PSEL

PSEL Controller

Operation Manual Fourth Edition

IAI America, Inc.

PSEL	CAUTION

Optional System-Memory Backup Battery

The PSEL controller can be used with the optional system-memory backup battery.

Caution:	When installing the system-memory backup battery, “Other parameter No. 20” must be set to “2.”

Installing the system-memory backup battery will add the following functions to the controller:

- Save SEL global data
  Data of global variables, flags and strings will be retained even after the main power is turned off.
- Save RAM position data
  Position data changed by SEL programs will be retained even after the main power is turned off.
- Save an error list
  An error list containing up to 100 most recent errors will be retained even after the main power is turned off.

If you need any or all of the above functions, you must install the optional system-memory backup battery.

PSEL	CAUTION

Operator Alarm on Low Battery Voltage

This controller is equipped with the following backup batteries for retention of data in the event of power failure:

- System-memory backup battery
  For retention of position data, global variables/flags, error list, strings, etc.

Since these batteries are not rechargeable, they will be eventually consumed. Unless the batteries are replaced in a timely manner, the voltage will drop to a level where the data can no longer be retained. If a power failure occurs in this condition, the data will be lost. (The life of each battery varies depending on the operating time.)
Once the data is lost, the controller will not operate normally the next time the power is turned on, and recovery will take time.

To prevent this problem, this controller can output a low battery voltage alarm from its I/O port.
To output this alarm signal from an I/O port, you must set the applicable I/O parameter.

> You can specify a desired output port to issue a voltage-low warning for the system-memory backup battery.
> Set “15” as the input function specification value in the I/O parameter corresponding to the output port number you want to specify.
> Setting example)
> To specify output port No. 306 to issue a voltage-low warning for the system-memory backup battery, set “15” in I/O parameter No. 52.

It is recommended that this function be utilized to prevent unnecessary problems resulting from low battery voltage (consumption of battery life).
In particular, the person in charge of system design should utilize this function to provide a design means for issuing an operator alarm using an output signal from an I/O port, while the person in charge of electrical design should provide an electrical means for achieving the same effect.
For the battery replacement procedure, refer to the applicable section in the operating manual.

It is recommended that you always backup the latest data to a PC in case of voltage drop in the system-memory battery or unexpected controller failure.

Using Rotary Actuators in Multi-rotation Mode

With rotary actuator models supporting the multi-rotation mode, multi-rotation operation or limit rotation operation can be specified using parameters.
For more information of setting the relevant parameters, refer to the following sections in 2, “Utilization Examples of Axis-specific Parameters” under “Parameter utilization” provided in the Appendix:

- About the axis operation type and rotational axis mode
- Operate a rotational axis in the multi-rotation mode or with short-cut control

However, take note of the following points:

Rotational axes using a simplified absolute unit do not support multi-rotation operation. This is because with these axes, you cannot set the infinite stroke mode or select the short-cut control in the index mode.

Applicable models

Actuator	RCP2-RTBL-I-28P-20-360-*
	RCP2-RTBL-I-28P-30-360-*
	RCP2-RTCL-I-28P-20-360-*
	RCP2-RTCL-I-28P-30-360-*

Safety Precautions

Please read the information in “Safety Precautions” carefully before selecting a model and using the product.
The precautions described below are designed to help you use the product safely and avoid bodily injury and/or property damage.

Directions are classified as “danger,” “warning,” “caution” and “note,” according to the degree of risk.

Danger	Failure to observe the instruction will result in an imminent danger leading to death or serious injury.
Warning	Failure to observe the instruction may result in death or serious injury.
Caution	Failure to observe the instruction may result in injury or property damage.
Note	The user should take heed of this information to ensure the proper use of the product, although failure to do so will not result in injury.

This product has been designed and manufactured as a component for use in general industrial machinery.

Devices must be selected and handled by a system designer, personnel in charge of the actual operation using the product or similar individual with sufficient knowledge and experience, who has read both the catalog and operation manual (particularly the “Safety Precautions” section). Mishandling of the product poses a risk.
Please read the operation manuals for all devices, including the main unit and controller.
It is the user’s responsibility to verify and determine the compatibility of this product with the user’s system, and to use them properly.
After reading the catalog, operation manual and other materials, be sure to keep them in a convenient place easily accessible to the personnel using this product.
When transferring or loaning this product to a third party, be sure to attach the catalog, operation manual and other materials in a conspicuous location on the product, so that the new owner or user can understand its safe and proper use.
The danger, warning and caution directions in this “Safety Precautions” do not cover every possible case. Please read the catalog and operation manual for the given device, particularly for descriptions unique to it, to ensure its safe and proper handling.

Danger

[General]

- Do not use this product for the following applications:
  1. Medical equipment used to maintain, control or otherwise affect human life or physical health
  2. Mechanisms and machinery designed for the purpose of moving or transporting people
  3. Important safety parts of machinery

  This product has not been planned or designed for applications requiring high levels of safety. Use of this product in such applications may jeopardize the safety of human life. The warranty covers only the product as it is delivered.

[Installation]

- Do not use this product in a place exposed to ignitable, inflammable or explosive substances. The product may ignite, burn or explode.
- Avoid using the product in a place where the main unit or controller may come in contact with water or oil droplets.
- Never cut and/or reconnect the cables supplied with the product for the purpose of extending or shortening the cable length. Doing so may result in fire.

[Operation]

- Do not pour water onto the product. Spraying water over the product, washing it with water or using it in water may cause the product to malfunction, resulting in injury, electric shock, fire, etc.

[Maintenance, Inspection, Repair]

- Never modify the product. Unauthorized modification may cause the product to malfunction, resulting in injury, electric shock, fire, etc.
- Do not disassemble and reassemble the components relating to the basic structure of the product or its performance and function. Doing so may result in injury, electric shock, fire, etc.

⚠ Warning

[General]

- Do not use the product outside the specifications. Using the product outside the specifications may cause it to fail, stop functioning or sustain damage. It may also significantly reduce the service life of the product.
  In particular, observe the maximum acceleration/deceleration and speed.

[Installation]

- If the machine will stop in the case of system problem such as emergency stop or power failure, design a safety circuit or other device that will prevent equipment damage or injury.
  Also provide safety measures so that the product will not be started simply by canceling the emergency stop or upon recovery of power.
- Be sure to provide Class D grounding for the controller and actuator (formerly Class 3 grounding: Grounding resistance at 100 Ω or less). Leakage current may cause electric shock or malfunction.
- Before supplying power to and operating the product, always check the operation area of the equipment to ensure safety. Supplying power to the product carelessly may cause electric shock or injury due to contact with the moving parts.
- Wire the product correctly by referring to the operation manual. Securely connect the cables and connectors so that they will not be disconnected or come loose. Failure to do so may cause the product to malfunction or cause fire.

[Operation]

- Do not touch the terminal block or various switches while the power is supplied to the product. Failure to observe this instruction may result in electric shock or malfunction.
- Before operating the moving parts of the product by hand (for the purpose of manual positioning, etc.), confirm that the servo is turned off (using the teaching pendant or PC software). Failure to observe this instruction may result in injury.
- The cables supplied with the product are flexible, but they are not robot cables. Do not store the cables in a movable cable duct (cable bearer, etc.) that bends more than the specified bending radius.
- Do not scratch the cables. Scratching, forcibly bending, pulling, winding, crushing with heavy object or pinching a cable may cause it to leak current or lose continuity, resulting in fire, electric shock, malfunction, etc.

- Turn off the power to the product in the event of power failure. Failure to do so may cause the product to suddenly start moving when the power is restored, thus resulting in injury or product damage.
- If the product is generating heat, smoke or a strange smell, turn off the power immediately. Continuing to use the product may result in product damage or fire.
- If any of the internal protective devices (alarms) of the product has actuated, turn off the power immediately. Continuing to use the product may result in product damage or injury due to malfunction. Once the power supply is cut off, investigate and remove the cause and then turn on the power again.

[Maintenance, Inspection, Repair]

- Before conducting maintenance/inspection, parts replacement or other operations on the product, completely shut down the power supply. At this time, take the following measures:
  1. Display a sign that reads, “WORK IN PROGRESS. DO NOT TURN ON POWER” at a conspicuous place, in order to prevent a person other than the operator from accidentally turning on the power while the operation is working.
  2. When two or more operators are to perform maintenance/inspection together, always call out every time the power is turned on/off or an axis is moved in order to ensure safety.

[Disposal]

- Do not throw the product into fire. The product may burst or generate toxic gases.

⚠ Caution

[Installation]

- Do not use the product under direct sunlight (UV ray), in a place exposed to dust, salt or iron powder, in a humid place, or in an atmosphere of organic solvent, phosphate-ester machine oil, sulfur dioxide gas, chlorine gas, acids, etc. The product may lose its function over a short period of time, or exhibit a sudden drop in performance or its service life may be significantly reduced.
- Do not use the product in an atmosphere of corrosive gases (sulfuric acid or hydrochloric acid), inflammable gases or ignitable liquids. Rust may form and reduce the structural strength or the motor may ignite or explode.
- When using the product in any of the places specified below, provide a sufficient shield. Failure to do so may result in malfunction:
  1. Place where large current or high magnetic field is present
  2. Place where welding or other operations are performed that cause arc discharge
  3. Place subject to electrostatic noise
  4. Place with potential exposure to radiation
- Install the main unit and controller in a place subject to as little dust as possible. Installing them in a dusty place may result in malfunction.
- Do not install the product in a place subject to large vibration or impact (4.9 m/s^2 or more). Doing so may result in the malfunctioning of the product.
- Provide an emergency-stop device in a readily accessible position so the device can be actuated immediately upon occurrence of a dangerous situation during operation. Lack of such device in an appropriate position may result in injury.
- Provide sufficient maintenance space when installing the product. Routine inspection and maintenance cannot be performed without sufficient space, which will eventually cause the equipment to stop or the product to sustain damage.
- Do not hold the moving parts of the product or its cables during installation. It may result in injury.
- Always use IAI’s genuine cables for connection between the controller and the actuator. Also use IAI’s genuine products for the key component units such as the actuator, controller and teaching pendant.
- Before installing or adjusting the product or performing other operations on the product, display a sign that reads, “WORK IN PROGRESS. DO NOT TURN ON POWER.” If the power is turned on inadvertently, injury may result due to electric shock or sudden activation of an actuator.

[Operation]

- Provide safety measures so that the product will not be started simply by turning on the power. Failure to do so may cause the product to start suddenly, resulting in injury or product damage.
- Do not insert a finger or object in the openings in the product. It may cause fire, electric shock or injury.
- Do not bring a floppy disk or other magnetic media within one meter of the product. The magnetic field generated by the magnet may destroy the data in the floppy disk, etc.

[Maintenance, Inspection, Repair]

- When the power was turned off and the cover was opened to replace the battery, etc., do not touch the condenser terminal in the product immediately after the power was turned off (within 30 seconds). Residual voltage may cause electric shock.
- Do not touch the terminals when performing an insulation resistance test. Electric shock may result. (Do not perform any withstand voltage test, since the product uses DC voltage.)

Note

[General]

- If you are planning to use the product under a condition or environment not specified in the catalogs and operation manual, or in an application requiring strict safety such as aircraft facility, combustion system, entertainment machine, safety device or other equipment having significant impact on human life or property, design operating ranges with sufficient margins from the ratings and design specifications or provide sufficient safety measures such as fail-safes. Whatever you do, always consult IAI's sales representative.

[Installation]

- Do not place objects around the controller that will block airflows. Insufficient ventilation may damage the controller.
- Do not configure a control circuit that will cause the load to drop in case of power failure. Configure a control circuit that will prevent the table or load from dropping when the power to the machine is cut off or an emergency stop is actuated.

[Installation, Operation, Maintenance]

- When installing the actuator or carrying out similar tasks, wear protective gloves, protective goggles, safety shoes, etc., to ensure safety.

[Disposal]

- When the product becomes no longer usable or necessary, dispose of it properly as an industrial waste.

Others

- IAI shall not be liable whatsoever for any loss or damage arising from a failure to observe the items specified in "Safety Precautions."
- This product cannot be used in any way except in accordance with the methods explained in this operation manual. IAI shall assume no responsibility for any outcome resulting from using the product in a manner not conforming to this manual.

CE Mark

1. European Union EC Directives

The European Union EC Directives are designed to protect the users and consumers of products sold in the EU (European Union) block against health and safety risks associated with use of such products, while ensuring free distribution of products within the EU block in compliance with the New Approach Directives issued by the European Commission. Accordingly, companies exporting to Europe or having production sites in Europe must meet the CE Marking requirements.

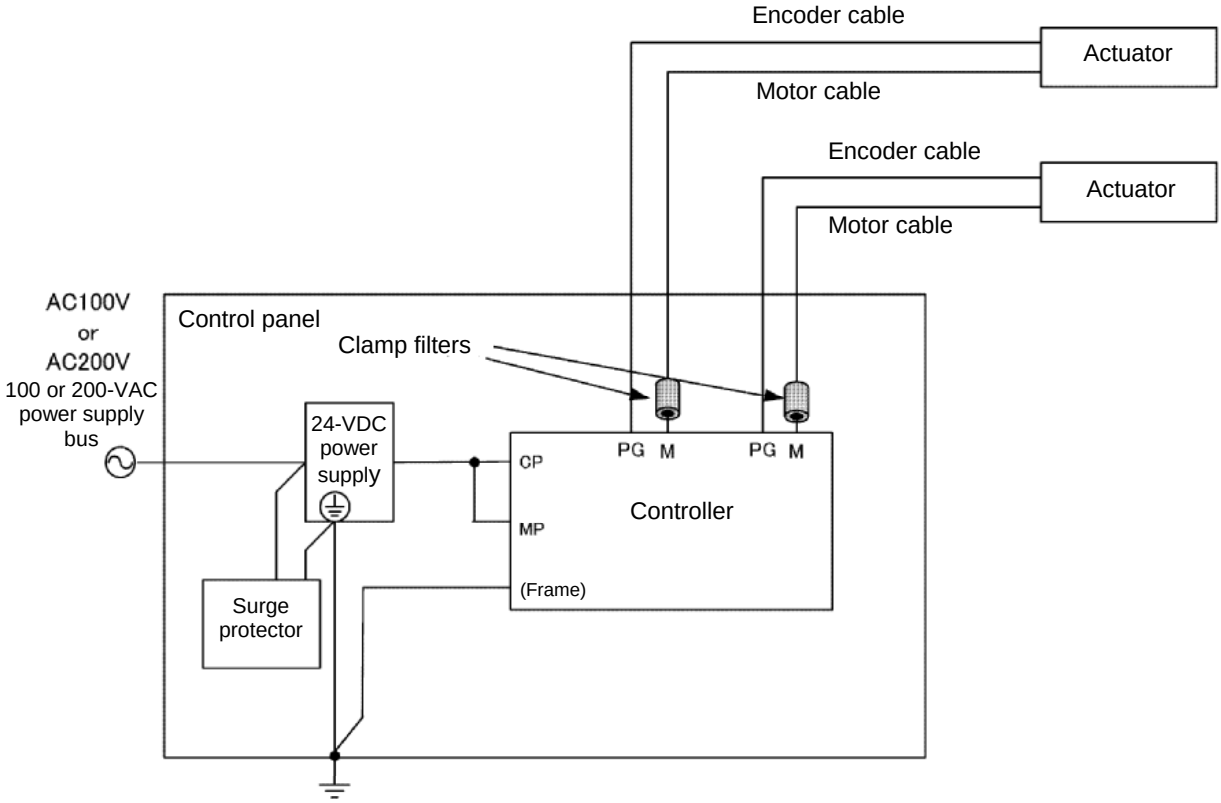
With the PSEL controller, models (conditions) of controller/actuator/peripheral connection & installation are determined and these models are used to confirm compliance with the standards relating to the applicable EMC Directives.

2. Conforming Standards

<EMC Directives>
(EMI)
EN 61000-6-4/EN 55011 Group 1, Class A
(EMS)
EN 61000-6-2 (Immunity in Industrial Environment)
EN61000-4-2 (Electrostatic Discharge Immunity)
EN61000-4-3 (Radiated Radio-frequency, Electromagnetic Field Immunity)
EN61000-4-4 (Electrical Fast Transient/Burst Immunity)
EN61000-4-5 (Surge Immunity)
EN61000-4-6 (Immunity to Conducted Disturbances Induced by Radio-frequency Electromagnetic Fields)
EN61000-4-8 (Power-frequency Magnetic Field Immunity)

CE Mark

3. Configuration of Peripherals



(1) Environment

Item	Standard
Overvoltage category	I
Pollution degree	II
IP code	IP20
Altitude	2000 m or less

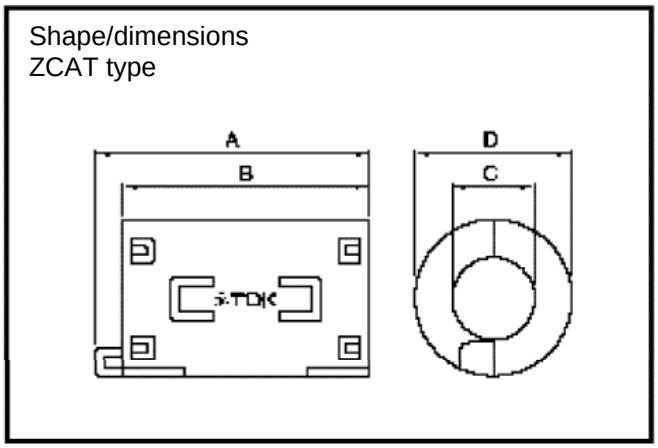
- Use the PSEL controller in an environment of pollution degree 2 or 1 as specified in IEC 60664-1.
  Example) Install the controller in a control panel having a structure capable of shutting out ingress of water, oil, carbon, dust, etc. (IP54)

(2) Power supply
For the controller and I/O power supplies, use CE Mark-compliant 24-VDC power supplies whose input and output circuits are protected by reinforced insulation (SELV).

CE Mark

(3) Clamp filter

Install the following clamp filter on the motor cable:
Manufacturer: TDK Corporation
Model: ZCAT3035-1330



Shape/dimensions (mm)

A：39±1
B：34±1
φC：13±1
φD：30±1

Fig. 1 External View of Clamp Filter

(4) Surge protector

Install the following surge protector on the primary side of the 24-VDC power supply:
Manufacturer: Okaya Electric Industries Co., Ltd.
Model: R·A·V-481BWZ-2A

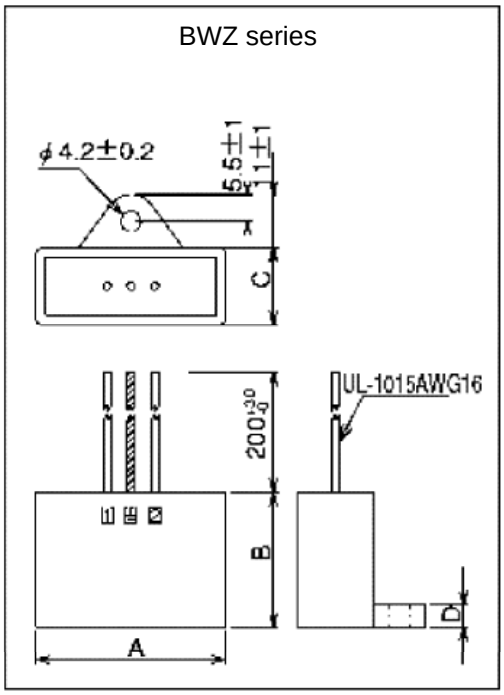


Fig.2 External View of Surge Protector

CE Mark

(5) Cables

Take note of the following limitations on cables:

A) All cables connected to the PSEL controller, including various network cables, must be less than 30 m long.

B) For the controller power (24-VDC) cable, use a 2-core (1-pair) twisted pair cable with a wire size of AWG16 to 18 (1.25 mm^2 to 0.75 mm^2).

Table of Contents

Part 1 Installation

Chapter 1 Overview

1. Introduction

Thank you for purchasing the PSEL Controller.

Please read this manual carefully, and handle the product with due care and operate it correctly.
Keep this manual in a safe place and reference relevant items when needed.

When actually starting up your system or if you have encountered a problem, you should also refer to the manuals for the teaching pendant, PC software and other components used with the system, in addition to this manual.

This manual does not cover all possible operations other than normal operations, or unexpected events such as complex signal changes resulting from use of critical timings. Accordingly, you should consider items not specifically explained in this manual as “prohibited.”

* Utmost effort has been made to ensure accuracy and completeness of the information contained in this manual. However, should you find any error in the manual or if you have any comment regarding its content, please contact IAI.
Keep this manual in a convenient place so that you can quickly reference it whenever necessary.

2. Type

Refer to the following table for details on type specification.

Example of type specification

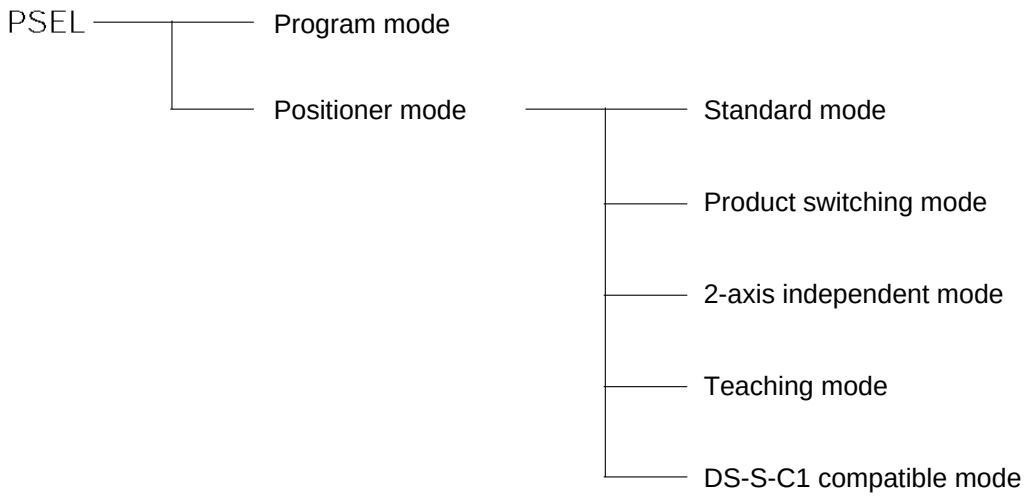
PSEL	C	2	20PI	20PIB	NP	2	0
❶	❷	❸	❹	❹	❺	❻	❼

Type specification table

❶ Series	❷ Controller type	❸ Number of axes	❹ Details of axis 1 to axis 2			❺ Standard I/O	❻ I/O flat cable length	❼ Power-source voltage
			Motor output (W)	Encoder type	Brake			
PSEL	C (Standard specification)	1 (Axis 1) 2 (Axis 2)	20P (22, square) 28P (28, square) 42P (42, square) 56P (56, square)	I (Incremental)	Blank (Without brake) B (With brake)	NP Standard PIO 24 inputs/8 outputs NPN specification PN Standard PIO 24 inputs/8 outputs PNP specification	2 : 2m (Standard) 3 : 3m 5 : 5m 0 : None	0: 24 VDC

3. PSEL Controller Functions

The functions provided by the PSEL controller are structured in the following manner.



The PSEL controller has the “program mode” in which SEL programs are input to operate the actuator(s), and the “positioner mode” in which position numbers are specified from the host PLC to operate the actuator(s).
The positioner mode provides five sub-modes to meet the needs of various applications.
The program mode has been selected at the factory prior to the shipment of the controller (Other parameter No. 25 = 0).

Caution: Two modes cannot be selected at the same time.

This controller can be configured with one axis and two axes. Just like other conventional SEL controllers, this controller can be combined with various actuators. When connecting an actuator, be sure to use a dedicated cable.

- Turn on the I/O power before or simultaneously with the main power (control power + motor power).
- Take the control power and motor power from the same power supply and turn on both powers simultaneously.
- Before performing a check or inserting/removing a connector, turn off the power and wait for at least 10 minutes. Even after the power is turned off, the internal circuits will continue to carry high voltages for a short period.
- About actuator duty
  IAI recommends that our actuators be used at a duty of 50% or less as a guideline in view of the relationship of service life and accuracy:

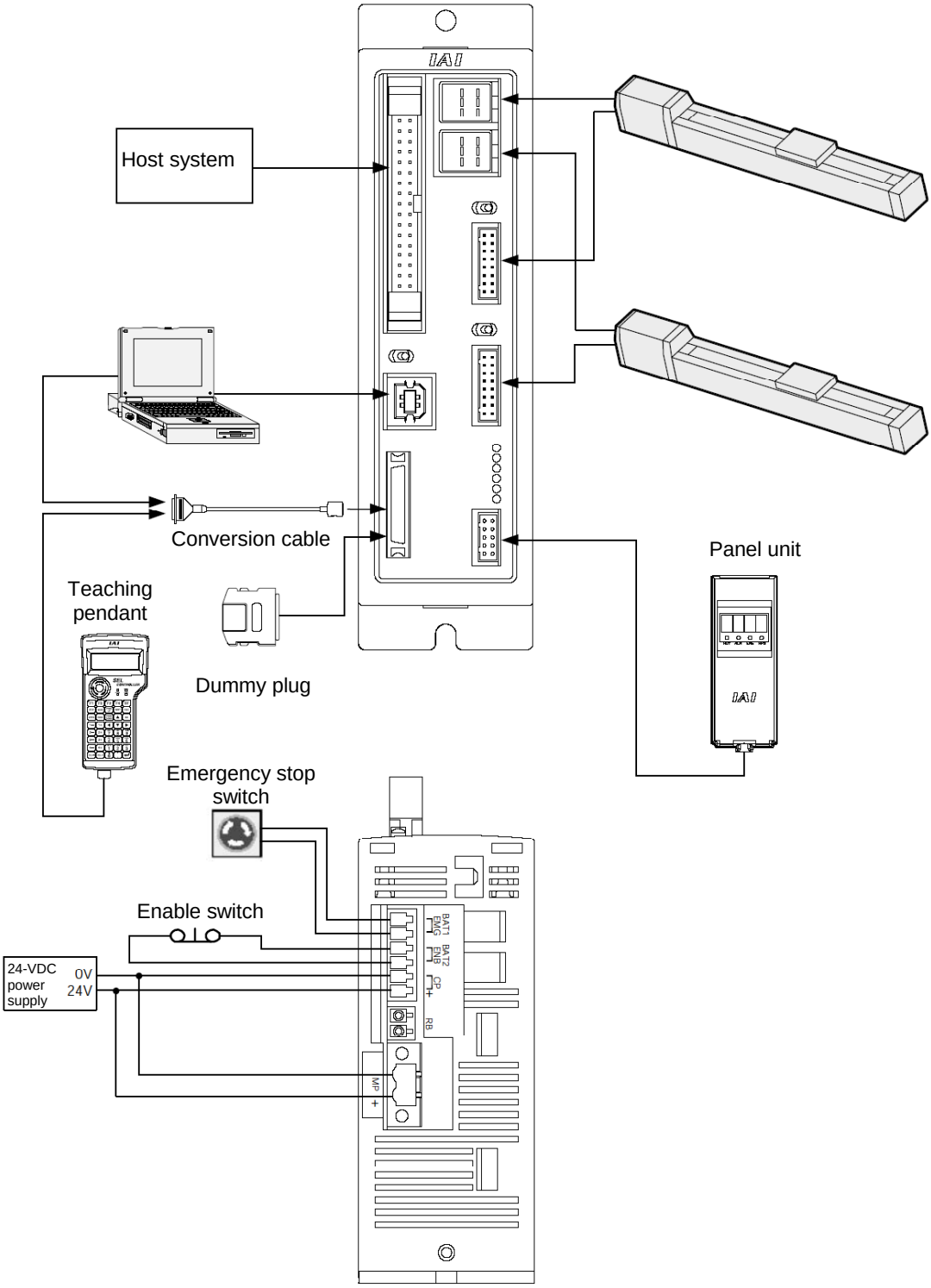
$$\text{Duty (\%)} = \frac{\text{Acceleration / Deceleration Time}}{\text{Motion time} + \text{Inactivity}} \text{ X } 100$$

- After turning off the control power, be sure to wait for at least 5 seconds before turning it back on.
- Do not insert or remove connectors while the controller power is on. Doing so may cause malfunction.

Read the operation manual for each actuator. If you have purchased our optional PC software and/or teaching pendant, read the respective operation manuals, as well.

* Utmost effort has been made to ensure that the information contained in this manual is true and correct. However, should you find any error or if you have any comment regarding the content, please contact IAI.

4. System Setup



5. Warranty Period and Scope of Warranty

The PSEL Controller you have purchased passed our strict outgoing inspection. This unit is covered by the following warranty:

1. Warranty Period

The warranty period shall be either of the following periods, whichever ends first:

- 18 months after shipment from our factory
- 12 months after delivery to a specified location

2. Scope of Warranty

Should the product fail during the above period under a proper use condition due to a fault on the part of the manufacturer, IAI will repair the defect free of charge. However, the following cases are excluded from the scope of warranty:

- Discoloration of paint or other normal aging
- Wear of consumable parts due to use
- Subjective imperfection, such as noise not affecting mechanical function
- Defect caused by inappropriate handling or use by the user
- Defect caused by inappropriate or erroneous maintenance/inspection
- Defect caused by use of a part other than IAI's genuine part
- Defect caused by unauthorized modification, etc., not approved by IAI or its agent
- Defect due to an act of God, accident, fire, etc.

The warranty covers only the product as it is delivered. IAI shall not be liable for any loss arising in connection with the delivered product. The user must bring the defective product to our factory to receive a warranty repair.

3. Scope of Service

The price of the delivered product does not include costs incurred in association with program generation, dispatch of technician, etc. Therefore, a separate fee will be chargeable in the following cases even during the warranty period:

- Guidance on installation/adjustment and witnessing of test operation
- Maintenance/inspection
- Technical guidance and training on operation, wiring method, etc.
- Technical guidance and training regarding programs, such as program generation
- Other services and operations where IAI finds a need to charge a separate fee

Chapter 2 Specifications

1. Controller Specifications

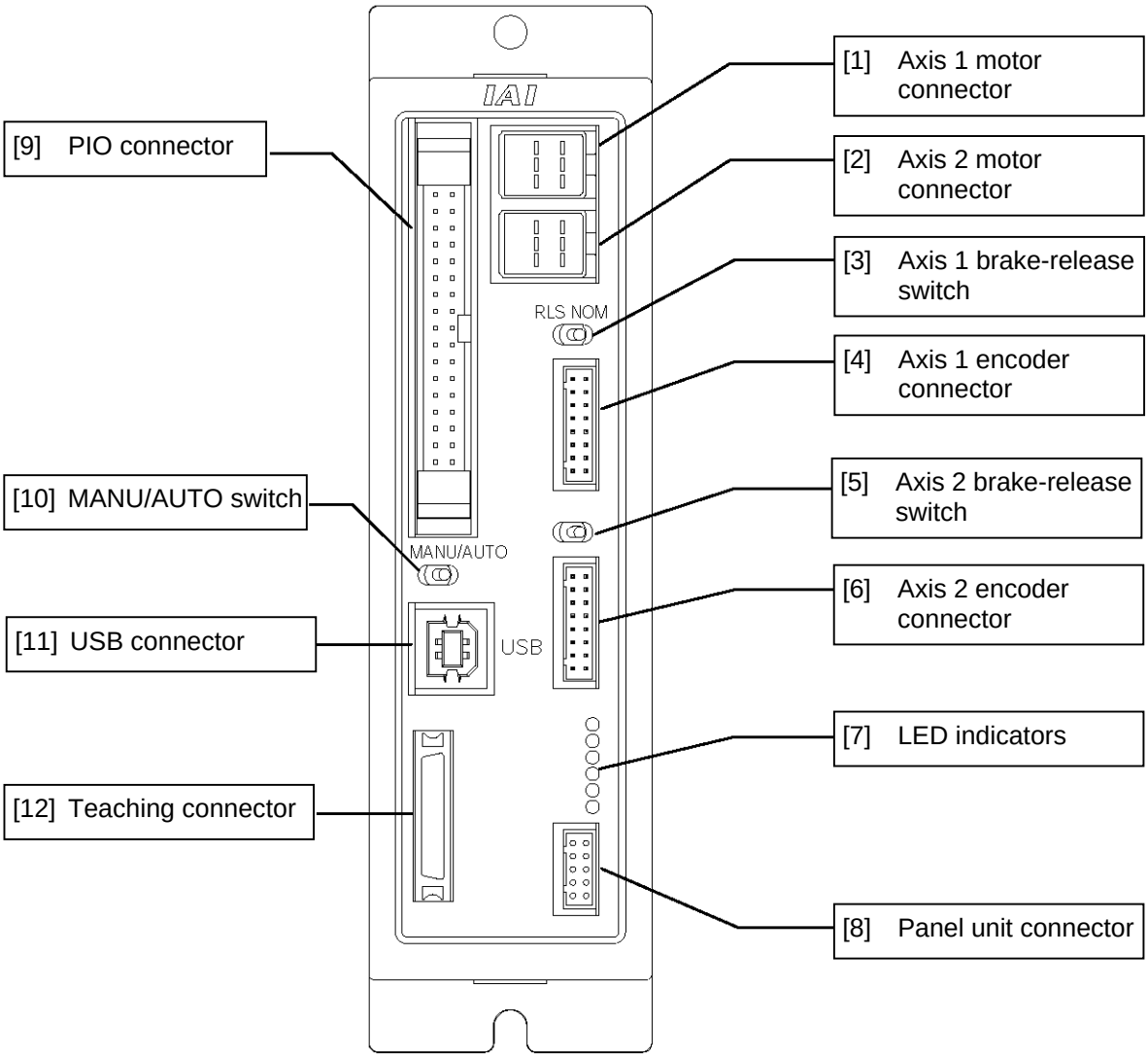
Base specifications of this product

Total output when maximum number of axes are connected	30 W x 2 axes
Control power input	24 VDC ± 10%
Motor power input	24 VDC ± 10%
Resistance against momentary power failure	Maximum 0.5 msec
Withstand voltage	1500 VAC for 1 minute (Measured between all power-supply terminals and FG)
Insulation resistance	500 VDC, 10 MΩ or more
Drive-source cutoff method	Internal relay
Emergency stop input	Contact B input (Internal power-supply type)
Emergency stop action	Deceleration stop + Regenerative brake by timer
Enable input	Contact B input (Internal power-supply type)
Position detection method	Incremental encoder of A/B two-phase output type
Battery	System-memory backup battery (Optional) Lithium battery: AB-5 by IAI, 3.6 V/2000 mAh
Programming language	Super SEL language
Number of program steps	2000 steps (total)
Number of positions	1500 positions (total)
Number of programs	64 programs
Multi-tasking capability	8 programs
Storage device	Flash ROM
Data input method	Teaching pendant or PC software
PIO power input	24 VDC ± 10%
Safety category	Category B (Built-in relay)
PIO inputs	24 points, NPN or PNP (Selectable as factory setting)
PIO outputs	8 points, NPN or PNP (Selectable as factory setting)
Air cooling method	Natural convection method
Weight	440 g
External dimensions	43 (W) x 159 (H) x 110 (D); mounting pitch 151 mm
Accessories	I/O flat cable Motor power connector Control power & system I/O connector

2. Name and Function of Each Part

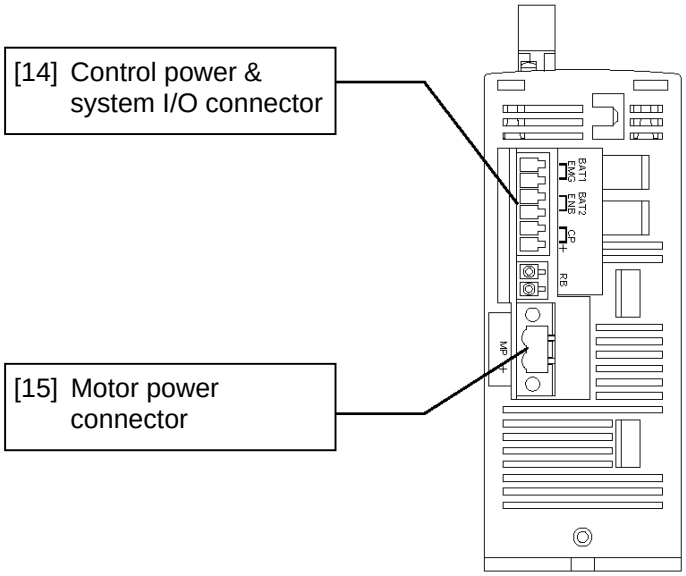
2.1 Name of Each Part

2.1.1 Front View

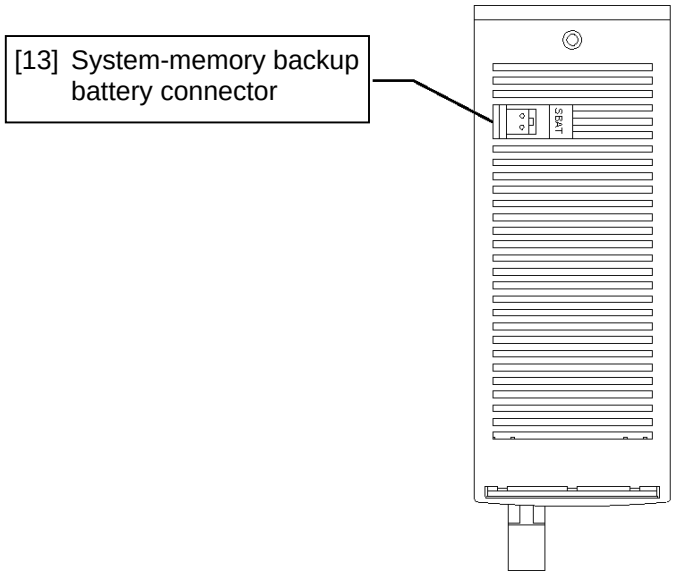


*1 For the 1-axis specification, [2], [5] and [6] are not installed and the front panel is masked.

2.1.2 Down View



2.1.3 Top View



[1] Axis 1 motor connector (M1): This connector is used to connect the motor cable for axis 1.

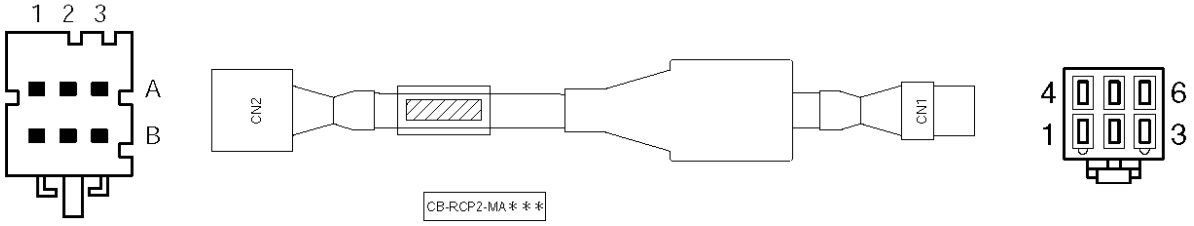
Motor Connector Specifications

Item	Specification	Remarks
Applicable connector	AMP Dynamic D2100, 6 pins	0-1376136-1 (AMP)
	Cable-end connector	0-1318119-3 (AMP) Contact: 1318107-1 (AMP)
Connector name	M1	
Maximum connection distance	20 m	
Connected unit	Actuator (motor)	
Connected cable	Motor cable	AWG22 X 6C

Motor cable

Controller end
CN2 pin assignments

Actuator end
CN1 pin assignments



CN2

Cable color	Signal symbol	Pin No.
Blue	$\overline{A}$	A1
Black	VMM	A2
White	$\overline{B}$	A3
Red	A	B1
Black	VMM	B2
Green (yellow 3)	B	B3

Housing : 1-1318119-3 (AMP)
Receptacle contact : 1318107-1

CN1

Pin No.	Signal symbol	Cable color
1	A	Red
2	VMM	Black
3	$\overline{A}$	Blue
4	B	Green (yellow 3)
5	VMM	Black
6	$\overline{B}$	White

Housing : SLP-06V (JST)
Socket contact : BSF-21T-P1.4

[2] Axis 2 motor connector (M2): This connector is used to connect the motor drive-source cable for axis 2. The specifications are the same as those of the axis 1 motor connector.

[3] Axis 1 brake-release switch (BK1): This switch is used to forcibly release the electromagnetic brake of the actuator constituting axis 1.

RLS (left) NOM (right)

Name	Description
RLS	Supply the power to the brake and forcibly release the brake.
NOM	Turn the brake ON/OFF using an internal sequence. Normally this switch is set to the “NOM” side.

