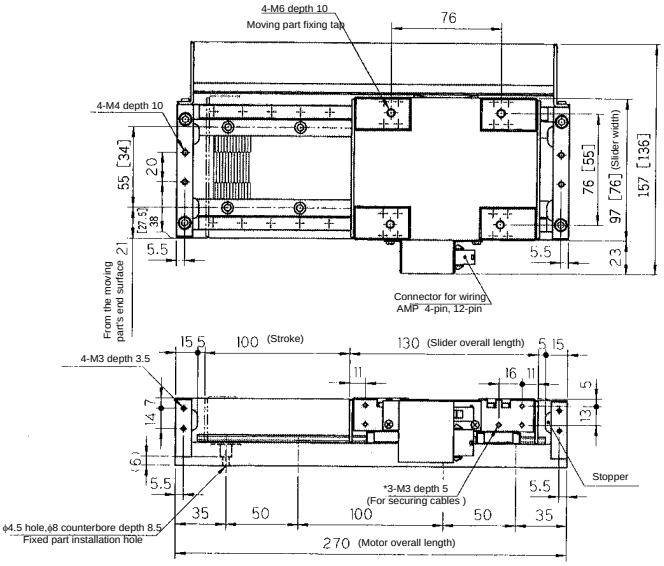
Note: The high-speed model is the same as the standard model, and the high-speed + high-rigidity model is the same as the high-rigidity model.

(1) 100N standard model (LM series) Dimensions in [ ] are for 50N series.

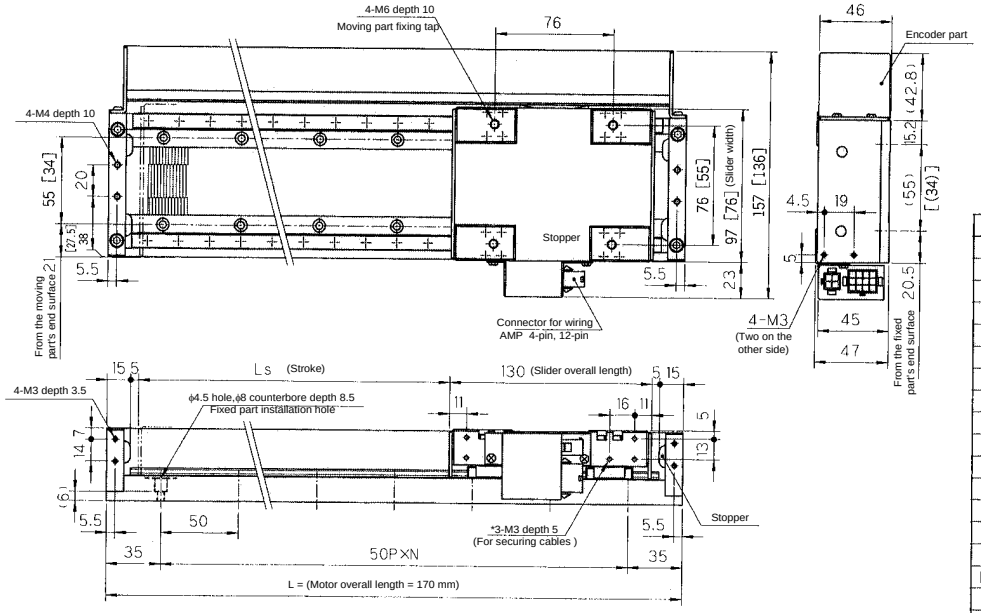
1) Stroke length 50 mm



2) Stroke length 100 mm



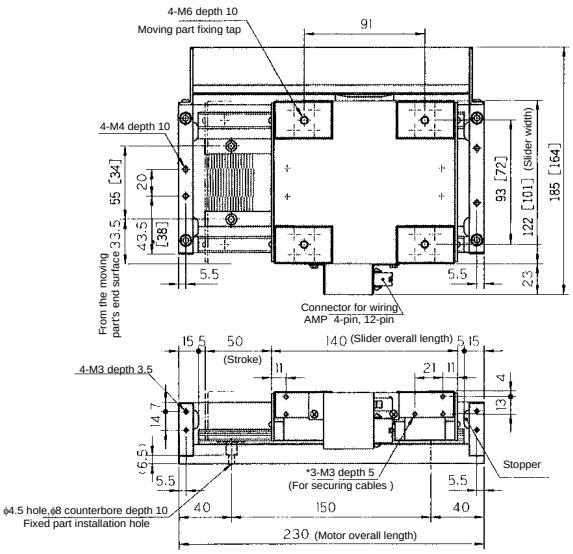
3) Stroke length 150 to 1800 mm



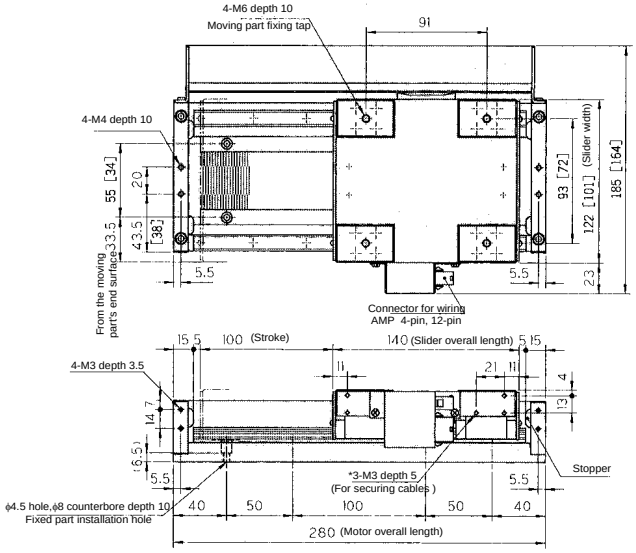
Part number	L	Ls	N	Fixed part installation hole
LM11010F00-0002	320	150	5	2 x 6 holes
LM11010G00-0002	370	200	6	2 x 7 holes
LM11010H00-0002	470	300	8	2 x 9 holes
LM11010J00-0002	570	400	10	2 x 11 holes
LM11010K00-0002	670	500	12	2 x 13 holes
LM11010L00-0002	770	600	14	2 x 15 holes
LM11010M00-0002	870	700	16	2 x 17 holes
LM11010N00-0002	970	800	18	2 x 19 holes
LM11010P00-0002	1070	900	20	2 x 21 holes
LM11010Q00-0002	1170	1000	22	2 x 23 holes
LM11010R00-0002	1270	1100	24	2 x 25 holes
LM11010S00-0002	1370	1200	26	2 x 27 holes
LM11010T00-0002	1470	1300	28	2 x 29 holes
LM11010U00-0002	1570	1400	30	2 x 31 holes
LM11010V00-0002	1670	1500	32	2 x 33 holes
LM11010W00-0002	1770	1600	34	2 x 35 holes
LM11010X00-0002	1870	1700	36	2 x 37 holes
LM11010Y00-0002	1970	1800	38	2 x 39 holes

(2) 100N high-rigidity model (LM210 series) Dimensions in [ ] are for 50N series.

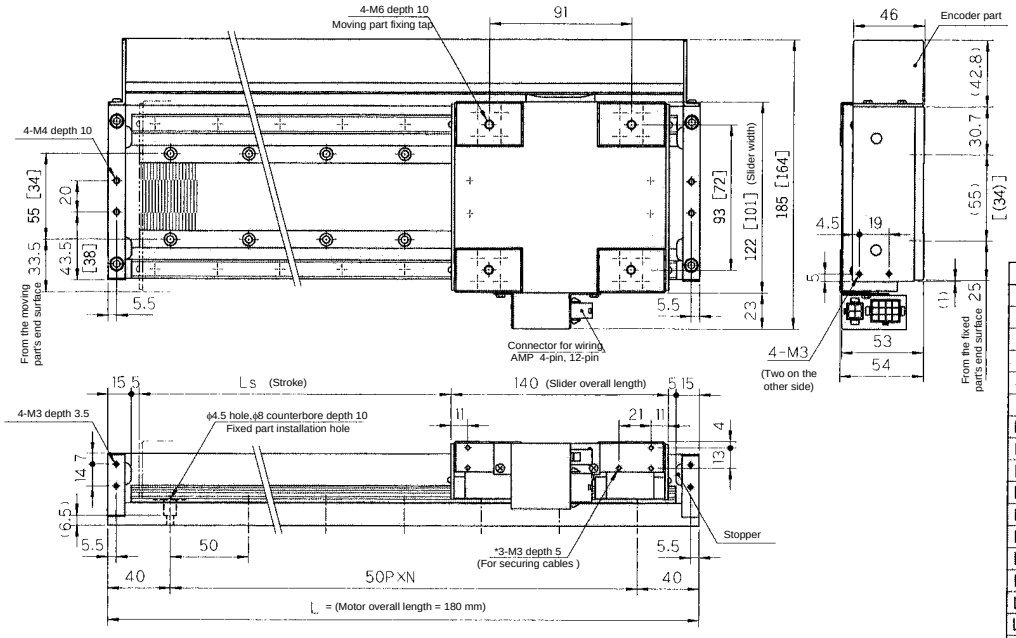
1) Stroke length 50 mm



2) Stroke length 100 mm



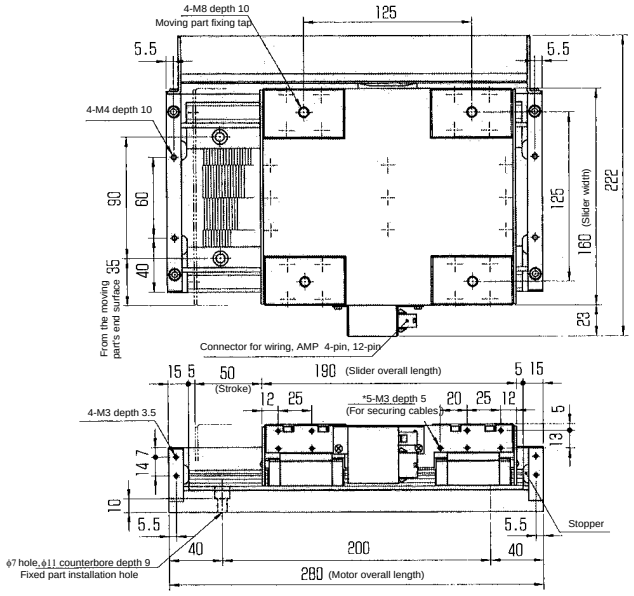
3) Stroke length 150 to 1800 mm



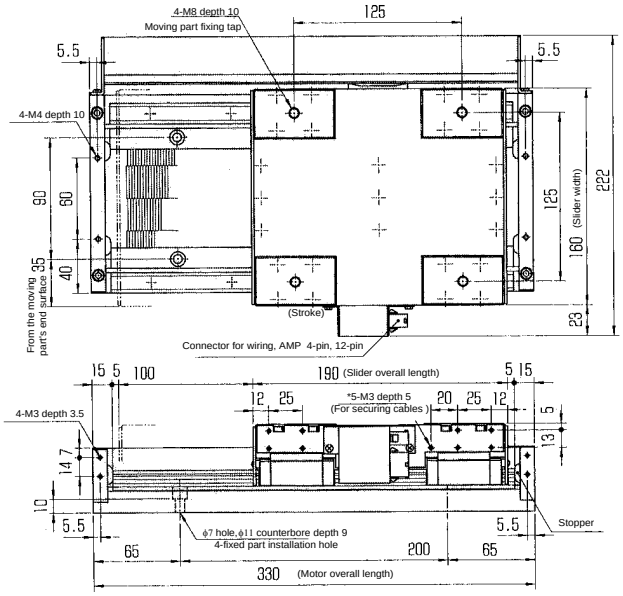
Part number	L	Ls	N	Fixed part installation hole
LM21010FU0-0002	330	150	5	2 x 6 holes
LM21010G00-0002	380	200	6	2 x 7 holes
LM21010H00-0002	480	300	8	2 x 9 holes
LM21010J00-0002	580	400	10	2 x 11 holes
LM21010K00-0002	680	500	12	2 x 13 holes
LM21010L00-0002	780	600	14	2 x 15 holes
LM21010M00-0002	880	700	16	2 x 17 holes
LM21010N00-0002	980	800	18	2 x 19 holes
LM21010P00-0002	1080	900	20	2 x 21 holes
LM21010Q00-0002	1180	1000	22	2 x 23 holes
LM21010R00-0002	1280	1100	24	2 x 25 holes
LM21010S00-0002	1380	1200	26	2 x 27 holes
LM21010T00-0002	1480	1300	28	2 x 29 holes
LM21010U00-0002	1580	1400	30	2 x 31 holes
LM21010V00-0002	1680	1500	32	2 x 33 holes
LM21010W00-0002	1780	1600	34	2 x 35 holes
LM21010X00-0002	1880	1700	36	2 x 37 holes
LM21010Y00-0002	1980	1800	38	2 x 39 holes

(3) 300N standard model (LM130 series)

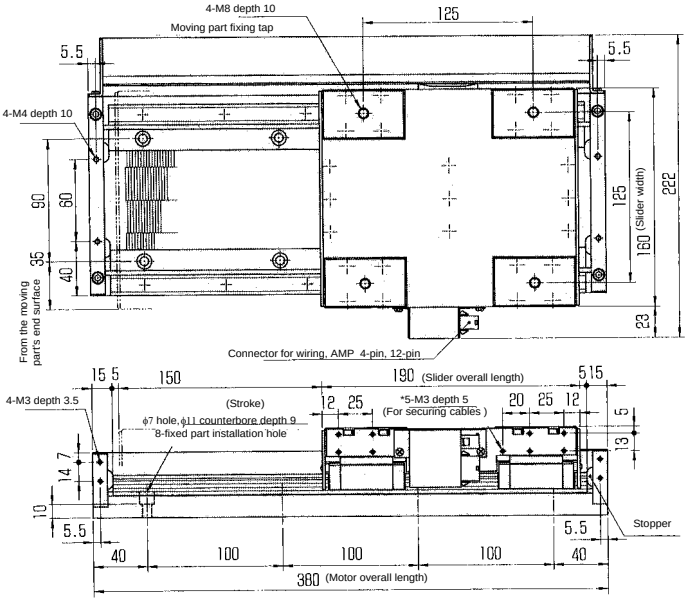
1) Stroke length 50 mm



2) Stroke length 100 mm

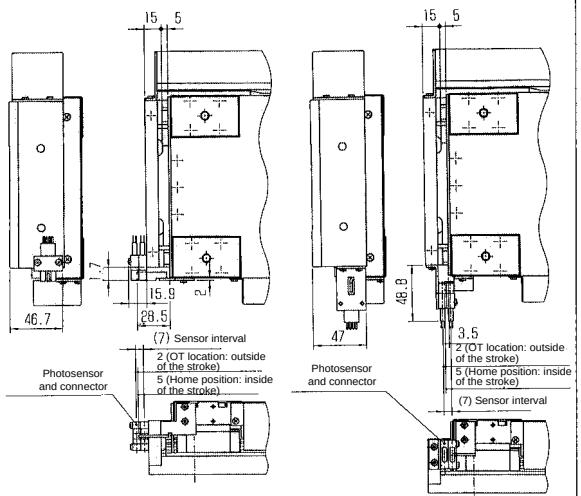


3) Stroke length 150 mm



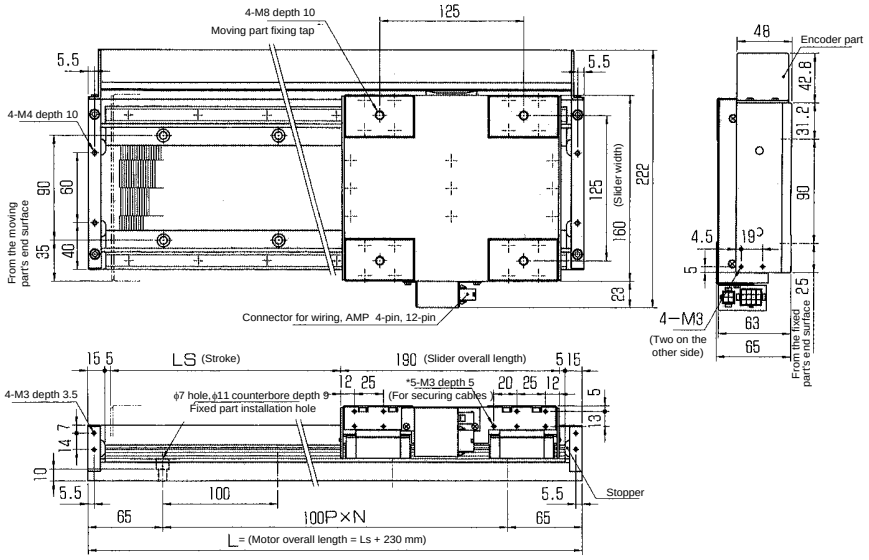
Sensor installation location
(applicable to all models)

(1) Outside of the sensor (2) Inside of the sensor



Photosensor: EE-SX670A (Omron)
Connector: EE-1001 (Omron)
Two on each side, four in total

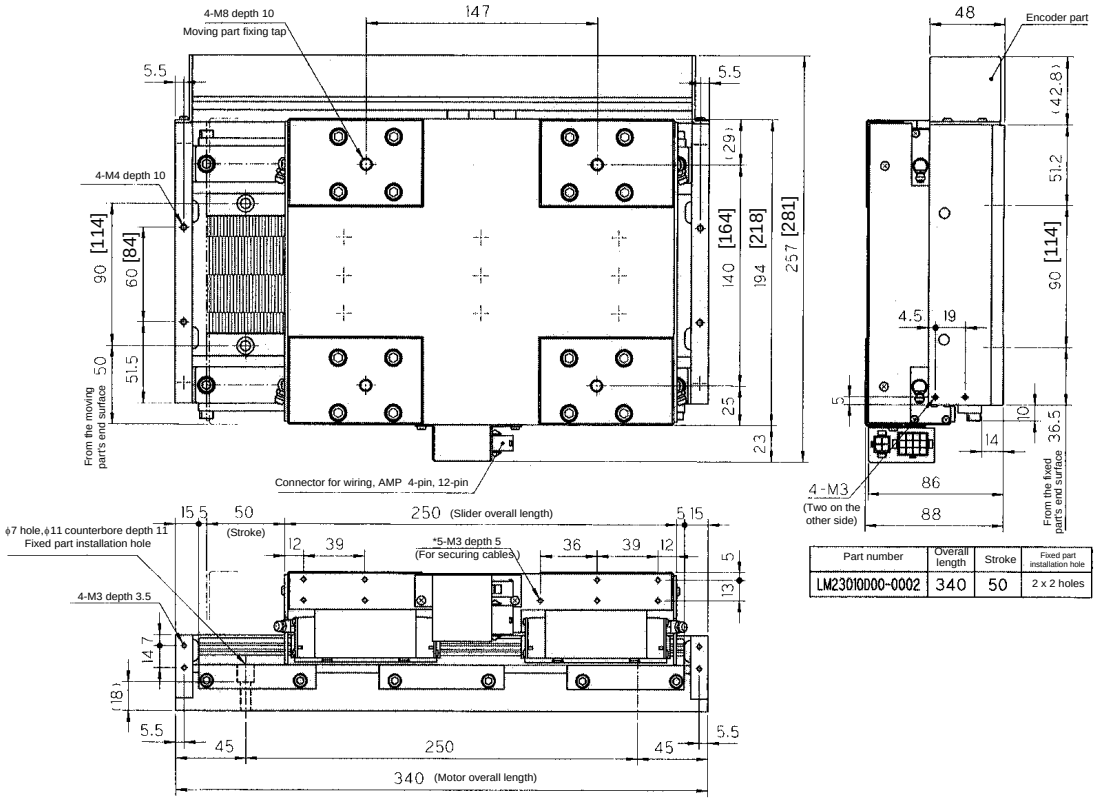
4) Stroke length 150 to 1800 mm



Part number	L	Ls	N	Fixed part installation hole
LM130 10G00-0002	430	200	3	2 x 4 holes
LM130 10H00-0002	530	300	4	2 x 5 holes
LM130 10J00-0002	630	400	5	2 x 6 holes
LM130 10K00-0002	730	500	6	2 x 7 holes
LM130 10L00-0002	830	600	7	2 x 8 holes
LM130 10M00-0002	930	700	8	2 x 9 holes
LM130 10N00-0002	1030	800	9	2 x 10 holes
LM130 10P00-0002	1130	900	10	2 x 11 holes
LM130 10Q00-0002	1230	1000	11	2 x 12 holes
LM130 10R00-0002	1330	1100	12	2 x 13 holes
LM130 10S00-0002	1430	1200	13	2 x 14 holes
LM130 10T00-0002	1530	1300	14	2 x 15 holes
LM130 10U00-0002	1630	1400	15	2 x 16 holes
LM130 10V00-0002	1730	1500	16	2 x 17 holes
LM130 10W00-0002	1830	1600	17	2 x 18 holes
LM130 10X00-0002	1930	1700	18	2 x 19 holes
LM130 10Y00-0002	2030	1800	19	2 x 20 holes

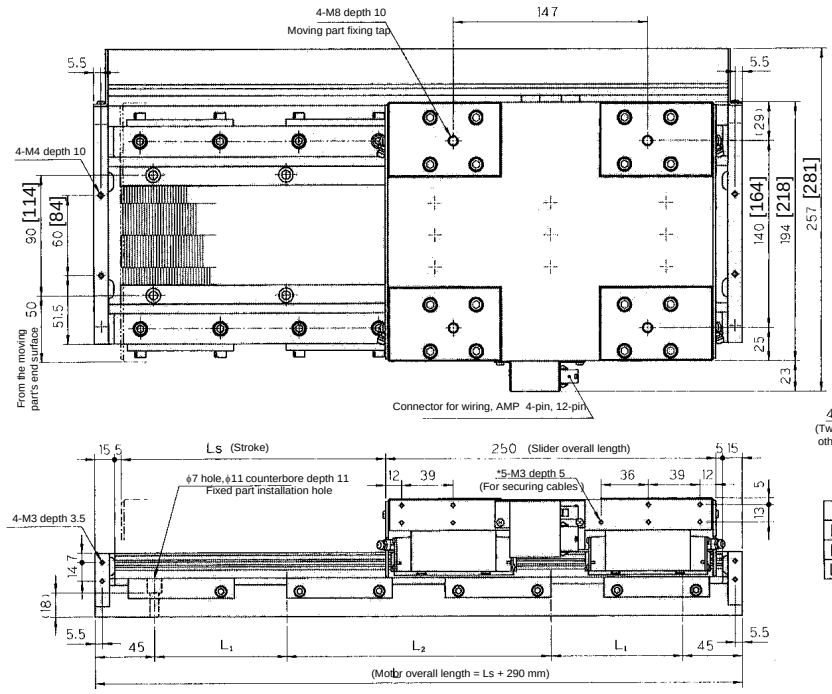
(4) 300N high-rigidity model (LM230 series) Dimensions in [ ] are for 400N high-rigidity model (LM240)

1) Stroke length 50 mm



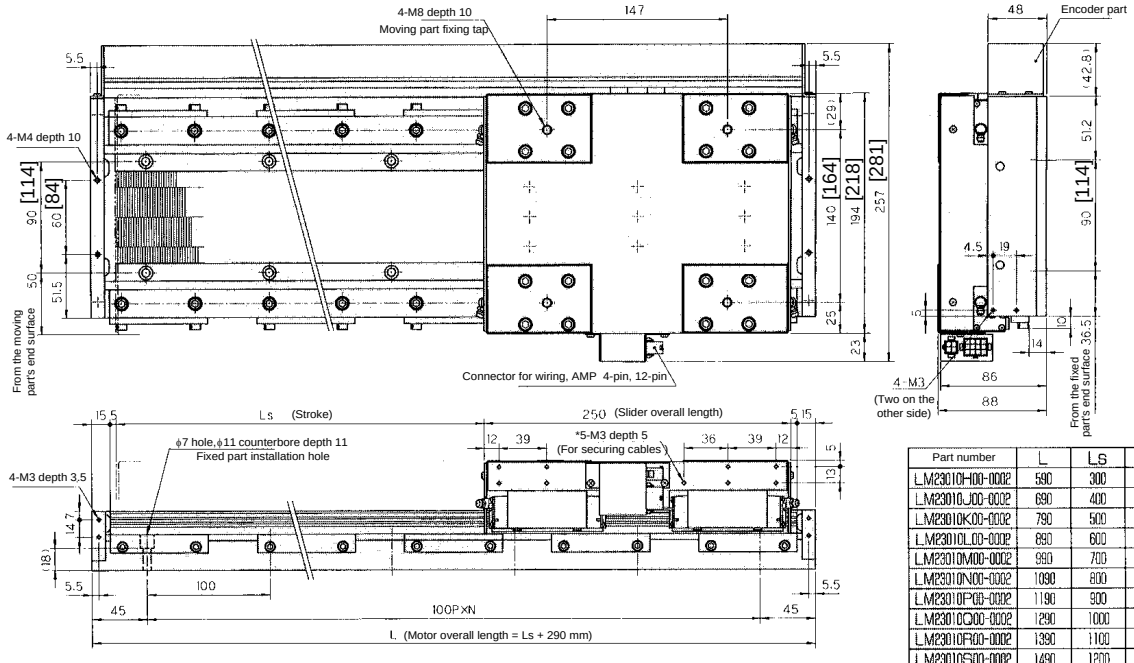
Part number	Overall length	Stroke	Fixed part installation hole
LM23010D00-0002	340	50	2 x 2 holes

2) Stroke length 100 to 200 mm



Part number	L	Ls	L1	L2	Fixed part installation hole
LM23010E00-0002	390	100	50	200	2 x 4 holes
LM23010F00-0002	440	150	100	150	2 x 4 holes
LM23010G00-0002	490	200	100	200	2 x 4 holes

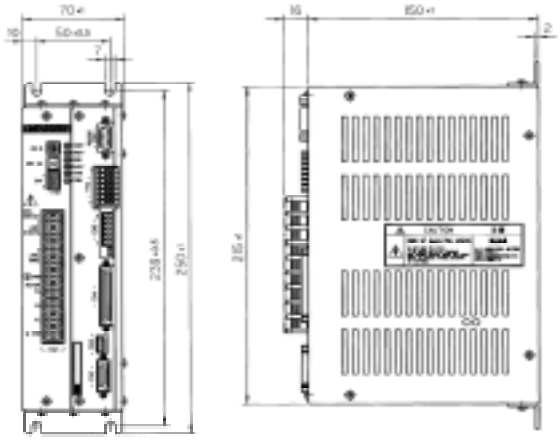
3) Stroke length 300 to 1800 mm



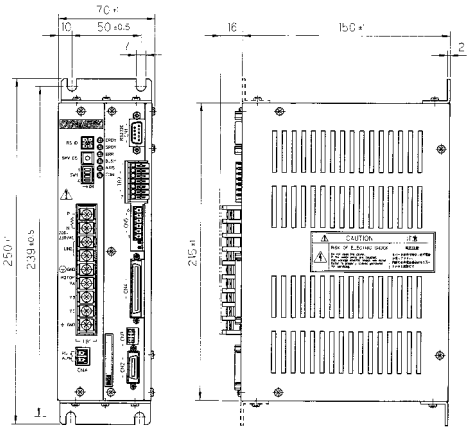
Part number	L	Ls	N	Fixed part installation hole
LM23010H00-0002	590	300	5	2 x 6 holes
LM23010J00-0002	690	400	6	2 x 7 holes
LM23010K00-0002	790	500	7	2 x 8 holes
LM23010L00-0002	890	600	8	2 x 9 holes
LM23010M00-0002	990	700	9	2 x 10 holes
LM23010N00-0002	1090	800	10	2 x 11 holes
LM23010P00-0002	1190	900	11	2 x 12 holes
LM23010Q00-0002	1290	1000	12	2 x 13 holes
LM23010R00-0002	1390	1100	13	2 x 14 holes
LM23010S00-0002	1490	1200	14	2 x 15 holes
LM23010T00-0002	1590	1300	15	2 x 16 holes
LM23010U00-0002	1690	1400	16	2 x 17 holes
LM23010V00-0002	1790	1500	17	2 x 18 holes
LM23010W00-0002	1890	1600	18	2 x 19 holes
LM23010X00-0002	1990	1700	19	2 x 20 holes
LM23010Y00-0002	2090	1800	20	2 x 21 holes

(5) Driver dimensions (applicable to all models) Unit: mm

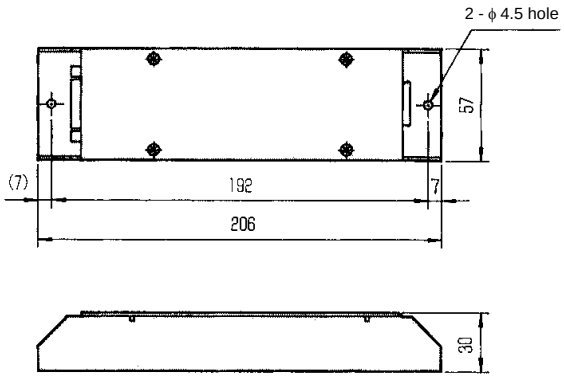
1) U□□□□□□□□A
(without regenerative unit)



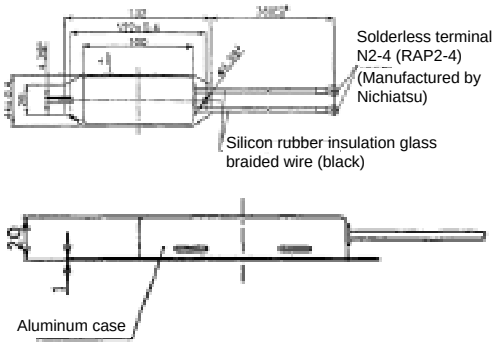
2) U□□□□□□□□B
(with regenerative unit)



(6) Head amp part (applicable to all models)



(7) Regenerative Resistor (LM330/240 only)