[4] Axis 1 encoder/sensor connector (PG1): This connector is used to connect the encoder cable for axis 1. It connects the encoder cable of the actuator constituting axis 1.

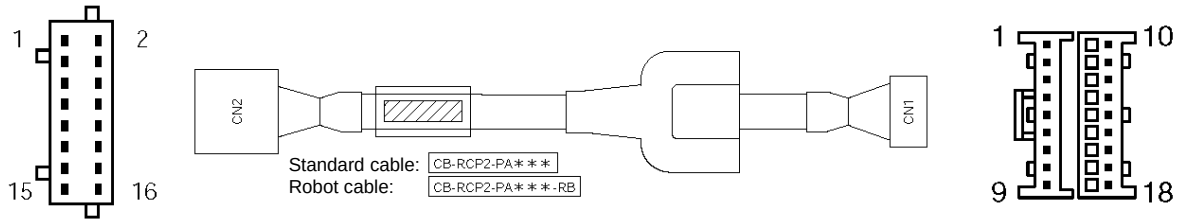
Encoder Connector Specifications

Item	Specification	Remarks
Applicable connector	2-mm pitch, double-row connector, 16 pins	S16B-PHDSS (JST)
	Cable-end connector	PHDR-16VS (JST) Contact: SPHD-001T-P0.5 (JST)
Connector name	PG1	
Maximum connection distance	20 m	
Connected unit	Actuator encoder	
Connected cable	Motor cable	AWG22 X 6P Shielded

Encoder cable

Controller end
CN2 pin assignments

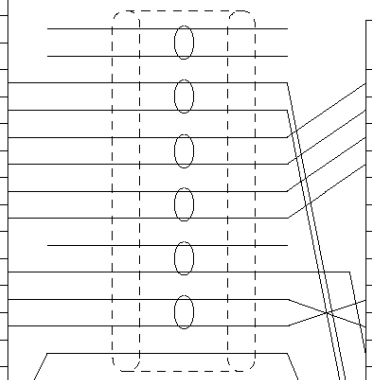
Actuator end
CN1 pin assignments



CN2

Cable color		Signal symbol	Pin No.
Robot cable	Standard cable		
–	–	(Reserved)	16
–	–	(Reserved)	15
Purple	Red	BK+	14
White (with purple)	Gray	BK–	13
Blue	Brown	ENA	12
White (with blue)	Green	$\overline{\text{ENA}}$	11
Yellow	Purple	ENB	10
White (with yellow)	Pink	$\overline{\text{ENB}}$	9
–	–	(Reserved)	8
Green	Yellow	VPS	7
Red	Orange	5V	6
White (with red)	Blue	GND	5
–	–	–	4
–	–	–	3
–	–	–	2
Drain	Drain	F.G	1

Contact : PHDR-16VS (JST)
Retainer : SPHD-001T-P0.5



CN1

Pin No.	Signal symbol	Cable color	
		Standard cable	Robot cable
1	ENA	Brown	Blue
2	$\overline{\text{ENA}}$	Green	White (with blue)
3	ENB	Purple	Yellow
4	$\overline{\text{ENB}}$	Pink	White (with yellow)
5	–	–	–
6	–	–	–
7	–	–	–
8	–	–	–
9	GND	Blue	White (with red)
10	5V	Orange	Red
11	VPS	Yellow	Green
12	–	–	–
13	–	–	–
14	–	–	–
15	–	–	–
16	BK+	Red	Yellow
17	BK–	Gray	White (with purple)
18	F.G	Drain	Drain

Housing : XMP-18V (JST)
Contact : BXA-001T-P0.6
Retainer : XMS-09V

[5] Axis 2 brake-release switch (BK2): This switch is used to forcibly release the electromagnetic brake of the actuator constituting axis 2. The specifications are the same as those of the axis 1 brake-release switch in [3].

[6] Axis 2 encoder/sensor connector (PG2): This connector is used to connect to the encoder cable for axis 2. The specifications are the same as those of the axis 1 encoder/sensor connector in [4].

[7] LED indicators: These indicators indicate the controller status.

PWR ○
RDY ○
ALM ○
EMG ○
SV1 ○
SV2 ○

Name	Color	Status when the LED is lit
PWR	Green	The controller has been started successfully and is receiving power.
RDY	Green	The controller is ready.
ALM	Orange	An alarm is present (an error of message level or higher has generated.)
EMG	Red	An emergency stop is being actuated.
SV1	Green	The servo for axis 1 is on.
SV2	Green	The servo for axis 2 is on.

[8] Panel unit connector: This connector is used to connect the optional panel unit.

[9] PIO connector: This 34-pin, flat DIO connector consists of 24 inputs and eight outputs.

Standard I/O Interface Specifications (key items)

Item	Description
Connector name	I/O
Applicable connector	Flat connector, 34 pins
Power supply	Power is supplied from connector pin Nos. 1 and 34.
Inputs	24 points (including general-purpose inputs and dedicated inputs)
Outputs	8 points (including general-purpose outputs and dedicated outputs)
Connected to	External PLC, sensor, etc.

I/O Interface List (Program mode)

Pin No.	Category	Port No.	Function	Cable color
1A		-	External power supply 24 V	1-Brown
1B	Input	016	Program specification (PRG No. 1)	1-Red
2A		017	Program specification (PRG No. 2)	1-Orange
2B		018	Program specification (PRG No. 4)	1-Yellow
3A		019	Program specification (PRG No. 8)	1-Green
3B		020	Program specification (PRG No. 10)	1-Blue
4A		021	Program specification (PRG No. 20)	1-Purple
4B		022	Program specification (PRG No. 40)	1-Gray
5A		023	Software reset (restart)	1-White
5B		000	Program start	1-Black
6A		001	General-purpose input	2-Brown
6B		002	General-purpose input	2-Red
7A		003	General-purpose input	2-Orange
7B		004	General-purpose input	2-Yellow
8A		005	General-purpose input	2-Green
8B		006	General-purpose input	2-Blue
9A		007	General-purpose input	2-Purple
9B		008	General-purpose input	2-Gray
10A		009	General-purpose input	2-White
10B		010	General-purpose input	2-Black
11A		011	General-purpose input	3-Brown
11B		012	General-purpose input	3-Red
12A		013	General-purpose input	3-Orange
12B		014	General-purpose input	3-Yellow
13A		015	General-purpose input	3-Green
13B	Output	300	Alarm output	3-Blue
14A		301	Ready output	3- Purple
14B		302	Emergency-stop output	3-Gray
15A		303	General-purpose output	3-White
15B		304	General-purpose output	3-Black
16A		305	General-purpose output	4-Brown
16B		306	General-purpose output	4-Red
17A		307	General-purpose output	4-Orange
17B	N		External power supply 0 V	4-Yellow

The above functions reflect the factory settings for the program mode.
These functions can be changed by changing the corresponding parameters.

[10] MANU/AUTO switch:

This switch is used to specify the controller operation mode.

MANU (left)  AUTO (right)

	MANU	AUTO
Teaching pendant/PC software operation (when the TP connector is used)	Possible	Not possible
PC software operation (when the USB connector is used)	Possible Note)	Not possible
Starting of an auto start program	Not possible	Possible

Note) When this switch is set to the “MANU” side and the USB connector is used, the servo cannot be turned on unless a dummy plug or teaching pendant is connected to the TP connector. When the USB connector is used, always keep a dummy plug or PC software cable connected to the TP plug while the controller is in use. (This is to cancel the disabled condition.)
If a dummy plug is used, always operate the controller in a condition where the emergency stop switch is within an easy reach.

[11] USB connector:

This connector is used to connect the PC software and the controller via a USB cable.
Connector: USB connector B (XM7B-0442)
Connected to: USB cable
The maximum USB cable length is 5 m.

Notes
- When the USB port is used, you must connect all required controllers one by one while installing the USB driver included in the “X-SEL PC Software IA-101-X-USB” CD-ROM. For information on how to install the driver, refer to the Operation Manual for X-SEL PC Software.
- When the USB port is used, a dummy plug must be plugged into the teaching connector [12]. Dummy plug model: DP-3

[12] Teaching connector (TP):

The teaching interface connects IAI's teaching pendant or a PC (PC software) to enable operation and setting of your equipment from the teaching pendant/PC.
The interface is a RS232C system based on a 26-pin, half-pitch I/O connector. The signal level conforms to RS232C, and a desired baud rate (maximum 115.2 kbps) can be selected based on the program. This connector can be used only when the mode switch is set to "MANU."

Interface Specifications of Teaching Serial Interface

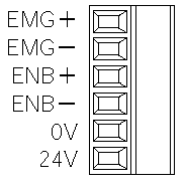
Item	Description	Details
Connector	26-pin, half-pitch I/O connector	TX20A-26R-D2LT1-A1LHE (by JAE)
	Mating connector	TX20A-26PH1-D2P1-D1E (by JAE)
Connector name	T.P.	Teaching connector
Baud rate	Up to 115.2 kbps	Half-duplex communication speeds of up to 115.2 kbps are supported.
Maximum wiring distance	10M	At 38.4 kbps
Interface standard	RS232C	
Connected unit	Dedicated teaching pendant	IAI's standard IA-T-X (D) for X-SEL
Connection cable		Dedicated cable
Power supply	5 VDC or 24 VDC	A multi-fuse (MF-R090) is installed to protect each line against short current (the fuse will trip with currents of between 1.1 A and 2.2 A).
Protocol	X-SEL teaching protocol	The connector supports the X-SEL teaching pendant interface protocol.
Emergency-stop control	Series emergency-stop relay drive (24 V)	An emergency-stop relay drive line is provided in the interface connector. This line is connected in series with other emergency-stop contact.
Enabling control	Enable switch line (24 V)	A line for connecting an enable switch is provided as an operator interlock..

Teaching pendant & dedicated communication cable connector

Item	Specification			Remarks
Terminal assignments	Pin No.	I/O	Signal name	
	1		SG	Signal ground
	2	Out	EMGS	Emergency-stop status
	3	Out	VCC	Power output (Standard IA-T-X/XD power supply (5 V))
	4	In	DTR	Data terminal ready (Shorted to DSR)
	5		NC	Not connected
	6		NC	Not connected
	7		NC	Not connected
	8	Out	RSVVCC	Power output (ANSI compliant IA-T-XA power supply (24 V))
	9	In	EMGIN	Emergency-stop contact output, negative
	10	Out	RSVVCC	Power output (ANSI compliant IA-T-XA power supply (24 V))
	11		NC	Not connected
	12	Out	EMGOUT2	Emergency-stop contact output, positive
	13	Out	RTS	Request to send (Not used; fixed to 0 V)
	14	In	CTS	Clear to send (Not used / Used as the TP-connection detection terminal)
	15	Out	TXD	Transmitted data
	16	In	RXD	Received data
	17	Out	DSR	Data set ready (Shorted to DTR)
	18		NC	Not connected
	19		NC	Not connected
	20		NC	Not connected
	21		NC	Not connected
	22		NC	Not connected
	23	In	ENBTB	Enable input
	24	Out	ENBVCC	Enable drive power (24 V)
	25		NC	Not connected (Reserved by ENBTBX2)
	26		SG	Signal ground

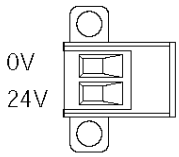
[13] System-memory backup battery connector: This connector is used to install the system-memory backup battery.

[14] Control power & system I/O connector: This connector is used to input the 24-VDC control power and connect the emergency stop switch and enable switch.
The power supply connected to this connector is used for the controller internal power, brake power, and so on, and not used as the motor drive source.
The 0-V input is connected to the ground for the controller's internal power supply and is not insulated.



Item	Specification		Remarks
Applicable connector	3.5 mm, 2-piece COMBICON, 6 pins		MC1.5/6-G-3.5 by Phoenix Contact
	Cable-end connector		MC1.5/6-ST-3.5 by Phoenix Contact
	Applicable wire size		AWG20 ~ 16 (0.5 ~ 1.25 sq)
	Recommended stripped-wire length		7 mm
Connector name	CP EMG ENB		
Input voltage	24 VDC + 10%/-10%		
Maximum input current	1.2 A		
Terminal assignments	No.	Name	Function
	1	EMG+	Emergency stop switch +
	2	EMG-	Emergency stop switch -
	3	ENB+	Enable switch +
	4	ENB-	Enable switch -
	5	0 V	Control power input ground (Connected to the internal ground)
	6	24 V	Control power input +24 V

[15] Motor power connector: This connector is used to input the 24-VDC motor power.
The power supply connected to this connector is used as the dedicated motor drive source.
Since the controller has a built-in drive-source cutoff relay, the power supply to the motor will be cut off internally if an emergency stop is actuated or other abnormality occurs.
Although the motor power and control power are input independently, the 0-V terminals of both are connected inside the controller. They are also connected to the ground for the controller's internal power supply and are not insulated.

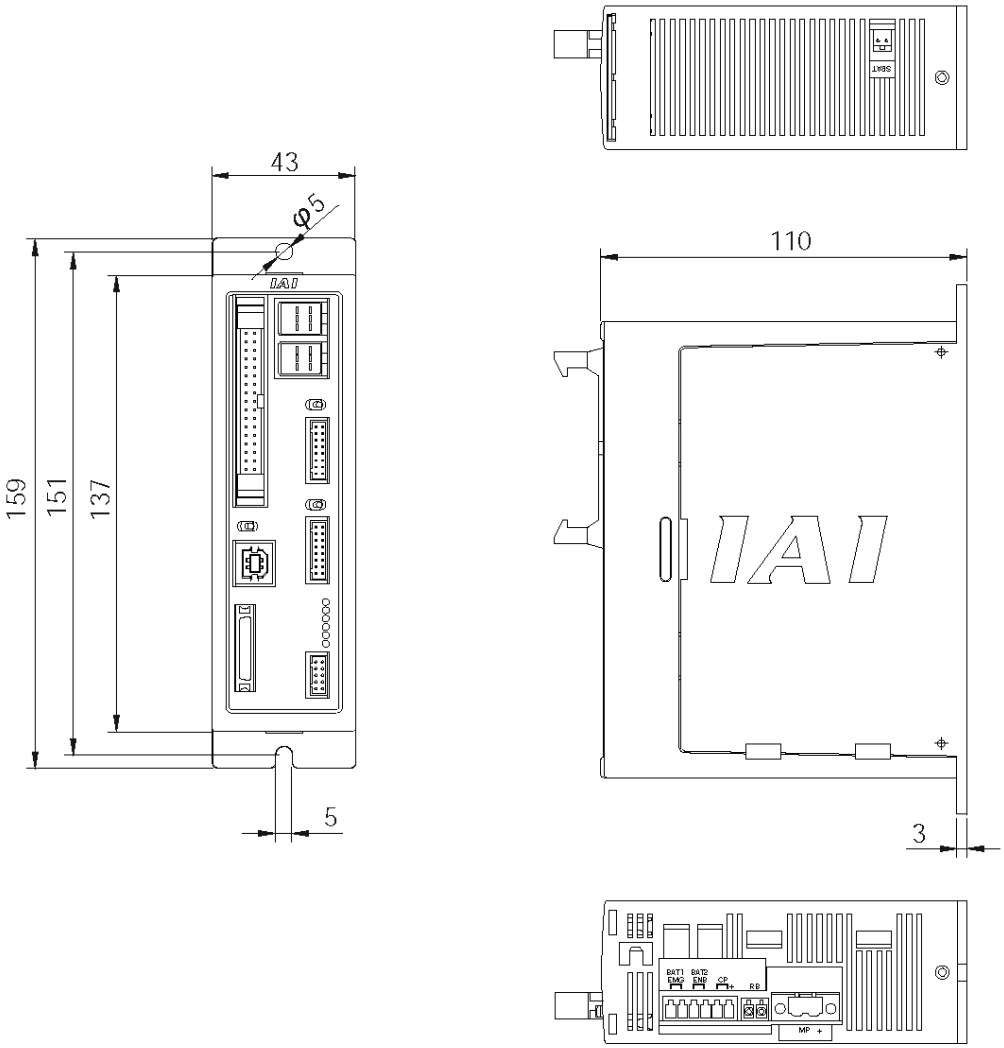


Item	Specification		Remarks
Applicable connector	5.08 mm, 2-piece COMBICON, 2 pins		MSTB2.5/2-GF-5.08 by Phoenix Contact
	Cable-end connector		MSTB2.5/2-STF-5.08 by Phoenix Contact
	Applicable wire size		AWG20 ~ 14 (0.5 ~ 2.0 sq)
	Recommended stripped-wire length		7 mm
Connector name	MP		
Input voltage	24 VDC ± 10%		
Maximum input current	4.0 A		2.0 A per axis
Terminal assignments	No.	Name	Function
	1	0 V	Motor power input ground (Connected to the internal ground)
	2	24 V	Motor power input +24 V

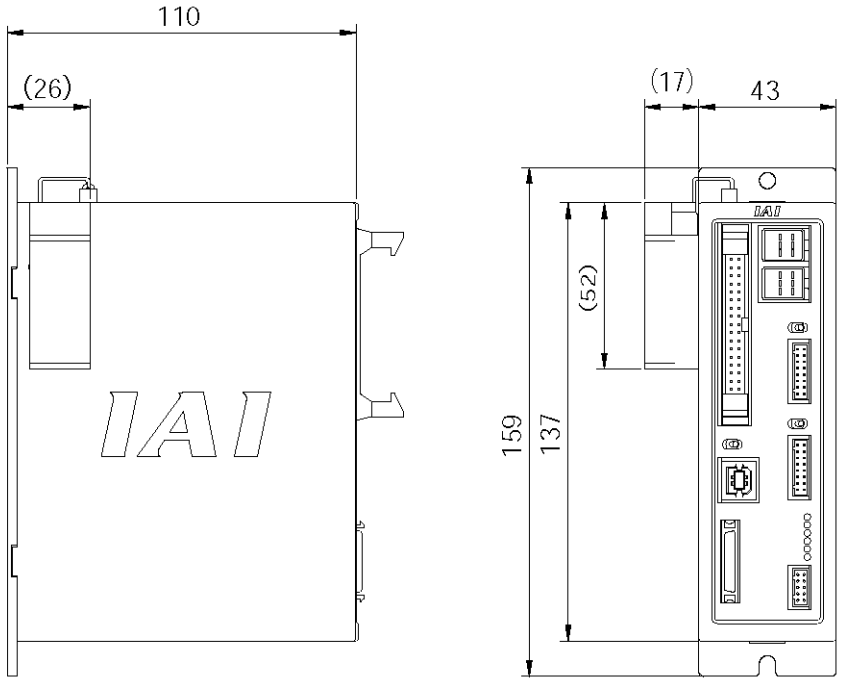
Chapter 3 Installation and Wiring

1. External Dimensions

(1) 2-axis specification
(The same external dimensions also apply to the 1-axis specification.)



(2) 2-axis specification with battery



2. Installation Environment

(1) When installing and wiring the controller, do not block the ventilation holes provided for cooling. (Insufficient ventilation will not only prevent the product from functioning fully, but it may also result in failure.)
(2) Prevent foreign matter from entering the controller through the ventilation holes. Since the controller is not designed as dustproof or waterproof (oilproof), avoid using it in a dusty place or place subject to oil mist or splashed cutting fluid.
(3) Do not expose the controller to direct sunlight or radiant heat from a high heat source such as a heat-treating furnace.
(4) Use the controller in a non-condensing environment free from corrosive or inflammable gases.
(5) Use the controller in an environment where it will not receive external vibration or impact.
(6) Prevent electrical noise from entering the controller or its cables.

Environmental Condition of Controller

Item	Specification and description
Surrounding air temperature range	0 ~ 40°C
Surrounding humidity range	10% ~ 95% (Non-condensing; conforming to JIS C3502 RH-2)
Storage temperature range	-25°C ~ 70°C (Excluding the battery)
Maximum operating altitude	2000 m
Protection class	IP20
Vibration	$10 \leq f < 57$: 0.035 mm (continuous), 0.075 mm (intermittent) $57 \leq f \leq 150$: 4.9 m/s^2 (continuous), 9.8 m/s^2 (intermittent) X, Y and Z directions
Impact	147 mm/s^2, 11 ms, half-sine pulse, 3 times each in X, Y and Z directions

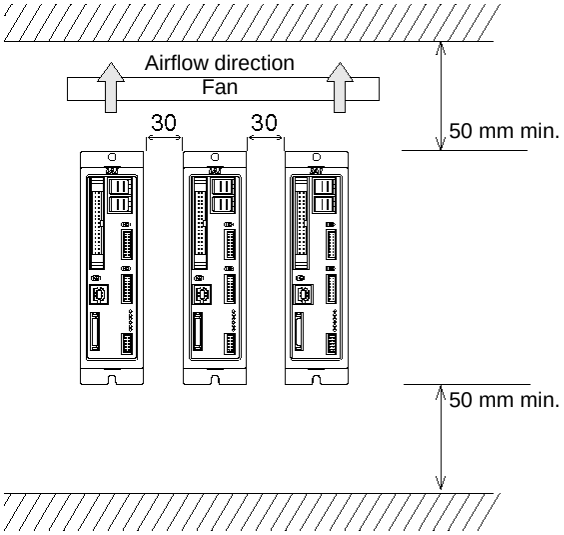
3. Heat Radiation and Installation

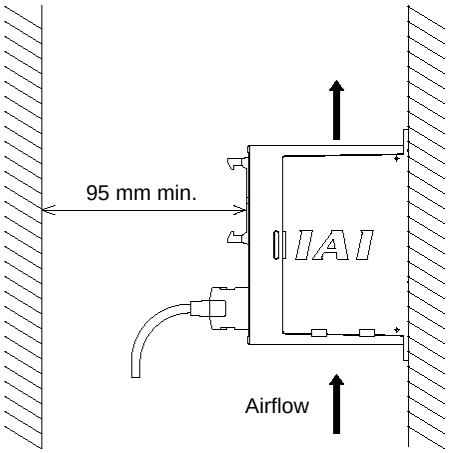
Design the control panel size, controller layout and cooling method so that the ambient temperature around the controller will be kept at or below 40°C.

Install the controller vertically on a wall, as illustrated below. The controller will be cooled by natural convection. Be sure to install the controller in the aforementioned direction and provide a minimum clearance of 50 mm above and below the controller.

If multiple controllers are to be installed side by side, providing additional suction fans on top of the controllers will help maintain a uniform ambient temperature.

Provide a minimum clearance of 95 mm between the front side of the controller and a wall (enclosure).



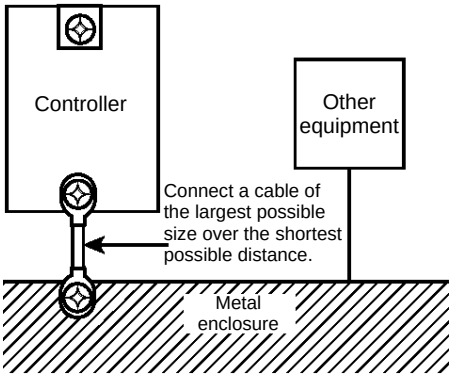


If multiple controllers are to be connected on top of one another, prevent the controller above from taking in the exhaust air from the controller below.

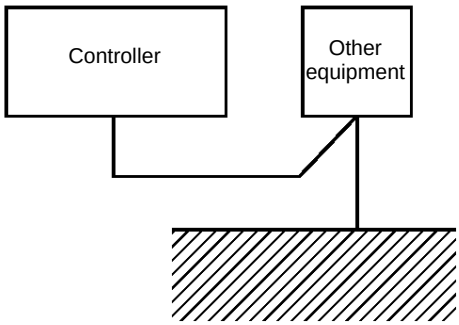
4. Noise Control Measures and Grounding

The PSEL controller has no dedicated terminal to connect the FG to ground. Accordingly, provide grounding using the controller mounting screw.

[1] Provide dedicated Class D grounding. The grounding wire should have a size of 2.0 to 5.5 mm^2 or larger.



Class D grounding Proper grounding



Avoid using this method.

[2] Notes on wiring method

Use twisted wires for the 24-VDC external power supply.

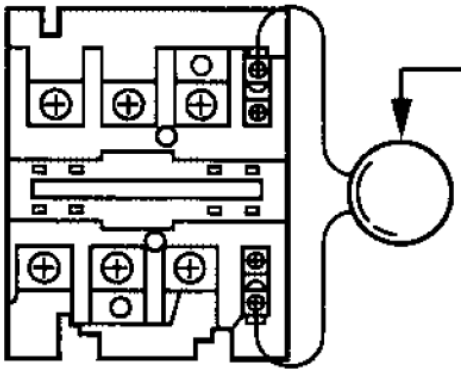
Wire the controller cables separately from lines creating a strong electric field such as power circuit lines (by not bundling them together or placing in the same cable duct).

If you wish to extend the motor cable or encoder cable beyond the length of each supplied cable, please contact IAI's Technical Service Section or Sales Engineering Section.

(3) Noise sources and noise elimination
There are many noise sources, but solenoid valves, magnet switches and relays are of particular concern when building a system. Noise from these parts can be eliminated using the measures specified below:

[1] AC solenoid valve, magnet switch, relay

Measure --- Install a surge killer in parallel with the coil.

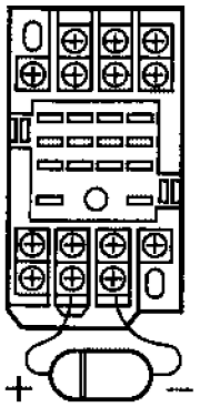


Surge killer

← Point
Wire from each coil over the shortest distance. Installing a surge killer on the terminal block, etc., will be less effective because of a longer distance from the coil.

[2] DC solenoid valve, magnet switch, relay

Measure --- Install a diode in parallel with the coil. Determine the diode capacity in accordance with the load capacity.

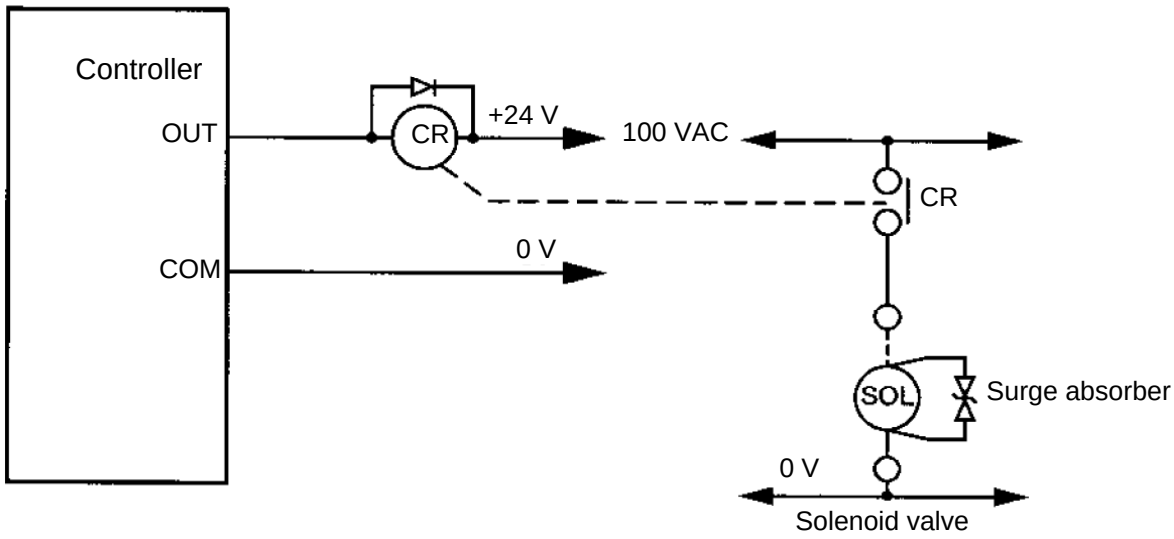


In a DC circuit, connecting a diode in reversed polarity will damage the diode, internal parts of the controller and DC power supply. Exercise due caution.

Diode

The above noise elimination measures are particularly important when a 24-VDC relay is driven directly by a controller output and there is also a 100-VAC solenoid valve, etc.

Reference Circuit Diagram



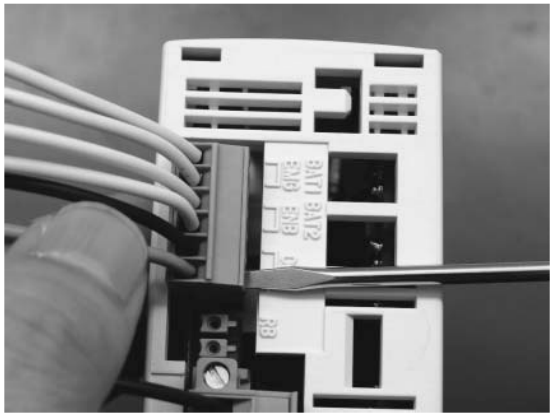
5. Supply Voltage

The supply voltage to the controller is 24 VDC ± 10%.
The power-supply current varies depending on the number of axes, as shown below.

	1-axis specification	2-axis specification
[1] Control power-supply current	1.2 A	
[2] Rated motor power-input current	1.2 A	2.4 A
[3] Maximum motor power-input current	2.0 A	4.0 A
[4] Rated current ([1] + [2])	2.4 A	3.6 A
[5] Maximum current ([1] + [3])	3.2 A	5.2 A

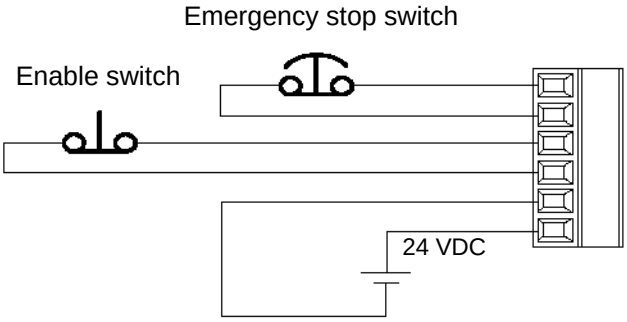
6. Wiring

6.1 Wiring the Control Power Supply, Emergency Stop Switch and Enable Switch

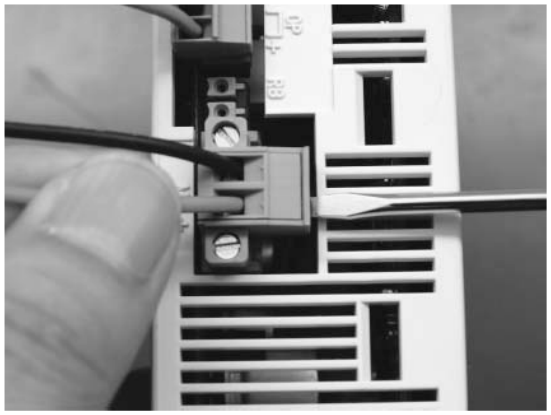


As shown to the left, insert the stripped end of each cable into the control power & system I/O connector, and tighten the screws with a screwdriver.

Recommended cable size: 0.75 mm^2 (AWG18)
Recommended stripped-wire length: 7 mm

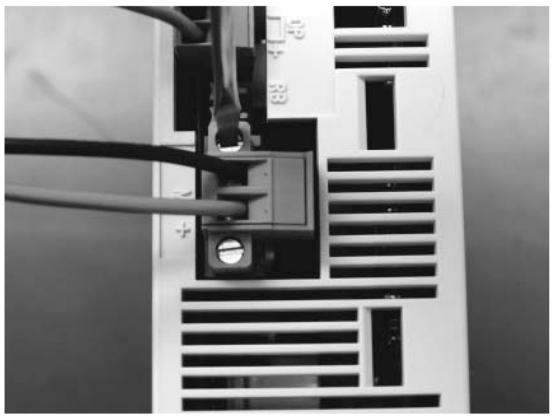


6.2 Wiring the Motor Power Cables

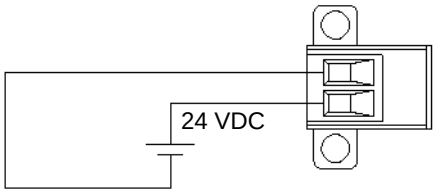


As shown to the left, insert the stripped end of each cable into the motor power connector, and tighten the screws with a screwdriver.

Recommended cable size: 2 mm^2 (AWG14)
Recommended stripped-wire length: 7 mm

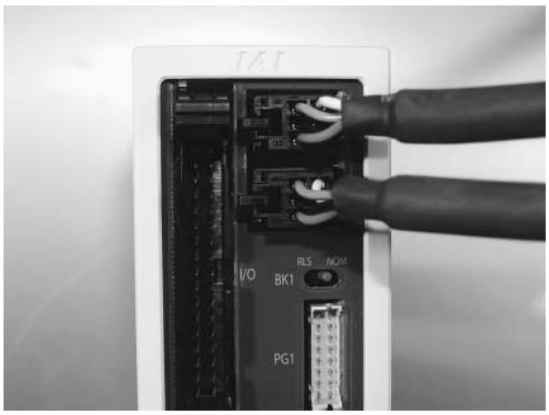


As shown to the left, tighten the screws to affix the connector.



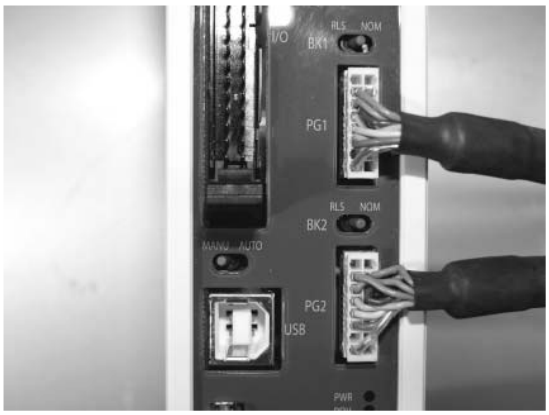
6.3 Connecting the Actuator

6.3.1 Connecting the Motor Cable (M1/M2)



Connect the motor cable from the actuator to the applicable motor connector on the front face of the controller.

6.3.2 Connecting the Encoder Cable (PG1/PG2)

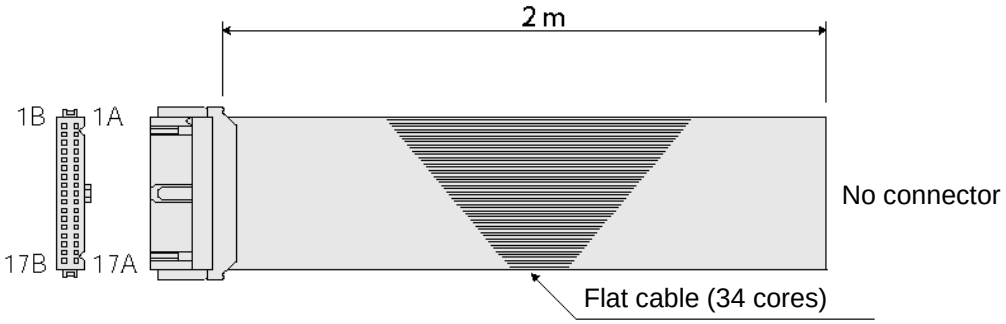


Connect the encoder cable from the actuator to the applicable encoder connector on the front face of the controller.

6.4 Connecting the PIO Cable (I/O)

Connect the supplied flat cable. Connect the opposite end (open end without connector) of the cable to a desired peripheral (host PLC, etc.).

I/O flat cable (supplied): Model CB-DS-P10020



No.	Color	Wire	No.	Color	Wire
1A	Brown 1	Flat cable, pressure-welded	9B	Gray 2	Flat cable, pressure-welded
1B	Red 1		10A	White 2	
2A	Orange 1		10B	Black 2	
2B	Yellow 1		11A	Brown-3	
3A	Green 1		11B	Red 3	
3B	Blue 1		12A	Orange 3	
4A	Purple 1		12B	Yellow 3	
4B	Gray 1		13A	Green 3	
5A	White 1		13B	Blue 3	
5B	Black 1		14A	Purple 3	
6A	Brown-2		14B	Gray 3	
6B	Red 2		15A	White 3	
7A	Orange 2		15B	Black 3	
7B	Yellow 2		16A	Brown-4	
8A	Green 2		16B	Red 4	
8B	Blue 2		17A	Orange 4	
9A	Purple 2		17B	Yellow 4	

6.4.1 I/O Connection Diagram

(1) NPN specification (Program mode)

0V

+24V

Pin No.	Category	Port No.	Function	Cable color
1A	P24		External power supply 24 V	1 – Brown
1B	Input	016	Program specification (PRG No. 1)	1 – Red
2A		017	Program specification (PRG No. 2)	1 – Orange
2B		018	Program specification (PRG No. 4)	1 – Yellow
3A		019	Program specification (PRG No. 8)	1 – Green
3B		020	Program specification (PRG No. 10)	1 – Blue
4A		021	Program specification (PRG No. 20)	1 – Purple
4B		022	Program specification (PRG No. 40)	1 – Gray
5A		023	Software reset (restart)	1 – White
5B		000	Program start	1 – Black
6A		001	General-purpose input	2 – Brown
6B		002	General-purpose input	2 – Red
7A		003	General-purpose input	2 – Orange
7B		004	General-purpose input	2 – Yellow
8A		005	General-purpose input	2 – Green
8B		006	General-purpose input	2 – Blue
9A		007	General-purpose input	2– Purple
9B		008	General-purpose input	2 – Gray
10A		009	General-purpose input	2 – White
10B		010	General-purpose input	2 – Black
11A		011	General-purpose input	3 – Brown
11B		012	General-purpose input	3 – Red
12A		013	General-purpose input	3 – Orange
12B		014	General-purpose input	3 – Yellow
13A		015	General-purpose input	3 – Green

Pin No.	Category	Port No.	Function	Cable color
13B	Output	300	Alarm output	3 – Blue
14A		301	Ready output	3 – Purple
14B		302	General-purpose output	3 – Gray
15A		303	General-purpose output	3 – White
15B		304	General-purpose output	3 – Black
16A		305	General-purpose output	4 – Brown
16B		306	General-purpose output	4 – Red
17A		307	General-purpose output	4 – Orange
17B	N		External power supply 0V	4 – Yellow

The above functions reflect the factory settings.

(2) PNP specification (Program mode)

+24V 0V

Pin No.	Category	Port No.	Function	Cable color
1A	P24		External power supply 24 V	1 – Brown
1B	Input	016	Program specification (PRG No. 1)	1 – Red
2A		017	Program specification (PRG No. 2)	1 – Orange
2B		018	Program specification (PRG No. 4)	1 – Yellow
3A		019	Program specification (PRG No. 8)	1 – Green
3B		020	Program specification (PRG No. 10)	1 – Blue
4A		021	Program specification (PRG No. 20)	1 – Purple
4B		022	Program specification (PRG No. 40)	1 – Gray
5A		023	Software reset (restart)	1 – White
5B		000	Program start	1 – Black
6A		001	General-purpose input	2 – Brown
6B		002	General-purpose input	2 – Red
7A		003	General-purpose input	2 – Orange
7B		004	General-purpose input	2 – Yellow
8A		005	General-purpose input	2 – Green
8B		006	General-purpose input	2 – Blue
9A		007	General-purpose input	2– Purple
9B		008	General-purpose input	2 – Gray
10A		009	General-purpose input	2 – White
10B		010	General-purpose input	2 – Black
11A		011	General-purpose input	3 – Brown
11B		012	General-purpose input	3 – Red
12A		013	General-purpose input	3 – Orange
12B		014	General-purpose input	3 – Yellow
13A		015	General-purpose input	3 – Green

Pin No.	Category	Port No.	Function	Cable color
13B	Output	300	Alarm output	3 – Blue
14A		301	Ready output	3 – Purple
14B		302	General-purpose output	3 – Gray
15A		303	General-purpose output	3 – White
15B		304	General-purpose output	3 – Black
16A		305	General-purpose output	4 – Brown
16B		306	General-purpose output	4 – Red
17A		307	General-purpose output	4 – Orange
17B	N		External power supply 0V	4 – Yellow

The above functions reflect the factory settings.