Parameter List

STD1

Parameter No.	Parameter name	Minimum value	Maximum value	Initial value	Unit	Possible to change
1	Enables the over-travel error function in the + direction	0	1	0	None	Always
2	Enables the over-travel error function in the - direction	0	1	0	None	Always
3	Selecting the type of cam profile move	0	1	1	None	Always
4	Selecting the acceleration type	0	1	0	None	Always
5	Selecting the deceleration type	0	1	0	None	Always
6	Selection of the cam profile	1	16	6	None	Always
7	Acceleration time during a trapezoidal move	1	9999	1000	msec	Always
8	Deceleration time during a trapezoidal move	1	9999	1000	msec	Always
9	Feeding Velocity	0	16000000	Motor dependent	Axis command unit/sec	Always
10	Jog Velocity	0	16000000	Motor dependent	Axis command unit/sec	Always
11	Over-travel search velocity during a homing move	1	16000000	Motor dependent	Axis command unit/sec	Always
12	Homing operation: Home sensor proximity signal search velocity	1	16000000	Motor dependent	Axis command unit/sec	Always
15	Homing operation: Origin position offset move feed velocity	1	16000000	Motor dependent	Axis command unit/sec	Always
16	Velocity override percentage 1	0	20000	10000	1/100 %	Always
17	Velocity override percentage 2	0	20000	10000	1/100 %	Always
20	Homing direction	0	1	0	None	Always
21	Enable/disable the over-travel signal under the homing mode	0	1	0	None	Always
27	Enabling the homing flag position error	0	1	1	None	Always
29	Offset distance from the Home position	-9999999	9999999	0	Axis command unit	Always
30	Homing complete operation command value	-9999999	9999999	0	Axis operation unit	Always
31	Operation width under testing mode	0	9999999	Motor dependent	Axis command unit	Always
32	Operation width under Auto-tuning	1	9999999	Motor dependent	Axis command unit	Always
33	Maximum acceleration/deceleration under Auto-tuning	100	9999	9999	msec	Always
34	Initializing the acceleration/deceleration time while under Auto-tuning	100	9999	1000	msec	Always
38	Servo stiffness settings	1	5	3	None	Always
39	Signal search direction	0	1	0	None	Always

Parameter No.	Parameter name	Minimum value	Maximum value	Initial value	Unit	Possible to change
41	Enabling the signal search mark signal	0	1	0	None	Always
42	Enabling the proximity signal during homing	0	1	0	None	Always
43	Enabling the (+) direction over travel signal during the signal search	0	1	0	None	Always
44	Enabling the (-) direction over travel signal during the signal search	0	1	0	None	Always
46	Signal search mark signal logic settings	0	1	0	None	Always
47	Logic for the proximity signal during homing	0	1	0	None	Always
48	Logic for the (+) direction over-travel signal during the signal search mode	0	1	0	None	Always
49	Logic for the (-) direction over-travel signal during the signal search mode	0	1	0	None	Always
50	Position control bandwidth	1	32	1	Hz	Always
51	Velocity control bandwidth	5	200	20	Hz	Always
53	Position integral limiting value	0	4999999	10000	None	Always
54	Position feed forward percentage	0	126	90	%	Always
55	Velocity feed forward percentage	0	126	100	%	Always
56	Acceleration feed forward gain	0	200	0	%	Always
58	Positioning settling width	0	9999999	5	Axis command unit	Always
59	Position deviation filter frequency	1	250	10	Hz	Always
60	Filter use when generating position settling signal	0	1	0	None	Always
65	Value causing an error detection in the (+) or CW direction	1	999999	999999	pulse	Always
66	Value causing an error detection in the (-) or CCW direction	-999999	-1	-999999	pulse	Always
68	Torque limit percentage	0	10000	10000	1/100 %	Always
69	Axis velocity monitoring gain (digital monitor)	0	8	0	None	Always
70	Analog monitor selection	0	5	4	None	Always
71	Axis positioning error monitoring gain (analog monitor)	0	8	0	None	Always
72	Test operation monitoring gain (analog monitor)	0	8	0	None	Always
73	Position monitoring gain (analog monitor)	0	14	0	None	Always
74	Position differential value monitoring gain (analog monitor)	0	7	0	None	Always
75	Velocity monitor selection (digital monitor)	0	1	0	None	Always
79	Move time during a cam profile selection	1	30000	2000	msec	Always

Parameter No.	Parameter name	Minimum value	Maximum value	Initial value	Unit	Possible to change
80	Acceleration time during a cam profile selection	1	9999	1000	msec	Always
81	Deceleration time during a cam profile selection	1	9999	1000	msec	Always
82	Enabling the Program step execution	0	1	0	None	Always
83	ABS in program/MDI operation	0	1	0	None	Always
84	Enabling the peak velocity during a cam profile move	0	1	0	None	Always
85	Enable the (+) direction soft limit error	0	1	0	None	Always
86	Enable the (-) direction soft limit error	0	1	0	None	Always
87	(+) direction soft limit settings	-999999999	999999999	0	Axis command unit	Always
88	(-) direction soft limit settings	-999999999	999999999	0	Axis command unit	Always
89	Brake turn OFF delay time upon Servo ON	0	2000	0	msec	Always
90	Advanced Brake turn ON before Servo OFF	0	2000	0	msec	Always
91	TBX_EMG Servo status	0	2	0	None	Always
92	Start-up program enable	0	1	0	None	Always
93	IFB_EMG Servo status	0	2	0	None	Always
100	M function enable in program and MDI operation	0	1	0	None	Always
101	M function enable during index operation	0	1	0	None	Always
102	Enabling the selection of serial communication type for the M function interface	0	1	0	None	Always
103	Optional stop enable	0	1	0	None	Always
104	ABS/INC setting during table and index operation	0	2	0	None	Always
105	Move direction option for rotation coordinates	0	4	0	None	Always
106	Settling wait enable	0	1	0	None	Always
108	Operation unit selection	0	2	0	None	Always
109	Index Type A divisions setting	-1	100	4	None	Always
110	Index Type B divisions setting	-1	0	0	None	Always
152	First order delay compensator setting	0	3	0	None	Always
153	Notch filter: Frequency 1 selection	50	1500	1500	Hz	Always
154	Notch filter: Frequency 2 selection	50	1500	1500	Hz	Always
155	Load inertial/load mass	0	200000	0	1/1000kgm^2 or 1/1000kg	Always

Parameter No.	Parameter name	Minimum value	Maximum value	Initial value	Unit	Possible to change
161	Area signal 0 ON	-999999999	999999999	0	Axis command unit	Always
162	Area signal 0 OFF	-999999999	999999999	0	Axis command unit	Always
163	Area signal 1 ON	-999999999	999999999	0	Axis command unit	Always
164	Area signal 1 OFF	-999999999	999999999	0	Axis command unit	Always
201	Selection of English/Japanese display	0	1	0	None	While in machine setting mode
202	Coordinate (+) direction setting	0	1	1	None	While in machine setting mode
208	Command unit selection	0	4	1	None	While in machine setting mode
209	Scaling data (command unit side)	1000	99999999	Motor dependent	Axis command unit	While in machine setting mode
210	Scaling data (pulse side)	1000	99999999	Motor dependent	pulse	While in machine setting mode
212	Straight line coordinate selection	0	1	Motor dependent	None	While in machine setting mode
213	Maximum velocity	1	16000000	Motor dependent	Axis command unit/sec	While in machine setting mode
215	IFB operation: Start signal processing speed selection	0	1	0	None	While in machine setting mode
216	Servo ON status upon power up during serial communication operations	0	1	1	None	While in machine setting mode
217	Jog move operation: Serial communication selection	0	1	0	None	While in machine setting mode
218	Enables error when over-load occurs	0	1	1	None	While in machine setting mode
219	Velocity feedback filter use	0	1	0	None	While in machine setting mode
220	Velocity feedback filter bandwidth	50	1000	1000	Hz	While in machine setting mode
221	Velocity command filter bandwidth	50	1000	1000	Hz	While in machine setting mode
222	Enables error when over-speed occurs	0	1	1	None	While in machine setting mode
224	Enables error when excessive position deviation occurs	0	1	1	None	While in machine setting mode
227	Over-speed error processing type	0	2	1	None	While in machine setting mode
228	Over-load error processing type	0	2	1	None	While in machine setting mode
229	Excessive position deviation error processing type	0	2	1	None	While in machine setting mode

Parameter details

STD1

1 Enables the over-travel error function in the + direction — Always

Minimum value:	0	**Long:**	+ﾎｳｺｳﾊｰﾄﾞ OT ｴﾗｰﾕｳｺｳ
Maximum value:	1		+HOT_ErrorEnable
Initial value:	0	**Short:**	+HOT_ErrEn
Unit:	None		+HOT_ErrEn

Specify whether or not to enable an error when the over-travel signal in the + direction is detected while commanding to move the axis in the + direction.

0: Does not enable an error.

1: Enables an error.

2 Enables the over-travel error function in the - direction — Always

Minimum value:	0	**Long:**	-ﾎｳｺｳﾊｰﾄﾞ OT ｴﾗｰﾕｳｺｳ
Maximum value:	1		-HOT_ErrorEnable
Initial value:	0	**Short:**	-HOT_ErrEn
Unit:	None		-HOT_ErrEn

Specify whether or not to enable an error when the over-travel signal in the - direction is detected while commanding to move the axis in the - direction.

0: Does not enable an error.

1: Enables an error.

3 Selecting the type of cam profile move — Always

Minimum value:	0	**Long:**	ｶﾑｹｲｼﾞｮｳｲﾄﾞｳｾﾝﾀｸ
Maximum value:	1		CamMoveSel
Initial value:	1	**Short:**	ｶﾑｲﾄﾞｳ
Unit:	None		CamMoveSel

Specify the operation curve type during move operation that accompanies positioning (displacement amount is already known).

0: Trapezoidal motion profile

1: Cam profile motion profile

4 Selecting the acceleration type — Always

Minimum value:	0	**Long:**	ｶｿｸｹｲｼﾞｮｳｾﾝﾀｸ
Maximum value:	1		AccProfeel
Initial value:	0	**Short:**	ｶｿｸｹｲｼﾞｮｳ
Unit:	None		AccProfeel

Select the acceleration type during trapezoidal move.

0: Constant acceleration

1: S shaped

5 Selecting the deceleration type — Always

Minimum value:	0	**Long:**	ｹﾞﾝｿｸｹｲｼﾞｮｳｾﾝﾀｸ
Maximum value:	1		DecProfeel
Initial value:	0	**Short:**	ｹﾞﾝｿｸｹｲｼﾞｮ
Unit:	None		DecProfeel

Select the deceleration type during trapezoidal move.

0: Constant deceleration

1: S shaped

6 Selection of the cam profile — Always

Minimum value:	1	**Long:**	ｶﾑｹｲｼﾞｮｳｾﾝﾀｸ
Maximum value:	16		CamProfeel
Initial value:	6	**Short:**	ｶﾑｹｲｼﾞｮｳ
Unit:	None		DecProfeel

Select the cam profile during cam profile move.

1 to 8: Standard installed cam profile

9 to 16: User registered cam profile

7 Acceleration time during a trapezoidal move — Always

Minimum value:	1	**Long:**	Tacc ﾀﾞｲｹｲｲﾄﾞｳｼﾞ
Maximum value:	9999		TaccTrapezoid
Initial value:	1000	**Short:**	Tacc ﾀﾞｲｹｲ
Unit:	msec		TaccTrapez

Specify the acceleration time required for velocity change for the maximum velocity during trapezoidal move.

8 Deceleration time during a trapezoidal move — Always

Minimum value:	1	**Long:**	Tdec ﾀﾞｲｹｲｲﾄﾞｳｼﾞ
Maximum value:	9999		TdecTrapezoid
Initial value:	1000	**Short:**	Tdec ﾀﾞｲｹｲ
Unit:	msec		TdecTrapez

Specify the deceleration time required for velocity change for the maximum velocity during trapezoidal move.

9 Feeding Velocity — Always

Minimum value:	0	**Long:**	ｵｸﾘｿｸﾄﾞ
Maximum value:	16000000		FeedVelocity
Initial value:	Motor dependent	**Short:**	ｵｸﾘｿｸﾄﾞ
Unit:	Axis command unit/sec		FeedVel

Specify the feeding velocity.

For trapezoidal move: Specify the feeding velocity.

For cam move: Specify the feeding velocity (peak velocity).

10 Jog Velocity — Always

Minimum value:	0	**Long:**	ｼﾞｮｸﾞｵｸﾘｿｸﾄﾞ
Maximum value:	16000000		JogVelocity
Initial value:	Motor dependent	**Short:**	ｼﾞｮｸﾞｿｸﾄﾞ
Unit:	Axis command unit/sec		JogVel

Specify the feeding velocity in jog mode.

11 Over-travel search velocity during a homing move — Always

Minimum value:	1	**Long:**	ORG-OT ｻｰﾁｵｸﾘｿｸﾄﾞ
Maximum value:	16000000		ORG-OT_SearchVel
Initial value:	Motor dependent	**Short:**	ORG-OT-VEL
Unit:	Axis command unit/sec		ORG-OT-Vel

Specify the velocity when executing over-travel signal search move in homing mode. This parameter is invalid if the over-travel signal search is not executed.

12 Homing operation: Home sensor proximity signal search velocity — Always

Minimum value:	1	**Long:**	ORG-ｷﾝﾎﾞｳｻｰﾁｵｸﾘｿｸﾄﾞ
Maximum value:	16000000		ORG-ORG_SearchVel
Initial value:	Motor dependent	**Short:**	ORG-ORGVel
Unit:	Axis command unit/sec		ORG-ORGVel

Specify the velocity to search the home proximity signal in homing mode. This parameter is invalid if the origin proximity signal is not used.

15 Homing operation: Origin position offset feed velocity **Always**

Minimum value:	1	**Long:**	ORG-ｵﾌｾｯﾄｲﾄﾞｳｵｸﾘｿｸﾄﾞ
Maximum value:	16000000		ORG-OffsetVel
Initial value:	Motor dependent	**Short:**	ORG-OffVel
Unit:	Axis command unit/sec		ORG-OffVel

Specify the velocity when executing origin offset move in homing mode. This parameter is invalid if the origin offset is 0.

16 Velocity override percentage 1 **Always**

Minimum value:	0	**Long:**	ｿｸﾄﾞｵｰﾊﾞﾗｲﾄﾞ 1
Maximum value:	20000		VelOverride1
Initial value:	10000	**Short:**	V ｵｰﾊﾞﾗｲﾄﾞ 1
Unit:	1/100 %		VelOvrrid1

Specify override 1 for feeding velocity.

17 Velocity override percentage 2 **Always**

Minimum value:	0	**Long:**	ｿｸﾄﾞｵｰﾊﾞﾗｲﾄﾞ 2
Maximum value:	20000		VelOverride2
Initial value:	10000	**Short:**	V ｵｰﾊﾞﾗｲﾄﾞ 2
Unit:	1/100 %		VelOvrrid2

Specify override 2 for feeding velocity.

20 Homing direction **Always**

Minimum value:	0	**Long:**	ORG-ｹﾞﾝﾃﾝﾌｯｷﾎｳｺｳ
Maximum value:	1		ORG-Direction
Initial value:	0	**Short:**	O-OrgDir
Unit:	None		O-OrgDir

Specify the homing direction (origin proximity signal search direction) in homing mode.

0: - direction

1: + direction

21 Enable/Disable the over-travel signal under the homing mode **Always**

Minimum value:	0	**Long:**	ORG-OT ｼﾝｺﾞｳｼﾖｳ
Maximum value:	1		ORG-OT_SignalUse
Initial value:	0	**Short:**	O-OT_Use
Unit:	None		O-OT_Use

Specify whether or not to execute the over-travel signal search in homing mode.

0: Disables the over-travel signal search.

1: Enables the over-travel signal search.