(3) NPN specification (Standard positioner mode)

0V +24V

Pin No.	Category	Port No.	Positioner mode: Standard mode	Model switching mode	2-axis independent mode	Teaching mode	CD-S-C1 compatible mode	Cable color
1A	P24		24-V input					1 – Brown
1B	Input	16	Position input 10	Input 10	Position input 7	Axis 1 jog	Position No. 1000 input	1 – Red
2A		17	Position input 11	Input 11	Position input 8	Axis 1 jog	—	1 – Orange
2B		18	Position input 12	Input 12	Position input 9	Axis 2 jog	—	1 – Yellow
3A		19	Position input 13	Input 13	Position input 10	Inching (0.01 mm)	—	1 – Green
3B		20	—	Input 14	Position input 11	Inching (0.1 mm)	—	1 – Blue
4A		21	—	Input 15	Position input 12	Inching (0.5 mm)	—	1 – Purple
4B		22	—	Input 16	Position input 13	Inching (1 mm)	—	1 – Gray
5A		23	Error reset	Error reset	Error reset	Error reset	CPU reset	1 – White
5B		0	Start	Start	Axis 1 start	Start	Start	1 – Black
6A		1	Home return	Home return	Home return	Servo ON	Pause	2 – Brown
6B		2	Servo ON	Servo ON	Axis 1 servo ON	* Pause	Cancellation	2 – Red
7A		3	Push motion	Push motion	* Axis 1 pause	Position input 1	Interpolation setting	2 – Orange
7B		4	* Pause	* Pause	* Axis 1 cancellation	Position input 2	Position No. 1 input	2 – Yellow
8A		5	* Cancellation	* Cancellation	Axis 2 start	Position input 3	Position No. 2 input	2 – Green
8B		6	Interpolation	Interpolation	Axis 2 home return	Position input 4	Position No. 4 input	2 – Blue
9A		7	Position input 1	Input 1	Axis 2 servo ON	Position input 5	Position No. 8 input	2– Purple
9B		8	Position input 2	Input 2	* Axis 2 pause	Position input 6	Position No. 10 input	2 – Gray
10A		9	Position input 3	Input 3	* Axis 2 cancellation	Position input 7	Position No. 20 input	2 – White
10B		10	Position input 4	Input 4	Position input 1	Position input 8	Position No. 40 input	2 – Black
11A		11	Position input 5	Input 5	Position input 2	Position input 9	Position No. 80 input	3 – Brown
11B		12	Position input 6	Input 6	Position input 3	Position input 10	Position No. 100 input	3 – Red
12A		13	Position input 7	Input 7	Position input 4	Position input 11	Position No. 200 input	3 – Orange
12B		14	Position input 8	Input 8	Position input 5	Teaching mode specification	Position No. 400 input	3 – Yellow
13A		15	Position input 9	Input 9	Position input 6	Axis 1 jog+	Position No. 800 input	3 – Green

Pin No	Category	Port No	Positioner mode: Standard mode	Model switching mode	2-axis independent mode	Teaching mode	CDC-S-C1 compatible mode	Cable color
13B	Output	300	* Alarm	* Alarm	* Alarm	* Alarm	Alarm	3 – Blue
14A		301	Ready	Ready	Ready	Ready	Ready	3 – Purple
14B		302	Positioning complete	Positioning complete	Axis 1 positioning complete	Positioning complete	Positioning complete	3 – Gray
15A		303	Home return complete	Home return complete	Axis 1 home return complete	Home return complete	—	3 – White
15B		304	Servo ON output	Servo ON output	Axis 1 servo ON	Servo ON output	—	3 – Black
16A		305	Push motion complete	Push motion complete	Axis 2 positioning complete		—	4 – Brown
16B		306	System battery error	System battery error	Axis 2 home return complete	System battery error	System battery error	4 – Red
17A		307	—	—	Axis 2 servo ON	—	—	4 – Orange
17B	N		0-V input					4 – Yellow

*: Contact B (always ON)

(4) PNP specification (Standard positioner mode)

+24V 0V

Pin No.	Category	Port No.	Positioner mode: Standard mode	Model switching mode	2-axis independent mode	Teaching mode	CD-S-C1 compatible mode	Cable color
1A	P24		24-V input					1 – Brown
1B	Input	16	Position input 10	Input 10	Position input 7	Axis 1 jog	Position No. 1000 input	1 – Red
2A		17	Position input 11	Input 11	Position input 8	Axis 1 jog	—	1 – Orange
2B		18	Position input 12	Input 12	Position input 9	Axis 2 jog	—	1 – Yellow
3A		19	Position input 13	Input 13	Position input 10	Inching (0.01 mm)	—	1 – Green
3B		20	—	Input 14	Position input 11	Inching (0.1 mm)	—	1 – Blue
4A		21	—	Input 15	Position input 12	Inching (0.5 mm)	—	1 – Purple
4B		22	—	Input 16	Position input 13	Inching 1 mm)	—	1 – Gray
5A		23	Error reset	Error reset	Error reset	Error reset		1 – White
5B		0	Start	Start	Axis 1 start	Start	Start	1 – Black
6A		1	Home return	Home return	Home return	Servo ON	Pause	2 – Brown
6B		2	Servo ON	Servo ON	Axis 1 servo ON	* Pause	Cancellation	2 – Red
7A		3	Push motion	Push motion	* Axis 1 pause	Position input 1	Interpolation setting	2 – Orange
7B		4	* Pause	* Pause	* Axis 1 cancellation	Position input 2	Position No. 1 input	2 – Yellow
8A		5	* Cancellation	* Cancellation		Position input 3	Position No. 2 input	2 – Green
8B		6	Interpolation	Interpolation	Axis 2 start	Position input 4	Position No. 4 input	2 – Blue
9A		7	Position input 1	Input 1	Axis 2 home return	Position input 5	Position No. 8 input	2– Purple
9B		8	Position input 2	Input 2	Axis 2 servo ON	Position input 6	Position No. 10 input	2 – Gray
10A		9	Position input 3	Input 3	* Axis 2 pause	Position input 7	Position No. 20 input	2 – White
10B		10	Position input 4	Input 4	Position input 1	Position input 8	Position No. 40 input	2 – Black
11A		11	Position input 5	Input 5	Position input 2	Position input 9	Position No. 80 input	3 – Brown
11B		12	Position input 6	Input 6	Position input 3	Position input 10	Position No. 100 input	3 – Red
12A		13	Position input 7	Input 7	Position input 4	Position input 11	Position No. 200 input	3 – Orange
12B		14	Position input 8	Input 8	Position input 5	Teaching mode specification	Position No. 400 input	3 – Yellow
13A		15	Position input 9	Input 9	Position input 6	Axis 1 jog+	Position No. 800 input	3 – Green

Pin No.	Category	Port No.	Positioner mode: Standard mode	Model switching mode	2-axis independent mode	Teaching mode	DC-S-C1 compatible mode	Cable color
13B	Output	300	* Alarm	* Alarm	* Alarm	* Alarm	Alarm	3 – Blue
14A		301	Ready	Ready	Ready	Ready	Ready	3 – Purple
14B		302	Positioning complete	Positioning complete	Axis 1 positioning complete	Positioning complete	Positioning	3 – Gray
15A		303	Home return complete	Home return complete	Axis 1 home return complete	Home return complete	—	3 – White
15B		304	Servo ON output	Servo ON output	Axis 1 servo ON	Servo ON output	—	3 – Black
16A		305	Push motion complete	Push motion complete	Axis 2 positioning complete		—	4 – Brown
16B		306	System battery error	System battery error	Axis 2 home return complete	System battery error	System battery error	4 – Red
17A		307	—	—	Axis 2 servo ON	—	—	4 – Orange
17B	N		0-V input					4 – Yellow

*: Contact B (always ON)

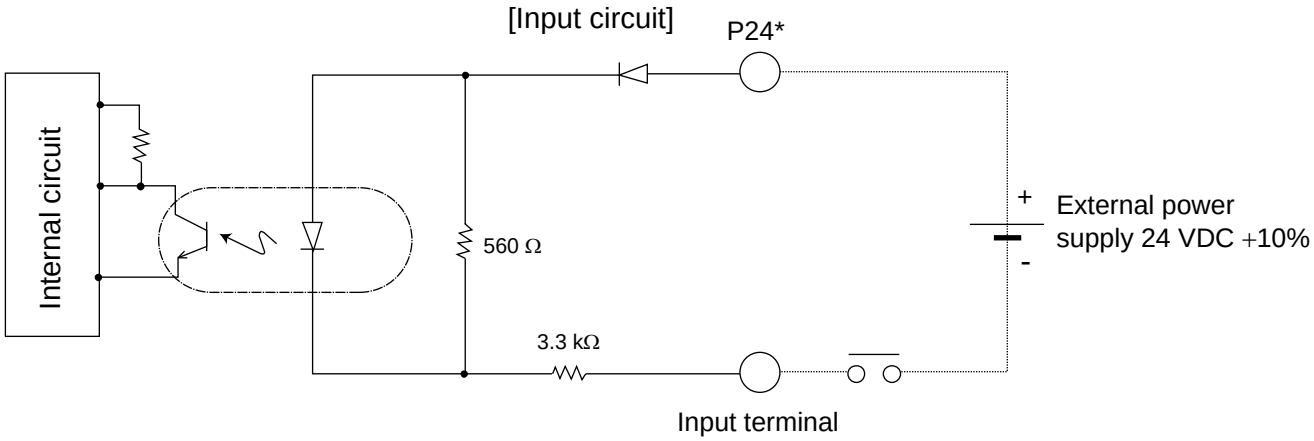
6.5 External I/O Specifications

6.5.1 NPN Specification

(1) Input part

External Input Specifications (NPN Specification)

Item	Specification
Input voltage	24 VDC ±10%
Input current	7 mA per circuit
ON/OFF voltage	ON voltage --- 16.0 VDC min. OFF voltage --- 5.0 VDC max.
Insulation method	Photocoupler insulation
External devices	[1] No-voltage contact (minimum load of approx. 5 VDC/1 mA) [2] Photoelectric/proximity sensor (NPN type) [3] Sequencer transistor output (open-collector type) [4] Sequencer contact output (minimum load of approx. 5 VDC/1 mA)

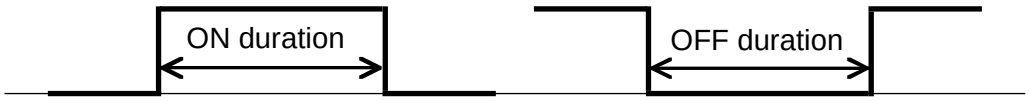


* P24: I/O interface pin No. 1

Caution

If a non-contact circuit is connected externally, malfunction may result from leakage current. Use a circuit in which leakage current in a switch-off state does not exceed 1 mA.

⊙ PSEL controller's input signal



At the default settings, the system recognizes the ON/OFF durations of input signals if they are approx. 4 msec or longer. The ON/OFF duration settings can also be changed using I/O parameter No. 20 (input filtering frequency).

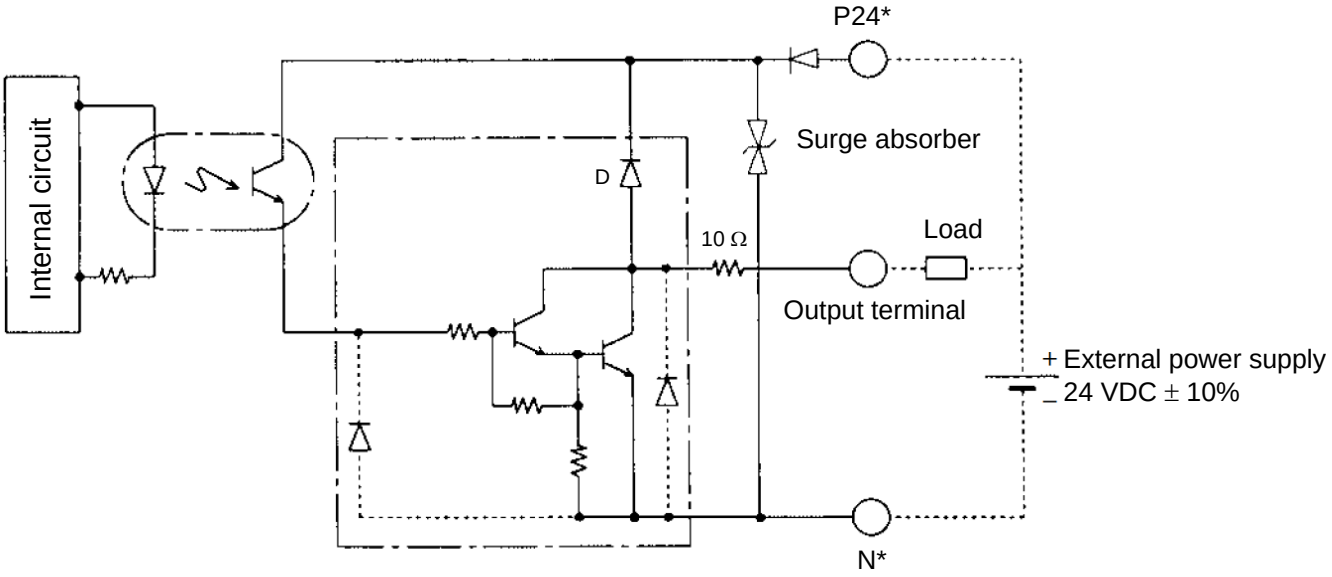
(2) Output part

External Output Specifications (NPN Specification)

Item	Specification	
Load voltage	24 VDC	TD62084 (or equivalent)
Maximum load current	100 mA per point, 400 mA per 8 ports Note)	
Leakage current	0.1 mA max. per point	
Insulation method	Photocoupler insulation	
External devices	[1] Miniature relay [2] Sequencer input unit	

Note) 400 mA is the maximum total load current of output port Nos. 300 to 307.

[Output circuit]



* P24: I/O interface pin No. 1A
* N: I/O interface pin No. 17B

Caution

In the event that the load is short-circuited or current exceeding the maximum load current is input, the overcurrent protection circuit will be actuated to cut off the circuit. However, give due consideration to the circuit connection layout to prevent short-circuit or overcurrent.

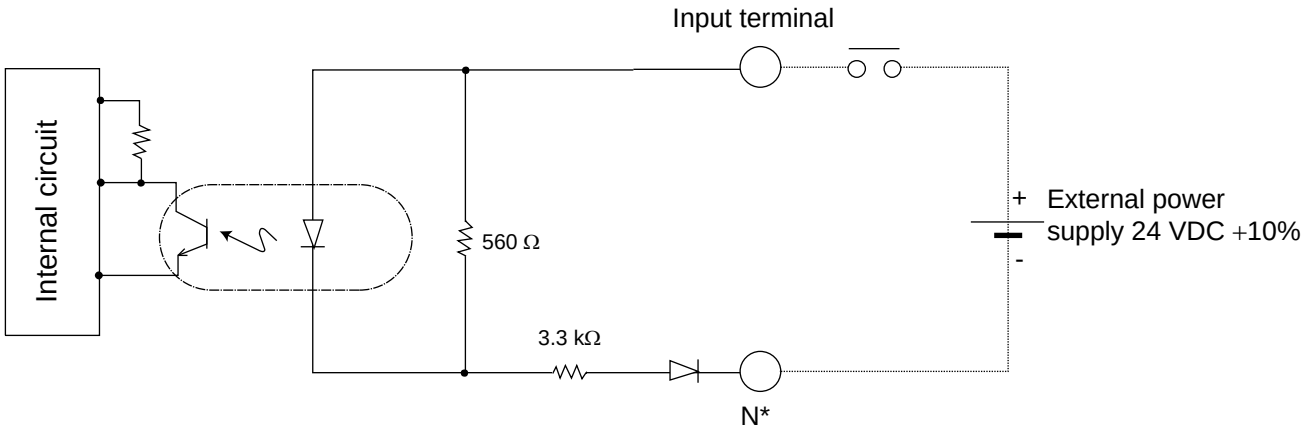
6.5.2 PNP Specification

(1) Input part

External Input Specifications (PNP Specification)

Item	Specification
Input voltage	24 VDC ±10%
Input current	7 mA per circuit
ON/OFF voltage	ON voltage --- 8 VDC max. OFF voltage --- 19 VDC min.
Insulation method	Photocoupler insulation
External devices	[1] No-voltage contact (minimum load of approx. 5 VDC/1 mA) [2] Photoelectric/proximity sensor (PNP type) [3] Sequencer transistor output (open-collector type) [4] Sequencer contact output (minimum load of approx. 5 VDC/1 mA)

[Input circuit]



* N: I/O interface pin No. 17B

Caution

If a non-contact circuit is connected externally, malfunction may result from leakage current. Use a circuit in which leakage current in a switch-off state does not exceed 1 mA.

⊙ PSEL controller's input signal

At the default settings, the system recognizes the ON/OFF durations of input signals if they are approx. 4 msec or longer. The ON/OFF duration settings can also be changed using I/O parameter No. 20 (input filtering frequency).

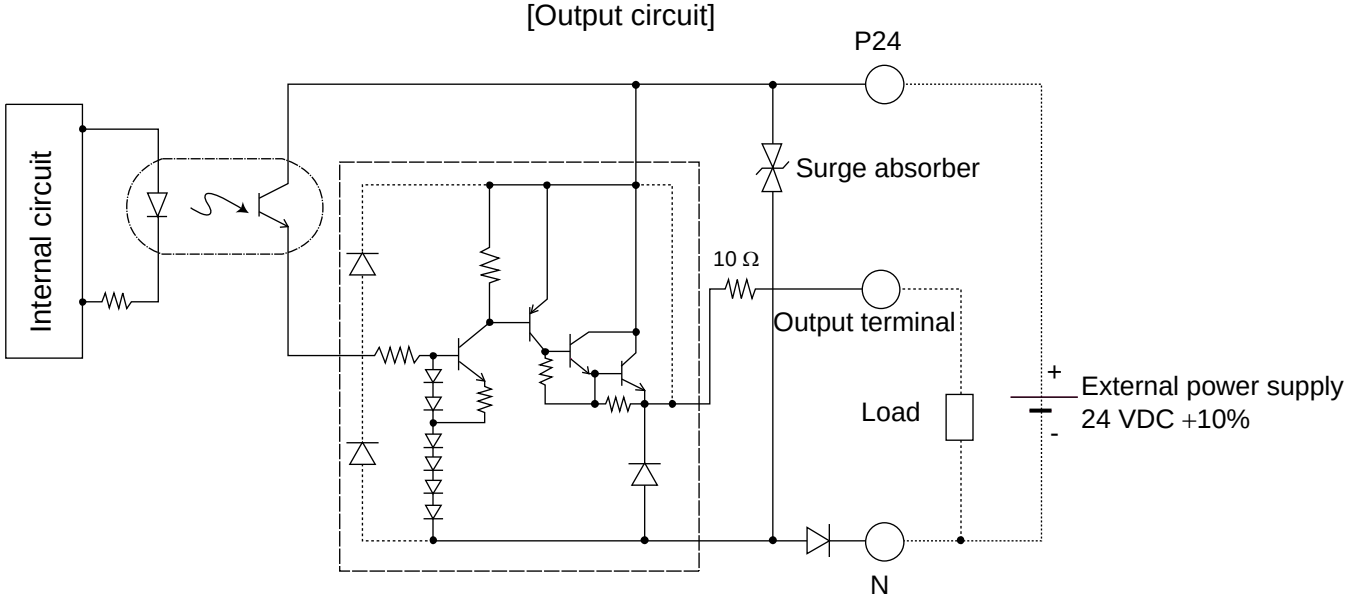
(2) Output part

External Output Specifications (PNP Specification)

Item	Specification	
Load voltage	24 VDC	TD62784 (or equivalent)
Maximum load current	100 mA per point, 400 mA per 8 ports Note)	
Leakage current	0.1 mA max. per point	
Insulation method	Photocoupler insulation	
External devices	[1] Miniature relay [2] Sequencer input unit	

Note) 400 mA is the maximum total load current of output port Nos. 300 to 307.

[Output circuit]



* P24: I/O interface pin No. 1A
* N: I/O interface pin No. 17B

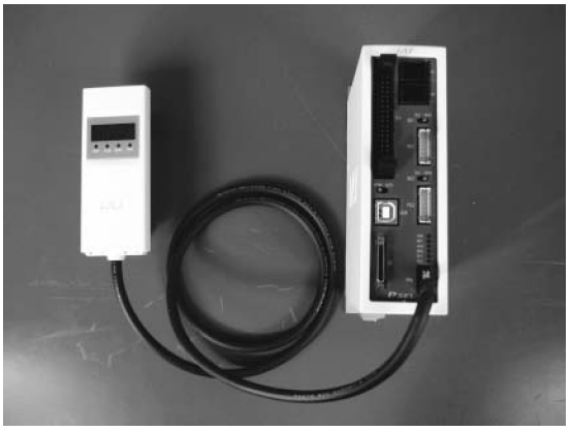
Caution

In the event that the load is short-circuited or a current exceeding the maximum load current is input, the overcurrent protection circuit will be actuated to cut off the circuit. However, give due consideration to the circuit connection layout to prevent short-circuit or overcurrent.

6.6 Connecting the Teaching Pendant/PC (Software) (TP) (Optional)

The PSEL controller's teaching connector (TP) is a small, half-pitch connector. If you are using a teaching pendant or PC software cable, connect the cable to a connector conversion cable, and then connect the conversion cable to the teaching connector on the controller.

6.7 Connecting the Panel Unit (Optional)



When the optional panel unit is connected, the controller status (program number of each active program, error codes, etc.) can be monitored.

6.7.1 Explanation of Codes Displayed on the Panel Unit (Optional)

(1) Application

Display	Priority (*1)	Description
cIIE	1	Control power cut off
EFXX	1	System-down level error
PrSd	2	Writing data to the flash ROM.
EStC	3	Emergency stop is being actuated (except during the update mode).
EnbC	4	Enable switch (deadman switch/safety gate) OFF (except in the update mode)
EEXX	5	Cold-start level error
EdXX	5	Cold-start level error
ECXX	5	Operation-cancellation level error
EbXX	5	Operation-cancellation level error
— rP	6	Waiting for a drive-source cutoff reset input (except during the update mode).
— rS	6	Operation is in pause (waiting for restart) (except during the update mode).
— IIIC	7	All servo axes are interlocked (except during the update mode).
EMXX	8	Message level error
EqXX	8	Message level error
rUdC	9	Core update mode
UdC	9	Core update is in progress.
FUdC	9	Core update has completed.
rUdS	9	Slave update mode
UdS	9	Slave update is in progress.
FUdS	9	Slave update has completed.
PNo.	9	Running a program (last started program); “No.” indicates program number.
InXX	9	Initialization sequence number
dbG	9	Debug mode

(*1) The priority increases as the number decreases.

Display	Priority (*1)	Description
Ready	9	Ready status (auto mode)　(Program mode)
Ready	9	Ready status (manual mode)　(Program mode)
PrNo.	9	Operating in positioner mode; “No.” indicates positioner mode number.
APr-Y	9	Ready status (auto mode)　(Positioner mode)
Pr-Y	9	Ready status (manual mode)　(Positioner mode)

(*1) The priority increases as the number decreases.

(2) Core

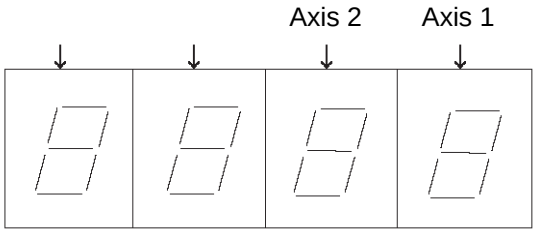
Display	Priority (*1)	Description
dCF	1	Control power cut off
EC-XX	1	Cold-start level error
EdXX	1	Cold-start level error
EC-XX	1	Operation-cancellation level error
EbXX	1	Operation-cancellation level error
EmXX	2	Message level error
EmXX	2	Message level error
-UdM	2	Application update mode
UdM	2	Application update is in progress.
-UdM	2	Application update has completed.
-m	2	Hardware test mode process
EC-m	2	Clearing the application flash ROM.
-EC-m	2	Application flash ROM has been cleared.
JPm	2	Jump to the application
-HF-C	2	Core flash-ROM check process
-HFm	2	Application flash-ROM check process
-HSd	2	SDRAM check process

(*1) The priority increases as the number decreases.

6.7.2 Current Monitor and Variable Monitor

By setting other parameter Nos. 49 and 50 appropriately, the optional panel unit can be used to monitor either current levels or variables.

(1) Current monitor
Currents of up to four axes having continuous axis numbers can be monitored.
Parameter settings
Other parameter No. 49 = 1
Other parameter No. 50 = Smallest axis number among the axes to be monitored
Example) If other parameter No. 49 is set to “1” and other parameter No. 50 to “1” for a 2-axis controller, the far-right segment digit will show the current for axis 1.



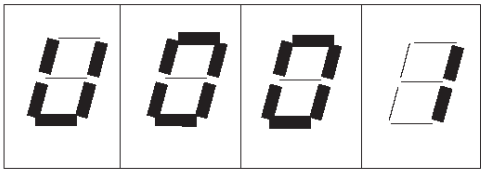
When data is written to the flash ROM or a software reset (restart) is executed after the parameter values have been input, the panel window will show the motor current to rating ratio (%) by a segment pattern, instead of “ready status” or “program run number.”
The segment display patterns and corresponding motor current to rating ratios (%) are shown below.

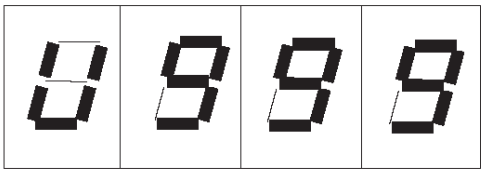
	0 < Motor current to rating ratio (%) ≤ 25		100 < Motor current to rating ratio (%) ≤ 150
	25 < Motor current to rating ratio (%) ≤ 50		150 < Motor current to rating ratio (%) ≤ 200
	50 < Motor current to rating ratio (%) ≤ 75		200 < Motor current to rating ratio (%)
	75 < Motor current to rating ratio (%) ≤ 100		

Thick lines indicate illuminated segments.

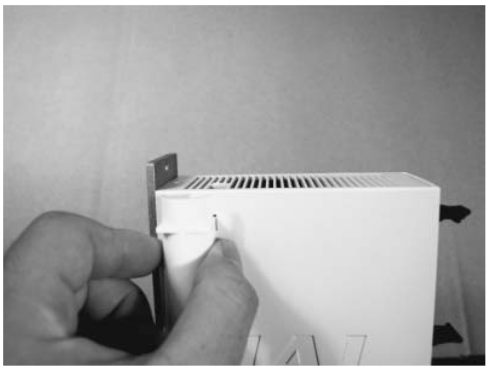
(2) Variable monitor
The contents of global integer variables can be displayed on the panel window.
Positive integers of 1 to 999 can be displayed.
Parameter settings
Other parameter No. 49 = 2
Other parameter No. 50 = Variable number of the global integer variable to be monitored

When data is written to the flash ROM or a software reset (restart) is executed after the parameter values have been input, the panel window will show the content of the global integer variable, instead of “ready status” or “program run number.” The far-left segment digit should read “U.”
Display example)

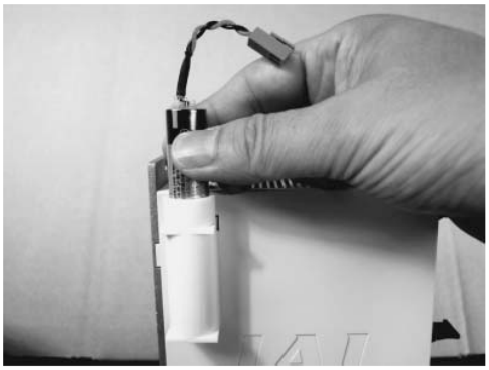




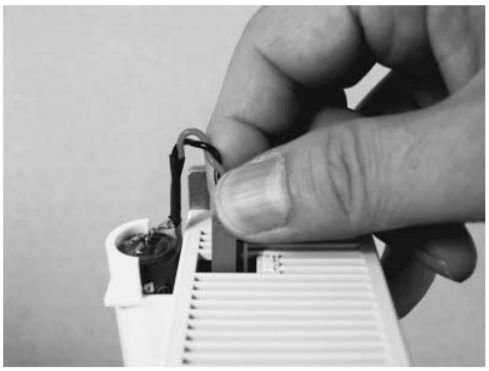
6.8 Installing the System-memory Backup Battery (Optional)



As shown to the left, install the supplied battery holder on the left side face of the controller.



Insert the battery into the holder.



Connect the battery connector.
Pay attention to the connector orientation.
(The connector hook should face the right side.)

Chapter 4 Operation

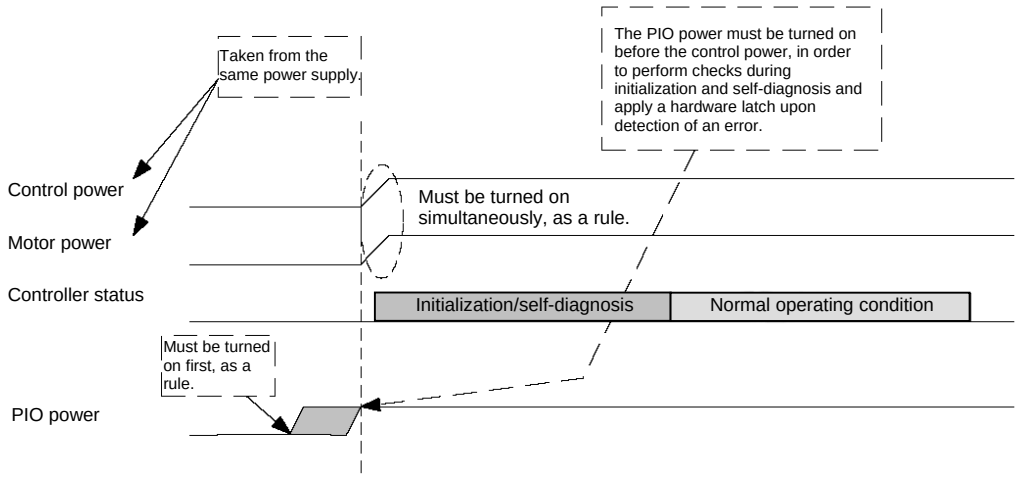
1. Startup

(1) Connect the motor cable and encoder cable to the controller.
(2) Connect the PIO connector to the host PLC using the supplied flat cable.
(3) Execute an emergency stop.
(4) Connect the PC or teaching pendant.
Set the AUTO/MANU switch to the “MANU” side.
(5) Supply the 24-V PIO power through the flat cable.
(6) Turn on the control power and motor power at the same time. (They should be taken from the same power supply.)
(7) Reset the emergency stop.
★ The EMG lamp turns off.
★ If the ALM lamp is lit, an error is present. Check the error list to identify the problem.

If the 24-V PIO power is not supplied, an “E69” error will generate.
To check for errors, connect the teaching pendant, PC software or panel unit.

1.1 Power ON Sequence

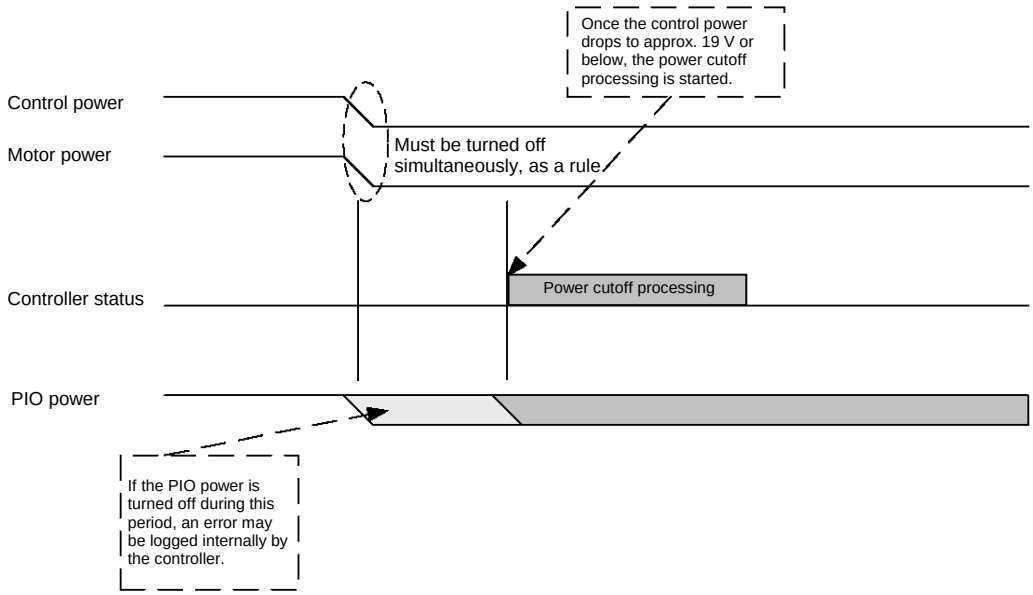
- Although separate inputs are provided for the control power and motor power, they should be supplied from the same power-supply terminal.
- Turn on the PIO power first. You can turn on the PIO power much earlier than the control power and motor power, as long as it is turned on before the control power/motor power.



* If the PIO power is not turned on before the control power is turned on, an error will be detected.

1.2 Power Cutoff Sequence

- If the PIO power is turned off before the control power and motor power (before the power cutoff processing is performed), a PIO power error may be logged internally by the controller.
- The PIO power can be turned off much later than the control power and motor power, as long as it is turned off after the control power/motor power.



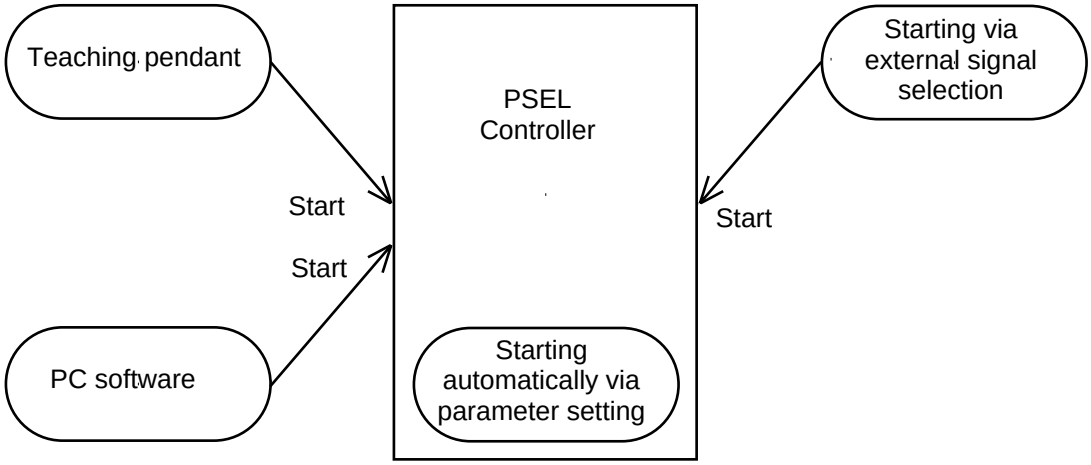
2. How to Start a Program

With the PSEL Controller, the stored programs can be started (run) using four methods. Of these methods, two are mainly used to debug programs or perform trial operations, while the remaining two are used in general applications on site.

The former two methods are “starting from the teaching pendant” and “starting from the PC software.”

These methods provide simple means of checking the operation. For details on “starting from the teaching pendant,” read the operation manual for the optional teaching pendant. For “starting from the PC software,” read the applicable explanation in the manual supplied with the PC software.

The latter two methods are “starting automatically via parameter setting” and “starting via external signal selection.” This chapter only explains the methods for “starting automatically via parameter setting” and “starting via external signal selection.”



2.1 Starting a Program by Auto-Start via Parameter Setting

Other parameter No. 7 (Auto program start setting) = 1 (Standard factory setting)

This parameter is set using the teaching pendant or PC software.

Set an auto-start program number

Set the number of the program you wish to start automatically in other parameter No. 1 (auto-start program number).
Set the controller mode to AUTO.

⇩

Reset the controller

Reconnect the power or execute a software reset, and the controller will be reset.

⇩

Automatically starting the program

Once the controller is reset in the above step, the program of the set number will start automatically.
*

⚠ Caution

[Note on starting a program by auto-start]
The automatic operation will begin immediately after the controller is reset, so the user may be surprised by unexpected movements of the equipment, particularly those caused by a sudden activation of the servo actuator. To ensure safety, always provide an interlocking function, such as allowing the program execution to proceed only after receiving a confirmation signal at the beginning of the program.

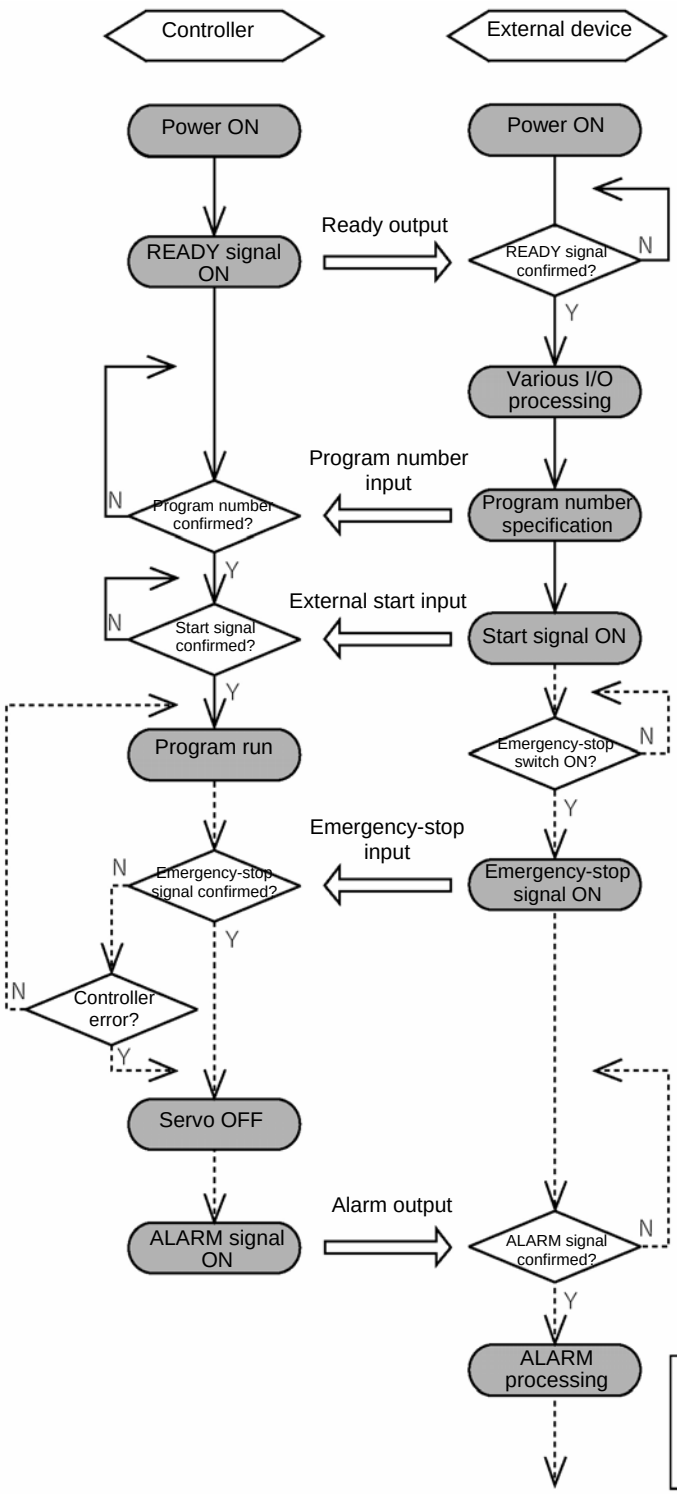
If you wish to start multiple programs at the same time, write multiple “EXPG” commands at the beginning of the main program to start the remaining programs. Provide safety measures for each program to be started.

* If the following setting is performed, the program of the selected program number will start automatically at the ON edge of the signal received by the selected input port. The program will be aborted at the OFF edge.
You can set a desired input port for receiving the auto program start signal (dedicated function).
Set the input function setting value “5” in the I/O parameter corresponding to the desired input port number (Nos. 30 through 45, 251 through 258).
(Refer to “I/O Function Lists” and “I/O Parameters.”)

2.2 Starting via External Signal Selection

Select a desired program number externally and then input a start signal.

(1) Flow chart



When the READY signal (Output port No. 301) turns ON, the RDY lamp (green) on the controller front panel will illuminate.

Input a desired program number as a BCD code from the external device (Input port Nos. 16 through 22).

Input a start signal (input port No. 0) from the external device.

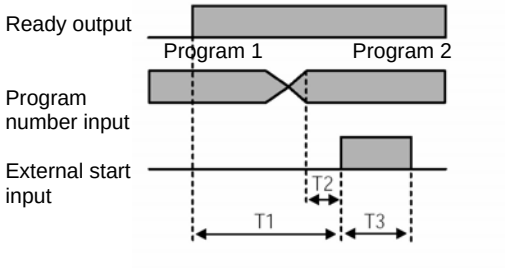
If the optional panel unit is connected, the CODE display area indicates the program number of each program that has been started.

If an emergency-stop signal was input from the external device or a controller error occurred, the controller will turn off the servo power. (The RDY lamp will turn off.)

Note) The assignments of dedicated input/output port functions (such as RDY output start signal) reflect the factory settings.

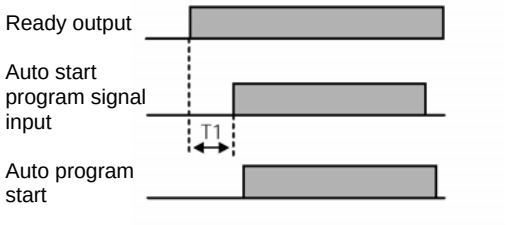
(2) Timing chart

[1] Starting a program



T1: Duration after the ready output turns ON until input of external start signal is permitted
T1 = 10 msec min.
T2: Duration after the program number is input until input of external start signal is permitted
T2 = 50 msec min.
T3: Input duration of external start signal
T3 = 100 msec min.

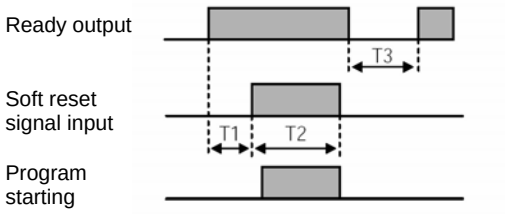
[2] Starting a program by auto start
* Set the input function specification value “5” (auto program start signal) for input port No. *.



T1: Time after the ready output turns ON until the auto start program signal can be input to input port No. *.
T1 = 10 msec min.

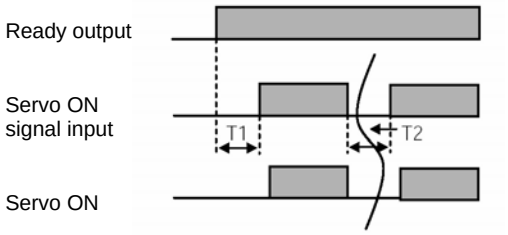
* Auto program start:
Set “0” for the auto start program setting of the other parameter No. 7.

[3] Soft reset signal
* Set the input function specification value “3” (soft reset signal) for input port No. *.



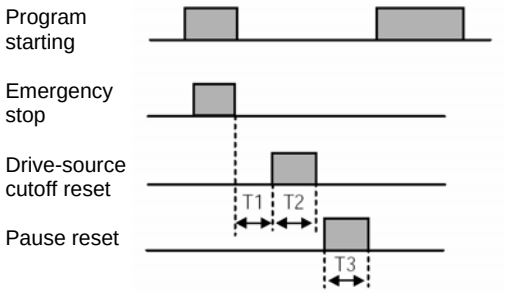
T1: Time after the ready output turns ON until the input function specification value “3” (soft reset signal) can be input to input port No. *.
T1 = 10 msec min.
T2: Time until the soft reset signal starts functioning.
T2 = 1 sec min.
T3: Time after the soft reset signal is cancelled until the ready signal is output.

[4] Servo ON signal
* Set the input function specification value “4” (servo ON signal) for input port No. *.



T1: Time after the ready output turns ON until the input function specification value “4” (servo ON signal) can be input
T1 = 10 msec min.
T2: Interval after the servo turns OFF until it turns ON again
T2 = 1.5 sec min.

[5] When the recovery type after emergency stop or enable operation is set to “Operation continued”
* Set other parameter No. 10 to “2” and set the input function specification value “7” (operation-pause reset signal) for input port No. *.
Set the input function specification value “17“ (drive-source cutoff reset input signal) for other input port No. *.



T1: Time after the emergency stop input is reset until the drive-source cutoff reset signal can be input
T1 = 2 sec min.
T2: Drive-source cutoff reset input time
T1 = 10 msec min.
T3: Pause reset input time
T1 = 10 msec min.

3. Drive-Source Recovery Request and Operation-Pause Reset Request

(1) Drive-source recovery request

[1] Case where a drive-source request is required

A drive-source recovery request is required in the following case:

- Specify a desired input port for receiving the drive-source cutoff reset input signal (dedicated function).
  Occurrence of a drive-source cutoff factor → Recovery after the cutoff factor is removed.

[2] How to request a drive-source recovery

A drive-source recovery request can be issued using one of the following methods:

- Set the input function specification value “17” in the I/O parameter corresponding to the desired input port number (Nos. 30 through 45, 251 through 258). (Refer to “I/O Function Lists” and “I/O Parameters.”)
  Input the ON edge to the input port of the specified number.
- Select [Drive-Source Recovery Request (P)] from the [Controller (C)] menu on the PC software screen.
- Select Ctl (controller operation) and RPwr (drive-source recovery request) on the mode selection screen of the teaching pendant.

(2) Operation-pause reset request

[1] Cases where an operation-pause reset request is required

An operation-pause reset request is required in any of the following cases:

- An emergency stop was actuated during automatic operation when other parameter No. 10 was set to “2” (Emergency-stop recovery type = Continued operation) (only during automatic operation) → Recovery (reset of operation pause) after the emergency stop is reset.
- The automatic operation was stopped using the deadman switch or enable switch when other parameter No. 11 was set to “2” (Deadman/enable switch recovery type = Continued operation) (only during automatic operation) → Recovery (reset of operation pause) after the stop is reset.
- Specify a desired input port for receiving the operation-pause input signal (dedicated function). Set the input function specification value “8” in the I/O parameter corresponding to the desired input port number (Nos. 30 through 45, 251 through 258). (Refer to “I/O Function Lists” and “I/O Parameters.”)
  OFF level signal input is received by the import port of the specified number during auto operation (operations pause) → Recovery after detection of ON signal level by the input port (operation pause is reset).

[2] How to request an operation-pause reset

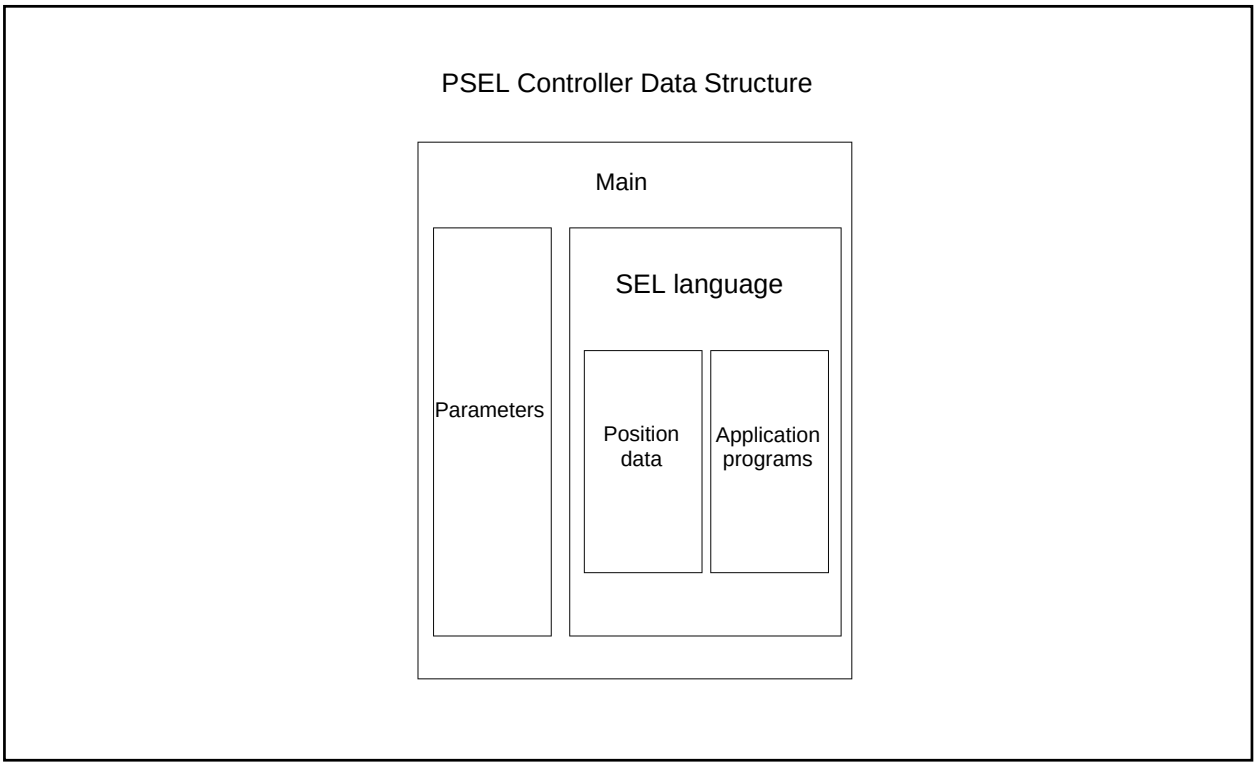
An operation-pause reset request can be issued using one of the following methods:

- Specify a desired input port for receiving the operation-pause input signal (dedicated function). Set the input function specification value “7” in the I/O parameter corresponding to the desired input port number (Nos. 30 through 45, 251 through 258). (Refer to “I/O Function Lists” and “I/O Parameters.”)
  Input the ON edge to the input port of the specified number.
- Select [Operation-Pause Reset Request (L)] from the [Controller (C)] menu on the PC software screen.
- Select Ctl (controller operation) and RAct (operation-pause reset request) on the mode selection screen of the teaching pendant.

* If the case in [1] of (1) and any of the cases in [1] of (2) are present at the same time, a drive-source recovery request must be issued first, followed by an operation-pause reset request.

4. Controller Data Structure

The controller data consists of parameters as well as position data and application programs used to implement SEL language.



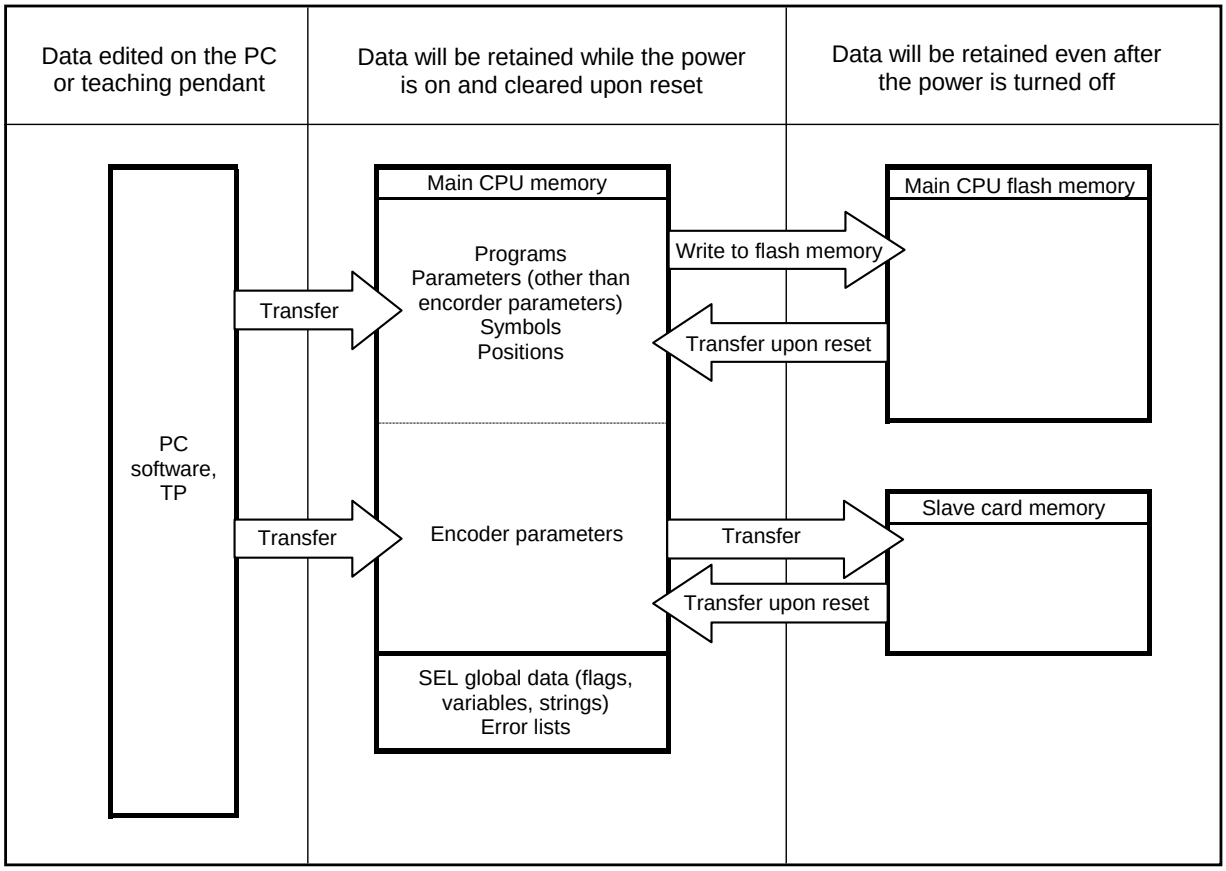
The user must create position data and application programs. The parameters are predefined, but their settings can be changed in accordance with the user's system.
Refer to Appendix, "List of Parameters," for details on the parameters.

4.1 How to Save Data

The flow to save data in the PSEL controller is illustrated below.
When data is transferred from the PC software or teaching pendant to the controller, the data is only written to the main CPU memory as shown in the diagram below and will be erased once the controller is powered down or reset.
For important data, always write to the flash memory so that they will not be lost.

4.1.1 Factory Settings: When the System-Memory Backup Battery is Not Used

Other parameter No. 20 = 0 (System-memory backup battery not installed)



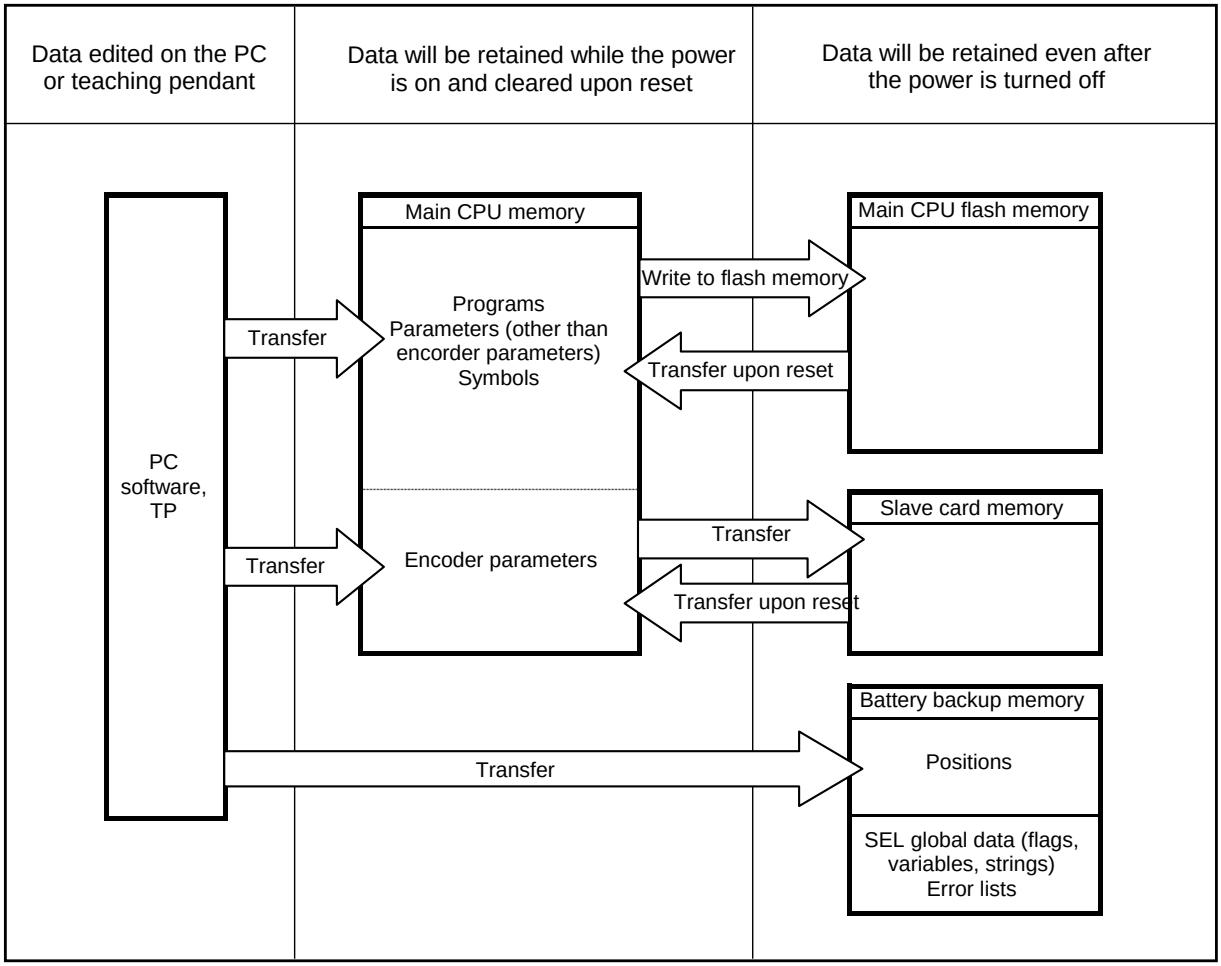
Since the programs, parameters and symbols are read from the flash memory at restart, the data in the temporary memory will remain the same as the original data before edit unless the edited data are written to the flash memory.

The controller always operates in accordance with the data in the main CPU memory (excluding the parameters).

Note: SEL global data cannot be retained if the backup battery is not installed.
SEL global data will be cleared once the control power is turned off or a software reset is executed.
The error list will be cleared once the control power is turned off.

4.1.2 When the System-Memory Backup Battery (Optional) is Used

Change the setting of other parameter No. 20 to 2 (System-memory backup battery installed).



Since the programs, parameters and symbols are read from the flash memory at restart, the data in the temporary memory will remain the same as the original data before edit unless the edited data are written to the flash memory.

The controller always operates in accordance with the data in the main CPU memory (excluding the parameters).

4.2 Points to Note

Point to note when transferring data and writing to the flash memory

Never turn off the main power while data is being transferred or written to the flash memory. The data will be lost and the controller operation may be disabled.

Point to note when saving parameters to a file

The encoder parameters are stored in the EEPROM of the actuator's encoder itself (unlike other parameters, they are not stored in the EEPROM of the controller). The encoder parameters will be read from the encoder's EEPROM to the controller when the power is turned on or upon software reset.

Therefore, if the parameters are saved to a file after turning on the controller (or restarting it via a software reset) without an actuator (encoder) connected, the encoder parameters saved to the file will become invalid.

Point to note when transferring a parameter file to the controller

When a parameter file is transferred to the controller, the encoder parameters will be transferred to the EEPROM of the encoder (excluding manufacturing/function information).

Therefore, if the parameter file transferred to the controller has been read from a controller that was started without an actuator connected, invalid encoder parameters will be written to the encoder's EEPROM (provided that an actuator is connected to the controller to which the file was transferred).

When saving the parameters to a file, do so with an actuator connected to the controller.

Chapter 5 Maintenance

- Routine maintenance and inspection are necessary so that the system will operate properly at all times. Be sure to turn off the power before performing maintenance or inspection.
- The standard inspection interval is six months to one year. If the environment warrants, however, the interval should be shortened.

1. Inspection points

- Check to see if the supply voltage to the controller is inside the specified range.
- Inspect the ventilation holes in the controller and remove dirt, dust and other foreign attachments, if any.
- Inspect the controller cables (controller → actuator) and check for any loose screws or cable disconnection.
- Check the controller mounting screws, etc., for looseness.
- Inspect each cable (axis link cable, general-purpose I/O cable, system I/O cable, power cable) for loose connection, disconnection, play, etc.

2. Spare consumable parts

Without spare parts, a failed controller cannot be repaired even when the problem is identified quickly. We recommend that you keep the following consumable parts as spares:

Consumable parts
- Cables
- System-memory backup battery: (optional): AB-5 by IAI -- Must be replaced after approx. 5 years*
  *: The actual replacement timing will vary depending on the use condition. For details, refer to "◎ Battery Backup Function" in Appendix.

When the battery voltage drops, an applicable error code will be displayed on the panel window.

Error Codes Indicating Low Battery Voltage

System-memory backup battery	A01 or A02

3. Replacement Procedure for System-Memory Backup Battery (Optional)

Backing up the system memory
If the optional system-memory backup battery is installed in the PSEL controller and “Other parameter No. 20: Backup battery installation function type” is set to “2” (Installed), the following SRAM data will be retained even after the power is turned off:

- Position data
- SEL global data (flags, integer/real variables, string variables)
- Error list

For this reason, always follow the procedure below when replacing the system-memory backup battery:

Note:	If the system-memory backup battery is removed while other parameter No. 20, “Backup battery installation function type” is set to “2 (Installed),” above contents in the SRAM will be lost. So that the position data can be restored even after the SRAM data has been cleared, save the position data to a file using the PC software before removing the battery. For information on how to save position data to a file, refer to 6, “Position Data Edit Window” in “Operation Manual for X-SEL PC Software.”

(1) Turn on the controller power.

(2) Record (write down) the current setting of “Other parameter No. 20, Backup-battery installation function type.” (This will be used when reverting the parameter to its original setting following the replacement of system-memory backup battery.)

(3) If the PC software is installed in your PC, save the position data to a file using the PC software. The data will be used as a backup in case the SRAM data saved to the flash ROM fails.

(4) Change “Other parameter No. 20, Backup-battery installation function type” to “1” and transfer the setting to the controller, and then perform a flash ROM write. (The point data will be saved to the flash ROM.)

* Confirm that the flash ROM writing process has completed.

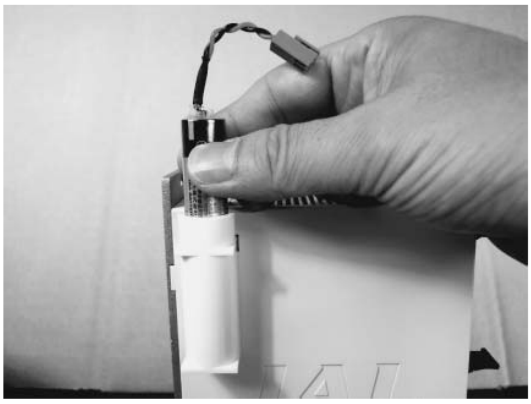
(5) Perform a software reset to restart the controller. (The SEL global data and error lists will be saved to the special area in the flash ROM.)

(6) When the controller has been restarted, turn off the power.

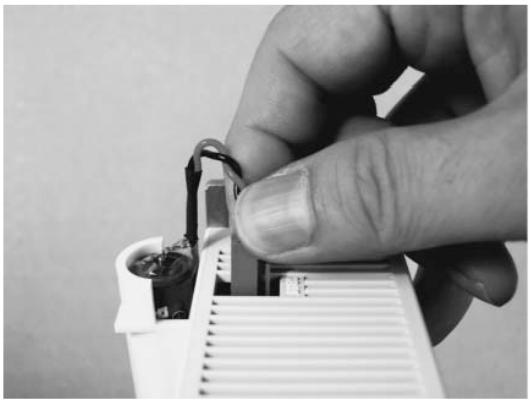
* Be sure to keep the power on from the start of controller restart until the RDY LED lamp on teh controller illuminates.

(7) Replace the system-memory backup battery. SRAM data will be lost if steps (1) through (6) are not performed properly.

Battery Replacement Procedure



[1] Remove the battery connector and pull out the battery.



[2] Insert a new battery into the holder and plug in the battery connector. The connector hook should face the right side.

(8) When the replacement of system-memory backup battery is complete, confirm that the battery is installed securely and then turn on the controller power.

(9) Revert “Other parameter No. 20, Backup-battery installation function type” to the value recorded in step (2), transfer the setting to the controller, and then perform a flash ROM write.

* Confirm that the flash ROM writing process has completed.

(10) Perform a software reset (restart the controller).
(Note) Commencing the operation without first executing a software reset or reconnecting the power may generate the following errors:
Error No. C70: ABS coordinate non-confirmation error
Error No. C6F: Home-return incomplete error

(11) When the controller has been restarted, confirm that the SRAM data have been restored.

Part 2 Programs

Chapter 1 SEL Language Data

1. Values and Symbols Used in SEL Language

1.1 List of Values and Symbols Used

The various functions required in a program are represented by values and symbols.

Function	Global range	Local range	Remarks
Input port	000 ~ 299 (300)		Varies depending on the function.
Output port	300 ~ 599 (300)		Varies depending on the function.
Flag	600 ~ 899 (300)	900 ~ 999 (100)	
Variable (integer)	200 ~ 299 (100) 1200 ~ 1299 (100)	1 ~ 99 (99) 1001 ~ 1099 (99)	99 is used for IN, INB, OUT, OUTB, etc.
Variable (real)	300 ~ 399 (100) 1300 ~ 1399 (100)	100 ~ 199 (100) 1100 ~ 1199 (100)	199 is used for PPUT, PGET, PARG, etc.
String	300 ~ 999 (700)	1 ~ 299 (299)	
Tag number		1 ~ 256 (256)	
Subroutine number		1 ~ 99 (99)	
Zone number	1 ~ 4 (4)		
Pallet number		1 ~ 10 (10)	
Axis number	1 ~ 2 (2)		Varies depending on the function.
Axis pattern	0 ~ 11		
Position number	1 ~ 1500 (1500)		
Program number	1 ~ 64 (64)		
Step number	1 ~ 2000 (2000)		
Task level	NORMAL/HIGH (2)		
SIO channel number	0 (1)		
Wait timer		1	
1-shot pulse timer		16 (Number of timers that can be operated simultaneously)	
Ladder timer		Local flag (100)	
Virtual input port (SEL system → SEL user program)	7000 ~ 7299 (300)		
Virtual output port (SEL user program → SEL system)	7300 ~ 7599 (300)		
Number of symbol definitions	500		
Number of times symbol can be used in commands	2500 (including literals)		
	Used in common from any program.	Referenced separately in each program. Cleared when the program is started.	

Caution

- Variables 99 and 199 are special variables this system uses in operations. Avoid using these two variables for general purposes.
- The values in the table represent ranges that can be processed by software. Items that require physical devices, such as I/O ports and functions relating to axis number and SIO, will be determined by possible combinations and models of commercial boards, etc., available for each device application.

- If the optional system-memory backup battery is installed, data of global variables and flags will be retained even after the controller power is turned off.
  (Other parameter No. 20 must be set to “2.” Refer to 4.1.2, “When the System Memory Backup Battery is Used” in Chapter 4 of Part 1.)

- The variables and flags in the local range will be cleared when the program is started.

- Ranges of values that can be used in SEL language
  Integers and real numbers can be used. However, pay due attention to the following limitations:

(1) Numeric data
The PSEL Controller can handle values of maximum eight digits including a sign and a decimal point.
Integer: -9,999,999 to 99,999,999
Real number: Maximum eight digits including a sign and decimal point, regardless of the size of value
Example) 999999.9, 0.123456, -0.12345
If a floating point is used in operations, the number of valid digits will be limited to seven. Also note that operations using a floating point are subject to error.

(2) Position data
The input range of position data consists of four integer digits and three decimal digits.
–9999.999 to 9999.999
(The maximum value varies depending on the actuator model.)
If position data are used in internal operations as numeric data (repeated multiplications and divisions), the accuracy of the last digit may decrease.

Consider the above limitations fully when using values. Particularly when the CPEQ command is used in a comparison operation using real numbers, a match will rarely result. In this case, the CPLE or CPGE command that looks at the magnitude relationship of two terms must be used.

1.2 I/O Ports

(1) Input ports
Used as input ports for limit switches, sensor switches, etc.

Input number assignment
000 to 023 (standard)

(2) Output ports
Used as various output ports.

Output number assignment
300 to 307 (standard)

1.3 Virtual I/O Ports

(1) Virtual input ports

Port No.	Function
7000	Always OFF
7001	Always ON
7002	Voltage low warning for system-memory backup battery
7003	Abnormal voltage of system-memory backup battery
7004	(For future expansion = Use strictly prohibited)
7005	(For future expansion = Use strictly prohibited)
7006	Top-level system error = Message level error is present
7007	Top-level system error = Operation-cancellation level error is present
7008	Top-level system error = Cold-start level error is present
7009	(For future expansion = Use strictly prohibited)
7010	Drive-source cutoff factor is present (including when waiting for cutoff reset input)
7011	Latch signal indicating that all-operation-cancellation factor is present (latch signal for recognizing 1-shot cancellation factor; latch is cancelled by 7300-ON)
7012	All-operation-pause factor is present (including when waiting for restart switch signal) (Valid only during automatic operation recognition)
7013	All-servo-axis-interlock factor is present (all-operation-pause factor + interlock input-port factor)
7014	(For future expansion = Use strictly prohibited)
7015	Voltage low warning for axis-1 absolute-data backup battery
7016	Abnormal voltage of axis-1 absolute-data backup battery (latched until power-on reset or software reset)
7017	Voltage low warning for axis-2 absolute-data backup battery
7018	Abnormal voltage of axis-2 absolute-data backup battery (latched until power-on reset or software reset)
7019 ~ 7026	(For future expansion = Use strictly prohibited)
7027 ~ 7040	(For future expansion = Use strictly prohibited)
7041 ~ 7070	(For future expansion = Use strictly prohibited)
7071	In AUTO mode
7072	During automatic operation
7073 ~ 7100	(For future expansion = Use strictly prohibited)
7101	Running program No. 01 (including during pause)
~	~
7164	Running program No. 64 (including during pause)
7165 ~ 7299	(For future expansion = Use strictly prohibited)

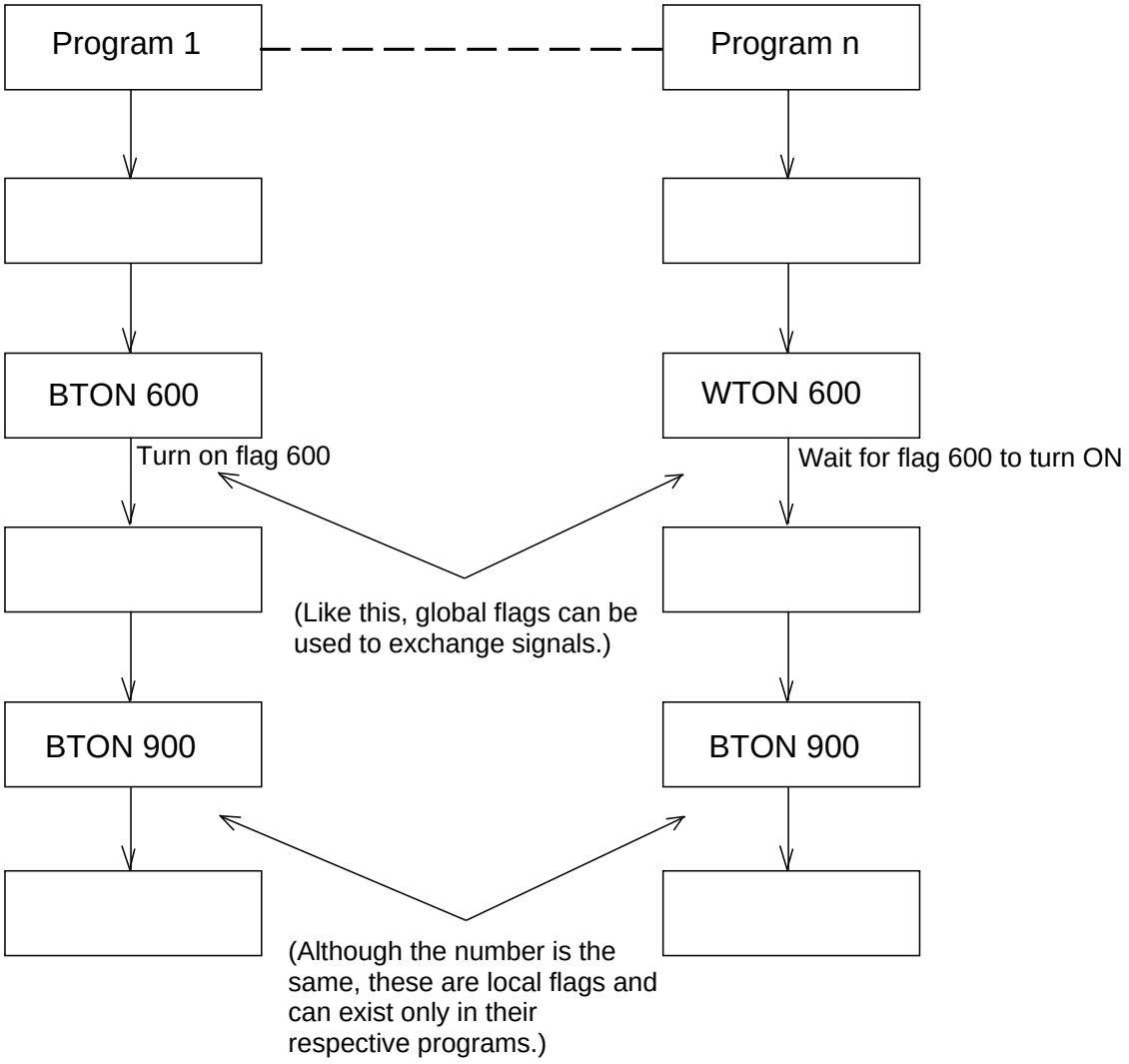
(2) Virtual output ports

Port No.	Function
7300	Latch cancellation output for a latch signal indicating that all-operation-cancellation factor is present (7011) (latch is cancelled only when operation-cancellation factor is no longer present) (7300 will be turned OFF following an attempt to cancel latch.)
7301 ~ 7380	(For future expansion = Use strictly prohibited)
7381 ~ 7399	(For future expansion = Use strictly prohibited)
7400 ~ 7599	(For future expansion = Use strictly prohibited)

1.4 Flags

Contrary to its common meaning, the term “flag” as used in programming means “memory.” Flags are used to set or reset data. They correspond to “auxiliary relays” in a sequencer.
Flags are divided into global flags (Nos. 600 to 899) that can be used in all programs, and local flags (Nos. 900 to 999) that can be used only in each program.
Global flags will be retained (backed up by battery) even after the power is turned off.
Local flags will be cleared when the power is turned off.

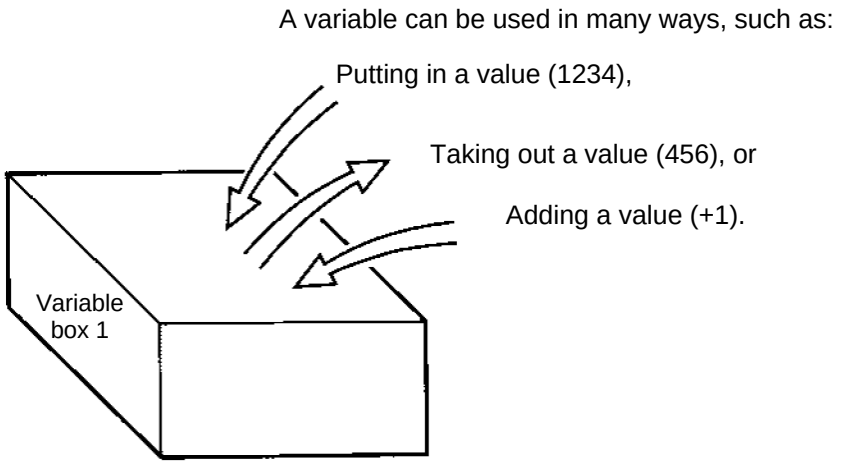
Flag number	600 ~ 899	Can be used in all programs	“Global flags”
Flag number	900 ~ 999	Used only in each program	“Local flags”



1.5 Variables

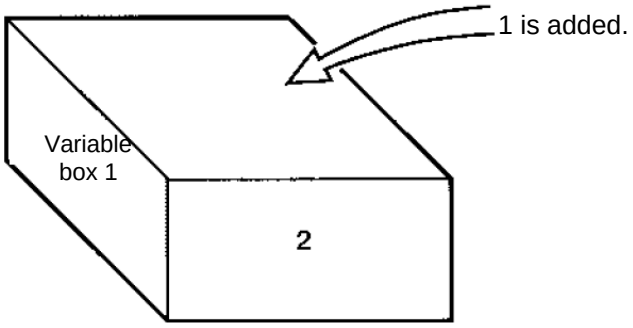
(1) Meaning of variable

"Variable" is a technical term used in software programming. Simply put, it means "a box in which a value is put." Variables can be used in many ways, such as putting in or taking out a value and performing addition or subtraction.



Command	Operand 1	Operand 2
ADD	1	1

If this command is applied to variable box 1, which already contains 2, then 1 will be added to the current value and 3 will result.

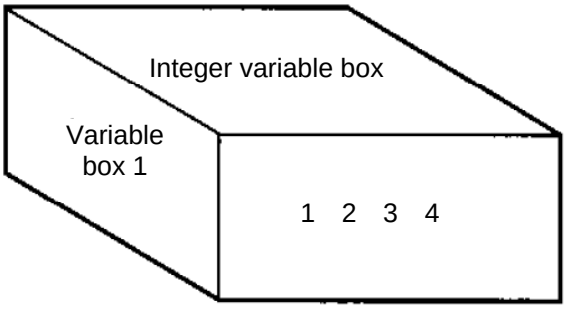


(Already contains 2)

(2) Types of variables

Variables are classified into two types, as follows:

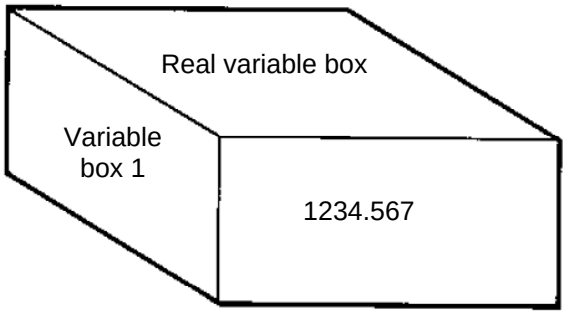
[1] Integer variables
These variables cannot handle decimal places.
[Example] 1234



Integer variable number	200 ~ 299 1200 ~ 1299	Can be used in all programs	“Global integer variables”
Integer variable number	1 ~ 99 1001 ~ 1099	Used only in each program	“Local integer variables”

Caution Integer 99 is a special register this system uses in integer operations. Any value in the range from –9,999,999 to 99,999,999 can be input in programs.

[2] Real variables
Actual values. These variables can handle decimal places.
[Example] 1234.567
(Decimal point)

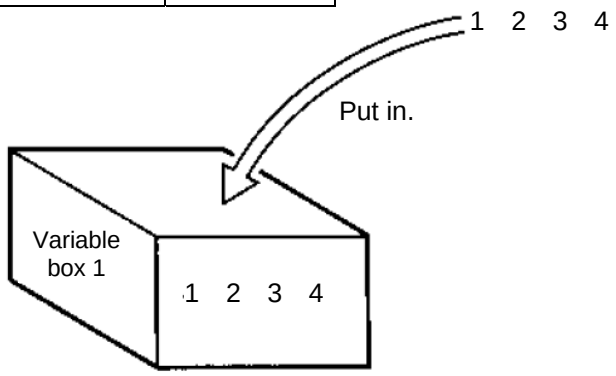


Real variable number	300 ~ 399 1300 ~ 1399	Can be used in all programs	“Global real variables”
Real variable number	100 ~ 199 1100 ~ 1199	Used only in each program	“Local real variables”

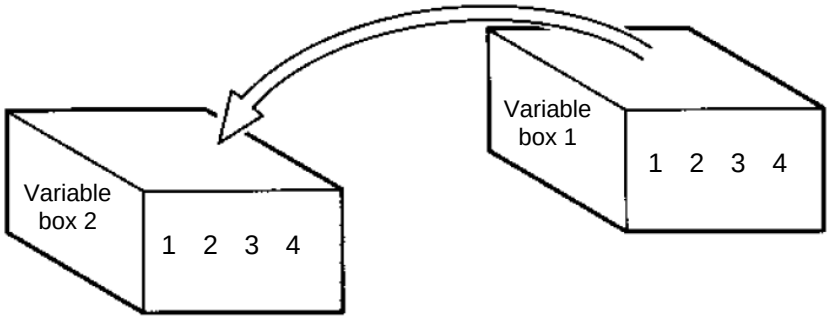
Caution Real number 199 is a special register this system uses in real-number operations. Any value in the range from –99,999.9 to 999,999.9 (eight digits including a sign) can be input in programs.

[3] Variables with “*” (asterisk) (indirect specification)
An “*” (asterisk) is used to specify a variable.
In the following example, the content of variable box 1 will be put in variable box 2. If variable box 1 contains “1234,” then “1234” will be put in variable box 2.