25 Homing operation: Origin inside selection **Always**

Minimum value:	0	**Long:**	ORG-ｹﾞﾝﾃﾝｳﾁｶﾞﾜｾﾝﾀｸ
Maximum value:	1		ORG-InsideSelect
Initial value:	1	**Short:**	O-Inside
Unit:	None		O-Inside

If the home position proximity signal is used in the home position mode, specify whether the Z signal inside the home position proximity signal is set to the origin or the Z signal outside the home position proximity signal is set to the origin. If the home position proximity signal is not used, this parameter is not valid.

0: The Z signal outside the home position proximity signal is set to the origin.

1: The Z signal inside the home position proximity signal is set to the origin.

27 Enabling the homing flag position error — Always

Minimum value:	0	**Long:**	ORG-ﾄﾞｸﾞｲﾁｴﾗｰﾕｳｺｳ
Maximum value:	1		ORG-DogErrorEnable
Initial value:	1	**Short:**	O-DogErrEn
Unit:	None		O-DogErrEn

When the homing mode is completed, an error or warning is generated when the relationship between the origin flag position and motor Z-phase position is as follows.

An error is generated if:

| Origin flag position - Z-phase position | > Z-phase gap x 0.45, or

| Origin flag position - Z-phase position | < Z-phase gap x 0.05

A warning is generated if :

| Origin flag position - Z-phase position | > Z-phase gap x 0.40, or

| Origin flag position - Z-phase position | < Z-phase gap x 0.10

29 Offset distance from the Home position — Always

Minimum value:	-9999999	**Long:**	ORG-ｵﾌｾｯﾄｲﾄﾞｳﾘｮｳ
Maximum value:	9999999		ORG-Offset
Initial value:	0	**Short:**	O-Offset
Unit:	Axis command unit		O-Offset

Specify the origin offset amount in homing mode.

30 Homing complete operation command value — Always

Minimum value:	-9999999	**Long:**	ORG-ｶﾝﾘｮｳｺﾞｳﾝﾃﾝｼﾚｲﾁ
Maximum value:	9999999		ORG-Pcmd[unit]
Initial value:	0	**Short:**	O-ｳﾝﾃﾝｼﾚｲﾁ
Unit:	Axis command unit		ORG-Pcmd

Specify the operation command coordinate value after homing operation is completed.

31 Operation width under testing mode — Always

Minimum value:	0	**Long:**	TEST-ﾄﾞｳｻﾊﾊﾞ
Maximum value:	9999999		TestWidth
Initial value:	Motor dependent	**Short:**	ﾃｽﾄﾄﾞｳｻﾊﾊﾞ
Unit:	Axis command unit		TestWidth

Specify the operation width in test mode.

32 Operation width under Auto-tuning — Always

Minimum value:	1	**Long:**	A_TUNE-ﾄﾞｳｻﾊﾊﾞ
Maximum value:	9999999		A-TUNE-Width
Initial value:	Motor dependent	**Short:**	AT-ﾄﾞｳｻﾊﾊﾞ
Unit:	Axis command unit		AT-Width

Specify the operating range in auto-tuning mode.

33 Maximum acceleration/deceleration under Auto-tuning — Always

Minimum value:	10	**Long:**	A_TUNE-Tacc ｻｲﾀﾞｲﾁ
Maximum value:	9999		A-TUNE_TaccMax
Initial value:	9999	**Short:**	AT-TaccMax
Unit:	msec		AT_TaccMax

Specify the maximum value of acceleration/deceleration time in auto-tuning mode.

34 Initializing the acceleration/deceleration time while under Auto-tuning — **Always**

Minimum value:	100	**Long:**	A_TUNE-Tacc ｼｮｷﾁ
Maximum value:	9999		A-TUNE_TaccIni
Initial value:	1000	**Short:**	AT-TaccIni
Unit:	msec		AT_TaccIni

Specify the initial value of acceleration/deceleration time in auto-tuning mode.

38 Servo stiffness settings — **Always**

Minimum value:	1	**Long:**	ｻｰﾎﾞｳｾｲｾｯﾃｲ
Maximum value:	5		ServoRigidity
Initial value:	3	**Short:**	ｻｰﾎﾞｺﾞｳｾｲ
Unit:	None		ServoRigit

Specify the servo stiffness. (The larger the number is specified, the stronger the servo stiffness becomes. However, the motor vibrates more.)

1: Velocity control width: 30Hz Position control width: 7Hz
2: Velocity control width: 40Hz Position control width: 10Hz
3: Velocity control width: 50Hz Position control width: 12Hz
4: Velocity control width: 60Hz Position control width: 15Hz
5: Velocity control width: 70Hz Position control width: 17Hz

39 Signal search direction — **Always**

Minimum value:	0	**Long:**	SIG-ﾎｳｺｳ
Maximum value:	1		SIG-Direction
Initial value:	0	**Short:**	S-ﾎｳｺｳ
Unit:	None		S-Dir

Specify the search direction in signal search mode.

0: - direction
1: + direction

41 Enabling the signal search mark signal — **Always**

Minimum value:	0	**Long:**	SIG-ﾏｰｸｼﾝｺﾞｳｭｳｺｳ
Maximum value:	1		SIG-MarkEnable
Initial value:	0	**Short:**	S-MrkEn
Unit:	None		S-MrkEn

Specify whether or not to search the mark signal in signal search mode.

0: Disables the mark signal search.
1: Enables the mark signal search.

42 Enabling the proximity signal during homing — **Always**

Minimum value:	0	**Long:**	SIG-ｹﾞﾝﾃﾝｷﾝﾎﾞｳｭｳｺｳ
Maximum value:	1		SIG-OrgEnable
Initial value:	0	**Short:**	S-OrgEn
Unit:	None		S-OrgEn

Specify whether or not to search the origin proximity signal in signal search mode.

0: Disables the origin proximity signal search.
1: Enables the origin proximity signal search.

43 Enabling the (+) direction over travel signal during the signal search — Always

Minimum value:	0	**Long:**	SIG-+ﾎｳｺｳ OT ﾕｳｺｳ
Maximum value:	1		SIG-+OT_Enable
Initial value:	0	**Short:**	S-OTEn+
Unit:	None		S-OTEn+

Specify whether or not to search the (+) direction over-travel signal in signal search mode.

0: Disables the (+) direction over-travel signal search.

1: Enables the (+) direction over-travel signal search.

44 Enabling the (-) direction over travel signal during the signal search — Always

Minimum value:	0	**Long:**	SIG--ﾎｳｺｳ OT ﾕｳｺｳ
Maximum value:	1		SIG--OT_Enable
Initial value:	0	**Short:**	S-OTEn-
Unit:	None		S-OTEn-

Specify whether or not to search the (-) direction over-travel signal in signal search mode.

0: Disables the (-) direction over-travel signal search.

1: Enables the (-) direction over-travel signal search.

46 Signal search mark signal logic settings — Always

Minimum value:	0	**Long:**	SIG-ﾏｰｸｼﾝｺﾞｳﾛﾝﾘ
Maximum value:	1		SIG-MarkLogic
Initial value:	0	**Short:**	S-MrkLogic
Unit:	None		S-MrkLogic

Specify the mark signal logic search in the signal search mode. This parameter is invalid if the mark signal search is not performed.

0: Mark signal OFF

1: Mark signal ON

47 Logic for the proximity signal during homing — Always

Minimum value:	0	**Long:**	SIG-ｷﾝﾎﾞｳｼﾝｺﾞｳﾛﾝﾘ
Maximum value:	1		SIG-OrgLogic
Initial value:	0	**Short:**	S-OrgLogic
Unit:	None		S-OrgLogic

Specify the origin proximity signal logic in the signal search mode. This parameter is invalid if the origin proximity signal search is not performed.

0: Origin proximity signal OFF

1: Origin proximity signal ON

48 Logic for the (+) direction over-travel signal during the signal search mode — Always

Minimum value:	0	**Long:**	SIG-+ﾎｳｺｳ OT ﾛﾝﾘ
Maximum value:	1		SIG-+OT_Logic
Initial value:	0	**Short:**	S-OTLogic+
Unit:	None		S-OTLogic+

Specify the + direction over-travel signal logic in the signal search mode. This parameter is invalid if the + direction over-travel signal search is not performed.

0: + direction over-travel signal OFF

1: + direction over-travel signal ON

49 Logic for the (-) direction over-travel signal during the signal search mode — **Always**

Minimum value:	0	**Long:**	SIG--ﾎｳｺｳ OT ﾛｼﾘ
Maximum value:	1		SIG--OT_Logic
Initial value:	0	**Short:**	S-OTLogic-
Unit:	None		S-OTLogic-

Specify the - direction over-travel signal logic in the signal search mode. This parameter is invalid if the - direction over-travel signal search is not performed.

0: - direction over-travel signal OFF

1: - direction over-travel signal ON

50 Position control bandwidth — **Always**

Minimum value:	1	**Long:**	ｲﾁｾｲｷﾞｮﾀｲｲｷ
Maximum value:	32		PosControlFreq
Initial value:	1	**Short:**	ｲﾁﾀｲｲｷ
Unit:	Hz		PosFreq

Specify the position control bandwidth of the axis position control section. This parameter is set automatically by either executing auto-tuning operation or changing the "servo stiffness settings" parameter.

51 Velocity control bandwidth — **Always**

Minimum value:	5	**Long:**	ｿｸﾄﾞｾｲｷﾞｮﾀｲｲｷ
Maximum value:	200		VelControlFreq
Initial value:	20	**Short:**	ｿｸﾄﾞﾀｲｲｷ
Unit:	Hz		VelFreq

Specify the control bandwidth of the velocity control section. This parameter is set automatically by either executing auto-tuning operation or changing the "servo stiffness settings" parameter.

53 Position integral limiting value — **Always**

Minimum value:	0	**Long:**	ｲﾁｾｷﾌﾞﾝﾘﾐｯﾀ
Maximum value:	4999999		PosIntegralLimit
Initial value:	10000	**Short:**	ｾｷﾌﾞﾝﾘﾐｯﾀ
Unit:	None		PosIntLim

Specify the limiter value of the position error integrator in the axis position control section. Specify a smaller value when a wind-up condition occurs during axis operation. This parameter is set automatically by either executing auto-tuning operation or changing the "servo stiffness settings" parameter.

54 Position feed forward percentage — **Always**

Minimum value:	0	**Long:**	ｲﾁﾌｨｰﾄﾞﾌｫﾜｰﾄﾞ_%
Maximum value:	126		Position_FF_%
Initial value:	90	**Short:**	ｲﾁ FF_%
Unit:	%		Inch_FF%

Specify the position feed forward of the axis control section.

55 Velocity feed forward percentage — **Always**

Minimum value:	0	Long:	ｿｸﾄﾞﾌｨｰﾄﾞﾌｫﾜｰﾄﾞ_%
Maximum value:	126		Velocity_FF_%
Initial value:	100	Short:	ｿｸﾄﾞ FF_%
Unit:	%		Vel_FF%

Specify the velocity feed forward.

56 Acceleration feed forward gain — Always

Minimum value:	0	**Long:**	ｶｿｸﾄﾞﾌｨｰﾄﾞﾌｫﾜｰﾄﾞ_%
Maximum value:	200		Accelaration_FF_%
Initial value:	0	**Short:**	ｶｿｸﾄﾞ FF_%
Unit:	%		Acc_FF%

Specify the acceleration feed forward. This parameter calculates the internal gain based on the load inertia/load mass.

58 Positioning settling width — Always

Minimum value:	0	**Long:**	ｲﾁｾｲﾃｲﾊﾊﾞ
Maximum value:	9999999		CoinWidth
Initial value:	5	**Short:**	ｲﾁｾｲﾃｲﾊﾊﾞ
Unit:	Axis command unit		Coin_width

Specify the settling width to be used for position settling check and position settling wait in the axis position control section.

59 Position deviation filter frequency — Always

Minimum value:	1	**Long:**	ｲﾁﾍﾝｻﾌｨﾙﾀｼｭｳﾊｽｳ
Maximum value:	250		PerrFilteFreq
Initial value:	10	**Short:**	PerrFilFrq
Unit:	Hz		PerrFilFrq

Specify the position deviation filter frequency.

60 Filter use when generating position settling signal — Always

Minimum value:	0	**Long:**	COIN ｾｲｾｲｼﾞﾌｨﾙﾀｼﾖｳ
Maximum value:	1		FilterEn_atCOIN
Initial value:	0	**Short:**	FilEn_COIN
Unit:	None		FilEn_COIN

Specify whether or not to use the position deviation filter when the position settling signal is generated.

0: Does not use.

1: Uses.

65 Value causing an error detection in the (+) or CW direction — Always

Minimum value:	1	**Long:**	ｲﾁﾍﾝｻｶﾀﾞｲｹﾝｼｭﾂﾁ+
Maximum value:	999999		PosDevErrLimit+
Initial value:	999999	**Short:**	ｲﾁﾍﾝｻﾘﾐｯﾄ+
Unit:	pulse		PerrLim+

Specify the + direction detection value when an excessive position deviation error occurs.

66 Value causing an error detection in the (-) or CCW direction — Always

Minimum value:	-999999	**Long:**	ｲﾁﾍﾝｻｶﾀﾞｲｹﾝｼｭﾂﾁ-
Maximum value:	-1		PosDevErrLimit-
Initial value:	-999999	**Short:**	ｲﾁﾍﾝｻﾘﾐｯﾄ-
Unit:	pulse		PerrLim-

Specify the - direction detection value when an excessive position deviation error occurs.

68 Torque limit percentage — Always

Minimum value:	0	**Long:**	ﾄﾙｸﾘﾐｯﾄﾊﾟｰｾﾝﾃｰｼﾞ
Maximum value:	10000		TorqLimit_%
Initial value:	10000	**Short:**	ﾄﾙｸﾘﾐｯﾄ%
Unit:	1/100 %		TorqLimit%

This parameter can limit the torque or thrust.

69 Axis velocity monitoring gain (digital monitor) — Always

Minimum value:	0	**Long:**	ｿｸﾄﾞﾓﾆﾀｹﾞｲﾝ(Digital)
Maximum value:	8		VelMonitorGain(Dig)
Initial value:	0	**Short:**	VdigMon_G
Unit:	None		VdigMon_G

Specify the axis velocity monitoring gain of the velocity monitor when controlling the digital velocity.

0: 6.55V / 32768 digits at digital detection velocity
1: 6.55V / 16384 digits
2: 6.55V / 8192 digits
3: 6.55V / 4096 digits
4: 6.55V / 2048 digits
5: 6.55V / 1024 digits
6: 6.55V / 512 digits
7: 6.55V / 256 digits
8: 6.55V/ 128 digits

70 Analog monitor selection — Always

Minimum value:	0	**Long:**	ｱﾅﾛｸﾞﾓﾆﾀｾﾝﾀｸ
Maximum value:	5		AnalogMonitorSelect
Initial value:	4	**Short:**	A_Mon ｾﾝﾀｸ
Unit:	None		A_MonSel

Select the content to be output to the analog monitor.

0: Position deviation [pulse]
1: Test operation response [pulse]
2: Position command value [pulse]
3: Current position value [pulse]
4: Position command differential value (command velocity) [pps]
5: Current position differential value (current velocity) [pps]

71 Axis positioning error monitoring gain (analog monitor) — Always

Minimum value:	0	**Long:**	ｲﾁﾍﾝｻﾓﾆﾀｹﾞｲﾝ
Maximum value:	8		PerrMonitorGain
Initial value:	0	**Short:**	PerrMon_G
Unit:	None		PerrMon_G

Specify the position deviation monitoring gain of the analog monitor.