Command	Operand 1	Operand 2
LET	1	1234



Command	Operand 1	Operand 2
LET	2	*1



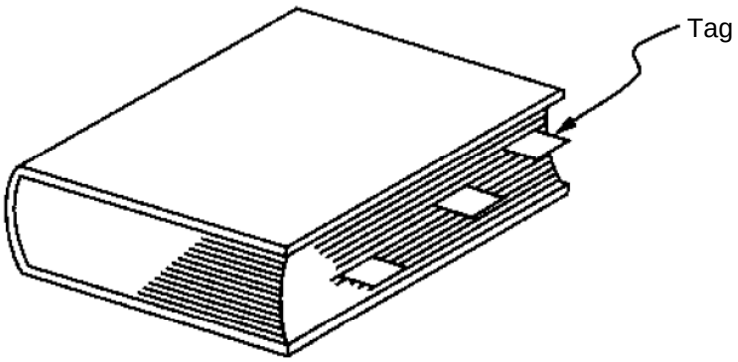
The above use of variables is called “indirect specification.”

An “*” is also used when indirectly specifying a symbol variable (refer to 1.8, “Symbols”).

Command	Operand 1	Operand 2	
LET	ABC	1	Put 1 in variable ABC.
LET	BCD	2	Put 2 in variable BCD.
ADD	ABC	*BCD	Add the content of variable BCD, or 2, to variable ABC. (The content of variable ABC becomes 3.)

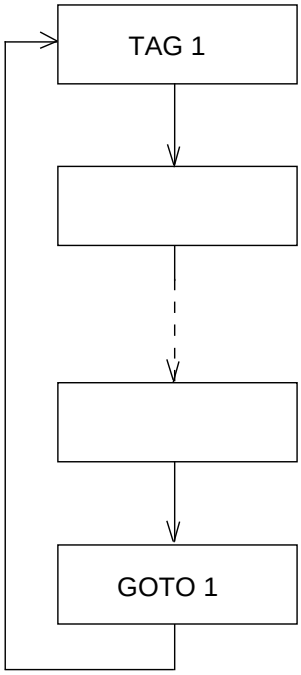
1.6 Tags

The term “tag” means “heading.”
Tags are used in the same way you attach labels to the pages in a book you want to reference frequently.
A tag is a destination specified in a jump command “GOTO.”



Command	Operand 1
TAG	Tag number (Integer between 1 and 256)

They are used only in each program.



1.7 Subroutines

By taking out the parts of a program that are used repeatedly and registering them as “subroutines,” the same processing can be performed with fewer steps. (A maximum of 15 nests are accommodated.)

They are used only in each program.

Command	Operand 1
EXSR	Subroutine number (Integer between 1 and 99; variable is also supported)

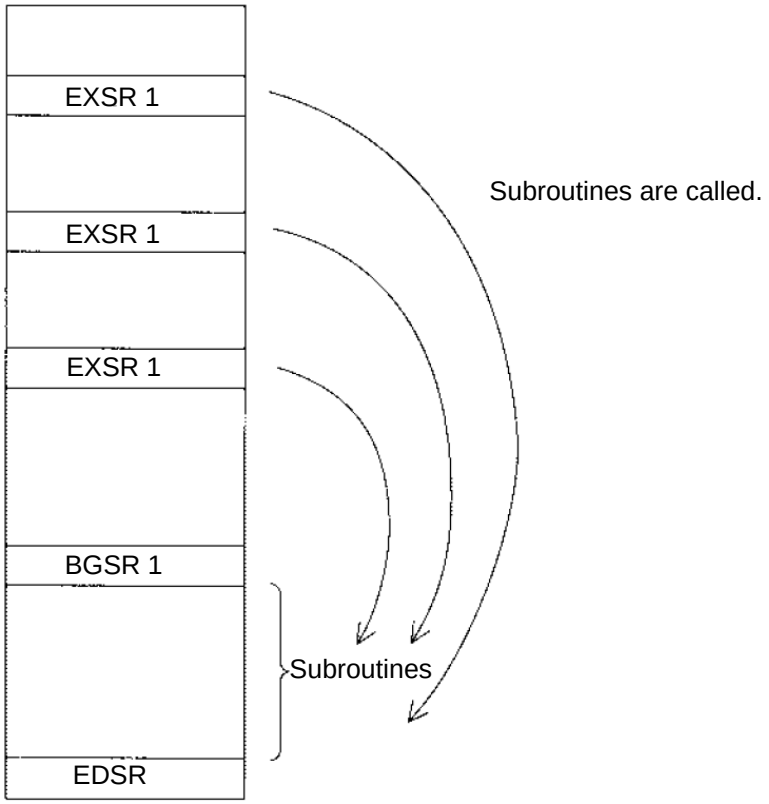
Subroutine execution command

Command	Operand 1
BGSR	Subroutine number (Integer between 1 and 99)

Subroutine start declaration

Command	Operand 1
EDSR	—

Subroutine end declaration



1.8 Symbols

In the PSEL Controller, values such as variable numbers and flag numbers can be handled as symbols. For the method to edit symbols, refer to “Editing Symbols” in the operation manual for PSEL teaching pendant or “Symbol Edit Window” in the operation manual for PSEL PC software.

(1) Supported symbols
The following items can be expressed using symbols:
Variable number, flag number, tag number, subroutine number, program number, position number, input port number, output port number, axis number, constant

(2) Description rules of symbols
[1] A maximum of nine single-byte alphanumeric characters or underscore starting with an alphabet (Note: The length of a character-string literal must not exceed eight single-byte characters.)
* Exercise caution that the same ASCII code may be expressed differently between the PC software and the teaching pendant because of the different fonts used by the two. (The same applies to character-string literals.)
5Ch --- PC software: Backslash \ (overseas specifications, etc.)
Teaching pendant: Yen mark ¥
7Eh --- PC software: ~
Teaching pendant: Right arrow →
[2] Symbols of the same name must not be defined within each function. (The same local symbol can be used in different programs.)
[3] Symbols of the same name must not be defined within the flag number, input-port number or output-port number group. (The same local symbol can be used in different programs.)
[4] Symbols of the same name must not be defined within the integer-variable number or real-variable number group. (The same local symbol can be used in different programs.)
[5] Symbols of the same name must not be defined within the integer constant or real constant group.

(3) Number of symbols that can be defined: Maximum 500

(4) Number of times symbols can be used in all SEL programs: Maximum 2500 times including character-string literals
* If symbol is used in all of the input condition, operand 1, operand 2 and output fields, it is deemed that symbol is used four times in one step.

1.9 Character-String Literals

Character-string literals are used in certain string-operation commands and consist of the portion enclosed by single quotation marks (‘ ‘) (maximum eight single-byte characters).
With the PC software, single-byte ASCII code characters from 20h to 7Eh (limited to those that can be input via keyboard) can be used inside the single quotation marks. With the teaching pendant, single-byte alphanumeric characters and single-byte underscores can be used.

1.10 Axis Specification

Axes can be specified based on axis number or axis pattern.

(1) Axis numbers and how axes are stated
Each of multiple axes is stated as follows:

Axis number	How axis is stated
1	Axis 1
2	Axis 2

The axis numbers stated above can also be expressed using symbols.

Use axis number if you wish to specify only one of multiple axes.

- Commands that use axis specification based on axis number
  BASE, PPUT, PGET, ACHZ, AXST, PASE, PARG, PRDQ, ECMD (1.5)

(2) Axis pattern
Whether or not each axis will be used is indicated by “1” or “0.”

	(Upper)	(Lower)
Axis number	Axis 2	Axis 1
Used	1	1
Not used	0	0

[Example] When axes 1 and 2 are used

Axis 2
↓
11
↑
Axis 1

[Example] When axes 2 is used

Axis 2
↓
10 (In this case, the 0s are needed to indicate the position of axis 2.)

Indirect specification of axis pattern in a variable
The axis pattern is considered a binary value, and a converted decimal value is assigned to a variable.

[Example] To perform home return for axis 2 only, you can specify as follows based on axis pattern:

```
HOME    10
```

In indirect specification, 10 (binary) is expressed as 2 (decimal), so the same operation can be specified as follows:

```
LET     6   2
HOME    *6
```

If you must select and specify multiple axes at the same time, use axis pattern.

- Commands that use axis specification based on axis pattern
  OFST, GRP, SVON, SVOF, HOME, JFWN, JFWF, JBWN, JBWF, STOP, PTST, PRED
  CHVL, PBND, WZNA, WZNO, WZFA, WZFO, MOVD, MVDI, PTRQ

SEL language consists of a position part (position data = coordinates, etc.) and a command part (application program).

2. Position Part

As position data, coordinates, speeds, accelerations and decelerations are set and stored.

± 2000000.000 mm

*1, 2
1 ~ 2000/mmsec

*2
Standard
0.3 G

*2
Standard
0.3 G

Position No.	Axis 1	Axis 2	Speed	Acceleration	Deceleration
1					
2					
3					
⋮					
1498					
1499					
1500					

*1 Varies depending on the actuator model.
*2 If speed, acceleration or deceleration is set in the position data, the setting will be given priority over the corresponding data set in the application program. Leave the position data fields empty if you wish to enable the corresponding data in the application program.

Priority	Speed	Acceleration (deceleration)
1	Value of position data set in operand 1	Value of position data set in operand 1
2	Value set by a VEL command	Value set by an ACC (DCL) command
3		All-axis parameter No. 11, “Default acceleration” (All-axis parameter No. 12, “Default deceleration”)

Values pertaining to a rotating axis are processed in degrees instead of millimeters.

If axis-specific parameter No. 1 (axis operation type) is set to “1” (rotational movement axis (angle control)) for a given axis, all millimeter values pertaining to that axis (including parameters, etc.) will be processed in degrees.

If the gear ratio parameters (axis-specific parameter Nos. 50 and 51) are set correctly, the angles (deg) will represent those of the body of rotation at the end.

Example)	Distance	1 mm → 1 deg
	Speed	1 mm/sec → 1 deg/sec
	Acceleration/deceleration	1 G = 9807 mm/sec^2 → 9807 deg/sec^2

3. Command Part

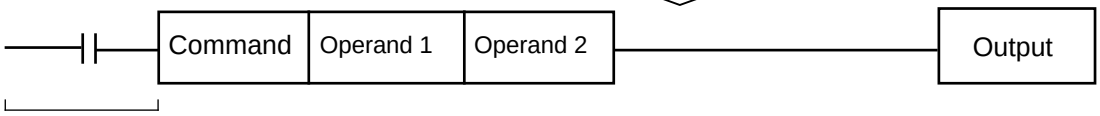
The primary feature of SEL language is its very simple command structure. Since the structure is simple, there is no need for a compiler (to translate into computer language) and high-speed operation is possible via an interpreter (the program runs as commands are translated).

3.1 SEL language Structure

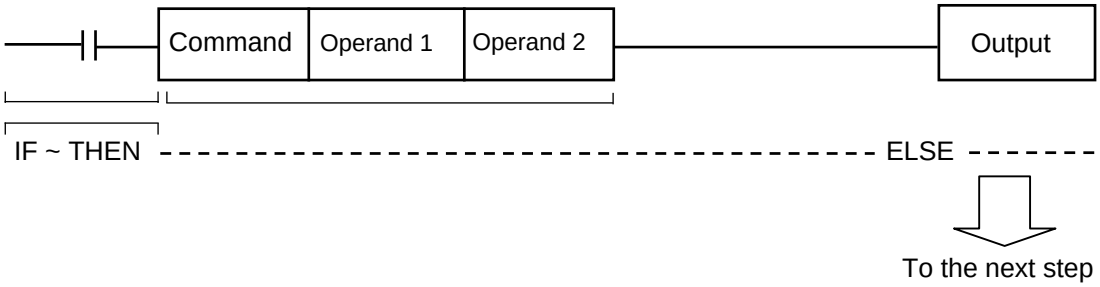
The table below shows the structure of one command step.

Extension condition (AND, OR)	Input condition (I/O, flag)	Command, declaration			Output (Output port, flag)
		Command, declaration	Operand 1	Operand 2	

Using a ladder diagram, this is expressed as follows:

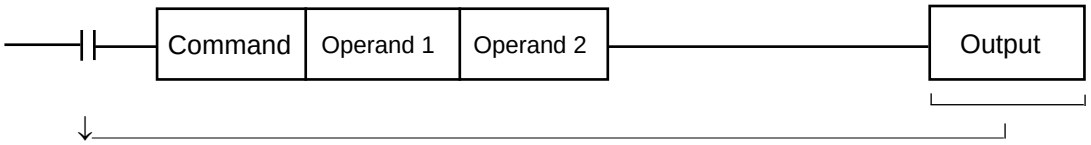


(1) The condition before the command is equivalent to “IF ~ THEN…” in BASIC.



1. If the input condition is satisfied, the command will be executed. If there is an output specification, the specified output port will be turned ON. If the input condition is not satisfied, the program will proceed to the next step regardless of the command that follows (e.g., WTON, WTOF). Obviously nothing will happen at the output port, but caution must be exercised.
2. If no condition is set, the command will be executed unconditionally.
3. To use the condition in reverse logic (so-called “contact b logic”), add "N" (NOT) to the condition.
4. The input condition supports input port, output port and flag.
5. The operand 1, operand 2 and output fields can be specified indirectly.

(2) The output field, which follows the command, operand 1 and operand 2 fields, will specify the following action:

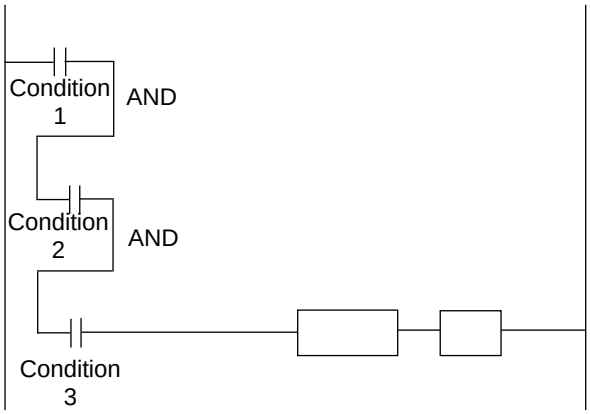


1. In the case of a control command relating to actuator operation, etc., the output will turn OFF the moment the execution of command is started, and turn ON when the execution is completed. In the case of a calculation operation command, etc., the output will turn ON if the result corresponds to a certain value, and turn OFF if not.
2. The output field supports output port and flag.

3.2 Extension Condition

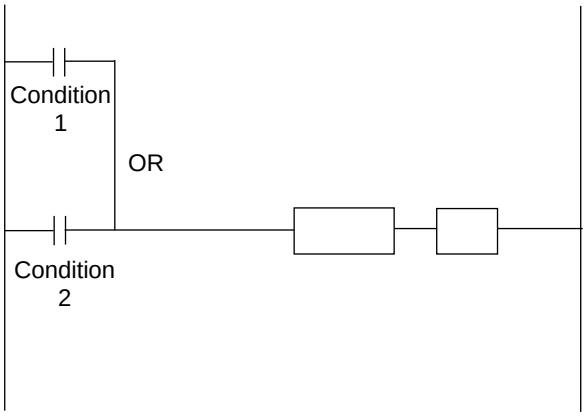
Conditions can be combined in a complex manner.

AND extension (Ladder diagram) (SEL language)



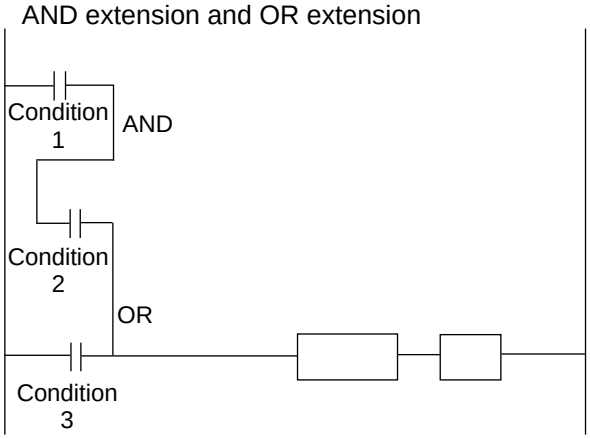
Extension condition	Input condition	Command			Output
		Command	Operand 1	Operand 2	
	Condition 1				
A	Condition 2				
A	Condition 3	Command	Operand 1	Operand 2	

OR extension



Extension condition	Input condition	Command			Output
		Command	Operand 1	Operand 2	
	Condition 1				
O	Condition 2	Command	Operand 1	Operand 2	

AND extension and OR extension



Extension condition	Input condition	Command			Output
		Command	Operand 1	Operand 2	
	Condition 1				
A	Condition 2				
O	Condition 3	Command	Operand 1	Operand 2	

Chapter 2 List of SEL Language Command Codes

1. By Function

Variables can be specified indirectly in the operand 1, operand 2 and output fields.
Symbols can be input in the condition, operand 1, operand 2 and output fields.
The input items in ( ) under operand 1 and operand 2 are optional.

Once an “actuator control declaration” command is executed in a program, the command will remain valid as long as the program is running. To change the values (in operand 1, operand 2, etc.) already set by the “actuator control declaration” command, the necessary parts of the program must be set again. In other words, the values set by the last executed command will prevail.
The output field will be turned OFF when the command is executed. Once the execution is completed, the output field may be turned ON depending on the operation type condition in the output field. (The output field will remain OFF if the condition is not satisfied.)
Note: The output field of a comparison command CP□□ (CPEQ, CPNE, CPGT, CPGE, CPLT and CPLE) will not be turned OFF when the command is executed.

Operation type in the output field
CC: Command was executed successfully,
ZR: Operation result is zero, PE: Operation is complete,
CP: Command part has passed, TU: Time up

EQ: Operand 1 = Operand 2, NE: Operand 1 ≠ Operand 2,
GT: Operand 1 > Operand 2, GE: Operand 1 ≥ Operand 2,
LT: Operand 1 < Operand 2, LE: Operand 1 ≤ Operand 2

Category	Condition	Command	Operand 1	Operand 2	Output	Function	Page
Variable assignment	Optional	LET	Assignment variable	Assigned value	ZR	Assign	85
	Optional	TRAN	Copy-destination variable	Copy-source variable	ZR	Copy	86
	Optional	CLR	Start-of-clear variable	End-of-clear variable	ZR	Clear variable	87
Arithmetic operation	Optional	ADD	Augend variable	Addend	ZR	Add	88
	Optional	SUB	Minuend variable	Subtrahend	ZR	Subtract	88
	Optional	MULT	Multiplicand variable	Multiplier	ZR	Multiply	89
	Optional	DIV	Dividend variable	Divisor	ZR	Divide	89
	Optional	MOD	Remainder assignment variable	Divisor	ZR	Calculate remainder	90
Function operation	Optional	SIN	Sine assignment variable	Operand [radian]	ZR	Sine	91
	Optional	COS	Cosine assignment variable	Operand [radian]	ZR	Cosine	91
	Optional	TAN	Tangent assignment variable	Operand [radian]	ZR	Tangent	92
	Optional	ATN	Inverse-tangent assignment operation	Operand	ZR	Inverse tangent	92
	Optional	SQR	Root assignment variable	Operand	ZR	Root	93
Logical operation	Optional	AND	AND operand variable	Operand	ZR	Logical AND	94
	Optional	OR	OR operand variable	Operand	ZR	Logical OR	95
	Optional	EOR	Exclusive-OR operand variable	Operand	ZR	Logical exclusive-OR	96
Comparison	Optional	CP□□	Comparison variable	Comparison value	EQ NE GT GE LT LE	Compare	97
Timer	Optional	TIMW	Wait time (sec)	Prohibited	TU	Wait	98
	Optional	TIMC	Program number	Prohibited	CP	Cancel waiting	99
	Optional	GTTM	Time assignment variable	Prohibited	CP	Get time	100
I/O, flag operation	Optional	BT□□	Start output, flag	(End output, flag)	CP	Output, flag [ON, OF, NT]	101
	Optional	BTPN	Output port, flag	Timer setting	CP	Output ON pulse	102
	Optional	BTPF	Output port, flag	Timer setting	CP	Output OFF pulse	103
	Optional	WT□□	I/O, flag	(Wait time)	TU	Wait for I/O, flag [ON, OF]	104
	Optional	IN	Head I/O, flag	End I/O, flag	CC	Input binary (32 bits max.)	105
	Optional	INB	Head I/O, flag	Conversion digits	CC	Input BCD (8 digits max.)	106
	Optional	OUT	Head output, flag	End I/O, flag	CC	Output binary (32 bits max.)	107
	Optional	OUTB	Head output, flag	Conversion digits	CC	Output BCD (8 digits max.)	108
	Optional	FMIO	Format type	Prohibited	CP	Set IN (B)/OUT (B) command format	109

Operation type in the output field
CC: Command was executed successfully, ZR: Operation result is zero,
PE: Operation is complete, CP: Command part has passed, TU: Time up
EQ: Operand 1 = Operand 2, NE: Operand 1 ≠ Operand 2,
GT: Operand 1 > Operand 2, GE: Operand 1 ≥ Operand 2,
LT: Operand 1 < Operand 2, LE: Operand 1 ≤ Operand 2

Category	Condition	Command	Operand 1	Operand 2	Output	Function	Page
Program control	Optional	GOTO	Jump-destination tag number	Prohibited	CP	Jump	112
	Prohibited	TAG	Declaration tag number	Prohibited	CP	Declare jump destination	112
	Optional	EXSR	Execution subroutine number	Prohibited	CP	Execute subroutine	113
	Prohibited	BGSR	Declaration subroutine number	Prohibited	CP	Start subroutine	113
	Prohibited	EDSR	Prohibited	Prohibited	CP	End subroutine	114
Task management	Optional	EXIT	Prohibited	Prohibited	CP	End program	115
	Optional	EXPG	Execution program number	(Execution program number)	CC	Start program	116
	Optional	ABPG	Stop program number	(Stop program number)	CC	Stop other program	117
	Optional	SSPG	Pause program number	(Pause program number)	CC	Pause program	118
	Optional	RSPG	Resumption program number	(Resumption program number)	CC	Resume program	119
Position operation	Optional	PGET	Axis number	Position number	CC	Assign position to variable 199	120
	Optional	PPUT	Axis number	Position number	CP	Assign value of variable 199	121
	Optional	PCLR	Start position number	End position number	CP	Clear position data	122
	Optional	PCPY	Copy-destination position number	Copy-source position number	CP	Copy position data	123
	Optional	PRED	Read axis pattern	Save-destination position number	CP	Read current axis position	124
	Optional	PRDQ	Axis number	Variable number	CP	Read current axis position (1 axis direct)	125
	Optional	PTST	Confirmation axis pattern	Confirmation position number	CC	Confirm position data	126
	Optional	PVEL	Speed [mm/sec]	Assignment-destination position number	CP	Assign position speed	127
	Optional	PACC	Acceleration [G]	Assignment-destination position number	CP	Assign position acceleration	128
	Optional	PDCL	Deceleration [G]	Assignment-destination position number	CP	Assign position deceleration	129
	Optional	PAXS	Axis-pattern assignment variable number	Position number	CP	Read axis pattern	130
	Optional	PSIZ	Size assignment variable number		CP	Confirm position size	131
	Optional	GVEL	Variable number	Position number	CP	Get speed data	132
	Optional	GACC	Variable number	Position number	CP	Get acceleration data	133
	Optional	GDCL	Variable number	Position number	CP	Get deceleration data	134
Actuator control declaration	Optional	VEL	Speed [mm/sec]	Prohibited	CP	Set speed	135
	Optional	OVRD	Speed ratio [%]	Prohibited	CP	Set speed coefficient	136
	Optional	ACC	Acceleration [G]	Prohibited	CP	Set acceleration	137
	Optional	DCL	Deceleration [G]	Prohibited	CP	Set deceleration	138
	Optional	SCRV	Ratio [%]	Prohibited	CP	Set sigmoid motion ratio	139
	Optional	OFST	Setting axis pattern	Offset value [mm]	CP	Set offset	140
	Optional	DEG	Division angle [deg]	Prohibited	CP	Set division angle	141
	Optional	BASE	Reference axis number	Prohibited	CP	Set reference axis	142
	Optional	GRP	Valid axis pattern	Prohibited	CP	Set group axes	143
	Optional	HOLD	(Input port to pause)	(HOLD type)	CP	Declare port to pause	144
	Optional	CANC	(Input port to abort)	(CANC type)	CP	Declare port to abort	145
	Optional	VLMX	Prohibited	Prohibited	CP	Specify VLMX speed	146
	Optional	DIS	Distance	Prohibited	CP	Set spline division distance	147
	Optional	POTP	0 or 1	Prohibited	CP	Set PATH output type	148
	Optional	PAPR	Distance	Speed	CP	Set PUSH command distance, speed	149
	Optional	QRTN	0 or 1	Prohibited	CP	Set quick-return mode	150

Operation type in the output field
CC: Command was executed successfully, ZR: Operation result is zero,
PE: Operation is complete, CP: Command part has passed, TU: Time up
EQ: Operand 1 = Operand 2, NE: Operand 1 ≠ Operand 2,
GT: Operand 1 > Operand 2, GE: Operand 1 ≥ Operand 2,
LT: Operand 1 < Operand 2, LE: Operand 1 ≤ Operand 2

Category	Condition	Command	Operand 1	Operand 2	Output	Function	Page
Actuator control command	Optional	SV□□	Operation axis pattern	Prohibited	PE	Servo [ON, OF]	151
	Optional	HOME	Home-return axis pattern	Prohibited	PE	Return to home	152
	Optional	MOVP	Destination position number	Prohibited	PE	Move to specified position	153
	Optional	MOVL	Destination position number	Prohibited	PE	Move to specified position via interpolation	154
	Optional	MVPI	Travel position number	Prohibited	PE	Move to relative position	155
	Optional	MVLI	Travel position number	Prohibited	PE	Move to relative position via interpolation	156
	Optional	MOVD	Target position	(Axis pattern)	PE	Move via direct value specification	157
	Optional	MVDI	Travel distance	(Axis pattern)	PE	Move relatively via direct value specification	158
	Optional	PATH	Start position number	End position number	PE	Move along path	159
	Optional	J□W□	Axis operation pattern	Start I/O, flag	PE	Jog [FN, FF, BN, BF]	160
	Optional	STOP	Axis stop pattern	Prohibited	CP	Decelerate and stop axis	161
	Optional	PSPL	Start position number	End position number	PE	Move along spline	162
	Optional	PUSH	Target position number	Prohibited	PE	Move by push motion	163
	Optional	PTRQ	Axis pattern	Ratio [%]	CC	Change push torque limit parameter	165
	Optional	CIR2	Passing position 1 number	Passing position 2 number	PE	Move along circle 2 (arc interpolation)	166
	Optional	ARC2	Passing position number	End position number	PE	Move along arc 2 (arc interpolation)	167
	Optional	CHVL	Axis pattern	Speed	CP	Change speed	168
	Optional	ARCD	End position number	Center angle [deg]	PE	Move along arc via specification of end position and center angle	169
	Optional	ARCC	Center position number	Center angle [deg]	PE	Move along arc via specification of center position and center angle	170
	Optional	PBND	Axis pattern	Distance	CP	Set positioning band	171
	Optional	CIR	Passing position 1 number	Passing position 2 number	PE	Move along circle (CIR2 is recommended)	172
	Optional	ARC	Passing position number	End position number	PE	Move along arc (ARC2 is recommended)	173
	Refer to the page on palletizing for commands relating to arch motion.						
	Optional	ARCH	Position number	Position number	PE	Arch motion	206
	Optional	ACHZ	Axis number	Prohibited	CP	Declare arch-motion Z-axis	208
	Optional	ATRG	Position number	Position number	CP	Set arch trigger	209
	Optional	OFAZ	Offset value	Prohibited	CP	Set arch-motion Z-axis offset	210
Structural IF	Optional	IF□□	Comparison variable	Comparison value	CP	Compare [EQ, NE, GT, GE, LT, LE]	174
	Optional	IS□□	Column number	Column number, character literal	CP	Compare strings	175
	Prohibited	ELSE	Prohibited	Prohibited	CP	Declare execution destination when IF command condition is not satisfied	176
	Prohibited	EDIF	Prohibited	Prohibited	CP	Declare end of IF	176
Structural DO	Optional	DW□□	Comparison variable	Comparison value	CP	Loop [EQ, NE, GT, GE, LT, LE]	177
	Optional	LEAV	Prohibited	Prohibited	CP	Pull out from DO	177
	Optional	ITER	Prohibited	Prohibited	CP	Repeat DO	178
	Prohibited	EDDO	Prohibited	Prohibited	CP	Declare end of DO	178
Multi-branching	Optional	SLCT	Prohibited	Prohibited	CP	Declare start of multi-branching	179
	Prohibited	WH□□	Comparison variable	Comparison value	CP	Branch value [EQ, NE, GT, GE, LT, LE]	180
	Prohibited	WS□□	Column number	Column number, character literal	CP	Branch character string [EQ, NE]	181
	Prohibited	OTHE	Prohibited	Prohibited	CP	Declare branching destination when condition is not satisfied	182
	Prohibited	EDSL	Prohibited	Prohibited	CP	Declare end of SLCT	182

Operation type in the output field
CC: Command was executed successfully, ZR: Operation result is zero,
PE: Operation is complete, CP: Command part has passed, TU: Time up
EQ: Operand 1 = Operand 2, NE: Operand 1 ≠ Operand 2,
GT: Operand 1 > Operand 2, GE: Operand 1 ≥ Operand 2,
LT: Operand 1 < Operand 2, LE: Operand 1 ≤ Operand 2

Category	Condition	Command	Operand 1	Operand 2	Output	Function	Page
System information acquisition	Optional	AXST	Variable number	Axis number	CP	Get axis status	183
	Optional	PGST	Variable number	Program number	CP	Get program status	184
	Optional	SYST	Variable number	Prohibited	CP	Get system status	185
Zone	Optional	WZNA	Zone number	Axis pattern	CP	Wait for zone ON, with AND	186
	Optional	WZNO	Zone number	Axis pattern	CP	Wait for zone ON, with OR	187
	Optional	WZFA	Zone number	Axis pattern	CP	Wait for zone OFF, with AND	188
	Optional	WZFO	Zone number	Axis pattern	CP	Wait for zone OFF, with OR	189
Communication	Optional	OPEN	Channel number	Prohibited	CP	Open channel	190
	Optional	CLOS	Channel number	Prohibited	CP	Close channel	190
	Optional	READ	Channel number	Column number	CC	Read from channel	191
	Optional	TMRW	Read timer setting	(Write timer setting)	CP	Set READ timeout value	193
	Optional	WRIT	Channel number	Column number	CC	Output to channel	195
	Optional	SCHA	Character code	Prohibited	CP	Set end character	196
String operation	Optional	SCPY	Column number	Column number, character literal	CC	Copy character string	197
	Optional	SCMP	Column number	Column number, character literal	EQ	Compare character strings	198
	Optional	SGET	Variable number	Column number, character literal	CP	Get character	199
	Optional	SPUT	Column number	Data	CP	Set character	200
	Optional	STR	Column number	Data	CC	Convert character string; decimal	201
	Optional	STRH	Column number	Data	CC	Convert character string; hexadecimal	202
	Optional	VAL	Variable number	Column number, character literal	CC	Convert character string data; decimal	203
	Optional	VALH	Variable number	Column number, character literal	CC	Convert character string data; hexadecimal	204
	Optional	SLEN	Character string length	Prohibited	CP	Set length	205

Operation type in the output field
CC: Command was executed successfully, ZR: Operation result is zero,
PE: Operation is complete, CP: Command part has passed, TU: Time up
EQ: Operand 1 = Operand 2, NE: Operand 1 ≠ Operand 2,
GT: Operand 1 > Operand 2, GE: Operand 1 ≥ Operand 2,
LT: Operand 1 < Operand 2, LE: Operand 1 ≤ Operand 2

Category	Condition	Command	Operand 1	Operand 2	Output	Function	Page
Palletizing-related	Optional	ARCH	Position number	Position number	PE	Arch motion	206
	Optional	ACHZ	Axis number	Prohibited	CP	Declare arch-motion Z-axis	208
	Optional	ATRG	Position number	Position number	CP	Set arch triggers	209
	Optional	OFAZ	Offset amount	Prohibited	CP	Set arch-motion Z-axis offset	210
	Optional	BGPA	Palletizing number	Prohibited	CP	Declare start of palletizing setting	211
	Prohibited	EDPA	Prohibited	Prohibited	CP	Declare end of palletizing setting	211
	Optional	PAPI	Count	Count	CP	Set palletizing counts	212
	Optional	PAPN	Pattern number	Prohibited	CP	Set palletizing pattern	212
	Optional	PASE	Axis number	Axis number	CP	Set palletizing axes	213
	Optional	PAPT	Pitch	Pitch	CP	Set palletizing pitches	213
	Optional	PAST	(Position number)	Prohibited	CP	Set palletizing reference point	214
	Optional	PAPS	Position number	Prohibited	CP	Set 3 palletizing points for teaching	215
	Optional	PSLI	Offset amount	(Count)	CP	Set zigzag	217
	Optional	PTNG	Palletizing number	Variable number	CP	Get palletizing position number	218
	Optional	PINC	Palletizing number	Prohibited	CC	Increment palletizing position number by 1	218
	Optional	PDEC	Palletizing number	Prohibited	CC	Decrement palletizing position number by 1	219
	Optional	PSET	Palletizing number	Data	CC	Set palletizing position number directly	219
	Optional	PARG	Palletizing number	Axis number	CP	Get palletizing angle	220
	Optional	PAPG	Palletizing number	Position number	CP	Get palletizing calculation data	220
	Optional	PMVP	Palletizing number	(Position number)	PE	Move to palletizing points via PTP	221
	Optional	PMVL	Palletizing number	(Position number)	PE	Move to palletizing points via interpolation	222
Building of pseudo-ladder task	Extension conditions LD (LOAD), A (AND), O (OR), AB (AND BLOCK) and OB (OR BLOCK) are supported.						
	Optional	CHPR	0 or 1	Prohibited	CP	Change task level	223
	Prohibited	TPCD	0 or 1	Prohibited	CP	Specify processing to be performed when input condition is not specified	223
	Prohibited	TSLP	Time	Prohibited	CP	Task sleep	224
	Optional	OUTR	Output, flag number	Prohibited	CP	Output relay for ladder	See 241
	Optional	TIMR	Local flag number	Timer setting	CP	Timer relay for ladder	See 241
Extended command	Optional	ECMD	1	Axis number	CC	Get motor current value	225
	Optional	ECMD	5	Axis number	CC	Get axis operation status	226
	Optional	ECMD	20	Variable number	CC	Get parameter value	227

2. Alphabetical Order

Operation type in the output field
CC: Command was executed successfully,
ZR: Operation result is zero, PE: Operation is complete,
CP: Command part has passed, TU: Time up

EQ: Operand 1 = Operand 2, NE: Operand 1 ≠ Operand 2,
GT: Operand 1 > Operand 2, GE: Operand 1 ≥ Operand 2,
LT: Operand 1 < Operand 2, LE: Operand 1 ≤ Operand 2

Command	Page	Condition	Operand 1	Operand 2	Output	Function
A						
ABPG	117	Optional	Stop program number	(Stop program number)	CC	Stop other program
ACC	137	Optional	Acceleration	Prohibited	CP	Set acceleration
ACHZ	206	Optional	Axis number	Prohibited	CP	Declare arch-motion Z-axis
ADD	88	Optional	Augend variable	Addend	ZR	Add
AND	94	Optional	AND operand variable	Operand	ZR	Logical AND
ARC	173	Optional	Passing position number	End position number	PE	Move along arc
ARC2	167	Optional	Passing position number	End position number	PE	Move along arc 2
ARCC	170	Optional	Center position number	Center angle	PE	Move along arc via specification of center position and center angle
ARCD	169	Optional	End position number	Center angle	PE	Move along arc via specification of end position and center angle
ARCH	206	Optional	Position number	Position number	PE	Arch motion
ATN	92	Optional	Inverse-tangent assignment operation	Operand	ZR	Inverse tangent
ATRG	209	Optional	Position number	Position number	CP	Set arch trigger
AXST	183	Optional	Variable number	Axis number	CP	Get axis status
B						
BASE	142	Optional	Reference axis number	Prohibited	CP	Set reference axis
BGPA	211	Optional	Palletizing number	Prohibited	CP	Declare start of palletizing setting
BGSR	113	Prohibited	Declaration subroutine number	Prohibited	CP	Start subroutine
BTPF	103	Optional	Output port, flag	Timer setting	CP	Output OFF pulse
BTPN	102	Optional	Output port, flag	Timer setting	CP	Output ON pulse
BT□□	101	Optional	Start output, flag	(End output, flag)	CP	Output, flag [ON, OF, NT]
C						
CANC	145	Optional	(Input port to abort)	(CANC type)	CP	Declare port to abort
CHPR	223	Optional	0 or 1	Prohibited	CP	Change task level
CHVL	168	Optional	Axis pattern	Speed	CP	Change speed
CIR	172	Optional	Passing position 1 number	Passing position 2 number	PE	Move along circle
CIR2	166	Optional	Passing position 1 number	Passing position 2 number	PE	Move along circle 2
CLOS	190	Optional	Channel number	Prohibited	CP	Close channel
CLR	87	Optional	Start-of-clear variable	End-of-clear variable	ZR	Clear variable
COS	91	Optional	Cosine assignment variable	Operand	ZR	Cosine
CP□□	97	Optional	Comparison variable	Comparison value	EQ/GE NE/LT GT/LE	Compare
D						
DCL	138	Optional	Deceleration	Prohibited	CP	Set deceleration
DEG	141	Optional	Division angle	Prohibited	CP	Set division angle
DIS	147	Optional	Distance	Prohibited	CP	Set spline division distance
DIV	89	Optional	Dividend variable	Divisor	ZR	Divide
DW□□	177	Optional	Comparison variable	Comparison value	CP	Loop [EQ, NE, GT, GE, LT, LE]
E						
ECMD	225	Optional	1	Axis number	CC	Get motor current value
ECMD	226	Optional	5	Axis number	CC	Get axis operation status
ECMD	227	Optional	20	Variable number	CC	Get parameter value
EDDO	178	Prohibited	Prohibited	Prohibited	CP	Declare end of DO
EDIF	176	Prohibited	Prohibited	Prohibited	CP	Declare end of IF
EDPA	211	Prohibited	Prohibited	Prohibited	CP	Declare end of palletizing setting
EDSL	182	Prohibited	Prohibited	Prohibited	CP	Declare end of SLCT

Operation type in the output field
CC: Command was executed successfully, ZR: Operation result is zero,
PE: Operation is complete, CP: Command part has passed, TU: Time up
EQ: Operand 1 = Operand 2, NE: Operand 1 ≠ Operand 2,
GT: Operand 1 > Operand 2, GE: Operand 1 ≥ Operand 2,
LT: Operand 1 < Operand 2, LE: Operand 1 ≤ Operand 2

Command	Page	Condition	Operand 1	Operand 2	Output	Function
EDSR	114	Prohibited	Prohibited	Prohibited	CP	End subroutine
ELSE	176	Prohibited	Prohibited	Prohibited	CP	Declare execution destination when IF command condition is not satisfied
EOR	96	Optional	Exclusive-OR operand variable	Operand	ZR	Logical exclusive-OR
EXIT	115	Optional	Prohibited	Prohibited	CP	End program
EXPG	116	Optional	Execution program number	(Execution program number)	CC	Start program
EXSR	113	Optional	Execution subroutine number	Prohibited	CP	Execute subroutine
F						
FMIO	109	Optional	Format type	Prohibited	CP	Set IN (B)/OUT (B) command format
G						
GACC	133	Optional	Variable number	Position number	CP	Get acceleration data
GDCL	134	Optional	Variable number	Position number	CP	Get deceleration data
GOTO	112	Optional	Jump-destination tag number	Prohibited	CP	Jump
GRP	143	Optional	Valid axis pattern	Prohibited	CP	Set group axes
GTTM	100	Optional	Time assignment variable	Prohibited	CP	Get time
GVEL	132	Optional	Variable number	Position number	CP	Get speed data
H						
HOLD	144	Optional	(Input port to pause)	(HOLD type)	CP	Declare port to pause
HOME	152	Optional	Home-return axis pattern	Prohibited	PE	Return to home
I						
IF□□	174	Optional	Comparison variable	Comparison value	CP	Compare [EQ, NE, GT, GE, LT, LE]
INB	106	Optional	Head I/O, flag	Conversion digits	CC	Input BCD (8 digits max.)
IN	105	Optional	Head I/O, flag	End I/O, flag	CC	Input binary (32 bits max.)
IS□□	175	Optional	Column number	Column number, character literal	CP	Compare strings
ITER	178	Optional	Prohibited	Prohibited	CP	Repeat DO
J						
J□W□	160	Optional	Axis operation pattern	Start I/O, flag	PE	Jog [FN, FF, BN, BF]
L						
LEAV	177	Optional	Prohibited	Prohibited	CP	Pull out from DO
LET	85	Optional	Assignment variable	Assigned value	ZR	Assign
M						
MOD	90	Optional	Remainder assignment variable	Divisor	ZR	Calculate remainder
MOVD	157	Optional	Target position	(Axis pattern)	PE	Move via direct value specification
MOVL	154	Optional	Destination position number	Prohibited	PE	Move to specified position via interpolation
MOVP	153	Optional	Destination position number	Prohibited	PE	Move to specified position
MULT	89	Optional	Multiplicand variable	Multiplier	ZR	Multiply
MVDI	158	Optional	Travel distance	(Axis pattern)	PE	Move relatively via direct value specification
MVLI	156	Optional	Travel position number	Prohibited	PE	Move to relative position via interpolation
MVPI	155	Optional	Travel position number	Prohibited	PE	Move to relative position

Operation type in the output field
CC: Command was executed successfully, ZR: Operation result is zero,
PE: Operation is complete, CP: Command part has passed, TU: Time up
EQ: Operand 1 = Operand 2, NE: Operand 1 ≠ Operand 2,
GT: Operand 1 > Operand 2, GE: Operand 1 ≥ Operand 2,
LT: Operand 1 < Operand 2, LE: Operand 1 ≤ Operand 2

Command	Page	Condition	Operand 1	Operand 2	Output	Function
O						
OFAZ	210	Optional	Offset amount	Prohibited	CP	Set arch-motion Z-axis offset
OFST	140	Optional	Setting axis pattern	Offset value	CP	Set offset
OPEN	190	Optional	Channel number	Prohibited	CP	Open channel
OR	95	Optional	OR operand variable	Operand	ZR	Logical OR
OTHE	182	Prohibited	Prohibited	Prohibited	CP	Declare branching destination when condition is not satisfied
OUT	107	Optional	Head output, flag	End I/O, flag	CC	Output binary (32 bits max.)
OUTB	108	Optional	Head output, flag	Conversion digits	CC	Output BCD (8 digits max.)
OUTR	214	Optional	Output, flag number	Prohibited	CP	Output relay for ladder
OVRD	136	Optional	Speed ratio	Prohibited	CP	Set speed ratio
P						
PACC	128	Optional	Acceleration	Assignment-destination position number	CP	Assign position acceleration
PAPG	220	Optional	Palletizing number	Position number	CP	Get palletizing calculation data
PAPI	212	Optional	Count	Count	CP	Set palletizing counts
PAPN	212	Optional	Pattern number	Set palletizing position by type	CP	Set palletizing pattern
PAPR	149	Optional	Distance	Speed	CP	Set PUSH command distance, speed
PAPS	215	Optional	Position number	Prohibited	CP	Set 3 palletizing points for teaching
PAPT	213	Optional	Pitch	Pitch	CP	Set palletizing pitches
PARG	220	Optional	Palletizing number	Axis number	CP	Get palletizing angle
PASE	213	Optional	Axis number	Axis number	CP	Set palletizing axes
PAST	214	Optional	(Position number)	Prohibited	CP	Set palletizing reference point
PATH	159	Optional	Start position number	End position number	PE	Move along path
PAXS	130	Optional	Axis-pattern assignment variable number	Position number	CP	Read axis pattern
PBND	171	Optional	Axis pattern	Distance	CP	Set positioning band
PCLR	122	Optional	Start position number	End position number	CP	Clear position data
PCPY	123	Optional	Copy-destination position number	Copy-source position number	CP	Copy position data
PDCL	129	Optional	Deceleration	Assignment-destination position number	CP	Assign position deceleration
PDEC	219	Optional	Palletizing number	Prohibited	CC	Decrement palletizing position number by 1
PGET	120	Optional	Axis number	Position number	CC	Assign position to variable 199
PGST	184	Optional	Variable number	Program number	CP	Get program status
PMVL	222	Optional	Palletizing number	Prohibited	PE	Move to palletizing points via interpolation
PMVP	221	Optional	Palletizing number	Prohibited	PE	Move to palletizing points via PTP
POTP	148	Optional	0 or 1	Prohibited	CP	Set PATH output type
PPUT	121	Optional	Axis number	Position number	CP	Assign value of variable 199
PRDQ	125	Optional	Axis number	Variable number	CP	Read current axis position (1 axis direct)
PRED	124	Optional	Read axis pattern	Save-destination position number	CP	Read current axis position
PSET	219	Optional	Palletizing number	Data	CC	Set palletizing position number directly
PSIZ	131	Optional	Size assignment variable number		CP	Confirm position size
PSLI	217	Optional	Offset amount	(Count)	CP	Set zigzag
PSPL	162	Optional	Start position number	End position number	PE	Move along spline
PTRQ	165	Optional	Axis pattern	Ratio	CC	Change push torque limit parameter
PTST	126	Optional	Confirmation axis pattern	Confirmation position number	CP	Confirm position data
PUSH	163	Optional	Target position number	Prohibited	PE	Move by push motion
PVEL	127	Optional	Speed	Assignment-destination position number	CP	Assign position speed

Operation type in the output field
CC: Command was executed successfully, ZR: Operation result is zero,
PE: Operation is complete, CP: Command part has passed, TU: Time up
EQ: Operand 1 = Operand 2, NE: Operand 1 ≠ Operand 2,
GT: Operand 1 > Operand 2, GE: Operand 1 ≥ Operand 2,
LT: Operand 1 < Operand 2, LE: Operand 1 ≤ Operand 2

Command	Page	Condition	Operand 1	Operand 2	Output	Function
Q						
QRTN	150	Optional	0 or 1	Prohibited	CP	Set quick-return mode
R						
READ	191	Optional	Channel number	Column number	CC	Read from channel
RSPG	119	Optional	Resumption program number	(Resumption program number)	CC	Resume program
S						
SCHA	196	Optional	Character code	Prohibited	CP	Set end character
SCMP	198	Optional	Column number	Column number, character literal	EQ	Compare character strings
SCPY	197	Optional	Column number	Column number, character literal	CC	Copy character string
SCRV	139	Optional	Ratio	Prohibited	CP	Set sigmoid motion ratio
SGET	199	Optional	Variable number	Column number, character literal	CP	Get character
SIN	91	Optional	Sine assignment variable	Operand	ZR	Sine
SLCT	179	Optional	Prohibited	Prohibited	CP	Declare start of multi-branching
SLEN	205	Optional	Character string length	Prohibited	CP	Set length
SPUT	200	Optional	Column number	Data	CP	Set character
SQR	93	Optional	Root assignment variable	Operand	ZR	Root
SSPG	118	Optional	Pause program number	(Pause program number)	CC	Pause program
STOP	161	Optional	Axis stop pattern	Prohibited	CP	Decelerate and stop axis
STR	201	Optional	Column number	Data	CC	Convert character string; decimal
STRH	202	Optional	Column number	Data	CC	Convert character string; hexadecimal
SUB	88	Optional	Minuend variable	Subtrahend	ZR	Subtract
SV□□	151	Optional	Operation axis pattern	Prohibited	PE	Servo [ON, OF]
SYST	185	Optional	Variable number	Prohibited	CP	Get system status

Operation type in the output field
CC: Command was executed successfully, ZR: Operation result is zero,
PE: Operation is complete, CP: Command part has passed, TU: Time up
EQ: Operand 1 = Operand 2, NE: Operand 1 ≠ Operand 2,
GT: Operand 1 > Operand 2, GE: Operand 1 ≥ Operand 2,
LT: Operand 1 < Operand 2, LE: Operand 1 ≤ Operand 2

Command	Page	Condition	Operand 1	Operand 2	Output	Function
T						
TAG	112	Prohibited	Declaration tag number	Prohibited	CP	Declare jump destination
TAN	92	Optional	Tangent assignment variable	Operand	ZR	Tangent
TIMC	99	Optional	Program number	Prohibited	CP	Cancel waiting
TIMR	241	Optional	Local flag number	Timer setting	CP	Timer relay for ladder
TIMW	98	Optional	Wait time	Prohibited	TU	Wait
TMRW	193	Optional	Read timer setting	(Write timer setting)	CP	Set READ timeout value
TPCD	223	Prohibited	0 or 1	Prohibited	CP	Specify processing to be performed when input condition is not specified
TRAN	86	Optional	Copy-destination variable	Copy-source variable	ZR	Copy
TSLP	224	Prohibited	Time	Prohibited	CP	Task sleep
V						
VAL	203	Optional	Variable number	Column number, character literal	CC	Convert character string data; decimal
VALH	204	Optional	Variable number	Column number, character literal	CC	Convert character string data; hexadecimal
VEL	135	Optional	Speed	Prohibited	CP	Set speed
VLMX	146	Optional	Prohibited	Prohibited	CP	Specify VLMX speed
W						
WH□□	180	Prohibited	Comparison variable	Comparison value	CP	Branch value [EQ, NE, GT, GE, LT, LE]
WRIT	195	Optional	Channel number	Column number	CC	Output to channel
WS□□	181	Prohibited	Column number	Column number, character literal	CP	Branch character string [EQ, NE]
WT□□	104	Optional	I/O, flag	(Wait time)	TU	Wait for I/O, flag [ON, OF]
WZFA	188	Optional	Zone number	Axis pattern	CP	Wait for zone OFF, with AND
WZFO	189	Optional	Zone number	Axis pattern	CP	Wait for zone OFF, with OR
WZNA	186	Optional	Zone number	Axis pattern	CP	Wait for zone ON, with AND
WZNO	187	Optional	Zone number	Axis pattern	CP	Wait for zone ON, with OR

Chapter 3 Explanation of Commands

1. Commands

1.1 Variable Assignment

- LET (Assign)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	LET	Variable number	Data	ZR

[Function] Assign the value specified in operand 2 to the variable specified in operand 1.
The output will turn ON when 0 is assigned to the variable specified in operand 1.

[Example 1]	LET	1	10	Assign 10 to variable 1.
[Example 2]	LET	3	10	Assign 10 to variable 3.
	LET	1	*3	Assign the content of variable 3 (10) to variable 1.

(Note) When data in a real variable is assigned to an integer variable, all decimal fractions are rounded to the nearest integer.

LET	100	13.5	Assign 13.5 to real variable 100.
LET	1	*100	Assign 14, which is a rounded result of the content of real variable 100 (13.5), to integer variable 1.

- TRAN (Copy)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	TRAN	Variable number	Variable number	ZR

[Function] Assign the content of the variable specified in operand 2 to the variable specified in operand 1.
The output will turn ON when 0 is assigned to the variable specified in operand 1.

[Example 1]	TRAN	1	2	Assign the content of variable 2 to variable 1.
	LET	1	*2	A LET command of the same effect as the above operation
[Example 2]	LET	3	4	Assign 4 to variable 3.
	LET	4	10	Assign 10 to variable 4.
	TRAN	1	*3	Assign the content of variable 3 (which is variable 4, or 10) to variable 1.

(Note) When data in a real variable is assigned to an integer variable, all decimal fractions are rounded to the nearest integer.

LET	100	13.5	Assign 13.5 to real variable 100.
TRAN	1	100	Assign 14, which is a rounded result of the content of real variable 100 (13.5), to integer variable 1.

- CLR (Clear variable)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	CLR	Variable number	Variable number	ZR

[Function] Clear the variables from the one specified in operand 1 through the other specified in operand 2.
The contents of the variables that have been cleared become 0.
The output will turn ON when 0 is assigned to the variable specified in operand 1.

[Example 1]	CLR	1	5	Clear variables 1 through 5.
[Example 2]	LET	1	10	Assign 10 to variable 1.
	LET	2	20	Assign 20 to variable 2.
	CLR	*1	*2	Clear the variables from the content of variable 1 (variable 10) through the content of variable 2 (variable 20).

1.2 Arithmetic Operation

- ADD (Add)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	ADD	Variable number	Data	ZR

[Function] Add the content of the variable specified in operand 1 and the value specified in operand 2, and assign the result to the variable specified in operand 1.
The output will turn ON when the operation result becomes 0.

[Example 1]

LET	1	3	Assign 3 to variable 1.
ADD	1	2	Add 2 to the content of variable 1 (3). 5 (3+2=5) will be stored in variable 1.

[Example 2]

LET	1	2	Assign 2 to variable 1.
LET	3	2	Assign 2 to variable 3.
ADD	1	*3	Add the content of variable 3 (2) to the content of variable 1 (2). 4 (2+2=4) will be stored in variable 1.

- SUB (Subtract)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	SUB	Variable number	Data	ZR

[Function] Subtract the value specified in operand 2 from the content of the variable specified in operand 1, and assign the result to the variable specified in operand 1.
The output will turn ON when the operation result becomes 0.

[Example 1]

LET	1	3	Assign 3 to variable 1.
SUB	1	2	Subtract 2 from the content of variable 1 (3). 1 (3–2=1) will be stored in variable 1.

[Example 2]

LET	1	3	Assign 3 to variable 1.
LET	3	2	Assign 2 to variable 3.
SUB	1	*3	Subtract the content of variable 3 (2) from the content of variable 1 (3). 1 (3–2=1) will be stored in variable 1.

- MULT (Multiply)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	MULT	Variable number	Data	ZR

[Function] Multiply the content of the variable specified in operand 1 by the value specified in operand 2, and assign the result to the variable specified in operand 1.
The output will turn ON when the operation result becomes 0.

[Example 1]

LET	1	3	Assign 3 to variable 1.
MULT	1	2	Multiply the content of variable 1 (3) by 2. 6 (3x2=6) will be stored in variable 1.

[Example 2]

LET	1	4	Assign 4 to variable 1.
LET	3	2	Assign 2 to variable 3.
MULT	1	*3	Multiply the content of variable 1 (4) by the content of variable 3 (2). 8 (4x2=8) will be stored in variable 1.

- DIV (Divide)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	DIV	Variable number	Data	ZR

[Function] Divide the content of the variable specified in operand 1 by the value specified in operand 2, and assign the result to the variable specified in operand 1.
The output will turn ON when the operation result becomes 0.

(Note) If the variable specified in operand 1 is an integer variable, any decimal places will be rounded off.

[Example 1]

LET	1	6	Assign 6 to variable 1.
DIV	1	2	Divide the content of variable 1 (6) by 2. 3 (6÷2=3) will be stored in variable 1.

[Example 2]

LET	1	6	Assign 6 to variable 1.
LET	3	2	Assign 2 to variable 3.
DIV	1	*3	Divide the content of variable 1 (6) by the content of variable 3 (2). 3 (6÷2=3) will be stored in variable 1.

- MOD (Remainder)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	MOD	Variable number	Data	ZR

[Function] Assign, to the variable specified in 1, the remainder obtained by dividing the content of the variable specified in operand 1 by the value specified in operand 2.
The output will turn ON when the operation result becomes 0.

(Note) A MOD command is used with integer variables.

[Example 1]

LET	1	7	Assign 7 to variable 1.
MOD	1	3	Obtain the remainder of dividing the content of variable 1 (7) by 3. 1 (7÷3=2 with a remainder of 1) will be assigned to variable 1.

[Example 2]

LET	1	7	Assign 2 to variable 1.
LET	3	3	Assign 3 to variable 3.
MOD	1	*3	Obtain the remainder of dividing the content of variable 1 (7) by the content of variable 3 (3). 1 (7÷3=2 with a remainder of 1) will be assigned to variable 1.

1.3 Function Operation

- SIN (Sine operation)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	SIN	Variable number	Data	ZR

[Function] Assign the sine of the data specified in operand 2 to the variable specified in operand 1.
The output will turn ON when the operation result becomes 0.
The setting in operand 1 must be a real variable in a range of 100 to 199, 1100 to 1199, 300 to 399 or 1300 to 1399.
The unit of data in operand 2 is radian.

(Note 1) Radian = Angle x $\pi \div 180$

[Example 1]

SIN	100	0.523599	Assign the sine of 0.523599 (0.5) to variable 100.

[Example 2]

LET	101	30	30 x $\pi \div 180$ (radian)
MULT	101	3.141592	(30° will be converted to radian and assigned to
DIV	101	180	variable 101.)
SIN	100	*101	Assign the sine of the content of variable 101 (0.5) to variable 100.

- COS (Cosine operation)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	COS	Variable number	Data	ZR

[Function] Assign the cosine of the data specified in operand 2 to the variable specified in operand 1.
The output will turn ON when the operation result becomes 0.
The setting in operand 1 must be a real variable in a range of 100 to 199, 1100 to 1199, 300 to 399 or 1300 to 1399.
The unit of data in operand 2 is radian.

(Note 1) Radian = Angle x $\pi \div 180$

[Example 1]

COS	100	1.047197	Assign the cosine of 1.047197 (0.5) to variable 100.

[Example 2]

LET	101	60	60 x $\pi \div 180$ (radian)
MULT	101	3.141592	(60° will be converted to radian and assigned to
DIV	101	180	variable 101.)
COS	100	*101	Assign the cosine of the content of variable 101 (0.5) to variable 100.

- TAN (Tangent operation)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	TAN	Variable number	Data	ZR

[Function] Assign the tangent of the data specified in operand 2 to the variable specified in operand 1.
The output will turn ON when the operation result becomes 0.
The setting in operand 1 must be a real variable in a range of 100 to 199, 1100 to 1199, 300 to 399 or 1300 to 1399.
The unit of data in operand 2 is radian.

(Note 1) Radian = Angle x π ÷ 180

[Example 1]

```
TAN   100   0.785398
```

Assign the tangent of 0.785398 (1) to variable 100.

[Example 2]

```
LET   101   45
MULT  101   3.141592
DIV   101   180
TAN   100   *101
```

45 x π ÷ 180 (radian)
(45° will be converted to radian and assigned to variable 101.)
Assign the tangent of the content of variable 101 (1) to variable 100.

- ATN (Inverse-tangent operation)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	ATN	Variable number	Data	ZR

[Function] Assign the inverse tangent of the data specified in operand 2 to the variable specified in operand 1.
The output will turn ON when the operation result becomes 0.
The setting in operand 1 must be a real variable in a range of 100 to 199, 1100 to 1199, 300 to 399 or 1300 to 1399.
The unit of inverse tangent is radian.

(Note 1) Radian = Angle x π ÷ 180

[Example 1]

```
ATN   100   1
```

Assign the inverse tangent of 1 (0.785398) to variable 100.

[Example 2]

```
LET   101   1
ATN   100   *101
```

Assign 1 to variable 101.
Assign the inverse tangent of the content of variable 101 (0.785398) to variable 100.

- SQR (Root operation)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	SQR	Variable number	Data	ZR

[Function] Assign the root of the data specified in operand 2 to the variable specified in operand 1. The output will turn ON when the operation result becomes 0.

[Example 1]	SQR	1	4	Assign the root of 4 (2) to variable 1.
[Example 2]	LET	2	5	Assign 5 to variable 2.
	SQR	100	*2	Assign the root of the content of variable 2 (5) to variable 100.

1.4 Logical Operation

- AND (Logical AND)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	AND	Variable number	Data	ZR

[Function] Assign the logical AND operation result of the content of the variable specified in operand 1 and the value specified in operand 2, to the variable specified in operand 1.
The output will turn ON when the operation result becomes 0.

[Example 1]

LET	1	204	Assign 204 to variable 1.
AND	1	170	Assign the logical AND operation result (136) of the content of variable 1 (204) and 170, to variable 1.

[Example 2]

LET	1	204	Assign 204 to variable 1.
LET	3	170	Assign 170 to variable 3.
AND	1	*3	Assign the logical AND operation result (136) of the content of variable 1 (204) and the content of variable 3 (170) to variable 1.

	Decimal		Binary
	204		11001100
AND	170	AND	10101010
	136		10001000

- OR (Logical OR)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	OR	Variable number	Data	ZR

[Function] Assign the logical OR operation result of the content of the variable specified in operand 1 and the value specified in operand 2, to the variable specified in operand 1.
The output will turn ON when the operation result becomes 0.

[Example 1]

LET	1	204	Assign 204 to variable 1.
OR	1	170	Assign the logical OR operation result (238) of the content of variable 1 (204) and 170, to variable 1.

[Example 2]

LET	1	204	Assign 204 to variable 1.
LET	3	170	Assign 170 to variable 3.
OR	1	*3	Assign the logical OR operation result (238) of the content of variable 1 (204) and the content of variable 3 (170) to variable 1.

Decimal	Binary
204	11001100
OR 170	OR 10101010
238	11101110

- EOR (Logical exclusive-OR)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	EOR	Variable number	Data	ZR

[Function] Assign the logical exclusive-OR operation result of the content of the variable specified in operand 1 and the value specified in operand 2, to the variable specified in operand 1. The output will turn ON when the operation result becomes 0.

[Example 1]

LET	1	204	Assign 204 to variable 1.
EOR	1	170	Assign the logical exclusive-OR operation result (102) of the content of variable 1 (204) and 170, to variable 1.

[Example 2]

LET	1	204	Assign 204 to variable 1.
LET	3	170	Assign 170 to variable 3.
EOR	1	*3	Assign the logical exclusive-OR operation result (102) of the content of variable 1 (204) and the content of variable 3 (170) to variable 1.

Decimal	Binary
204	11001100
EOR 170	EOR 10101010
102	01100110

1.5 Comparison Operation

● CP□□ (Compare)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	CP□□	Variable number	Data	EQ NE GT GE LT LE

[Function] The output will be turned ON if the comparison result of the content of the variable specified in operand 1 and the value specified in operand 2 satisfies the condition.
The value in the variable does not change.
The output will be turned OFF if the condition is not satisfied.

(Note) The output will not be turned OFF when the command is executed.

CP□□
- EQ ..... Operand 1 = Operand 2
- NE ..... Operand 1 ≠ Operand 2
- GT ..... Operand 1 > Operand 2
- GE ..... Operand 1 ≥ Operand 2
- LT ..... Operand 1 < Operand 2
- LE ..... Operand 1 ≤ Operand 2

[Example 1]

	LET	1	10		Assign 10 to variable 1.
	CPEQ	1	10	600	Turn ON flag 600 if the content of variable 1 is 10.
600	ADD	2	1		Add 1 to variable 2 if flag 600 is ON.

[Example 2]

LET	1	10		Assign 10 to variable 1.
LET	3	10		Assign 10 to variable 3.
CPEQ	1	*3	310	Turn ON output 310 if the content of variable 1 (10) is equal to the content of variable 3.

1.6 Timer

- TIMW (Timer)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	TIMW	Time	Prohibited	TU

[Function] Stop the program and wait for the time specified in operand 1.
The setting range is 0.01 to 99, and the unit is second.
The output will turn ON when the specified time has elapsed and the program proceeds to the next step.

[Example 1]	TIMW	1.5		Wait for 1.5 seconds.
[Example 2]	LET	1	10	Assign 10 to variable 1.
	TIMW	*1		Wait for the content of variable 1 (10 seconds).

- TIMC (Cancel timer)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	TIMC	Program number	Prohibited	CP

[Function] Cancel a timer in other program running in parallel.

(Note) Timers in TIMW, WTON, WTOF and READ commands can be cancelled. In the case of WTON, WTOF and READ commands, even if timeout is not specified it is assumed that an unlimited timer has been specified and the wait time will be cancelled.

```
[Example 1]   TIMC     10                  Cancel the wait time in program 10.

[Example 2]   LET      1     10            Assign 10 to variable 1.
              TIMC     *1                  Cancel the wait time in the content of variable 1
                                           (program 10).

[Example 3]   Program 1        Program 10

                 :                 :
                 :              WTON 8 20      Program 10 waits for input 8 for 20 seconds.
                 :              (Wait for input 8)
              TIMC     10     (Wait for input 8)  Cancel the wait time in program 10.
                 :                 :
```

(Note) The steps shown in the above example represent those executed simultaneously in different programs.

- GTTM (Get time)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	GTTM	Variable number	Prohibited	CP

[Function] Read system time to the variable specified in operand 1. The time is specified in units of 10 milliseconds.
The time obtained here has no base number. Therefore, this command is called twice and the difference will be used to calculate the elapsed time.

[Example 1]

```
GTTM  1        Read the reference time to variable 1.
ADD   1   500  Set the ending time to 5 seconds later.
GTTM  2        Read the current system time to variable 2.
DWLE  2   *1   Proceed to the step next to EDDO when 5 seconds elapsed.
  :            The above process will be repeated for 5 seconds.
  :
GTTM  2        Read the current system time to variable 2.
EDDO
```

[Example 2]

```
LET   1   5    Assign 5 to variable 1.
GTTM  *1       Store the current system time in the content of variable 1
               (variable 5).
```

1.7 I/O, Flag Operation

● BT□□ (Output port, flag operation)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	BT□□	Output, flag	(Output, flag)	CP

[Function] Reverse the ON/OFF status of the output ports or flags from the one specified in operand 1 through the other specified in operand 2.

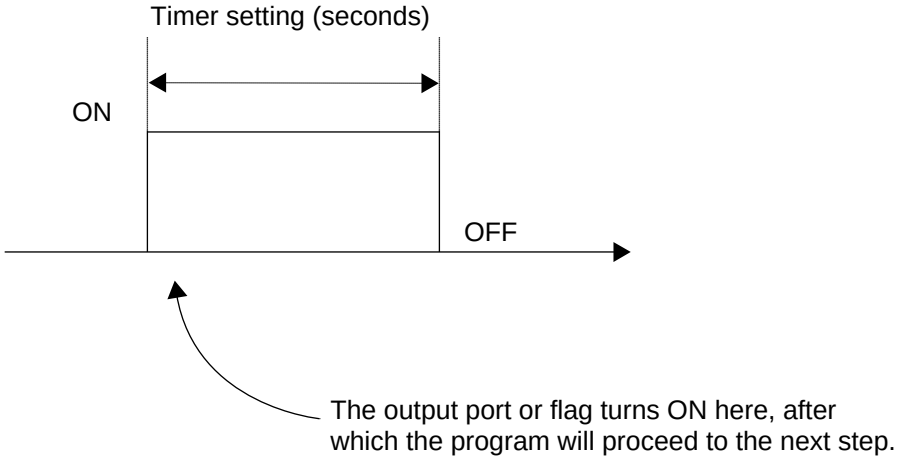
BT□□
- ON ..... Switch the status to ON.
- OF ..... Switch the status to OFF.
- NT ..... Reverse the status.

[Example 1]	BTON	300		Turn ON output port 300.
[Example 2]	BTOF	300	307	Turn OFF output ports 300 through 307.
[Example 3]	LET	1	600	Assign 600 to variable 1.
	BTNT	*1		Reverse the content of variable 1 (flag 600).
[Example 4]	LET	1	600	Assign 600 to variable 1.
	LET	2	607	Assign 607 to variable 2.
	BTON	*1	*2	Turn ON the flags from the content of variable 1 (flag 600) through the content of variable 2 (flag 607).

- BTPN (Output ON pulse)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	BTPN	Output port, flag	Timer setting	CP

[Function] Turn ON the specified output port or flag for the specified time.
When this command is executed, the output port or flag specified in operand 1 will be turned ON and then the program will proceed to the next step. The output port or flag will be turned OFF automatically upon elapse of the timer setting specified in operand 2.
The timer is set in a range from 0.01 to 99.00 seconds (including up to two decimal places).



(Note 1) If this command is executed with respect to an output port or flag already ON, the output port or flag will be turned OFF upon elapse of the timer setting.

(Note 2) If the program ends after the command has been executed but before the timer is up, the output port or flag will not be turned OFF.

(Note 3) This command will not be cancelled by a TIMC command.

(Note 4) A maximum of 16 timers, including BTPN and BTPF, can be operated simultaneously in a single program. (There is no limitation as to how many times these timers can be used in a single program.)

[Example]

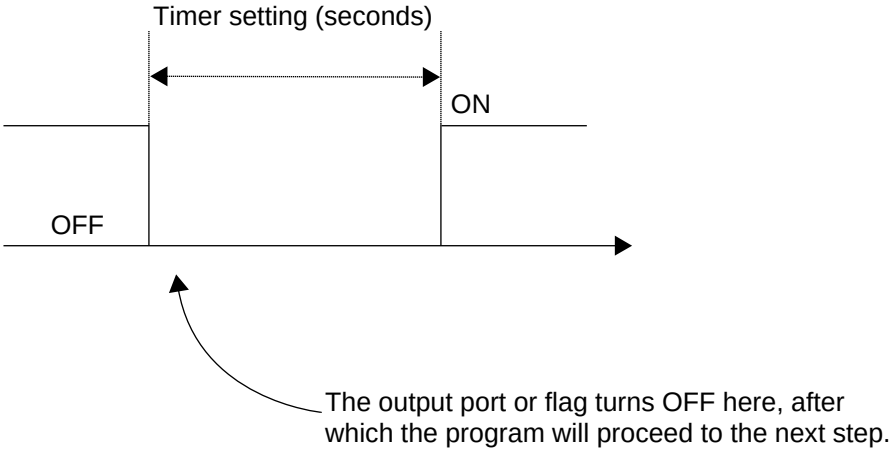
```
BTPN   300   1    Turn ON output port 300 for 1 second.
BTPN   600   10   Turn ON flag 600 for 10 seconds.
```

(Note 5) This command cannot be used to output pulses for a specified period because a margin of error occurs in the pulse output time if other task or interruption process occurs after the port is turned ON until it is turned OFF.

- BTPF (Output OFF pulse)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	BTPF	Output port, flag	Timer setting	CP

[Function] Turn OFF the specified output port or flag for the specified time.
When this command is executed, the output port or flag specified in operand 1 will be turned OFF and then the program will proceed to the next step. The output port or flag will be turned ON automatically upon elapse of the timer setting specified in operand 2.
The timer is set in a range from 0.01 to 99.00 seconds (including up to two decimal places).



(Note 1) If this command is executed with respect to an output port or flag already OFF, the output port or flag will be turned ON upon elapse of the timer setting.

(Note 2) If the program ends after the command has been executed but before the timer is up, the output port or flag will not be turned ON.

(Note 3) This command will not be cancelled by a TIMC command.

(Note 4) A maximum of 16 timers, including BTPN and BTPF, can be operated simultaneously in a single program. (There is no limitation as to how many times these timers can be used in a single program.)

[Example]

```
BTPF   300   1    Turn OFF output port 300 for 1 second.
BTPF   600   10   Turn OFF flag 600 for 10 seconds.
```

- WT□□ (Wait for I/O port, flag)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	WT□□	I/O, flag	(Time)	TU

[Function] Wait for the I/O port or flag specified in operand 1 to turn ON/OFF.
The program can be aborted after the specified time by setting the time in operand 2.
The setting range is 0.01 to 99 seconds.
The output will turn ON upon elapse of the specified time (only when operand 2 is specified).
Note) A local flag cannot be entered in operand 1.

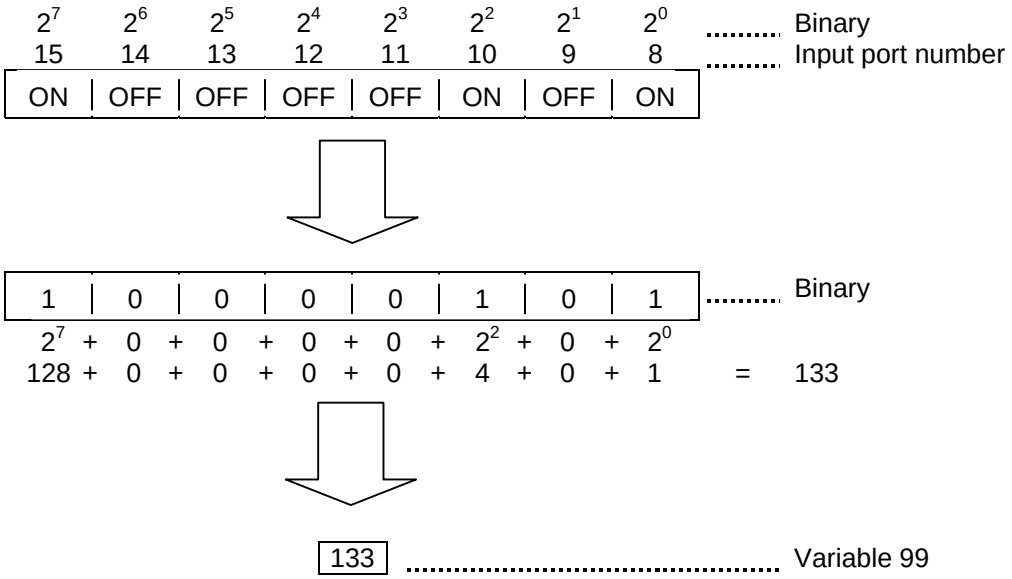
WT□□
- ON ..... Wait for the applicable I/O port or flag to turn ON.
- OF ..... Wait for the applicable I/O port or flag to turn OFF.

[Example 1]	WTON	15		Wait for input port 15 to turn ON.
[Example 2]	WTOF	307	10	Wait for 10 seconds for output port 307 to turn OFF.
[Example 3]	LET	1	600	Assign 600 to variable 1.
	WTON	*1		Wait for the content of variable 1 (flag 600) to turn ON.
[Example 4]	LET	1	8	Assign 8 to variable 1.
	LET	2	5	Assign 5 to variable 2.
	WTOF	*1	*2	Wait for the content of variable 2 (5 seconds) for the content of variable 1 (input port 8) to turn OFF.

- IN (Read I/O, flag as binary)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	IN	I/O, flag	I/O, flag	CC

[Function] Read the I/O ports or flags from the one specified in operand 1 through the other specified in operand 2, to variable 99 as a binary.



(Note 1) A maximum of 32 bits can be input.

(Note 2) When 32 bits have been input and the most significant bit is ON, the value read to variable 99 will be treated as a negative value.

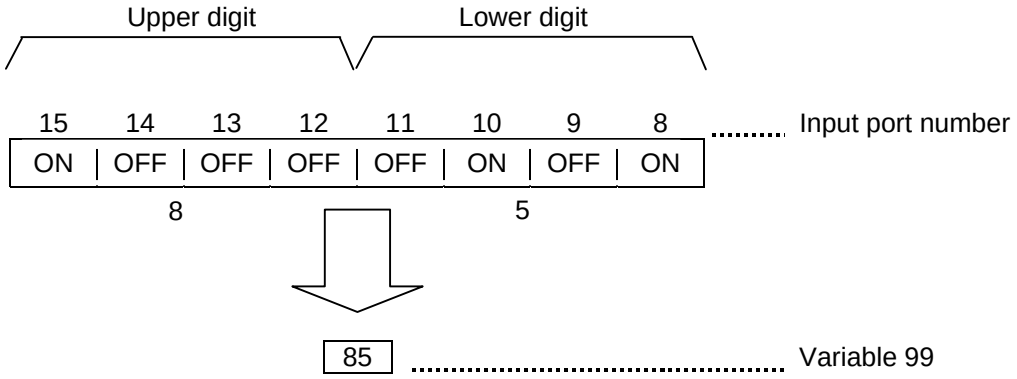
(Note 3) The read data format can be changed using a FMIO command (refer to the section on FMIO command).

[Example 1]	IN	8	15	Read input ports 8 through 15, to variable 99 as a binary.
[Example 2]	LET	1	8	Assign 8 to variable 1.
	LET	2	15	Assign 15 to variable 2.
	IN	*1	*2	Read the input ports from the content of variable 1 (input port 8) through the content of variable 2 (input port 15), to variable 99 as a binary.

- INB (Read I/O, flag as BCD)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	INB	I/O, flag	BCD digits	CC

[Function] Read the I/O ports or flags from the one specified in operand 1 for the number of digits specified in operand 2, to variable 99 as a BCD.



(Note 1) A maximum of eight digits (32 bits) can be input.

(Note 2) The number of I/O ports and flags that can be used is 4 x n (digits).

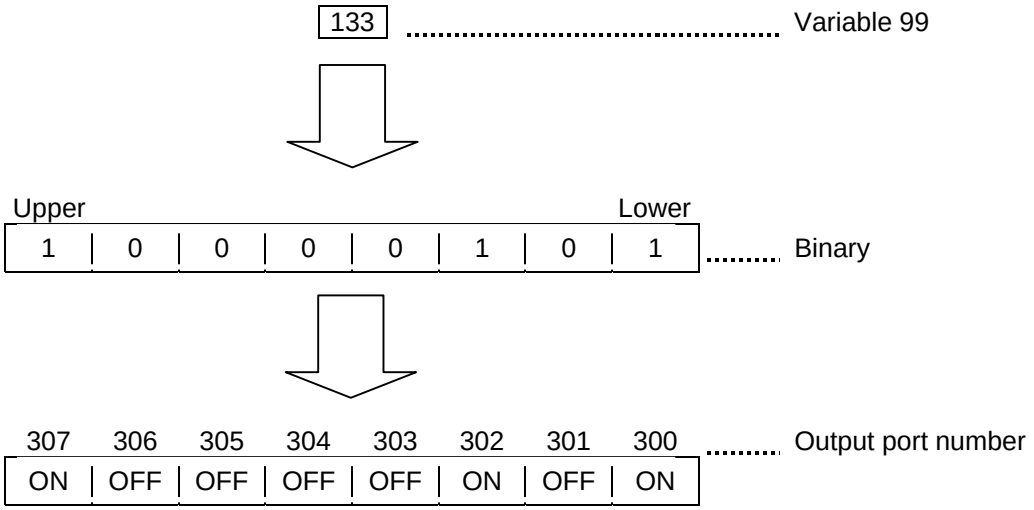
(Note 3) The read data format can be changed using a FMIO command (refer to the section on FMIO command).

[Example 1]	INB	8	2	Read input ports 8 through 15, to variable 99 as a BCD.
[Example 2]	LET	1	8	Assign 8 to variable 1.
	LET	2	2	Assign 2 to variable 2.
	INB	*1	*2	Read the input ports from the content of variable 1 (input port 8) for the content of variable 2 (two digits) (until input port 15), to variable 99 as a BCD.

- OUT (Write output, flag as binary)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	OUT	Output, flag	Output, flag	CC

[Function] Write the value in variable 99 to the output ports or flags from the one specified in operand 1 through the other specified in operand 2.



(Note 1) A maximum of 32 bits can be output.

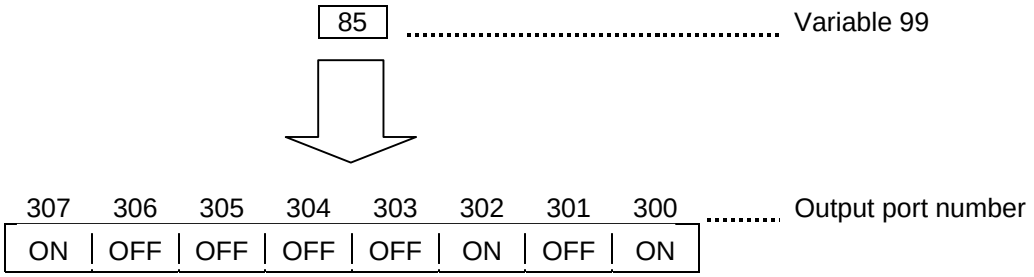
(Note 2) The write data format can be changed using a FMIO command (refer to the section on FMIO command).

[Example 1]	OUT	300	307	Write the value in variable 99 to output ports 300 through 307 as a binary.
[Example 2]	LET	1	300	Assign 300 to variable 1.
	LET	2	307	Assign 307 to variable 2.
	OUT	*1	*2	Write the value in variable 99 to the output ports from the content of variable 1 (output port 300) through the content of variable 2 (output port 307) as a binary.

- OUTB (Write output, flag as BCD)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	OUTB	Output, flag	BCD digits	CC

[Function] Write the value in variable 99 to the output ports or flags from the one specified in operand 1 for the number of digits specified in operand 2 as a BCD.



(Note 1) A maximum of eight digits (32 bits) can be output.

(Note 2) The number of output ports and flags that can be used is 4 x n (digits).

(Note 3) The write data format can be changed using a FMIO command (refer to the section on FMIO command).