0: 6.55V / 32768 pulses
1: 6.55V / 16384 pulses
2: 6.55V / 8192 pulses
3: 6.55V / 4096 pulses
4: 6.55V / 2048 pulses
5: 6.55V / 1024 pulses
6: 6.55V / 512 pulses
7: 6.55V / 256 pulses
8: 6.55V/ 128 pulses

72 Test operation monitoring gain (analog monitor) — Always

Minimum value: 0
Maximum value: 8
Initial value: 0
Unit: None

Long: ﾃｽﾄﾄﾞｳｻﾓﾆﾀｹﾞｲﾝ
TestMonitorGain
Short: PerrMon_G
PerrMon_G

Specify the test operation response monitoring gain of the analog monitor.

0: 6.55V / 32768 pulses
1: 6.55V / 16384 pulses
2: 6.55V / 8192 pulses
3: 6.55V / 4096 pulses
4: 6.55V / 2048 pulses
5: 6.55V / 1024 pulses
6: 6.55V / 512 pulses
7: 6.55V / 256 pulses
8: 6.55V/ 128 pulses

73 Position monitoring gain (analog monitor) — Always

Minimum value: 0
Maximum value: 14
Initial value: 0
Unit: None

Long: ｲﾁﾓﾆﾀｹﾞｲﾝ
PosMonitorGain
Short: PosMon_G
PosMon_G

Specify the position monitoring (position command value and current position value) gain of the analog monitor.

0: 6.55V / 4194304 pulses
1: 6.55V / 2097152 pulses
2: 6.55V / 1048576 pulses
3: 6.55V / 524288 pulses
4: 6.55V / 262144 pulses
5: 6.55V / 131072 pulses
6: 6.55V / 65536 pulses
7: 6.55V / 32768 pulses
8: 6.55V / 16384 pulses
9: 6.55V / 8192 pulses
10: 6.55V / 4096 pulses
11: 6.55V / 2048 pulses
12: 6.55V / 1024 pulses
13: 6.55V / 512 pulses
14: 6.55V / 256 pulses
15: 6.55V/ 128 pulses

74 Position differential value monitoring gain (analog monitor) — Always

Minimum value: 0
Maximum value: 7
Initial value: 0
Unit: None

Long: ｿｸﾄﾞﾓﾆﾀｹﾞｲﾝ
VelMonitorGain
Short: VelMon_G
VelMon_G

Specify the position differential value monitoring (position command differential value and current position differential value) gain of the analog monitor.

0: 6.55V / 8192000 pulses
1: 6.55V / 4096000 pulses
2: 6.55V / 2048000 pulses
3: 6.55V / 1024000 pulses
4: 6.55V / 512000 pulses
5: 6.55V / 256000 pulses
6: 6.55V / 128000 pulses
7: 6.55V / 64000 pulses

75 Velocity monitor selection (digital monitor) — Always

Minimum value: 0
Maximum value: 1
Initial value: 0
Unit: None

Long: ｿｸﾄﾞﾓﾆﾀｾﾝﾀｸ(Digital)
VelMonSel(Digital)
Short: VelMonSelD
VelMonSelD

This parameter switches the velocity monitoring output content when controlling the digital velocity.

0: Velocity monitor
1: Velocity monitor AC

79 Move time during a cam profile selection **Always**

Minimum value:	1	**Long:**	ｶﾑｲﾄﾞｳｼﾞｲﾄﾞｳｼﾞｶﾝ
Maximum value:	30000		CamMoveTime
Initial value:	2000	**Short:**	ｶﾑｲﾄﾞｳｼﾞｶﾝ
Unit:	msec		CamMoveTim

Specify the move time during cam profile move.

80 Acceleration time during a cam profile selection **Always**

Minimum value:	1	**Long:**	ｶﾑｲﾄﾞｳｼﾞｶｿｸｼﾞｶﾝ
Maximum value:	9999		TaccCam
Initial value:	1000	**Short:**	Tacc ｶﾑ
Unit:	msec		TaccCam

Specify the acceleration time required for the 100% change of velocity override during cam profile move.

81 Deceleration time during a cam profile selection **Always**

Minimum value:	1	**Long:**	ｶﾑｲﾄﾞｳｼﾞｹﾞﾝｿｸｼﾞｶﾝ
Maximum value:	9999		TdecCam
Initial value:	1000	**Short:**	Tdec ｶﾑ
Unit:	msec		TdecCam

Specify the deceleration time required for the 100% change of velocity override during cam profile move.

82 Enabling the Program step execution **Always**

Minimum value:	0	**Long:**	ﾌﾟﾛｸﾞﾗﾑｽﾃｯﾌﾟｼﾞｯｺｳ En
Maximum value:	1		ProgramStepEnable
Initial value:	0	**Short:**	PrgStepEn
Unit:	None		PrgStepEn

Specify whether or not to use step execution during program operation.

0: Normal execution

1: Step execution

83 ABS in program/MDI operation **Always**

Minimum value:	0	**Long:**	ﾌﾟﾛｸﾞﾗﾑ･MDI_ABS ﾕｳｺｳ
Maximum value:	1		Prg ･MDI_ABS_En
Initial value:	0	**Short:**	PrgMDI_ABS
Unit:	None		PrgMDI_ABS

Specify whether or not to use absolute move during program operation or MDI operation.

0: Incremental move

1: Absolute move

84 Enabling the peak velocity during a cam profile move **Always**

Minimum value:	0	**Long:**	ｶﾑﾋﾟｰｸｿｸﾄﾞﾊﾟﾗﾒｰﾀﾕｳｺｳ
Maximum value:	1		CamPeakVelEnable
Initial value:	0	**Short:**	CamPeakEn
Unit:	None		CamPeakEn

Specify whether or not to operate by setting the peak velocity during cam profile move.

0: Operates using the move time during cam profile move operation.

1: Operates using the feeding velocity (peak velocity).

85 Enable the (+) direction soft limit error — Always

Minimum value:	0	**Long:**	+ﾎｳｺｳｿﾌﾄ OT ｴﾗｰﾕｳｺｳ
Maximum value:	1		+SOT_ErrorEnable
Initial value:	0	**Short:**	+SOT_ErrEn
Unit:	None		+SOT_ErrEn

Specify whether or not to set the + direction soft limit error when the specified value exceeds the + direction soft limit command value.

0: Sets no error.

1: Sets an error.

86 Enable the (-) direction soft limit error — Always

Minimum value:	0	**Long:**	-ﾎｳｺｳｿﾌﾄ OT ｴﾗｰﾕｳｺｳ
Maximum value:	1		-SOT_ErrorEnable
Initial value:	0	**Short:**	-SOT_ErrEn
Unit:	None		-SOT_ErrEn

Specify whether or not to set the - direction soft limit error when the specified value exceeds the - direction soft limit command value.

0: Sets no error.

1: Sets an error.

87 (+) direction soft limit settings — Always

Minimum value:	-999999999	**Long:**	+ﾎｳｺｳｿﾌﾄ OT ﾘﾐｯﾄﾁ
Maximum value:	999999999		+SOT_Limit
Initial value:	0	**Short:**	+SOT_Limit
Unit:	Axis command unit		+SOT_Limit

Specify the + direction soft limit value.

88 (-) direction soft limit settings — Always

Minimum value:	-999999999	**Long:**	-ﾎｳｺｳｿﾌﾄ OT ﾘﾐｯﾄﾁ
Maximum value:	999999999		-SOT_Limit
Initial value:	0	**Short:**	-SOT_Limit
Unit:	Axis command unit		-SOT_Limit

Specify the - direction soft limit value.

89 Brake turn OFF delay time upon Servo ON — Always

Minimum value:	0	**Long:**	ｻｰﾎﾞｵﾝｼﾞﾌﾞﾚｰｷｵﾌﾁｴﾝ
Maximum value:	2000		TimeSrvOn_toBrkOff
Initial value:	0	**Short:**	TimeBrkOff
Unit:	msec		TimeBrkOff

Specify the delay time from servo ON to brake OFF.

90 Advanced Brake turn ON before Servo OFF — Always

Minimum value:	0	**Long:**	ｻｰﾎﾞｵﾌｼﾞﾌﾞﾚｰｷｵﾝﾁｴﾝ
Maximum value:	000		TimeBrkOn_toSrvOff
Initial value:	0	**Short:**	TimeBrkOn
Unit:	msec		TimeBrkOn

Specify the advanced time for brake ON before servo OFF.

91 TBX_EMG Servo status — **Always**

Minimum value:	0	**Long:**	TBX_EMG_ｻｰﾎﾞｼﾞｮｳﾀｲ
Maximum value:	2		TbxEmgServoCondition
Initial value:	0	**Short:**	TbxEmg ｻｰﾎﾞ
Unit:	None		TbxEmgServ

Specify the servo status of the motor when EMG from TBX is executed.

0: Maintains the servo status after the axis operation stops.
1: Turns the servo OFF after the axis operation stops.
2: Stops the axis operation and immediately turns the servo OFF.

92 Start-up program enable — **Always**

Minimum value:	0	**Long:**	ｽﾀｰﾄｱｯﾌﾟﾌﾟﾛｸﾞﾗﾑﾕｳｺｳ
Maximum value:	1		StartUpProgramEnable
Initial value:	0	**Short:**	ｽﾀｰﾄ ProgEn
Unit:	None		StartPrgEn

Specify whether or not to execute the start-up program when program number 0 is registered.

0: Does not execute the start-up program.
1: Executes the start-up program.

93 IFB_EMG Servo status — **Always**

Minimum value:	0	**Long:**	IFB_EMG_ｻｰﾎﾞｼﾞｮｳﾀｲ
Maximum value:	2		IfbEmgServoCondition
Initial value:	0	**Short:**	IfbEmg ｻｰﾎﾞ
Unit:	None		IfbEmgServ

Specify the servo status of the motor when executing EMG from the interface board.

0: Maintains the servo status after the axis operation stops.
1: Turns the servo OFF after the axis operation stops.
2: Stops the axis operation and immediately turns the servo OFF.

100 M function enable in program and MDI operation — **Always**

Minimum value:	0	**Long:**	M ｷﾉｳﾕｳｺｳ(PRG ・MDI)
Maximum value:	1		M_Enable(PRG ・MDI)
Initial value:	0	**Short:**	PrgMDI_MEn
Unit:	None		PrgMDI_MEn

Specify whether or not to validate the M function during program and MDI operation.

0: Invalidates the M function.
1: Validates the M function.

101 M function enable during index operation — **Always**

Minimum value:	0	**Long:**	M ｷﾉｳﾕｳｺｳ(INDEX)
Maximum value:	1		M_Enable(Index)
Initial value:	0	**Short:**	Index_M_En
Unit:	None		Index_M_En

Specify whether or not to validate the M function during index operation.

0: Invalidates the M function.
1: Validates the M function.

102 Enabling the selection of serial communication type for the M function interface — **Always**

Minimum value:	0	**Long:**	M ｲﾝﾀﾌｪｰｽｳｼｼﾝｶﾞﾜｾﾝﾀｸ
Maximum value:	1		M_Interface_Sel_RS
Initial value:	0	**Short:**	M_Sel_RS
Unit:	None		M_Sel_RS

Specify whether or not to set the M function interface destination at the serial communication side.

0: Sets the M function interface destination at the interface board side.
1: Sets the M function interface destination at the serial communication side.

103 Optional stop enable — Always

Minimum value: 0
Maximum value: 1
Initial value: 0
Unit: None

Long: ｵﾌﾟｼｮﾅﾙｽﾄｯﾌﾟﾕｳｺｳ
OptionalStopEn
Short: OptStopEn
OptStopEn

Specify whether or not to validate the optional stop function.

0: Invalidates the optional stop function.
1: Validates the optional stop function.

104 ABS/INC setting during table and index operation — Always

Minimum value: 0
Maximum value: 2
Initial value: 0
Unit: None

Long: Table ・Index_ABS/INC
Table ・Index_ABS/INC
Short: TI_ABS/INC
TI_ABS/INC

Specify the absolute move and incremental move during table reference and index operation.

	[During table reference operation]	[During index operation]
0:	Table data dependent	Incremental move
1:	Table data dependent	Absolute move
2:	Start-up option dependent	Start-up option dependent

105 Move direction option for rotation coordinates — Always

Minimum value: 0
Maximum value: 4
Initial value: 0
Unit: None

Long: ｶｲﾃﾝｻﾞﾋｮｳｼﾞｲﾄﾞｳﾎｳｺｳ
RotaryDirection
Short: ｶｲﾃﾝﾎｳｺｳ
RotDir

Specify the move direction setting when using the rotation coordinates.

	[During G code operation]	[During table reference operation]	[During index operation]
0:	Type 0	Type 0	Type 0
1:	Type 1	Type 1	Type 1
2:	Type 2	Type 2	Type 2
3:	Type 0	Start-up option dependent	Start-up option dependent
4:	Type 0	Table data dependent	Type 0

Type0: Short-cut move (not possible for multiple rotations)
Type1: Does not cross the rotation coordinate origin (not possible for multiple rotations)
Type2: Determines the direction by processing the target position using the current rotation coordinate origin as the reference (possible for multiple rotations).

106 Settling wait enable — Always

Minimum value: 0
Maximum value: 1
Initial value: 0
Unit: None

Long: ｾｲﾃｲﾏﾁﾕｳｺｳ
CoinEnable
Short: ｾｲﾃｲﾏﾁﾕｳｺｳ
CoinEnable

Specify whether or not to execute a settling wait for the move followed by positioning when the axis move operation is completed. For the move that is not followed by positioning, the settling wait is not executed regardless of this parameter setting. The settling wait is executed in homing mode regardless of this parameter setting.

0: Does not execute settling wait.
1: Executes settling wait.

108 Operation unit selection — Always

Minimum value: 0
Maximum value: 2
Initial value: 0
Unit: None

Long: ｳﾝﾃﾝﾀﾝｲｾﾝﾀｸ
DriveCmdUnit
Short: ｳﾝﾃﾝﾀﾝｲ
DrvCmdUnit

Specify the axis operation unit.

0: Command unit
1: Index Type A
2: Index Type B

109 Index Type A divisions setting — Always

Minimum value:	-1	**Long:**	ｲﾝﾃﾞｯｸｽﾌﾞﾝｶﾂｽｳ_A
Maximum value:	100		IndexResolution_A
Initial value:	4	**Short:**	IndexResA
Unit:	None		IndexResA

Specify the number of divisions during index Type A operation.

-1: Selects file A.

0: Selects file B.

1 to 100: Sets the number of divisions.

110 Index Type B divisions setting — Always

Minimum value:	-1	**Long:**	ｲﾝﾃﾞｯｸｽﾌﾞﾝｶﾂｽｳ_B
Maximum value:	0		IndexResolution_B
Initial value:	0	**Short:**	IndexResB
Unit:	None		IndexResB

Specify the number of divisions during index Type B operation.

-1: Selects file A.

0: Selects file B.

152 First order delay compensator setting — Always

Minimum value:	0	**Long:**	ｲﾁｼﾞｵｸﾚﾎｼｮｳ
Maximum value:	3		CompFilterSel
Initial value:	0	**Short:**	ｲﾁｼﾞｵｸﾚ
Unit:	None		FilterSel

Specify the first order delay compensator.