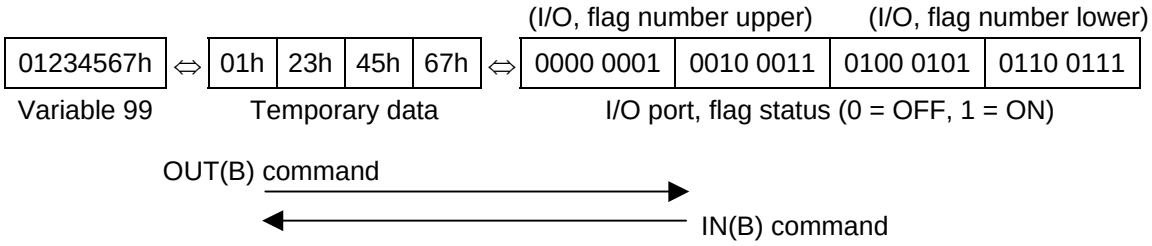
[Example 1]	OUTB	300	2	Write the value in variable 99 to the output ports from 300 for two digits (until output port 307) as a BCD.
[Example 2]	LET	1	300	Assign 300 to variable 1.
	LET	2	2	Assign 2 to variable 2.
	OUTB	*1	*2	Write the value in variable 99 to the output ports from the content of variable 1 (output port 300) for the content of variable 2 (two digits) (until output port 307) as a BCD.

- FMIO (Set IN, INB, OUT, OUTB command format)

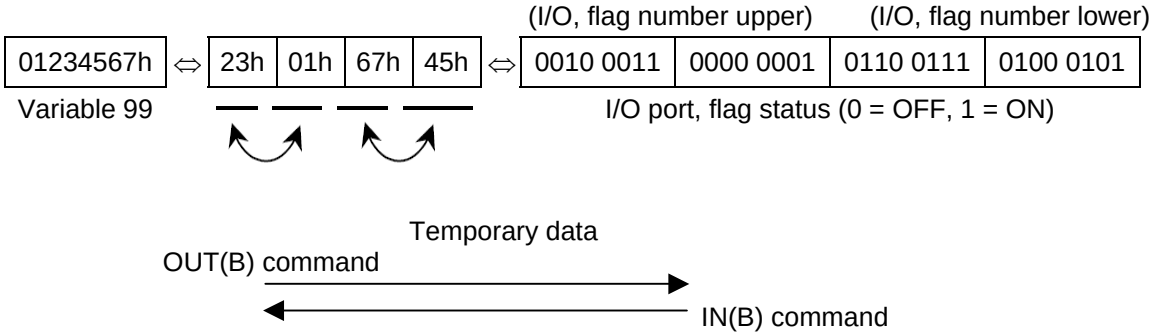
Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	FMIO	Format type	Prohibited	CP

[Function] Set the data format for reading or writing I/O ports and flags with an IN, INB, OUT or OUTB command.

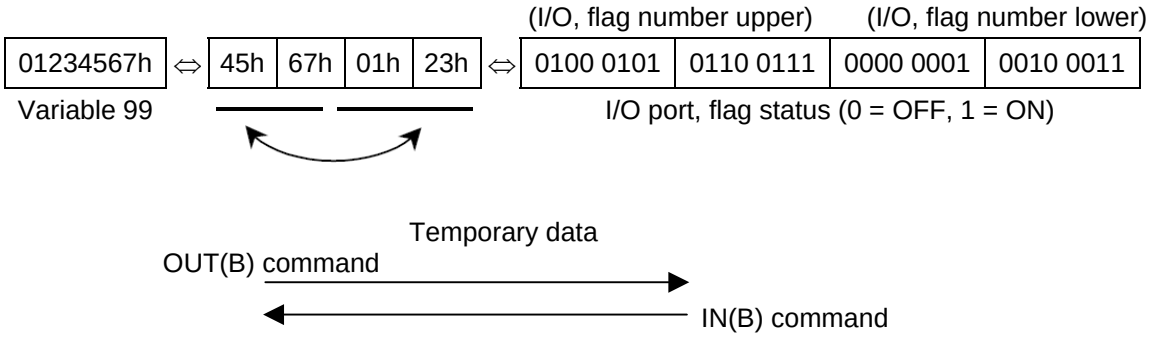
(1) Operand 1 = 0 (Default status when a FMIO command has not been executed)
Data is read or written without being reversed.



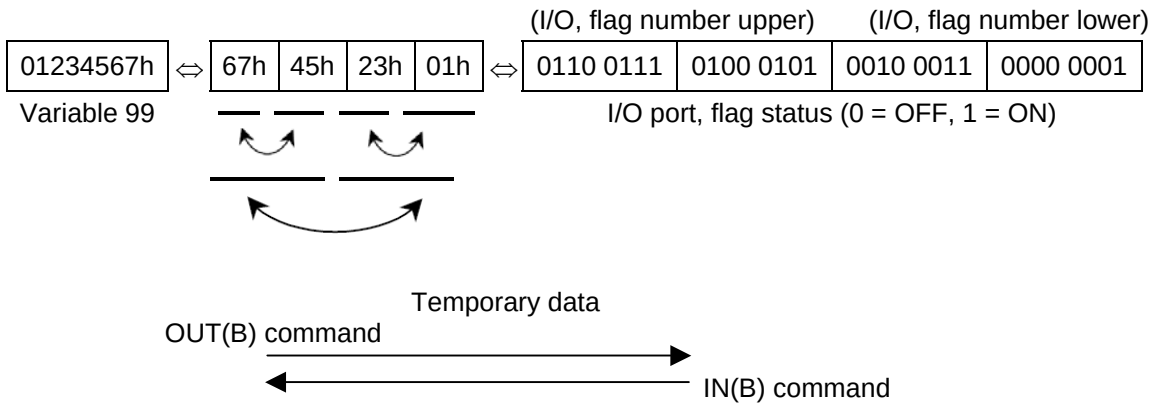
(2) Operand 1 = 1
Data is read or written after its upper eight bits and lower eight bits are reversed every 16 bits.



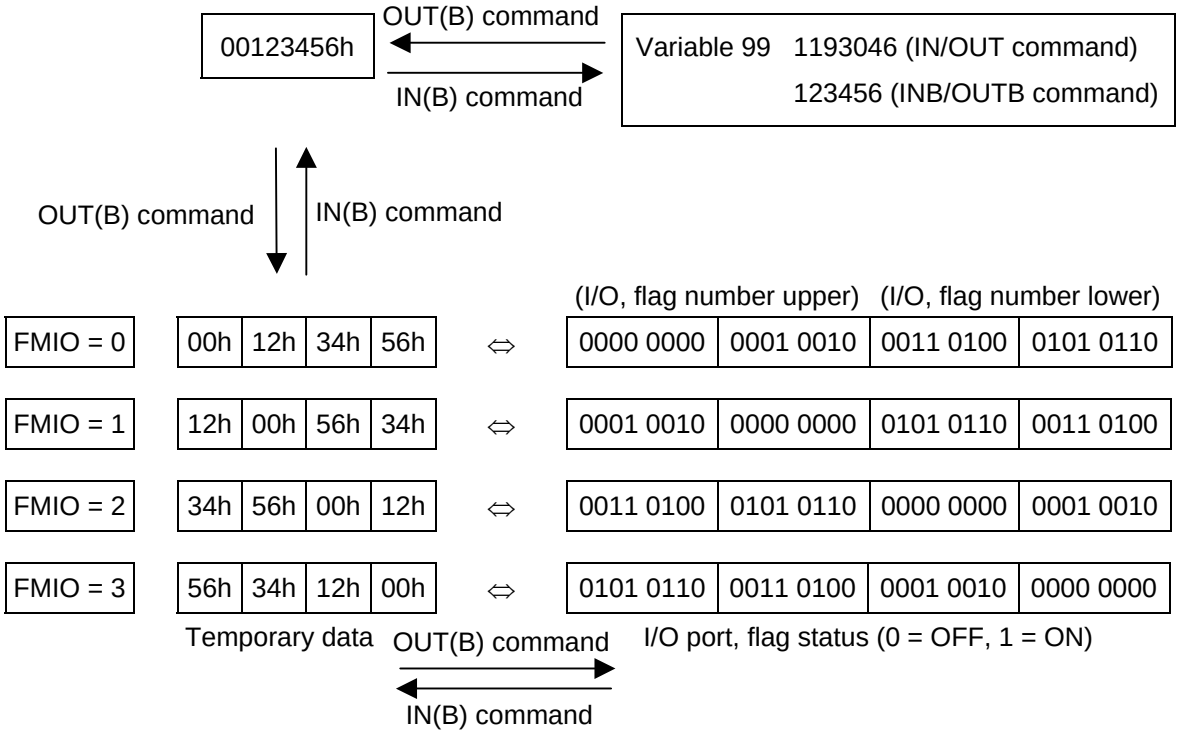
(3) Operand 1 = 2
Data is read or written after its upper 16 bits and lower 16 bits are reversed every 32 bits.



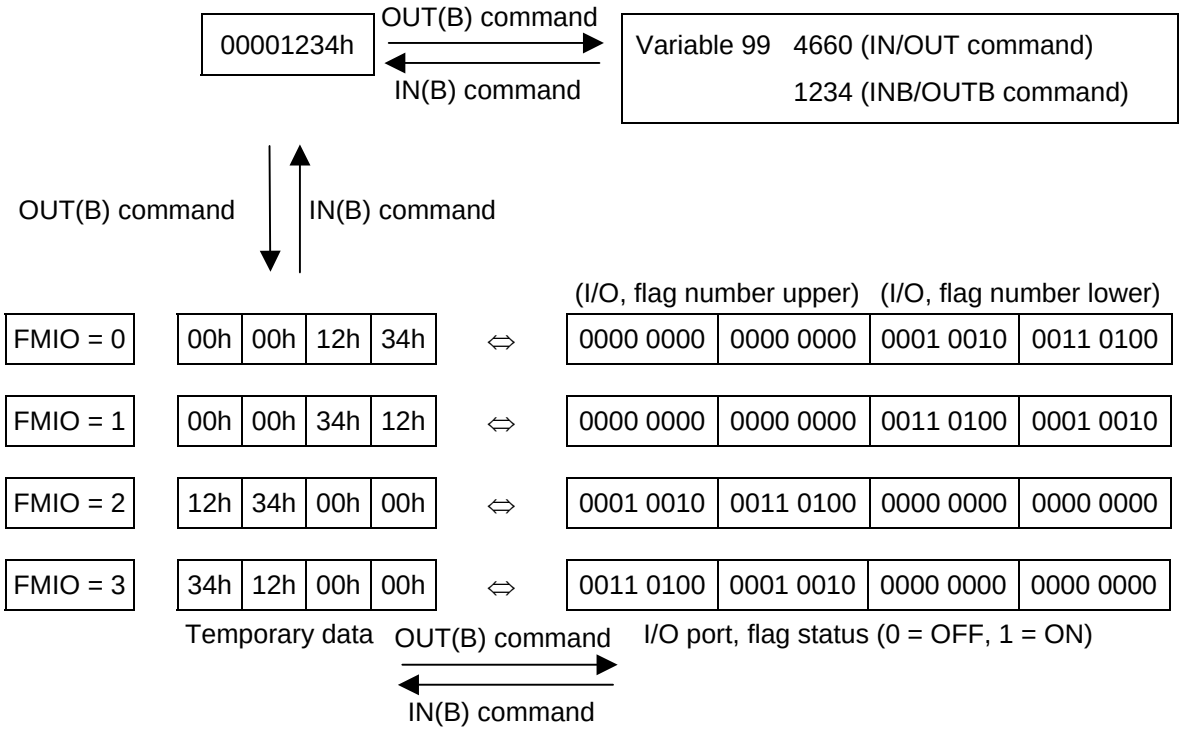
(4) Operand 1 = 3
Data is read or written after its upper 16 bits and lower 16 bits are reversed every 32 bits and its upper eight bits and lower eight bits are reversed every 16 bits.



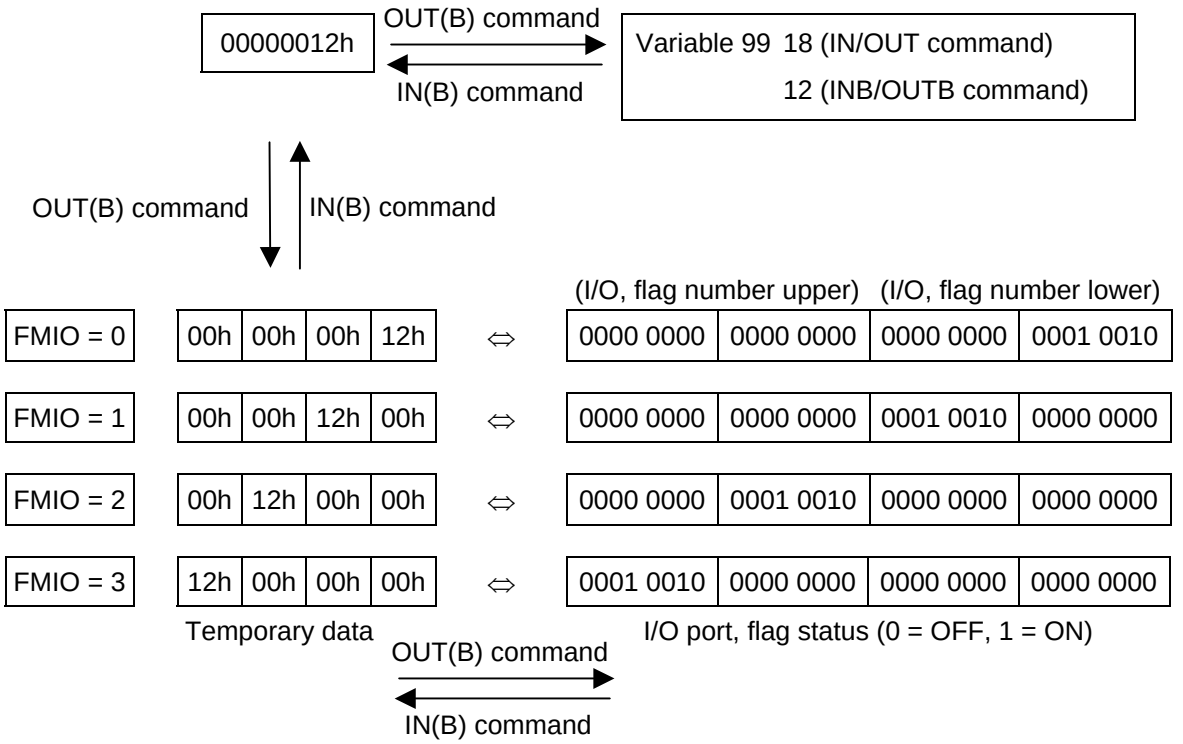
[Example 1] Variable 99 = 00123456h (Decimal: 1193046, BCD: 123456)



[Example 2] Variable 99 = 00001234h (Decimal: 4660, BCD: 1234)



[Example 3] Variable 99 = 00000012h (Decimal: 18, BCD: 12)



1.8 Program Control

- GOTO (Jump)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	GOTO	Tag number	Prohibited	CP

[Function] Jump to the position of the tag number specified in operand 1.

(Note) A GOTO command is valid only within the same program.

[Example 1]

```
TAG     1     Set a tag.
  :
  :
  :
GOTO    1     Jump to tag 1.
```

Using a GOTO command to branch out of or into any of the syntaxes listed below is prohibited.

Since the maximum number of nests is defined for each conditional branching command or subroutine call, a nest will be infinitely repeated if an ED□□ is not passed, and a nest overflow error will generate. In the case of palletizing setting, an error will generate if the second BGPA is declared after the first BGPA declaration without passing an EDPA.

(1) IF□□ or IS□□ and EDIF syntax
(2) DW□□ and EDDO syntax
(3) SLCT and EDSL syntax
(4) BGSR and EDSR syntax
(5) BGPA and EDPA syntax

- TAG (Declare tag)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Prohibited	Prohibited	TAG	Tag number	Prohibited	CP

[Function] Set the tag number specified in operand 1.

[Example 1] Refer to the section on GOTO command.

- EXSR (Execute subroutine)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	EXSR	Subroutine number	Prohibited	CP

[Function] Execute the subroutine specified in operand 1.
A maximum of 15 nested subroutine calls are supported.

(Note) This command is valid only for subroutines within the same program.

[Example 1]

```
EXSR   1          Execute subroutine 1.
  :
  :
EXIT
BGSR   1          Start subroutine 1.
  :
  :
  :
EDSR              End subroutine 1.
```

[Example 2]

```
LET    1    10     Assign 10 to variable 1.
EXSR   *1          Execute the content of variable 1 (subroutine 10).
```

- BGSR (Start subroutine)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Prohibited	Prohibited	BGSR	Subroutine number	Prohibited	CP

[Function] Declare the start of the subroutine specified in operand 1.

[Example 1] Refer to the section on EXSR command.

(Note) Using a GOTO command to branch out of or into a BGSR-EDSR syntax is prohibited.

- EDSR (End subroutine)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Prohibited	Prohibited	EDSR	Prohibited	Prohibited	CP

[Function] Declare the end of a subroutine.
This command is always required at the end of a subroutine.
Thereafter, the program will proceed to the step next to the EXSR that has been called.

[Example 1] Refer to the section on EXSR command.

1.9 Task Management

- EXIT (End program)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	EXIT	Prohibited	Prohibited	CP

[Function] End the program.
If the last step has been reached without encountering any EXIT command, the program will return to the beginning.

(Note) Status at program end

- Output ports ..... Retained
- Local flags ..... Cleared
- Local variables ..... Cleared
- Current values ..... Retained
- Global flags ..... Retained
- Global variables ..... Retained

[Example 1]

```
      :
      :
    EXIT        End the program.
```

- EXPG (Start other program)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	EXPG	Program number	(Program number)	CC

[Function] Start the programs from the one specified in operand 1 through the other specified in operand 2, and run them in parallel. Specification in operand 1 only is allowed.

[Example 1] EXPG 10 12 Start program Nos. 10, 11 and 12.

Error-generation/output-operation conditions
When one EXPG program is specified (only operand 1 is specified)

Status of the specified program	No program number error *1			Program number error *1
	Program already registered		Program not yet registered	
	Program running	Program not running		
Error	A57 “Multiple program start error”	None	C03 “Non-registered program specification error”	C2C “Program number error”
Output operation	ON	ON	OFF	OFF

* The errors shown in the table represent those that generate in accordance with the status of the specified program. Errors caused by other factors are excluded.
* 1 --- Program number error indicates specification of a number smaller than 1 or exceeding 64.

When multiple EXPG programs are specified (both operands 1 and 2 are specified)

Status of the specified program	No program number error *2			Program number error *1
	Registered program exists inside the specified range *3		None of programs inside the specified range are registered	
	Running program exists inside the specified range	None of programs inside the specified range are running		
Error	A57 “Multiple program start error”	None	C03 “Non-registered program specification error”	C2C “Program number error”
Output operation	ON	ON	OFF	OFF

* The errors shown in the table represent those that generate in accordance with the status of the specified program. Errors caused by other factors are excluded.
* 2 --- Program number error indicates specification of a number smaller than 1 or exceeding 64.
* 3 --- In this case, non-registered programs inside the specified range are not treated as a target of operation. This will not affect error generation or output operation.

- ABPG (Abort other program)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	ABPG	Program number	(Program number)	CC

[Function] Forcibly end the programs from the one specified in operand 1 to the other specified in operand 2. Specification in operand 1 only is allowed.

(Note 1) If an ABPG command is issued while a movement command is being executed, the axes will immediately decelerate and stop.
(Note 2) Not only the operation but also the execution of the step itself will be terminated.

[Example 1] ABPG 10 12 End program Nos. 10, 11 and 12.

Error-generation/output-operation conditions
When one ABPG program is specified (only operand 1 is specified)

Status of the specified program	No program number error *1			Program number error *1
	Program already registered		Program not yet registered	
	Program running	Program not running		
Error	None	None	None	C2C "Program number error"
Output operation	ON (OFF *2)	ON	ON	OFF

* The errors shown in the table represent those that generate in accordance with the status of the specified program. Errors caused by other factors are excluded.
* 1 --- Program number error indicates specification of a number smaller than 1 or exceeding 64.
* 2 --- If an own task (own program) is specified in an ABPG command, the own task will be terminated and then deleted. The output will turn OFF.

When multiple ABPG programs are specified (both operands 1 and 2 are specified)

Status of the specified program	No program number error *3			Program number error *1
	Registered program exists inside the specified range *4		None of programs inside the specified range are registered	
	Running program exists inside the specified range	None of programs inside the specified range are running		
Error	None	None	None	C2C "Program number error"
Output operation	ON (OFF *5)	ON	ON	OFF

* The errors shown in the table represent those that generate in accordance with the status of the specified program. Errors caused by other factors are excluded.
* 3 --- Program number error indicates specification of a number smaller than 1 or exceeding 64.
* 4 --- In this case, non-registered programs inside the specified range are not treated as a target of operation. This will not affect error generation or output operation.
* 5 --- If an own task (own program) is included in the specified range, the own task will be terminated, upon which the processing of the ABPG command will end. Since the own task will be deleted, the result of ending the processing of specified programs will become indeterminable. Exercise caution. The output will always turn OFF regardless of the result.

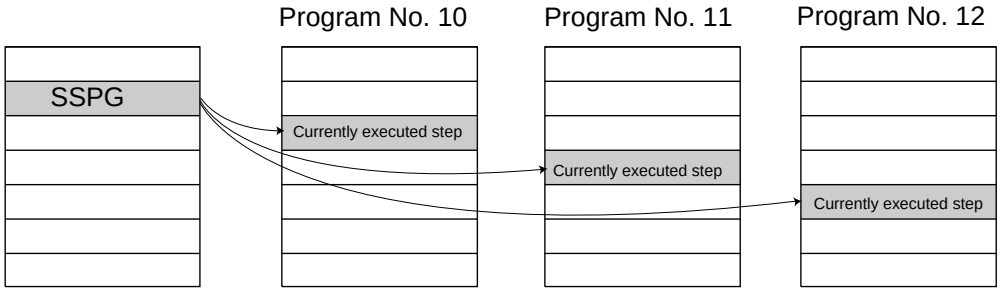
- SSPG (Pause program)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	SSPG	Program number	(Program number)	CC

[Function] Pause the program from the one specified in operand 1 through the other specified in operand 2, at the current step. Specification in operand 1 only is allowed.

(Note 1) Pausing a program will also pause the operation the program has been executing.
(Note 2) Not only the operation but also the execution of the step itself will be paused.

[Example 1] SSPG 10 12 Pause program Nos. 10, 11 and 12 at the current step.



Error-generation/output-operation conditions
When one SSPG program is specified (only operand 1 is specified)

Status of the specified program	No program number error *1			Program number error *1
	Program already registered		Program not yet registered	
	Program running	Program not running		
Error	None	None	C03 “Non-registered program specification error”	C2C “Program number error”
Output operation	ON	OFF	OFF	OFF

* The errors shown in the table represent those that generate in accordance with the status of the specified program. Errors caused by other factors are excluded.
* 1 --- Program number error indicates specification of a number smaller than 1 or exceeding 64.

When multiple SSPG programs are specified (both operands 1 and 2 are specified)

Status of the specified program	No program number error *2			Program number error *1
	Registered program exists inside the specified range *3		None of programs inside the specified range are registered	
	Running program exists inside the specified range *4	None of programs inside the specified range are running		
Error	None	None	C03 “Non-registered program specification error”	C2C “Program number error”
Output operation	ON	OFF	OFF	OFF

* The errors shown in the table represent those that generate in accordance with the status of the specified program. Errors caused by other factors are excluded.
* 2 --- Program number error indicates specification of a number smaller than 1 or exceeding 64.
* 3 --- In this case, non-registered programs inside the specified range are not treated as a target of operation with EXPG, ABPG, SSPG and PSPG commands. This will not affect error generation or output operation.
* 4 --- In this case, programs not running (but already registered) inside the specified range are not treated as a target of operation with SSPG and RSPG commands. This will not affect error generation or output operation.

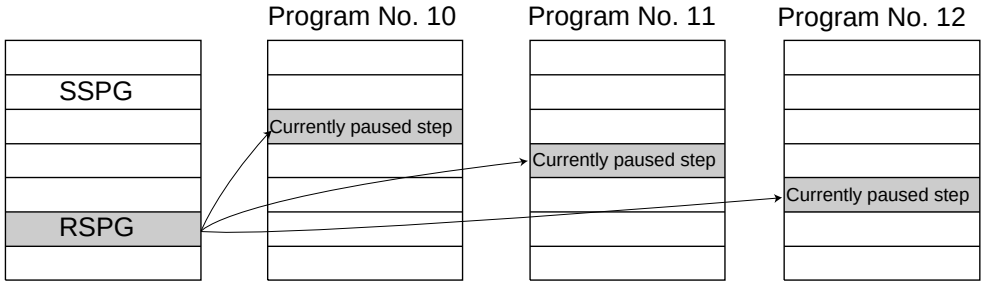
- RSPG (Resume program)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	RSPG	Program number	(Program number)	CC

[Function] Resume the programs from the one specified in operand 1 through the other specified in operand 2. Specification in operand 1 only is allowed.

(Note 1) Resuming a program will also resume the operation the program had been executing before the pause.

[Example 1] RSPG 10 12 Resume program Nos. 10, 11 and 12 from the paused step.



Error-generation/output-operation conditions
When one RSPG program is specified (only operand 1 is specified)

Status of the specified program	No program number error *1			Program number error *1
	Program already registered		Program not yet registered	
	Program running	Program not running		
Error	None	None	C03 “Non-registered program specification error”	C2C “Program number error”
Output operation	ON	OFF	OFF	OFF

* The errors shown in the table represent those that generate in accordance with the status of the specified program. Errors caused by other factors are excluded.
* 1 --- Program number error indicates specification of a number smaller than 1 or exceeding 64.

When multiple RSPG programs are specified (both operands 1 and 2 are specified)

Status of the specified program	No program number error *2			Program number error *1
	Registered program exists inside the specified range *3		None of programs inside the specified range are registered	
	Running program exists inside the specified range *4	None of programs inside the specified range are running		
Error	None	None	C03 “Non-registered program specification error”	C2C “Program number error”
Output operation	ON	OFF	OFF	OFF

* The errors shown in the table represent those that generate in accordance with the status of the specified program. Errors caused by other factors are excluded.
* 2 --- Program number error indicates specification of a number smaller than 1 or exceeding 64.
* 3 --- In this case, non-registered programs inside the specified range are not treated as a target of operation. This will not affect error generation or output operation.
* 4 --- In this case, programs not running (but already registered) inside the specified range are not treated as a target of operation with SSPG and RSPG commands. This will not affect error generation or output operation.

1.10 Position Operation

- PGET (Read position data)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PGET	Axis number	Position number	CC

[Function] Read to variable 199 the data of the axis number specified in operand 1 in the position data specified in operand 2.
Data will not be stored in variable 199 (the PGET command will not be executed), if no data is available in the position data table (the position data table shows “X.XXX” on the poison data display of the teaching pendant is blank in the PC software) when the PGET command is executed.

[Example 1]	PGET	2	3	Read to variable 199 the data of axis 2 at position 3.
[Example 2]	LET	1	2	Assign 2 to variable 1.
	LET	2	3	Assign 3 to variable 2.
	PGET	*1	*2	Read to variable 199 the data of the content of variable 1 (axis 2) at the content of variable 2 (position 3).

- PPUT (Write position data)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PPUT	Axis number	Position number	CP

[Function] Write the value in variable 199 to the axis number specified in operand 1 in the position data specified in operand 2.

[Example 1]	LET	199	150	Assign 150 to variable 199.
	PPUT	2	3	Write the content of variable 199 (150) to axis 2 at position 3.
[Example 2]	LET	199	150	Assign 150 to variable 199.
	LET	1	2	Assign 2 to variable 1.
	LET	2	3	Assign 3 to variable 2
	PPUT	*1	*2	Write the content of variable 199 (150) to the content of variable 1 (axis 2) at the content of variable 2 (position 3).

- PCLR (Clear position data)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PCLR	Position number	Position number	CP

[Function] Clear the position data from the one specified in operand 1 through the other specified in operand 2.
The cleared data is no longer stored, which is different from storage of “0.000.” The position data table shows “X.XXX” on the teaching pendant or the poison data display becomes blank in the PC software.

[Example 1]	PCLR	10	20	Clear the data from position Nos. 10 through 20.
[Example 2]	LET	1	10	Assign 10 to variable 1.
	LET	2	20	Assign 20 to variable 2.
	PCLR	*1	*2	Clear the data of the content of variable 1 (position 10) through the content of variable 2 (position 20).

- PCPY (Copy position data)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PCPY	Position number	Position number	CP

[Function] Copy the position data specified in operand 2 to the position number specified in operand 1.

[Example 1]	PCPY	20	10	Copy the data of position No. 10 to position No. 20.
[Example 2]	LET	1	20	Assign 20 to variable 1.
	LET	2	10	Assign 10 to variable 2.
	PCPY	*1	*2	Copy the data of the content of variable 2 (position 10) to the content of variable 1 (position 20).

- PRED (Read current position)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PRED	Axis pattern	Position number	CP

[Function] Read the current position of the axis specified in operand 1 to the position specified in operand 2.

[Example 1]

```
PRED   11   10   Read the current positions of axes 1 and 2 to position No.
                 10.
```

[Example 2]

The axis pattern can be specified indirectly using a variable.
When the command in [Example 1] is rephrased based on indirect specification using a variable:
11 (binary) → 3 (decimal)

```
LET    1    3    Assign 3 to variable 1.
PRED   *1   10
```

[Example 3]

```
LET    1    10   Assign 10 to variable 1.
PRED   11   *1   Read the current positions of axes 1 and 2 to the content of
                 variable 1 (position 10).
```

- PRDQ (Read current axis position (1 axis direct))

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PRDQ	Axis number	Variable number	CP

[Function] Read the current position of the axis number specified in operand 1 to the variable specified in operand 2.
The current position can be obtained more quickly than when a PRED command is used.
The current position of a synchronized slave axis can also be read.

[Example] PRDQ 2 100 Read the current position of axis 2 to variable 100.

- PTST (Check position data)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PTST	Axis pattern	Position number	CC

[Function] Check if valid data is contained in the axis pattern specified in operand 1 at the position number specified in operand 2.
The output will turn ON if none of the data specified by the axis pattern is available (the position data table shows “X.XXX” on the teaching pendant or the poison data display is blank in the PC software). “0” is recognizes as valid data.

[Example 1]

```
PTST  11  10  300
```

Turn ON output 300 if there are no valid values of axes 1 and 2 at position 10.
Output 300 will turn OFF if the position data is given as follows:

[Example 2] The axis pattern can be specified indirectly using a variable.
When the command in [Example 1] is rephrased based on indirect specification using a variable:
11 (binary) → 3 (decimal)

```
LET   1   3
PTST  *1  10  300
```

Assign 3 to variable 1.

[Example 3]

```
LET   1   11
PTST  11  *1  600
```

Assign 11 to variable 1.
Turn ON flag 600 if there are no valid values in the data of axes 1 and 2 at the content of variable 1 (position 11).
Flag 600 will turn ON if the position data is given as follows:

Position data display in PC software

No.	Axis1	Axis2	Vel	Acc	Dcl
10	100.000	50.000			
11					

- PVEL (Assign speed data)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PVEL	Speed	Position number	CP

[Function] Write the speed specified in operand 1 to the position number specified in operand 2.

(Note) If a negative value is written with a PVEL command, an alarm will generate when that position is specified in a movement operation, etc. Exercise caution.

[Example 1]	PVEL	100	10	Write speed 100 mm/s to position No. 10.
[Example 2]	LET	1	100	Assign 100 to variable 1.
	LET	2	10	Assign 10 to variable 2.
	PVEL	*1	*2	Write the content of variable 1 (speed 100 mm/s) to the content of variable 2 (position 10).

- PACC (Assign acceleration data)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PACC	Acceleration	Position number	CP

[Function] Write the acceleration specified in operand 1 to the position number specified in operand 2.

(Note) Range check is not performed for a PACC command. Be careful not to exceed the limit set for each actuator.

[Example 1]	PACC	0.3	10	Write acceleration 0.3 G to position No. 10.
[Example 2]	LET	100	0.3	Assign 0.3 to variable 100.
	LET	2	10	Assign 10 to variable 2.
	PACC	*100	*2	Write the content of variable 100 (acceleration 0.3 G) to the content of variable 2 (position 10).

- PDCL (Assign deceleration data)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PDCL	Deceleration	Position number	CP

[Function] Assign the deceleration data specified in operand 1 to the deceleration item in the position data specified in operand 2.
The deceleration is set in G and may include up to two decimal places.

[Example 1] PDCL 0.3 3 Assign 0.3 to the deceleration data at position No. 3.

- PAXS (Read axis pattern)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PAXS	Variable number	Position number	CP

[Function] Store the axis pattern at the position specified in operand 2 to the variable specified in operand 1.

[Example 1]

PAXS	1	99	Read the axis pattern at position 99 to variable 1. If the position is given as follows, “1” (binary 01) will be read to variable 1.

[Example 2]

LET	1	3	Assign 3 to variable 1.
LET	2	101	Assign 101 to variable 2.
PAXS	*1	*2	Read the axis pattern at the content of variable 2 (position 101) to the content of variable 1 (variable 3). If the point is given as follows, “3” (binary 11) will be stored in variable 3.

The table below shows different positions and corresponding values stored in a variable.

Position data display in PC software

No.	Axis1	Axis2		
98			.....	0 0 = 0 + 0 = 0
99	100.000		.....	0 1 = 0 + 1 = 1
100		150.000	.....	1 0 = 2 + 0 = 2
101	100.000	50.000	.....	1 1 = 2 + 1 = 3

- PSIZ (Check position data size)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PSIZ	Variable number	Prohibited	CP

[Function] Set an appropriate value in the variable specified in operand 1 in accordance with the parameter setting.

- When “Other parameter No. 23, PSIZ function type” = 0
  The maximum number of position data that can be stored in the controller will be set. (Regardless of whether the data are used or not.)
- When “Other parameter No. 23, PSIZ function type” = 1
  The number of point data used will be set.

[Example] PSIZ 1
When “Other parameter No. 23, PSIZ function type” = 0
The maximum number of position data that can be stored in variable 1 will be set.
When “Other parameter No. 23, PSIZ function type” = 1
The number of point data currently used will be set in variable 1.

- GVEL (Get speed data)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	GVEL	Variable number	Position number	CP

[Function] Obtain speed data from the speed item in the position data specified in operand 2, and set the value in the variable specified in operand 1.

[Example] GVEL 100 10 Set the speed data at position No. 10 in variable 100.

Position data display in PC software

Position No.	Axis1	Axis2	Vel	Acc	Dcl
1					
2					
• • •					
10	50.000	100.000	200	0.30	0.30
• •					

If the position data is set as above when the command is executed, 200 will be set in variable 100.

- GACC (Get acceleration data)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	GACC	Variable number	Position number	CP

[Function] Obtain acceleration data from the acceleration item in the position data specified in operand 2, and set the value in the variable specified in operand 1.

[Example] GACC 100 10 Set the acceleration data at position No. 10 in variable 100.

Position data display in PC software

No.	Axis1	Axis2	Vel	Acc	Dcl
1					
2					
• • •					
10	50.000	100.000	200	0.30	0.30
• •					

If the position data is set as above when the command is executed, 0.3 will be set in variable 100.

- GDCL (Get deceleration data)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	GDCL	Variable number	Position number	CP

[Function] Obtain deceleration data from the deceleration item in the position data specified in operand 2, and set the value in the variable specified in operand 1.

[Example] GDCL 100 10 Set the deceleration data at position No. 10 in variable 100.

Position data display in PC software

No.	Axis1	Axis2	Vel	Acc	Dcl
1					
2					
• • •					
10	50.000	100.000	200	0.30	0.30
• •					

If the position data is set as above when the command is executed, 0.3 will be set in variable 100.

1.11 Actuator Control Declaration

- VEL (Set speed)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	VEL	Speed	Prohibited	CP

[Function] Set the actuator travel speed in the value specified in operand 1.
The unit is mm/s.
The maximum speed will vary depending on the model of the actuator connected. Set a speed not exceeding the applicable maximum speed.

(Note 1) Decimal places cannot be used. An error will generate
(Note 2) The minimum speed is 1 mm/s.

```
[Example 1]   VEL    100          Set the speed to 100 mm/s.
              MOVP   1            Move to point 1 at 100 mm/s.

[Example 2]   VEL    500          Set the speed to 500 mm/s.
              MOVP   2            Move to point 2 at 500 mm/s.

[Example 3]   LET    1     300    Assign 300 to variable 1.
              VEL    *1           Set the speed to the content of variable 1 (300 mm/s).
```

- OVRD (Override)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	OVRD	Speed ratio	Prohibited	CP

[Function] Reduce the speed in accordance with the ratio specified in operand 1 (speed coefficient setting). The speed ratio is set in a range from 1 to 100%.
A speed command specifying a speed below 1 mm/sec can be generated using OVRD.

[Example 1]

VEL	100	Set the speed to 100 mm/s.
OVRD	50	Reduce the speed to 50%. As a result, the actual speed will become 50 mm/s.

Command limit speed for smooth operation: Travel distance per encoder pulse [mm/pulse]/time [msec]
Command limit speed that can be generated: Travel distance per encoder pulse [mm/pulse]/time [msec]
(Smoothness of actual operation cannot be guaranteed. Movement must be checked on the actual machine.)
[Calculation formula of travel distance per encoder pulse]
Rotary encoder
Travel distance per encoder pulse [mm/pulse] = (Screw lead [0.001 mm] x Gear ratio numerator)
/ (Encoder resolution [pulses/rev] x Gear ratio denominator
/ (2 ^ Encoder division ratio)

Linear encoder
Travel distance per encoder pulse [mm/pulse] = Encoder resolution (0.001 μm/pulse) x 1000
/ (2 ^ Encoder division ratio)

(Reference) Use the values of the following parameters for the above calculation formulas:

Encoder resolution:	Axis-specific parameter No. 42
Encoder division ratio:	Axis-specific parameter No. 43
Screw lead:	Axis-specific parameter No. 47
Gear ratio numerator:	Axis-specific parameter No. 50
Gear ratio denominator:	Axis-specific parameter No. 51

- ACC (Set acceleration)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	ACC	Acceleration	Prohibited	CP

[Function] Set the travel acceleration of the actuator.
The maximum acceleration will vary depending on the load and model of the actuator connected.
The acceleration is set in G and may include up to two decimal places.

(Note) If the position data contains no acceleration AND acceleration is not set by an ACC command, the actuator will move based on the default value set in “All-axis parameter No. 11, Default acceleration.”

[Example 1] ACC 0.3 Set the acceleration to 0.3 G.

(Note) Setting an acceleration exceeding the specified range for the actuator may generate an error. It may also result in a failure or shorter product life.

- DCL (Set deceleration)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	DCL	Deceleration	Prohibited	CP

[Function] Set the travel deceleration of the actuator.
The maximum deceleration will vary depending on the load and model of the actuator connected.
The deceleration is set in G and may include up to two decimal places.

(Note) If the position data contains no deceleration AND deceleration is not set by a DCL command, the actuator will move based on the default value set in “All-axis parameter No. 12, Default deceleration.”
A DCL command cannot be used with CIR and ARC commands.

[Example] DCL 0.3 Set the deceleration to 0.3 G.

(Note) Setting a deceleration exceeding the specified range for the actuator may generate an error. It may also result in a failure or shorter product life.

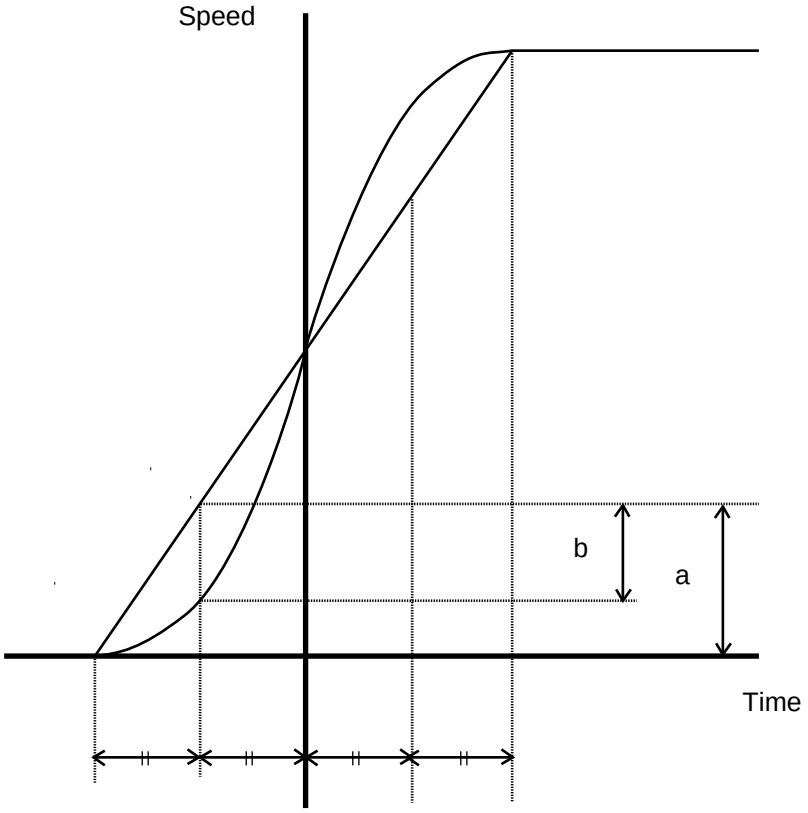
- SCRV (Set sigmoid motion ratio)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	SCRV	Ratio	Prohibited	CP

[Function] Set the ratio of sigmoid motion control of the actuator in the value specified in operand 1. The ratio is set as an integer in a range from 0 to 50 (%).

$$\frac{b}{a} \text{ X } 100\ (\%)$$

If the ratio is not set using this command or 0% is set, a trapezoid motion will be implemented.
A SCRV command can be used with the following commands:
MOVP, MOVL, MVPI, MVLI, JBWF, JBWN, JFWF, JFWN
MOVD, MVDI



[Example 1] SCRV 30 Set the sigmoid motion ratio to 30%.