0: No first order delay compensator

1: 20Hz/80Hz

2: 30Hz/120Hz

3: 40Hz/160Hz

153 Notch filter: Frequency 1 selection — Always

Minimum value:	50	**Long:**	ﾉｯﾁﾌｨﾙﾀｼｭｳﾊｽｳ 1
Maximum value:	1500		NotchFilterFreq1
Initial value:	1500	**Short:**	ﾉｯﾁｼｭｳﾊｽｳ 1
Unit:	Hz		NotchFreq1

Specify the frequency of notch filter channel 1 for the driver equipped with the notch filter option. This parameter is no valid for the driver without the notch filter option.

154 Notch filter: Frequency 2 selection — Always

Minimum value:	50	**Long:**	ﾉｯﾁﾌｨﾙﾀｼｭｳﾊｽｳ 2
Maximum value:	1500		NotchFilterFreq2
Initial value:	1500	**Short:**	ﾉｯﾁｼｭｳﾊｽｳ 2
Unit:	Hz		NotchFreq2

Specify the frequency of notch filter channel 2 for the driver equipped with the notch filter option. This parameter is no valid for the driver without the notch filter option.

155 Load inertia/load mass — Always

Minimum value:	0	**Long:**	ﾌｶ_J_or_M
Maximum value:	200000		Load_J_or_M
Initial value:	0	**Short:**	ﾌｶ_J_or_M
Unit:	1/1000kgm^2 or 1/1000kg		LoadJ_or_M

Specify the load inertia or load mass mounted on the motor. If an auto-tuning operation is executed, the measured value will be set automatically.

161 Area signal 0 ON — **Always**

Minimum value:	-999999999	**Long:**	ｴﾘｱｼﾝｺﾞｳ 0_ｵﾝ
Maximum value:	999999999		AreaSignal0_On
Initial value:	0	**Short:**	ｴﾘｱ Sig0_ｵﾝ
Unit:	Axis command unit		Area0_On

Specify the position where area signal 0 turns ON. For the rotation coordinate operation, the value must be 0 or greater and less than the scaling data (command unit side) value.

162 Area signal 0 OFF — **Always**

Minimum value:	-999999999	**Long:**	ｴﾘｱｼﾝｺﾞｳ 0_ｵﾌ
Maximum value:	999999999		AreaSignal0_Off
Initial value:	0	**Short:**	ｴﾘｱ Sig0_ｵﾌ
Unit:	Axis command unit		Area0_Off

Specify the position where area signal 0 turns OFF. For the rotation coordinate operation, the value must be 0 or greater and less than the scaling data (command unit side) value.

163 Area signal 1 ON — **Always**

Minimum value:	-999999999	**Long:**	ｴﾘｱｼﾝｺﾞｳ 1_ｵﾝ
Maximum value:	999999999		AreaSignal1_On
Initial value:	0	**Short:**	ｴﾘｱ Sig1_ｵﾝ
Unit:	Axis command unit		Area1_On

Specify the position where area signal 1 turns ON. For the rotation coordinate operation, the value must be 0 or greater and less than the scaling data (command unit side) value.

164 Area signal 1 OFF — **Always**

Minimum value:	-999999999	**Long:**	ｴﾘｱｼﾝｺﾞｳ 1_ｵﾌ
Maximum value:	999999999		AreaSignal1_Off
Initial value:	0	**Short:**	ｴﾘｱ Sig1_ｵﾌ
Unit:	Axis command unit		Area1_Off

Specify the position where area signal 1 turns OFF. For the rotation coordinate operation, the value must be 0 or greater and less than the scaling data (command unit side) value.

201 Selection of English/Japanese display — **While in machine setting mode**

Minimum value:	0	**Long:**	ｴｲｺﾞﾋｮｳｼﾞｾﾝﾀｸ
Maximum value:	1		EnglishDisplay
Initial value:	0	**Short:**	ｴｲｺﾞﾋｮｳｼﾞ
Unit:	None		EnglishDsp

Specify whether English display or Japanese display is used.

0: Japanese display

1: English display

202 Coordinate (+) direction setting — **While in machine setting mode**

Minimum value:	0	**Long:**	ｻﾞﾋｮｳｹｲｾｲﾎｳｺｳ
Maximum value:	1		AxisCoordinateDir
Initial value:	1	**Short:**	ｻﾞﾋｮｳﾎｳｺｳ
Unit:	None		AxCoordDir

Specify the coordinate system direction.

0:

1:

208 Axis command unit selection — While in machine setting mode

Minimum value: 0
Maximum value: 4
Initial value: 1
Unit: None

Long: ｼﾚｲﾀﾝｲｾﾝﾀｸ
AxisCmdUnit
Short: ｼﾞｸｼﾚｲﾀﾝｲ
AxisCmdUni

Specify the parameter unit related to the axis:

0: Pulse: Does not use the axis scaling data (command unit side) or axis scaling data (pulse side), but the axis motor pulse becomes the command unit.

1: Angle (DYNASERV): 360-degree angle is equivalent to the axis motor pulse specified with the axis scaling data (pulse side). The command unit is 1/1000 degrees.
Length (LINEARSERV): 1 m length is equivalent to the axis motor pulse specified with the axis scaling data (pulse side). The command unit is 1 microns.

2: Angle (DYNASERV): 360-degree angle is equivalent to the axis motor pulse specified with the axis scaling data (pulse side). The command unit is 1/100 degrees.
Length (LINEARSERV): 1 m length is equivalent to the axis motor pulse specified with the axis scaling data (pulse side). The command unit is 10 microns.

3: Scaling: The specified value set by the axis scaling data (command unit side) is equivalent to the axis motor pulse set by the axis scaling data (pulse side). The command unit is the unit of the numeric value entered as the axis scaling data (command unit side).

4: Angle (DYNASERV): 360-degree angle is equivalent to the axis motor pulse specified with the axis scaling data (pulse side). The command unit is 1/10000 degrees.
Length (LINEARSERV): 1 m length is equivalent to the axis motor pulse specified with the axis scaling data (pulse side). The command unit is 0.1 microns.

209 Scaling data (command unit side) — While in machine setting mode

Minimum value: 1000
Maximum value: 99999999
Initial value: Motor dependent
Unit: Axis command unit

Long: ｽｹｰﾘﾝｸﾞ-ﾀﾝｲｶﾞﾜ
ScaleValue(AxisUnit)
Short: ScaleUnit
ScaleUnit

Specify the data for converting the unit of the parameters related to the axis. Specify the value equivalent to the axis scaling data (pulse side). This parameter is used when scaling is selected by the "axis command unit selection." If the rotation coordinate system has been selected, this parameter is set to one rotation of the axis command unit coordinate system.

Example: Rotation: Circumference [micron]
Linear: Ball screw lead pitch [micron]

210 Scaling data (pulse side) — While in machine setting mode

Minimum value: 1000
Maximum value: 99999999
Initial value: Motor dependent
Unit: pulse

Long: ｽｹｰﾘﾝｸﾞ-ﾊﾟﾙｽ
ScaleValue(AxisPls)
Short: ScalePulse
ScalePulse

Specify the data for converting the unit of the parameters related to the axis. Specify the value equivalent to the axis scaling data (command unit side). This parameter is used when angle and scaling are selected by "axis command unit selection." If the rotation coordinate system has been selected, this parameter is set to one rotation of the axis pulse coordinate system.

212 Straight line coordinate selection — While in machine setting mode

Minimum value: 0
Maximum value: 1
Initial value: Motor dependent
Unit: None

Long: ﾁｮｸｾﾝｻﾞﾋｮｳｹｲｾﾝﾀｸ
LinearCoodinate
Short: ﾁｮｸｾﾝｻﾞﾋｮｳ
LinerCoord

Select the linear coordinate system.

0: Rotation coordinate system
1: linear coordinate system

213 Axis maximum velocity — While in machine setting mode

Minimum value:	1	**Long:**	ｻｲﾀﾞｲｿｸﾄﾞ
Maximum value:	16000000		Vmax
Initial value:	Motor dependent	**Short:**	ｻｲﾀﾞｲｿｸﾄﾞ
Unit:	Axis command unit/sec		Vmax

Specify the maximum velocity during operation. The actual maximum velocity is determined by the smaller value of this parameter or the maximum velocity [axis command unit/sec] converted from the maximum velocity [rps, mps] determined by the motor and driver. This maximum velocity value is displayed on the monitor.

215 IFB operation: Start signal processing speed selection — While in machine setting mode

Minimum value:	0	**Long:**	IFB ｳﾝﾃﾝｶｲｼｺｳｿｸｼｮﾘ
Maximum value:	1		IFB_ScanHighSpeed
Initial value:	0	**Short:**	IFB ｺｳｿｸｼｮﾘ
Unit:	None		IFB_High

Specify the start signal processing method for interface board operation.

0: Normal processing (10msec cyclic scan, auto delay read executed)

1: High speed processing (2msec cyclic scan, auto delay read not executed)

216 Servo ON status upon power up during serial communication operations — While in machine setting mode

Minimum value:	0	**Long:**	ｼﾘｱﾙｿｳｻｼﾞ PWR ｻｰﾎﾞｵﾝ
Maximum value:	1		Serial_PWR_ServoOn
Initial value:	1	**Short:**	PWRServoOn
Unit:	None		PWRServoOn

Specify whether or not to set the servo ON status when the power is turned on during serial communication.

0: Servo OFF status

1: Servo ON status

217 Jog move operation: Serial communication selection — While in machine setting mode

Minimum value:	0	**Long:**	ｼﾞｮｸﾞｵｸﾘﾂｳｼﾝｶﾞﾜｾﾝﾀｸ
Maximum value:	1		Jog_com_select_
Initial value:	0	**Short:**	Jog_byRS
Unit:	None		Jog_byRS

Select whether or not to perform jog feed operation at the serial communication side.

0: Interface board side

1: Serial communication side

218 Enables error when over-load occurs — While in machine setting mode

Minimum value:	0	**Long:**	ｵｰﾊﾞﾛｰﾄﾞｴﾗｰﾕｳｺｳ
Maximum value:	1		OverloadErrorEn
Initial value:	1	**Short:**	OVL_ErrEn
Unit:	None		OVL_ErrEn

Specify whether or not to process as an error when over-load occurs.

0: Does not process as an error.

1: Processes as an error.

219 Velocity feedback filter use — While in machine setting mode

Minimum value:	0	**Long:**	VfbFilter ｼﾖｳ
Maximum value:	1		UseVfbFilter
Initial value:	0	**Short:**	VfbFil ｼﾖｳ
Unit:	None		UseVfbFil

220 Velocity feedback filter bandwidth — **While in machine setting mode**

Minimum value:	50	**Long:**	VfbFilter ﾀｲｲｷ
Maximum value:	1000		VfbFilterFreq
Initial value:	1000	**Short:**	VfbFilFreq
Unit:	Hz		VfbFilFreq

221 Velocity command filter bandwidth — **While in machine setting mode**

Minimum value:	50	**Long:**	VcmdFilter ﾀｲｲｷ
Maximum value:	1000		VcmdFilterFreq
Initial value:	1000	**Short:**	VcmdFilFrq
Unit:	Hz		VcmdFilFrq

222 Enables error when over-speed occurs — **While in machine setting mode**

Minimum value:	0	**Long:**	ｵｰﾊﾞｽﾋﾟｰﾄﾞｴﾗｰﾕｳｺｳ
Maximum value:	1		OverSpeedErrorEn
Initial value:	1	**Short:**	OVS_ErrEn
Unit:	None		OVS_ErrEn

Specify whether or not to process as an error when over-speed occurs.

0: Does not process as an error.

1: Processes as an error.

224 Enables error when excessive position deviation occurs — **While in machine setting mode**

Minimum value:	0	**Long:**	ｲﾁﾍﾝｻｶﾀﾞｲｴﾗｰﾕｳｺｳ
Maximum value:	1		OverPerrErrorEn
Initial value:	1	**Short:**	OVPe_ErrEn
Unit:	None		OVPe_ErrEn

Specify whether or not to process as an error when an excessive position deviation occurs.

0: Does not process as an error.

1: Processes as an error.

227 Over-speed error processing type — **While in machine setting mode**

Minimum value:	0	**Long:**	ｵｰﾊﾞｽﾋﾟｰﾄﾞｴﾗｰｼｮﾘﾀｲﾌﾟ
Maximum value:	2		OverSpeedErrorType
Initial value:	1	**Short:**	OVS_ErrTyp
Unit:	None		OVS_ErrTyp

Specify the processing type when an over-speed error occurs.

0: Maintains the servo status after the axis operation stops.

1: Turns the servo OFF after the axis operation stops.

2: Stops the axis operation and immediately turns the servo OFF.

228 Over-load error processing type — **While in machine setting mode**

Minimum value:	0	**Long:**	ｵｰﾊﾞﾛｰﾄﾞｴﾗｰｼｮﾘﾀｲﾌﾟ
Maximum value:	2		OverloadErrorType
Initial value:	1	**Short:**	OVL_ErrTyp
Unit:	None		OVL_ErrTyp

Specify the processing type when an over-load error occurs.

0: Maintains the servo status after the axis operation stops.

1: Turns the servo OFF after the axis operation stops.

2: Stops the axis operation and immediately turns the servo OFF.

229 Excessive position deviation error processing type **While in machine setting mode**

Minimum value:	0	**Long:**	ｲﾁﾍﾝｻｶﾀﾞｲｴﾗｰｼｮﾘﾀｲﾌﾟ
Maximum value:	2		OverSpeedErrorType
Initial value:	1	**Short:**	OVPeErrTyp
Unit:	None		OVPeErrTyp

Specify the processing type when an excessive position deviation error occurs.

0: Maintains the servo status after the axis operation stops.

1: Turns the servo OFF after the axis operation stops.

2: Stops the axis operation and immediately turns the servo OFF.

Monitor List

STD1

Monitor No.	Monitor name	Unit
300	Currently under operation	None
301	Axis is under operation	None
302	Error status	None
303	Alarm status	None
304	Driver ready	None
305	Servo ready	None
306	Drive coordinate status	None
307	M function under operation	None
310	Display of program number under execution	None
311	Display of block number under execution in the program	None
312	Program nesting counter	None
313	Operation unit	None
314	Index Type A resolution current value	None
315	Index Type B resolution current value	None
316	Program auto-rewind enabled status	None
317	Completion of homing status display	None
318	Homing operation: The measured value	pulse
320	Pulse position command value	pulse
321	Pulse position current value	pulse
322	Pulse position deviation	pulse
323	Command unit command value	Axis command unit
324	Scaling data (command unit side)	Axis command unit
325	Scaling data (pulse side)	pulse
326	Axis under operation: Command value	Axis operation unit
328	Position settling status	None
329	Positioning status	None

Monitor No.	Monitor name	Unit
331	Area signal 0 status	None
332	Area signal 1 status	None
335	Interface ready	None
336	Operation hold status	None
337	Overload status	None
338	Zero signal status	None
339	Sensor group signal status	None
340	Zero signal fall edge side selection during homing operation	None
345	Error code (main)	None
346	Error code (sub)	None
347	Operation mode number	None
348	Multi-channel communication status	None
349	Multi-channel communication slave code	None
355	Monitor resolution	pulse/rev, pulse/m
356	Digital velocity sensitivity	digit/rps, digit/mps
357	Maximum velocity	Axis command unit/sec
358	Z-phase signal pulse interval	pulse/rev, pulse/m
360	Load ratio	x 1/100
361	Velocity ratio gain	x 1/100
362	Post-filter pulse position deviation	pulse
363	Velocity command value (digital)	1/16 digit
364	Post-filter velocity command value (digital)	1/16 digit
365	Present velocity value	1/16 digit
366	Present post-filter velocity value	1/16 digit
367	Velocity deviation (digital)	1/16 digit
368	Current command value (D/A)	digit
369	Present current value (A/D)	digit
370	Present command unit value	Axis command unit

Monitor No.	Monitor name	Unit
371	Command unit deviation	Axis command unit
372	Present velocity value DC	1/16 digit
387	Motor linear coordinate command value	pulse
388	Present motor linear coordinate value	pulse
390	Motor linear coordinate command differential value	pulse/ΔT
391	Present motor linear coordinate differential value	pulse/ΔT
392	Pre-filter current square duty	digit
393	Post-filter current square duty	digit
399	Time after power ON [msec]	msec

300 Currently under operation

Unit: None

Indicates that an operation is being performed.

301 Axis is under operation

Unit: None

Indicates that an axis operation is being performed.

302 Error status

Unit: None

Indicates the error status.

303 Alarm status

Unit: None

Indicates the alarm status.

304 Driver ready

Unit: None

Indicates that the driver is ready.

305 Servo ready

Unit: None

Indicates that the servo is ready.

306 Drive coordinate status

Unit: None

Indicates that the operation coordinate is ready to be used. If it cannot be used, an incremental move operation cannot be executed.

307 M function under operation

Unit: None

Indicates that the M function is being executed.

310 Display of program number under execution

Unit: None

Indicates the program number during execution or after execution.

311 Display of block number under execution in the program

Unit: None

Indicates the block number during execution or after execution.

312 Program nesting counter

Unit: None

Indicates the program nesting count during program execution.

313 Operation unit

Unit: None

Indicates the operation command unit.

0: Command unit
1: Index Type A
2: Index Type B

314 Index Type A resolution current value

Unit: None

Indicates the current value of the index Type A resolution.

315 Index Type B resolution current value

Unit: None

Indicates the current value of the index Type B resolution.

316 Program auto-rewind enabled status

Unit: None

Indicates the auto rewind enabled status in program operation.

317 Completion of homing status display

Unit: None

Indicates whether a homing operation has been completed after the power is turned on.

318 Homing operation: The measured value

Unit: pulse

Indicates the distance between the neighboring signal and origin that is measured during homing operation.

320 Pulse position command value

Unit: pulse

Displays the pulse position command value.

321 Pulse position current value

Unit: pulse

Displays the current pulse position value.

322 Pulse position deviation

Unit: pulse

Displays the pulse position deviation.

323 Command unit command value

Unit: Axis command unit

Displays the command unit command value.

324 Scaling data (command unit side)

Unit: Axis command unit

Displays the axis scaling data (command unit side) that is actually used.

325 Scaling data (pulse side)

Unit: pulse

Displays the axis scaling data (pulse side) that is actually used.

326 Axis under operation: Command value

Unit: Axis operation command

Displays the operation unit command value.

328 Position settling status

Unit: None

Indicates that the axis position deviation is within the specified range.

329 Positioning status

Unit: None

Indicates that the update of an axis position command value has stopped and the axis position deviation is within the specified range.

331 Area signal 0 status

Unit: None

Displays the area signal 0 status.

332 Area signal 1 status

Unit:

Displays the area signal 1 status.

335 Interface ready

Unit: None

Indicates that the PLC interface is ready.

336 Operation hold status

Unit: None

Indicates that the operation status is locked.

337 Overload status

Unit: None

Displays the overload status.

338 Zero signal status

Unit: None

Displays the zero signal status.

339 Sensor group signal status

Unit: None

Displays the sensor group signal status. Each bit in binary notation corresponds as follows:

bit0: (Reserved)
bit1: (Reserved)
bit2: (Reserved)
bit3: (Reserved)
bit4: ORG Origin neighboring signal
bit5: OTU + direction hardware over-travel signal
bit6: OTD - direction hardware over-travel signal
bit7: (Reserved)

340 Zero signal fall edge side selection during homing operation

Unit: None

Displays whether the fall edge side of the zero signal is selected as an edge during homing operation.

0: Selects the rising edge.
1: Selects the falling edge.

345 Error code (main)

Unit: None

Displays the error code (main code).

346 Error code (sub)

Unit: None

Displays the error code (subcode).

347 Operation mode number

Unit: None

Indicates the operation mode number during or after operation.

348 Multi-channel communication status

Unit: None

Indicates that the multi-channel communication status has been set.

0: Single channel communication status
1: multi-channel communication status

349 Multi-channel communication slave code

Unit: None

Indicates the slave station (self-station) code in multi-channel communication.

355 Monitor resolution

Unit: pulse/rev, pulse/m

Displays the monitor resolution.

356 Digital velocity sensitivity

Unit: digit/rps, digit/mps

Displays the digital velocity sensitivity.

357 Maximum velocity

Unit: Axis command unit/sec

This parameter is defined by the smaller value of the maximum velocity [axis command unit/sec] set by the user and the maximum velocity [axis command unit/sec] converted from the maximum velocity [rps, mps] set by the motor and driver. The position command velocity is restricted by this value. In addition, for trapezoidal (constant acceleration/deceleration) move, the inclination of acceleration/deceleration is calculated from the acceleration/deceleration time parameter, feed velocity parameter in operation mode, and this parameter value.

358 Z-phase signal pulse interval

Unit: pulse/rev, pulse/m

Displays the Z-phase signal pulse interval.

360 Load ratio

Unit: x 1/100

Displays the load inertia/self-inertia and load mass/self-mass.

361 Velocity ratio gain

Unit: x 1/100

Displays the velocity loop ratio gain.

362 Post-filter pulse position deviation

Unit: pulse

Displays the post-filter pulse position deviation.

363 Velocity command value (digital)

Unit: 1/16 digit

Displays the velocity command value when controlling the digital velocity. (1msec sample)

364 Post-filter velocity command value (digital)

Unit: 1/16 digit

Displays the post-filter velocity command value when controlling the digital velocity. (1msec sample)

365 Present velocity current value

Unit: 1/16 digit

Displays the present velocity value. (1msec sample)

366 Present post-filter velocity value

Unit: 1/16 digit

Displays the present post-filter velocity value. (1msec sample)

367 Velocity deviation (digital)

Unit: 1/16 digit

Displays the velocity deviation. (1msec sample)

368 Current command value (D/A)

Unit: digit

Displays the current command value D/A output value. The conversion rate is as follows:

1 digit = 0.0036%

27853 digits:	+100%
0 digits:	0%
-27853 digits:	-100%

369 Present current value (A/D)

Unit: digit

Displays the A/D input value of the present current command value.(1msec sample) The conversion rate is as follows:

1 digit = 0.0036%

27840digits:	+100%
0 digits:	0%
-27840 digits:	-100%

370 Present command unit value

Unit: Axis command unit

Displays the present command unit value.

371 Command unit deviation

Unit: Axis command unit

Displays the command unit deviation.

372 Present velocity value DC

Unit: 1/16 digit

Displays the present velocity value DC. (10msec sample)

387 Motor linear coordinate command value

Unit: pulse

Displays the motor linear coordinate command value.

388 Present motor linear coordinate value

Unit: pulse

Displays the present eccentricity-compensated motor linear coordinate value.

390 Motor linear coordinate command differential value

Unit: pulse/ΔT

Displays the differential value of the motor linear coordinate command value. (2msec sample)

391 Present motor linear coordinate differential value

Unit: pulse/ΔT

Displays the differential value of the present motor linear coordinate value. (2msec sample)

392 Pre-filter current square duty

Unit: digit

Displays the pre-filter current square duty (decimal point 15 bits).

393 Post-filter current square duty

Unit: digit

Displays the post-filter current square duty (decimal point 15 bits)

399 Time after power ON [msec]

Unit: msec

Indicates the time after the power has been turned on.

About Errors/Alarms

Error/Alarm Types

[KIND_POR]	Startup error	Error that cannot be set to CPU ready (unrecoverable)
[KIND_SYS]	System error	Unrecoverable error
[KIND_RGR]	Always error	
[KIND_SRV]	Servo error	
[KIND_ERR]	Error	
[KIND_ERRALM1]	Error/operation alarm	Error or operation warning depending on the operation status
[KIND_ERRALM2]	Error/operation alarm	Error or operation warning depending on the operation status
[KIND_ARM]	Operation alarm	
[KIND_ELS]	Others	

Error/Alarm Countermeasures

		Recovery	Servo	
[TYPE-POR]	Do not start up	Not possible	OFF	–
[TYPE-SRV]	Servo OFF	Possible	OFF	–
[TYPE-E1]	Stop deceleration/servo OFF	Possible	OFF	Stop deceleration
[TYPE-E2]	Stop deceleration	Possible	–	Stop deceleration
[TYPE-ELS]	Others	–	–	–

Each error code consists of a main code and a subcode.

Error/Alarm List

STD1

Error No.	Name	Type	Measures
1	Memory error	[KIND_POR] Start-up error	[TYPE_POR] Do not start up.
2	Interface board error	[KIND_POR] Start-up error	[TYPE_POR] Do not start up.
3	Battery alarm	[KIND_ELS] Others	[TYPE_ELS] Others
4	Watchdog error	[KIND_SYS] System error	[TYPE_ELS] Others
5	Kernel error	[KIND_SYS] System error	[TYPE_ELS] Others
10	Data sum error	[KIND_POR] Start-up error	[TYPE_POR] Do not start up.
11	Data error	[KIND_POR] Start-up error	[TYPE_POR] Do not start up.
15	Encoder error	[KIND_SYS] System error	[TYPE_SRV] Servo OFF
16	Coordinate error A	[KIND_SYS] System error	[TYPE_SRV] Servo OFF
17	Coordinate error B	[KIND_ERRALM2] Error/operation alarm	[TYPE_E2] Stop deceleration.
19	Slave drive error	[KIND_RGR] Always error	[TYPE_SRV] Servo OFF
20	Power module error	[KIND_RGR] Always error	[TYPE_SRV] Servo OFF
21	AC power error	[KIND_RGR] Always error	[TYPE_SRV] Servo OFF
22	Over load	[KIND_RGR] Always error	[TYPE_ELS] Others
23	Excessive position deviation	[KIND_SRV] Servo error	[TYPE_ELS] Others
24	Over speed	[KIND_RGR] Always error	[TYPE_ELS] Others
30	Servo not ready	[KIND_ERR] error	[TYPE_E2] Stop deceleration.
31	Excessive position command differential value	[KIND_ERR] error	[TYPE_E2] Stop deceleration.
42	+ direction hardware over-travel	[KIND_ERR] error	[TYPE_E2] Stop deceleration.
43	- direction hardware over-travel	[KIND_ERR] error	[TYPE_E2] Stop deceleration.
44	+ direction software over-travel	[KIND_ERR] error	[TYPE_E2] Stop deceleration.
45	- direction software over-travel	[KIND_ERR] error	[TYPE_E2] Stop deceleration.
46	Emergency stop	[KIND_ERR] error	[TYPE_E2] Stop deceleration.
49	Homing error	[KIND_ERRALM1] Error/operation alarm	[TYPE_E2] Stop deceleration.
50	Cannot execute	[KIND_ERRALM2] Error/operation alarm	[TYPE_E2] Stop deceleration.
51	Data not ready	[KIND_ERRALM2] Error/operation alarm	[TYPE_E2] Stop deceleration.
52	Timeout	[KIND_ERRALM1] Error/operation alarm	[TYPE_E2] Stop deceleration.
53	Cannot calculate	[KIND_ERRALM1] Error/operation alarm	[TYPE_E2] Stop deceleration.
56	M interface error	[KIND_ERRALM2] Error/operation alarm	[TYPE_E2] Stop deceleration.
60	Cannot interpret	[KIND_ERRALM2] Error/operation alarm	[TYPE_E2] Stop deceleration.
61	Command format error	[KIND_ERRALM2] Error/operation alarm	[TYPE_E2] Stop deceleration.
62	Data is out of range	[KIND_ERRALM2] Error/operation alarm	[TYPE_E2] Stop deceleration.
63	Operation error	[KIND_ALM] Operation alarm	[TYPE_E2] Stop deceleration.
65	Illegal parameter and monitor number	[KIND_ERRALM2] Error/operation alarm	[TYPE_E2] Stop deceleration.
66	Illegal device	[KIND_ALM] Operation alarm	[TYPE_E2] Stop deceleration.
67	Write protected	[KIND_ERRALM2] Error/operation alarm	[TYPE_E2] Stop deceleration.
80	No such command	[KIND_ALM] Operation alarm	[TYPE_E2] Stop deceleration.
81	Not registered	[KIND_ALM] Operation alarm	[TYPE_E2] Stop deceleration.
82	Out of memory	[KIND_ALM] Operation alarm	[TYPE_E2] Stop deceleration.
85	Device conflict	[KIND_ALM] Operation alarm	[TYPE_E2] Stop deceleration.

Error/Alarm Details

Error number 1 Memory error

Long: ﾒﾓﾘｴﾗｰ
MemoryError

Short: ﾒﾓﾘｴﾗｰ
MemoryErr

Error type: [KIND_POR] Start-up error
Measures: [TYPE_POR] Do not start up.
Main cause: An error is detected during memory check when the power is turned on.
[Subcode]
1: CPU built-in ROM sum error
2: RAM error
3: Flash ROM sum error
Action to take: Contact us.

Error number 2 Interface board error

Long: ｲﾝﾀﾌｪｰｽﾎﾞｰﾄﾞｴﾗｰ
InterfaceBoardError

Short: IFB ｴﾗｰ
IFB_Err

Error type: [KIND_POR] Start-up error
Measures: [TYPE_POR] Do not start up.
Main cause: Failed to initialize the interface board.
Action to take: Contact us.

Error number 3 Battery alarm

Long: ﾊﾞｯﾃﾘｱﾗｰﾑ
BatteryAlarm

Short: ﾊﾞｯﾃﾘｱﾗｰﾑ
BatteryAlm

Error type: [KIND_ELS] Others
Measures: [TYPE_ELS] Others
Main cause: Battery voltage for memory backup is low.
Action to take: Replace with a new lithium battery.

Error number 4 Watchdog error

Long: ｳｫｯﾁﾄﾞｯｸﾞｴﾗｰ
WatchdogError

Short: ｳｫｯﾁﾄﾞｯｸﾞ
Watchdog

Error type: [KIND_SYS] System error
Measures: [TYPE_ELS] Others
Main cause: A watchdog timer error occurred. The driver will be set in the reset status.
Action to take: Contact us.

Error number 5 Kernel error

Long: ｶｰﾈﾙｴﾗｰ
KernelError

Error type: [KIND_SYS] System error
Short: ｶｰﾈﾙｴﾗｰ
Measures: [TYPE_ELS] Others
KernelErr

Main cause: An error that should have not occurred in driver software has occurred.
[Subcode]
1: Axis operation handshake error
2: System program error

Action to take: Contact us.