- OFST (Set offset)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	OFST	Axis pattern	Offset value	CP

[Function] Reset the target value by adding the offset value specified in operand 2 to the original target value when performing the actuator movement specified in operand 1.
The offset is set in mm, and the effective resolution is 0.001 mm.
A negative offset may be specified as long as the operation range is not exceeded.
An OFST command is processed with respect to soft axes before a BASE shift.

(Note) An OFST command cannot be used outside the applicable program. To use OFST in multiple programs, the command must be executed in each program.
An OFST command cannot be used with MVPI, MVLI and MVDI commands.

[Example 1]

```
OFST   10   50   Add 50 mm to the specified position of axis 2.
  :
OFST   10   0    Return the offset of axis 2 to 0.
```

[Example 2] The axis pattern can be specified indirectly using a variable.
When the command in [Example 1] is rephrased based on indirect specification using a variable:
10 (binary) → 2 (decimal)

```
LET    1    2    Assign 2 to variable 1.
OFST   *1   50
  :
OFST   *1   0
```

[Example 3]

```
LET    2    50   Assign 50 to variable 2.
OFST   1    *2   Add the content of variable 2 (50 mm) to the specified
                 position of axis1.
```

- DEG (Set arc angle)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	DEG	Angle	Prohibited	CP

[Function] Set a division angle for the interpolation implemented by a CIR (move along circle) or ARC (move along arc) command.
When CIR or ARC is executed, a circle will be divided by the angle set here to calculate the passing points.
The angle is set in a range from 0 to 120 degrees.
If the angle is set to “0,” an appropriate division angle will be calculated automatically so that the actuator will operate at the set speed (maximum 180 degrees).
The angle is set in degrees and may include up to one decimal place.

(Note) If a CIR or ARC command is executed without setting an angle with this command, the default value registered in “All-axis parameter No. 30, Default division angle” will be used.

[Example] DEG 10 Set the division angle to 10 degrees.

- BASE (Specify axis base)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	BASE	Axis number	Prohibited	CP

[Function] Count the axes sequentially based on the axis number specified in operand 1 being the first axis.
A BASE command can be used with PRED, PRDQ, AXST, actuator-control and zone commands. Note that each zone range is assigned to the actuator via parameter.

[Example 1]

```
HOME  1        Axis 1 returns to the home.
BASE  2        Axis 2 is considered the first axis.
HOME  1        Axis 2 returns to the home.
```

[Example 2]

```
LET   1   2    Assign 2 to variable 1.
BASE  *1       The content of variable 1 (axis 2) will be considered as the
               first axis.
```

- GRP (Set group axes)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	GRP	Axis pattern	Prohibited	CP

[Function] Allow only the position data of the axis pattern specified in operand 1 to become valid.
The program assumes that there are no data for other axes not specified.
When multiple programs are run simultaneously, assigning axes will allow the same position data to be used effectively among the programs.
A GRP command can be used with operand axis-pattern specification commands excluding an OFST command, as well as with servo operation commands using position data.
A GRP command is processed with respect to soft axes before a BASE shift.

[Example 1]

```
GRP     10          Data of axis 2 becomes valid.
```

[Example 2] The axis pattern can be specified indirectly using a variable.
When the command in [Example 1] is rephrased based on indirect specification using a variable:
10 (binary) → 2 (decimal)

```
LET     1       2       Assign 2 to variable 1.
GRP     *1
```

- HOLD (Hold: Declare axis port to pause)

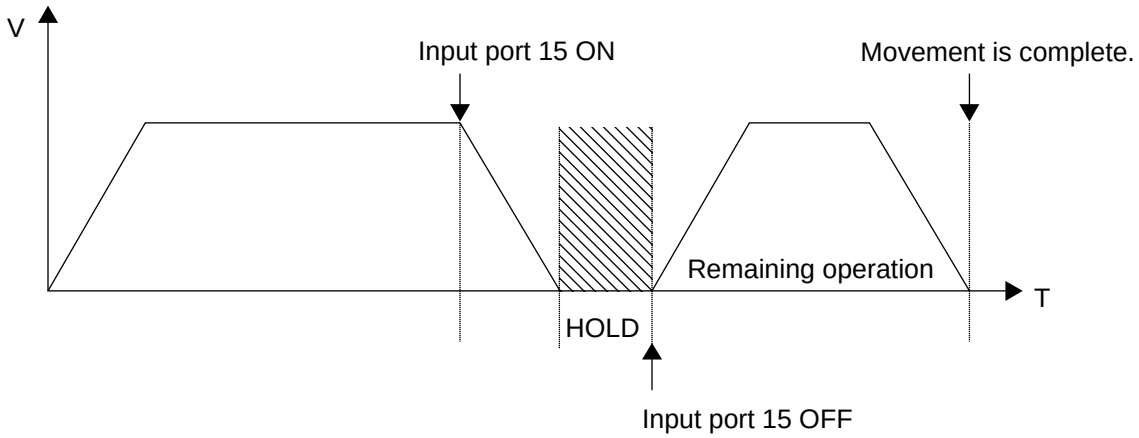
Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	HOLD	(Input port, global flag)	(HOLD type)	CP

[Function] Declare an input port or global flag to pause while a servo command is being executed. When operation is performed on the input port or global flag specified in operand 1, the current servo processing will pause. (If the axes are moving, they will decelerate to a stop.) If nothing is specified in operand 1, the current pause declaration will become invalid.

A HOLD type can be specified in operand 2.
[HOLD type]
0 = Contact a (Deceleration stop)
1 = Contact b (Deceleration stop)
2 = Contact b (Deceleration stop → Servo OFF (The drive source will not be cut off))
The HOLD type is set to “0” (contact a) when the program is started.
If nothing is specified in operand 2, the current HOLD type will be used.
Using other task to issue a servo ON command to any axis currently stopped via a HOLD servo OFF will generate an “Error No. C66, Axis duplication error.” If the servo of that axis was ON prior to the HOLD stop, the system will automatically turn on the servo when the HOLD is cancelled. Therefore, do not issue a servo ON command to any axis currently stopped via a HOLD servo OFF.
If any axis currently stopped via a HOLD servo OFF is moved by external force, etc., from the stopped position, and when the servo of that axis was ON prior to the HOLD stop, the axis will move to the original stopped position when the HOLD is cancelled before resuming operation.

(Note 1) The input port or global flag specified by a HOLD declaration will only pause the axes used in the task (program) in which the HOLD is declared. The declaration will not be valid on axes used in different tasks (programs).
(Note 2) An input port or global flag to pause is valid for all active servo commands other than a SVOF command. (A deceleration stop will also be triggered in J□W□ and PATH operations.)
(Note 3) Following a pause of home return, the operation will resume from the beginning of the home-return sequence.

[Example] HOLD 15 0 The axes will decelerate to a stop when input port 15 turns ON.



- CANC (Cancel: Declare axis port to abort)

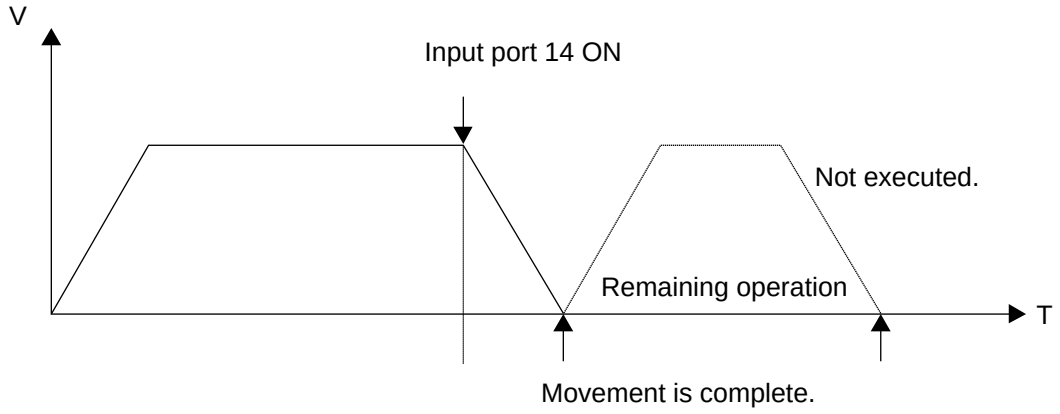
Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	CANC	(Input port, global flag)	(CANC type)	CP

[Function] Declare an input port or global flag to abort while a servo command is being executed. When operation is performed on the input port or global flag specified in operand 1, the current servo processing will be aborted. (If the axes are moving, they will decelerate to a stop before the processing is aborted.)
If nothing is specified in operand 1, the current abort declaration will become invalid.

A CANC type can be specified in operand 2.
[CANC type]
0 = Contact a (Deceleration stop)
1 = Contact b (Deceleration stop)
The CANC type is set to “0” (contact a) when the program is started.
If nothing is specified in operand 2, the current CANC type will be used.

(Note 1) The input port or global flag specified by a CANC command will only abort the axes used in the task (program) in which the CANC is declared. The declaration will not be valid on axes used in different tasks (programs).
(Note 2) An input port or global flag to pause is valid for all active servo commands other than a SVOF command. (A deceleration stop will also be triggered in JXWX and PATH operations.)

[Example] CANC 14 0 The axes will decelerate to a stop when input port 14 turns ON.



- VLMX (Specify VLMX speed)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	VLMX	Prohibited	Prohibited	CP

[Function] Set the actuator travel speed to the VLMX speed (normally maximum speed).
Executing a VLMX command will set the value registered in “Axis-specific parameter No. 29, VLMX speed” as the travel speed.

(Note) If the VLMX speed is specified in a continuous position travel command (PATH, PSPL), the target speed to each position will become a composite VLMX speed not exceeding the maximum speed of each axis set in “Axis-specific parameter No. 28, Maximum operating speed of each axis.” To make the target speed constant, a desired speed must be expressly specified using a VEL command.

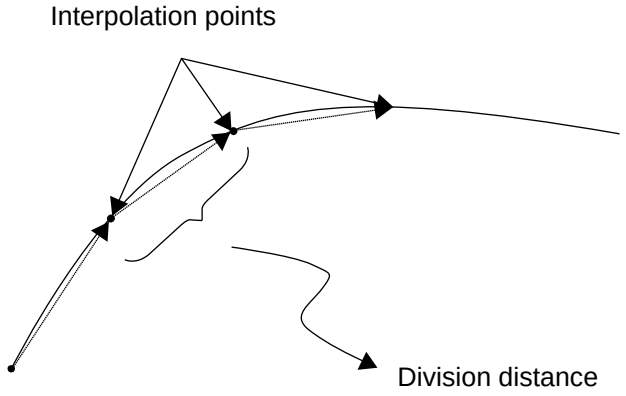
[Example]

```
VEL     1000 ┐
MOVP    1    │ The speed becomes 1000 mm/sec in this section.
MOVP    2    ┘
VLMX         ┐
MOVP    3    │ The speed becomes VLMX mm/sec in this section.
MOVP    4    ┘
```

- DIS (Set division distance at spline movement)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	DIS	Distance	Prohibited	CP

[Function] Set a division distance for the interpolation implemented by a PSPL (move along spline) command.
When a PSPL command is executed, a passing point will be calculated at each distance set here and the calculated passing points will be used as interpolation points.
If the distance is set to “0,” an appropriate division distance will be calculated automatically so that the actuator will operate at the set speed
The distance is input in mm.



(Note) If a PSPL command is executed without setting a distance with a DIS command, the default value registered in “All-axis parameter No. 31, Default division distance” will be used.

[Example] DIS 10 Set the division distance to 10 mm.

- POTP (Set PATH output type)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	POTP	0 or 1	Prohibited	CP

[Function] Set the output type in the output field to be used when a PATH or PSPL command is executed.
When a PATH or PSPL command is executed, the output will operate as follows in accordance with the setting of the POTP command.

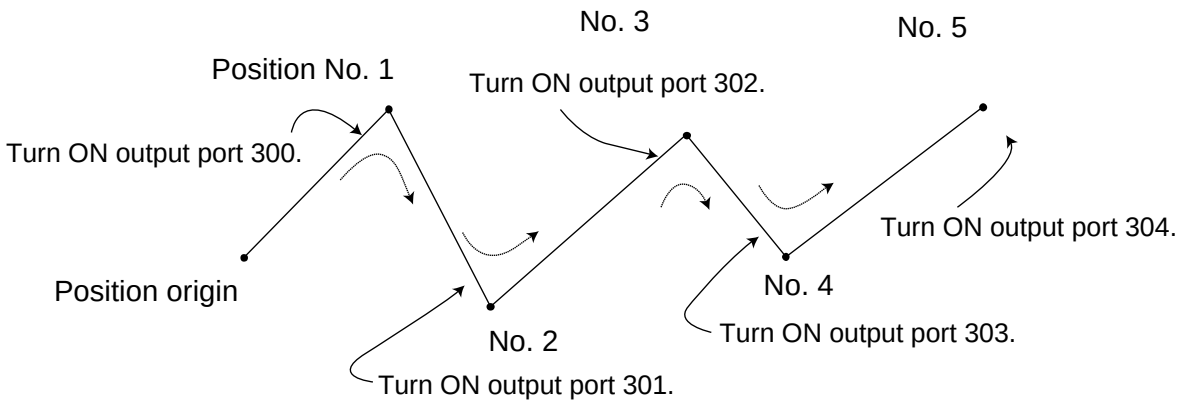
(1) POTP [Operand 1] = 0 (ON upon completion of operation)
The output port or flag will turn ON upon completion of operation.
(2) POTP [Operand 1] = 1 (Increment and output on approaching each position; ON upon completion of operation for the last position)
During PATH or PSPL operation, the output port number or flag number specified in the output field will be incremented and turned ON when each specified position approaches. At the last position, however, the output will turn ON upon completion of operation. This setting provides a rough guide for output in sequence control.

(Note 1) The default value of POTP, before it is set, is “0.”
(Note 2) If POTP = 1 and there is no valid data at the specified position, the output number will be incremented but the output will not turn ON. (The output number will be incremented regardless of the size of position numbers specified in operands 1 and 2 in a PATH or PSPL command.)

[Example]

```
POTP  1
PATH  1     5     300
```

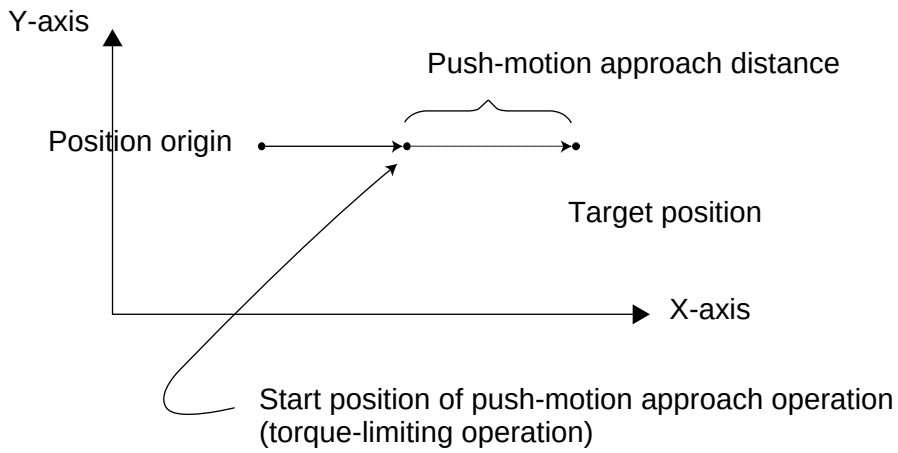
Turn ON output port Nos. 300 through 304 sequentially each time a specified position approaches during a pass movement from position Nos. 1 through 5, starting from the first position.



- PAPR (Set push-motion approach distance, speed)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PAPR	Distance	Speed	CP

[Function] Set the operation to be performed when a PUSH command is executed.
Set the distance (push-motion approach distance) over which push-motion approach operation (torque-limiting operation) will be performed in operand 1 (in mm), and set the speed (push-motion approach speed) at which push-motion approach operation (torque-limiting operation) will be performed in operand 2 (in mm/sec).
The push-motion approach distance specified in operand 1 may contain up to three decimal places, while the speed specified in operand 2 cannot contain any decimal place.



[Example]

PAPR	100	30	Set the push-motion approach distance in a PUSH command to 100 mm and the push-motion approach speed to 30 mm/sec.
MOVP	2		Move to position No. 2.
PUSH	10		Move by push motion from position No. 2 to position No. 10.

(Note) The push-motion approach speed in an OVRD command will be clamped by the minimum speed of 1 mm/sec. (Correct push-motion operation is not guaranteed at the minimum speed. Operation at slow push-motion approach must be checked on the actual machine by considering the effects of mechanical characteristics, etc.)

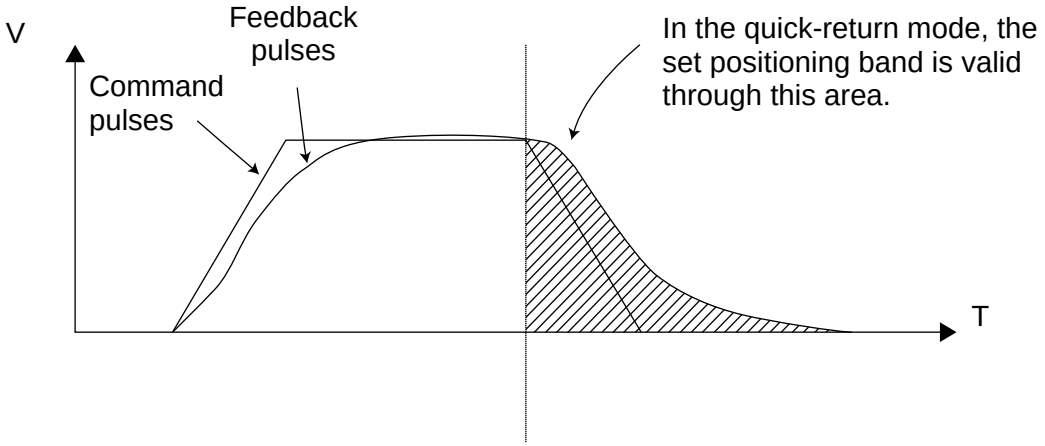
- QRTN (Set quick-return mode)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	QRTN	0 or 1	Prohibited	CP

[Function] Set and cancel the quick-return mode.

(1) QRTN [Operand 1] = 0 (Normal mode)
Positioning is deemed complete when all command pulses have been output and the current position is inside the positioning band.
* If a deceleration command is currently executed in the quick-return mode, the system will wait for all command pulses to be output.

(2) QRTN [Operand 1] = 1 (Quick-return mode)
Positioning is deemed complete when "a normal deceleration command is currently executed (excluding deceleration due to a stop command, etc.) or all command pulses have been output" AND "the current position is inside the positioning band." This setting is used to perform other processing during deceleration, in conjunction with a PBND command.



(Note 1) The quick-return mode will be cancelled when the program ends. (The positioning band set by a PBND command will not be cancelled.)

(Note 2) If a given axis is used even once in the quick-return mode, the program will not release the right to use the axis until the QRTN is set to "0" (normal mode) or the program ends. Any attempt to use the axis from other program will generate an "Error No. C66, Axis duplication error."

(Note 3) Following a return from a normal deceleration command in the quick-return mode, the next positioning will start after all command pulses for the previous positioning have been output. Therefore, in the quick-return mode a simple reciprocating operation will require a longer tact time because of the extra completion check. In this sense, this setting should be used only if you wish to reduce the overall tact time by performing other processing during deceleration.

(Note 4) The quick-return mode represents very irregular processing. Therefore, be sure to revert to the normal mode when the overlay processing is completed in the necessary section.

(Note 5) The quick-return mode cannot be used with a push-motion travel command or arc interpolation command.

1.12 Actuator Control Command

- SV□□ (Turn ON/OFF servo)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	SV□□	Axis pattern	Prohibited	PE

[Function] Turn ON/OFF the servos of the axes specified by the axis pattern in operand 1.

SV□□
- ON ..... Turn ON the servo.
- OF ..... Turn OFF the servo.

[Example 1]

```
SVON    11
```

Turn ON the servos of axes 1 and 2. Nothing will occur if the axis servos are already ON.

[Example 2]

The axis pattern can be specified indirectly using a variable.
When the command in [Example 1] is rephrased based on indirect specification using a variable:
11 (binary) → 3 (decimal)

```
LET     1       3
SVON    *1
```

Assign 3 to variable 1.

- HOME (Return to home)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	HOME	Axis pattern	Prohibited	PE

[Function] Perform home return of the axes specified by the axis pattern in operand 1.
The servo of each home-return axis will turn ON automatically.
The output will turn OFF at the start of home return, and turn ON when the home return is completed.

(Note) Following a pause of home return, the operation will resume from the beginning of the home-return sequence.
The home-return operation of an absolute-encoder axis is a movement to the rotation data reset position and may not necessarily be a movement to the preset home coordinate (including 0). If an output function specification value of “12” (All-valid-axed home (=0) output) or “14” (All-valid-axes preset home coordinate output) is stored in the I/O parameter “Output function setting nnn,” use a MOVP command, not a HOME command, when moving each absolute-encoder axis for the purpose of turning ON the applicable output.
If the operation is stopped or cancelled while a HOME command is being executed for an absolute-encoder axis in a mode other than the absolute reset mode provided by the PC software or teaching pendant, an “actual-position soft limit error” may generate depending on the position. It is not recommended to perform home return other than for the purpose of adjusting an absolute-encoder axis.

[Example 1]

```
HOME    11          Axes 1 and 2 return to the home.
```

[Example 2] The axis pattern can be specified indirectly using a variable.
When the command in [Example 1] is rephrased based on indirect specification using a variable:
11 (binary) → 3 (decimal)

```
LET     1     3     Assign 3 to variable 1.
HOME    *1
```

- MOVP (Move PTP by specifying position data)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	MOVP	Position number	Prohibited	PE

[Function] Move the actuator to the position corresponding to the position number specified in operand 1, without interpolation (PTP stands for “Point-to-Point”).
The output will turn OFF at the start of axis movement, and turn ON when the movement is complete.

[Example 1]

```
VEL   100
MOVP  1
```

Set the speed to 100 mm/s.
Move the axes to the position corresponding to position No. 1 (200, 100).

[Example 2]

```
VEL   100
LET   1     2
MOVP  *1
```

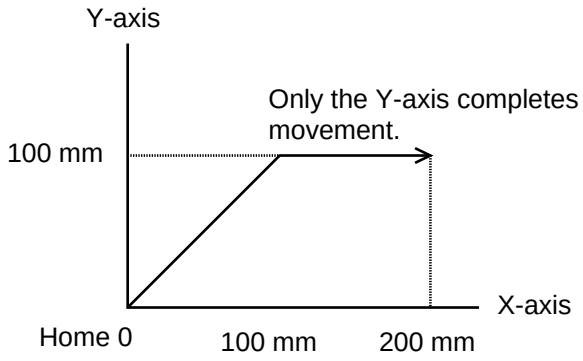
Set the speed to 100 mm/s.
Assign 2 to variable 1.
Move the axes to the position corresponding to the content of variable 1 (position No. 2, or (100, 100)).

Position data display in PC software

No.	Axis1 (X-axis)	Axis2 (Y-axis)	Vel	Acc	Dcl
1	200.000	100.000			
2	100.000	100.000			

(Note) If acceleration or deceleration is not specified in the position data table or by an ACC (DCL) command, the actuator will operate according to all-axis parameter No. 11, “Default acceleration” or all-axis parameter No. 12, “Default deceleration.”

Travel path from the home to the position corresponding to position No. 1 (200, 100)



Each axis moves at 100 mm/s.

- MOVL (Move by specifying position data)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	MOVL	Position number	Prohibited	PE

[Function] Move the actuator to the position corresponding to the position number specified in operand 1, with interpolation.
The output will turn OFF at the start of axis movement, and turn ON when the movement is complete.

[Example 1]

VEL	100		Set the speed to 100 mm/s.
MOVL	1		Move the axes to the position corresponding to position No. 1 (200, 100), with interpolation.

[Example 2]

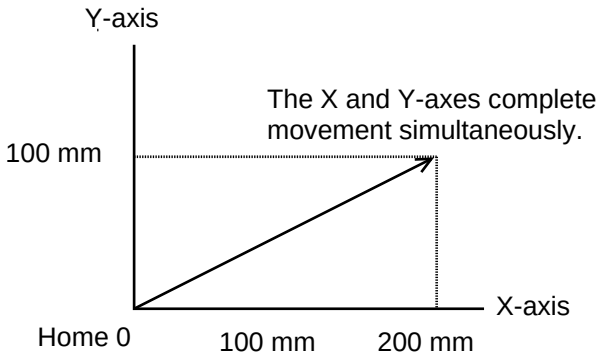
VEL	100		Set the speed to 100 mm/s.
LET	1	2	Assign 2 to variable 1.
MOVL	*1		Move the axes to the position corresponding to the content of variable 1 (position No. 2, or (100, 100)), with interpolation.

Position data display in PC software

No.	Axis1 (X-axis)	Axis2 (Y-axis)	Vel	Acc	Dcl
1	200.000	100.000			
2	100.000	100.000			

(Note) If acceleration or deceleration is not specified in the position data table or by an ACC (DCL) command, the actuator will operate according to all-axis parameter No. 11, “Default acceleration” or all-axis parameter No. 12, “Default deceleration.”

Travel path from the home to the position corresponding to position No. 1 (200, 100)



The tip of each axis moves at 100 mm/s.

- MVPI (Move via incremental PTP)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	MVPI	Position number	Prohibited	PE

[Function] Move the actuator, without interpolation, from the current position by the travel distance corresponding to the position number specified in operand 1.
The output will turn OFF at the start of axis movement, and turn ON when the movement is complete.

[Example 1]

VEL	100		Set the speed to 100 mm/s.
MVPI	1		If the current position is (50, 50) and position No. 1 is set to (150, 100), the axes will move 150 in the X direction and 100 in the Y direction (200, 150) from the current position.

[Example 2]

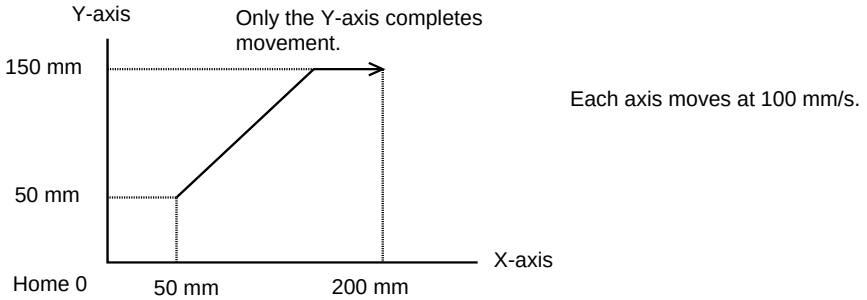
VEL	100		Set the speed to 100 mm/s.
LET	1	2	Assign 2 to variable 1.
MVPI	*1		Move from the current position by the travel distance corresponding to the content of variable 1 (position No. 2, or (100, 100)).

Position data display in PC software

No.	Axis1 (X-axis)	Axis 2 (Y-axis)	Vel	Acc	Dcl
1	150.000	100.000			
2	100.000	100.000			

(Note) If acceleration or deceleration is not specified in the position data table or by an ACC (DCL) command, the actuator will operate according to all-axis parameter No. 11, “Default acceleration” or all-axis parameter No. 12, “Default deceleration.”

Travel path from (50, 50) by the travel distance corresponding to position No. 1 (150, 100)



(Note) If the specified travel distance is equal to or less than the travel distance per encoder pulse [mm/pulse], the axis may not move.
[Calculation formula of travel distance per encoder pulse]
Rotary encoder
Travel distance per encoder pulse [mm/pulse] = (Screw lead [0.001 mm] x Gear ratio numerator) / (Encoder resolution [pulses/rev] x Gear ratio denominator / (2 ^ Encoder division ratio)
Linear encoder
Travel distance per encoder pulse [mm/pulse] = Encoder resolution (0.001 μm/pulse) x 1000 / (2 ^ Encoder division ratio)
(Reference) Use the values of the following parameters for the above calculation formulas:
Encoder resolution: Axis-specific parameter No. 42
Encoder division ratio: Axis-specific parameter No. 43
Screw lead: Axis-specific parameter No. 47
Gear ratio numerator: Axis-specific parameter No. 50
Gear ratio denominator: Axis-specific parameter No. 51

- MVLI (Move via incremental interpolation)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	MVLI	Position number	Prohibited	PE

[Function] Move the actuator, with interpolation, from the current position by the travel distance corresponding to the position number specified in operand 1.
The output will turn OFF at the start of axis movement, and turn ON when the movement is complete.

[Example 1]

VEL	100		Set the speed to 100 mm/s.
MVLI	1		If the current position is (50, 50) and position No. 1 is set to (150, 100), the axes will move 150 in the X direction and 100 in the Y direction (200, 150) from the current position, with interpolation.

[Example 2]

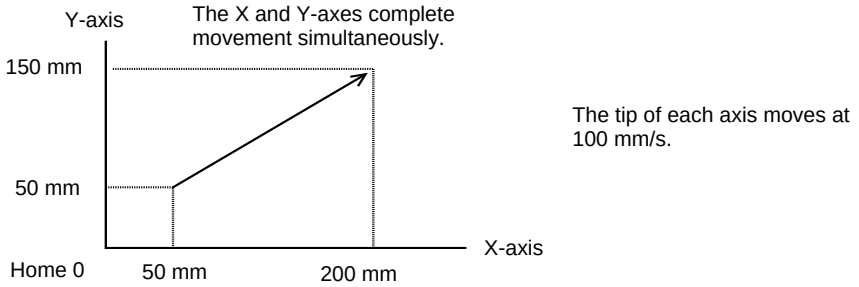
VEL	100		Set the speed to 100 mm/s.
LET	1	2	Assign 2 to variable 1.
MVLI	*1		Move from the current position by the travel distance corresponding to the content of variable 1 (position No. 2, or (100, 100)).

Position data display in PC software

No.	Axis1 (X-axis)	Axis 2 (Y-axis)	Vel	Acc	Dcl
1	150.000	100.000			
2	100.000	100.000			

(Note) If acceleration or deceleration is not specified in the position data table or by an ACC (DCL) command, the actuator will operate according to all-axis parameter No. 11, “Default acceleration” or all-axis parameter No. 12, “Default deceleration.”

Travel path from (50, 50) by the travel distance corresponding to position No. 1 (150, 100)



(Note) If the specified travel distance is equal to or less than the travel distance per encoder pulse [mm/pulse], the axis may not move.
[Calculation formula of travel distance per encoder pulse]
Rotary encoder
Travel distance per encoder pulse [mm/pulse] = (Screw lead [0.001 mm] x Gear ratio numerator) / (Encoder resolution [pulses/rev] x Gear ratio denominator / (2 ^ Encoder division ratio)
Linear encoder
Travel distance per encoder pulse [mm/pulse] = Encoder resolution (0.001 μm/pulse) x 1000 / (2 ^ Encoder division ratio)
(Reference) Use the values of the following parameters for the above calculation formulas:
Encoder resolution: Axis-specific parameter No. 42
Encoder division ratio: Axis-specific parameter No. 43
Screw lead: Axis-specific parameter No. 47
Gear ratio numerator: Axis-specific parameter No. 50
Gear ratio denominator: Axis-specific parameter No. 51

- MOVD (Move via direct value specification)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	MOVD	Target position	(Axis pattern)	PE

[Function] Move the axis specified by the axis pattern in operand 2, to the target position corresponding to the value specified in operand 1. If operand 2 is not specified, all axes will be moved.
The output will turn OFF at the start of axis movement, and turn ON when the movement is complete.
The target position is set in mm, and the set value is valid to the third decimal place.

[Example 1]	MOVD	100	10	Move axis 2 to position 100.
[Example 2]	LET	1	100	Assign 100 to variable 1.
	MOVD	*1	11	Move all axes to the content of variable 1 (100).

- MVDI (Move relatively via direct value specification)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	MVDI	Travel distance	(Axis pattern)	PE

[Function] Move the axis specified by the axis pattern in operand 2 from its current position by the travel distance corresponding to the value specified in operand 1. If operand 2 is not specified, all axes will be moved.
The output will turn OFF at the start of axis movement, and turn ON when the movement is complete.
The travel distance is set in mm, and the set value is valid to the third decimal place.

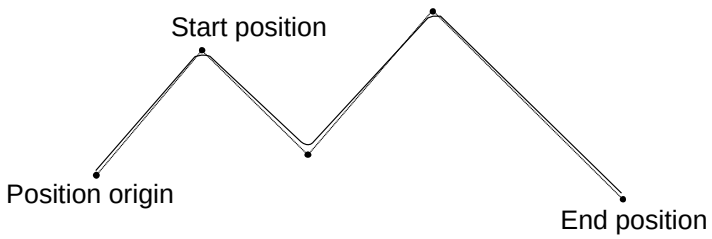
(Note) If the specified travel distance is equal to or less than the travel distance per encoder pulse [mm/pulse], the axis may not move.
[Calculation formula of travel distance per encoder pulse]
Rotary encoder
Travel distance per encoder pulse [mm/pulse] = (Screw lead [0.001 mm] x Gear ratio numerator)
/ (Encoder resolution [pulses/rev] x Gear ratio denominator
/ (2 ^ Encoder division ratio)
Linear encoder
Travel distance per encoder pulse [mm/pulse] = Encoder resolution (0.001 μm/pulse) x 1000
/ (2 ^ Encoder division ratio)
(Reference) Use the values of the following parameters for the above calculation formulas:
Encoder resolution: Axis-specific parameter No. 42
Encoder division ratio: Axis-specific parameter No. 43
Screw lead: Axis-specific parameter No. 47
Gear ratio numerator: Axis-specific parameter No. 50
Gear ratio denominator: Axis-specific parameter No. 51

[Example 1]	MVDI	30	11	Move all axes from the current position by 30 mm in the positive direction.
[Example 2]	LET	1	-100	Assign -100 to variable 1.
	MVDI	*1	1	Move axis 1 from the current position in accordance with the content of variable 1 (-100), or by 100 mm in the negative direction.

- PATH (Move along path)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PATH	Start position number	End position number	PE

[Function] Move continuously from the position specified in operand 1 to the position specified in operand 2.
The output type in the output field can be set using an actuator-declaration command POTP.
Increasing the acceleration will make the passing points closer to the specified positions.
If invalid data is set for any position number between the start and end position numbers, that position number will be skipped during continuous movement.

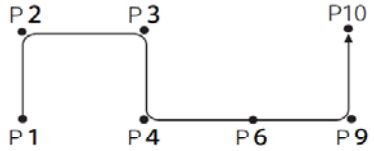


(Note 1) Multi-dimensional movement can be performed using a PATH command.
In this case, input in operand 1 the point number of the next target, instead of the predicted current position upon execution of the applicable command.
(Inputting a point number corresponding to the predicted current position will trigger movement to the same point during continuous movement, thereby causing the speed to drop.)

(Note 2) Continuous operation is possible even when all positions are not continuous.
As shown in the example, specify the position number of the discontinuous position for both the start position number and end position number in the PATH command. In this example, position No. 6 is discontinuous.

[Example] The actuator moves continuously in the sequence of position Nos. 1 → 2 → 3 → 4 → 6 → 9 → 10.

```
PATH   1   4
PATH   6   6     (Discontinuous position)
PATH   9   10
```



[Example 1]

```
VEL    100          Set the speed to 100 mm/s.
PATH   100   120    Move continuously from position Nos. 100 to 120.
```

[Example 2]

```
VEL    100          Set the speed to 100 mm/s.
LET    1     50     Assign 50 to variable 1.
LET    2     100    Assign 100 to variable 2.
PATH   *1    *2     Move continuously along the positions from the content of
                    variable 1 (position No. 50) to the content of variable 2
                    (position No. 100).
```

- J□W□ (Jog)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	J□W□	Axis pattern	Input, output, flag number	PE

[Function] The axes in the axis pattern specified in operand 1 will move forward or backward while the input or output port or flag specified in operand 2 is ON or OFF.

JBWF.....Move backward while the specified port is OFF.
JBWN.....Move backward while the specified port is ON.
JFWF.....Move forward while the specified port is OFF.
JFWN.....Move forward while the specified port is ON.

(Note 1) This command is also valid on an axis not yet completing home return. In this case, the maximum speed will be limited by “All-axis parameter No. 15, Maximum jog speed before home return.” Since coordinate values do not mean anything before home return, pay due attention to prevent contact with the stroke ends.

[Example 1]

```
VEL    100          Set the speed to 100 mm/s.
JBWF   11     10    Move axes 1 and 2 backward while input 10 is OFF.
```

[Example 2]

The axis pattern can be specified indirectly using a variable.
When the command in [Example 1] is rephrased based on indirect specification using a variable:
11 (binary) → 3 (decimal)

```
VEL    100          Set the speed to 100 mm/s.
LET    1      3     Assign 3 to variable 1.
JBWF   *1     10
```

[Example 3]

```
VEL    100          Set the speed to 100 mm/s.
LET    5      20    Assign 20 to variable 5.
JFWN   10     *5    Move axis 2 forward while the content of variable 5 (input
                    20), is ON.
```

- STOP (Stop movement)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	STOP	Axis pattern	Prohibited	CP

[Function] Decelerate and stop the axes specified by the axis pattern in operand 1.

(Note 1) A STOP command can be used with all active servo commands other than a SVOF command.

(Note 2) A STOP command only issues a deceleration-stop command (operation stop) to a specified axis pattern and does not wait for stopping to complete. Issuing other servo commands to a decelerating axis will either become invalid or generate an “axis duplication error,” etc.
Set a timer, etc., in the program so that the next servo command will be issued after a sufficient deceleration-stop processing time elapses.
Even when a STOP command is to be issued to an axis currently stopped, provide a minimum interval of 0.1 second before the next servo command is issued.

[Example 1]

```
STOP    11          Decelerate and stop axes 1 and 2.
```

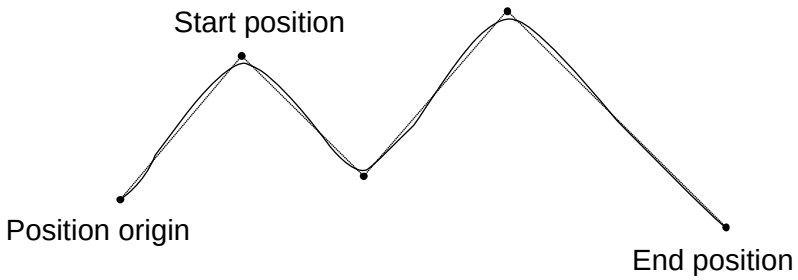
[Example 2] The axis pattern can be specified indirectly using a variable.
When the command in [Example 1] is rephrased based on indirect specification using a variable:
11 (binary) → 3 (decimal)

```
LET     1     3     Assign 3 to variable 1.
STOP    *1
```

- PSPL (Move along spline)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PSPL	Start position number	End position number	PE

[Function] Continuously move from the specified start position to end position via interpolation along a spline-interpolation curve.
The output type in the output field can be set using an actuator-declaration command POTP.
If invalid data is set for any position number between the start and end position numbers, that position number will be skipped during continuous movement.



(The above diagram is only an example.)

(Note) If the acceleration and deceleration are different between points, the speeds will not be connected smoothly.

In this case, input in operand 1 the point number of the next target, instead of the predicted current position upon execution of the applicable command.
(Inputting a point number corresponding to the predicted current position will trigger movement to the same point during continuous movement, thereby causing the speed to drop.)