Error number 10 Data sum error

Long: ﾃﾞｰﾀｻﾑｴﾗｰ
DataSumError

Error type: [KIND_POR] Start-up error
Short: ﾃﾞｰﾀｻﾑｴﾗｰ
Measures: [TYPE_POR] Do not start up.
DataSumErr

Main cause: Destroyed data was detected during data check when the power was turned on.
[Subcode]
1: Parameter file
2: Parts data file
3: Program file
4: Index correction file
5: I/O logic setting file
6: Index file Type B
7: Absolute accuracy compensation file

Action to take: Perform the Reset All operation, and download all backup data.

Error number 11 Data error

Long: ﾃﾞｰﾀｴﾗｰ
DataError

Error type: [KIND_POR] Start-up error
Short: ﾃﾞｰﾀｴﾗｰ
Measures: [TYPE_POR] Do not start up.
DataError

Main cause: Data that cannot be processed was detected when the power was turned on.
[Subcode]
1: Basic data and adjustment data problem
2: Servo constant problem

Action to take: Contact us.

Error number 15 Encoder error

Long: ｴﾝｺｰﾀﾞｴﾗｰ
EncoderError

Error type: [KIND_SYS] System error
Short: ｴﾝｺｰﾀﾞｴﾗｰ
Measures: [TYPE_SRV] Servo OFF
EncoderErr

Main cause: An encoder and revolver signal problem was detected.
[Subcode]
1: No SIG0 signal edge
2: No SIG1 signal edge
3: SIG0 signal cycle problem
4: SIG1 signal cycle problem

Action to take: Contact us.

Error number 16 Coordinate error A

Long: ｻﾞﾋｮｳｹｲｲｼﾞｮｳ A
CoordinateErrorA

Short: ｻﾞﾋｮｳｴﾗｰ A
CoordiErrA

Error type: [KIND_SYS] System error
Measures: [TYPE_SRV] Servo OFF
Main cause: An error occurred during coordinate processing.
[Subcode]
1: Eccentricity compensation computation problem
2: Conversion problem from the command unit to pulses
Action to take:

Error number 17 Coordinate error B

Long: ｻﾞﾋｮｳｹｲｲｼﾞｮｳ B
CoordinateErrorB

Short: ｻﾞﾋｮｳｴﾗｰ B
CoordiErrB

Error type: [KIND_ERRALM2] Error/operation alarm
Measures: [TYPE_E2] Stop deceleration.
Main cause: An error occurred during coordinate processing.
[Subcode]
1: Executed the process that was prohibited in the coordinate non-settling status.
2: Out of the command coordinate area
3: Executed a process using the setting that is prohibited.
Action to take:

Error number 19 Slave drive error

Long: ｽﾚｰﾌﾞﾄﾞﾗｲﾌﾞｴﾗｰ
SlaveDriveError

Short: ｽﾚｰﾌﾞｴﾗｰ
SlvDrvErr

Error type: [KIND_RGR] Always error
Measures: [TYPE_SRV] Servo OFF
Main cause: A tandem slave driver error occurred.
Action to take: Check the error code of the tandem slave driver, and take measures against the tandem slave driver problem. This error occurs only if the driver is the tandem master driver.

Error number 20 Power module error

Long: ﾊﾟﾜｰﾓｼﾞｭｰﾙｴﾗｰ
PowerModuleError

Short: PwrMdl ｴﾗｰ
PwrMdlErr

Error type: [KIND_RGR] Always error
Measures: [TYPE_SRV] Servo OFF
Main cause: A power module error in the driver was detected.
[subcode]
1: Over-voltage (over-voltage of main power supply)
2: Over-current (over-current detected, 1 PM fault)
Action to take:

Error number 21 AC power error

Long: ｼｭﾃﾞﾝｹﾞﾝｲｼﾞｮｳ
AC_PowerError

Error type: [KIND_RGR] Always error
Measures: [TYPE_SRV] Servo OFF
Short: ｼｭﾃﾞﾝｹﾞﾝ
AC_PwrErr

Main cause: Either the main power supply is not input or the input voltage has not reached the rating level.
Action to take: Verify the main power supply.

Error number 22 Over load

Long: ｵｰﾊﾞﾛｰﾄﾞ
Overload

Error type: [KIND_RGR] Always error
Measures: [TYPE_ELS] Others
Short: ｵｰﾊﾞﾛｰﾄﾞ
Overload

Main cause: The motor use is overloaded.
[Subcode]
1: The motor is overheated.
2: The power module heat sink is overheated.

Action to take:

Error number 23 Excessive position deviation

Long: ｲﾁﾍﾝｻｶﾀﾞｲ
OverPerr

Error type: [KIND_SRV] Servo error
Measures: [TYPE_ELS] Others
Short: ｲﾁﾍﾝｻｶﾀﾞｲ
OverPerr

Main cause: The position deviation was greater than the tolerance.
Action to take: Perform the servo adjustment again. If an error still occurs, extend the tolerance.

Error number 24 Over speed

Long: ｿｸﾄﾞｶﾀﾞｲ
OverSpeed

Error type: [KIND_RGR] Always error
Measures: [TYPE_ELS] Others
Short: ｿｸﾄﾞｶﾀﾞｲ
OverSpeed

Main cause: The velocity was higher than the maximum velocity.
Action to take:

Error number 30 Servo not ready

Long: ｻｰﾎﾞﾉｯﾄﾚﾃﾞｨ
ServoNotReady

Error type: [KIND_ERR] error
Measures: [TYPE_E2] Stop deceleration.
Short: ｻｰﾎﾞﾉｯﾄﾚﾃﾞ
SrvNotRdy

Main cause: The servo was not ready for the process that requires the servo to be ON.
Action to take: Perform an error reset operation, turn ON the servo, and execute again.

Error number 31 Excessive position command differential value

Long: ｲﾁｼﾚｲｻﾌﾞﾝﾁｶﾀﾞｲ
OverDiffPcmd

Short: ｲﾁｼﾚｲｻﾌﾞﾝ
OverDPcmd

Error type: [KIND_ERR] error
Measures: [TYPE_E2] Stop deceleration.
Main cause: The position command with the variation rate that was greater than the maximum velocity was given.
Action to take: The velocity override may be 100% or more. Verify it.

Error number 42 + direction hardware over-travel

Long: +ﾎｳｺｳﾊｰﾄﾞｵｰﾊﾞﾄﾗﾍﾞﾙ
Hard_OT_+_direction

Short: +ﾎｳｺｳ H_OT
+_Hard_OT

Error type: [KIND_ERR] error
Measures: [TYPE_E2] Stop deceleration.
Main cause: The + direction hardware over-travel signal was detected while moving to the + direction.
Action to take: Perform an error reset operation, then move to the - direction.

Error number 43 - direction hardware over-travel

Long: -ﾎｳｺｳﾊｰﾄﾞｵｰﾊﾞﾄﾗﾍﾞﾙ
Hard_OT_-_direction

Short: -ﾎｳｺｳ H_OT
-_Hard_OT

Error type: [KIND_ERR] error
Measures: [TYPE_E2] Stop deceleration.
Main cause: The - direction hardware over-travel signal was detected while moving to the - direction.
Action to take: Perform an error reset operation, then move to the + direction.

Error number 44 + direction software over-travel

Long: +ﾎｳｺｳｿﾌﾄｵｰﾊﾞﾄﾗﾍﾞﾙ
Soft_OT_+_direction

Short: +ﾎｳｺｳ S_OT
+_Soft_OT

Error type: [KIND_ERR] error
Measures: TYPE_E2] Stop deceleration.
Main cause: Attempted to move to outside of the + direction area via positioning move, but the command unit command value was outside of the + direction area.
Action to take: Perform an error reset operation. If the command unit command value is outside of the area, move to the - direction. This error occurs only for the linear coordinates.

Error number 45 - direction software over-travel

Long: -ﾎｳｺｳｿﾌﾄｵｰﾊﾞﾄﾗﾍﾞﾙ
Soft_OT_-_direction

Error type: [KIND_ERR] error
Short: -ﾎｳｺｳ S_OT

Measures: TYPE_E2] Stop deceleration.
-_Soft_OT

Main cause: Attempted to move to outside of the - direction area via positioning move, but the command unit command value was outside of the - direction area.

Action to take: Perform an error reset operation. If the command unit command value is outside of the area, move to the + direction. This error occurs only for the linear coordinates.

Error number 46 Emergency stop

Long: ﾋｼﾞｮｳﾃｲｼ
EmergencyStop

Error type: [KIND_ERR] error
Short: ﾋｼﾞｮｳﾃｲｼ
EMG_Stop

Measures: TYPE_E2] Stop deceleration.

Main cause: An emergency stop instruction was entered.

Action to take: Reset the error.

Error number 49 Homing error

Long: ｹﾞﾝﾃﾝﾌｯｷｲｼﾞｮｳ
OriginError

Error type: [KIND_ERRALM1] Error/operation alarm
Short: ｹﾞﾝﾃﾝｲｼﾞｮｳ
OriginErr

Measures: [TYPE_E2] Stop deceleration.

Main cause: An error occurred during homing operation.

[Subcode]

1: Locating fixture position problem

Action to take:

Error number 50 Cannot execute

Long: ｼﾞｯｺｳﾌｶ
CantExec

Error type: [KIND_ERRALM2] Error/operation alarm
Short: ｼﾞｯｺｳﾌｶ
CantExec

Measures: [TYPE_E2] Stop deceleration.

Main cause: Cannot execute.

[Subcode]

1: Prohibited command
2: Execution in progress
3: Error status
4: Data is being used.
5: Invalid data
6: Access timing
7: Driver mode
8: Invalid program file
9: Nesting overflow

Action to take:

Error number 51 Data not ready

Long: ﾃﾞｰﾀﾉｯﾄﾚﾃﾞｨ
DataNotReady

Short: ﾃﾞｰﾀﾉｯﾄﾚﾃﾞ
DataNotRdy

Error type: [KIND_ERRALM2] Error/operation alarm
Measures: [TYPE_E2] Stop deceleration.
Main cause: Data is not ready.
[Subcode]
2: Part data
3: Program file
4: Index A correction file
6: Index B file
Action to take:

Error number 52 Timeout

Long: ﾀｲﾑｱｳﾄ
TimeOut

Short: ﾀｲﾑｱｳﾄ
TimeOut

Error type: [KIND_ERRALM1] Error/operation alarm
Measures: [TYPE_E2] Stop deceleration.
Main cause: A timeout occurred during internal processing.
Action to take: Contact us.

Error number 53 Cannot calculate

Long: ｴﾝｻﾞﾝﾌｶ
CantCalc

Short: ｴﾝｻﾞﾝﾌｶ
CantCalc

Error type: [KIND_ERRALM1] Error/operation alarm
Measures: [TYPE_E2] Stop deceleration.
Main cause: The condition that cannot be calculated occurred during internal processing.
[Subcode]
1: Auto tuning
Action to take: Contact us.

Error number 56 M interface error

Long: M ｲﾝﾀﾌｪｰｽｲｼﾞｮｳ
M_InterfaceError

Short: M_I/F ｲｼﾞｮｳ
M_I/F

Error type: [KIND_ERRALM2] Error/operation alarm
Measures: [TYPE_E2] Stop deceleration.
Main cause: An M function interface error occurred.
Action to take:

Error number 60 Cannot interpret

Long: ｼﾚｲﾎﾝﾔｸﾌｶ
CantInterpret

Error type: [KIND_ERRALM2] Error/operation alarm
Measures: [TYPE_E2] Stop deceleration.
Main cause: The command cannot be interpreted.
Action to take:

Short: ｼﾚｲﾎﾝﾔｸﾌｶ
CantIntp

Error number 61 Command format error

Long: ｼﾚｲﾌｫｰﾏｯﾄｲｼﾞｮｳ
FormatError

Error type: [KIND_ERRALM2] Error/operation alarm
Measures: [TYPE_E2] Stop deceleration.
Main cause: A command format error occurred.
Action to take:

Short: ｼﾚｲﾌｫｰﾏｯﾄ
FormatErr

Error number 62 Data is out of range

Long: ﾃﾞｰﾀﾊﾝｲｶﾞｲ
Out_ofRange

Error type: [KIND_ERRALM2] Error/operation alarm
Measures: [TYPE_E2] Stop deceleration.
Main cause: The data is out of range.
Action to take:

Short: ﾃﾞｰﾀﾊﾝｲｶﾞｲ
OutOfRange

Error number 63 Operation error

Long: ｿｳｻｲｼﾞｮｳ
OperationError

Error type: [KIND_ALM] Operation alarm
Measures: [TYPE_E2] Stop deceleration.
Main cause: An operation error occurred.
[Subcode]
1: Prohibited operation in machine setting mode
2: Prohibited operation while holding an operation
Action to take:

Short: ｿｳｻｲｼﾞｮｳ
OperateErr

Error number 65 Illegal parameter and monitor number

Long: ﾌｾｲﾊﾞﾝｺﾞｳ
IllegalNo

Error type: [KIND_ERRALM2] Error/operation alarm
Measures: [TYPE_E2] Stop deceleration.
Main cause: The parameter and monitor of a non-existent number was specified.
Action to take:

Short: ﾌｾｲﾊﾞﾝｺﾞｳ
IllegalNo

Error number 66 Illegal device

Long: ﾌｾｲﾃﾞﾊﾞｲｽｼﾚｲ
IllegalDevice

Error type: [KIND_ALM] Operation alarm
Measures: [TYPE_E2] Stop deceleration.
Main cause: An operation was attempted using a device whose operation was prohibited.
Action to take:

Short: ﾌｾｲﾃﾞﾊﾞｲｽ
IlgDevice

Error number 67 Write protected

Long: ｶｷｺﾐﾌｶ
WriteProtected

Error type: [KIND_ERRALM2] Error/operation alarm
Measures: [TYPE_E2] Stop deceleration.
Main cause: A write access was attempted to an area that is prohibited to be written.
Action to take:

Short: ｶｷｺﾐﾌｶ
WrtProtect

Error number 80 No such command

Long: ｼﾚｲﾋｿﾝｻﾞｲ
NoSuchCommand

Error type: [KIND_ALM] Operation alarm
Measures: [TYPE_E2] Stop deceleration.
Main cause: A non-existent command was specified.
Action to take:

Short: ｼﾚｲﾋｿﾝｻﾞｲ
NoSuchCmd

Error number 81 Not registered

Long: ﾄｳﾛｸﾐｶﾝﾘｮｳ
NotRegistered

Error type: [KIND_ALM] Operation alarm
Measures: [TYPE_E2] Stop deceleration.
Main cause: Registration did not complete normally.
Action to take:

Short: ﾄｳﾛｸﾐｶﾝ
NotRegistd

Error number 82 Out of memory

Long: ﾖｳﾘｮｳﾌﾞｿｸ
Out_ofMemory

Error type: [KIND_ALM] Operation alarm
Measures: [TYPE_E2] Stop deceleration.
Main cause: Insufficient memory
Action to take:

Short: ﾖｳﾘｮｳﾌﾞｿｸ
MemoryLow

Error number **85** **Device conflict**

Long: ﾃﾞﾊﾞｲｽｷｮｳｺﾞｳ
DeviceConflict

Error type: [KIND_ALM] Operation alarm

Short: ﾃﾞﾊﾞｲｽｷｮｳ
DevConflic

Measures: [TYPE_E2] Stop deceleration.

Main cause: An operation was attempted by another device while other device was being operated.

Action to take: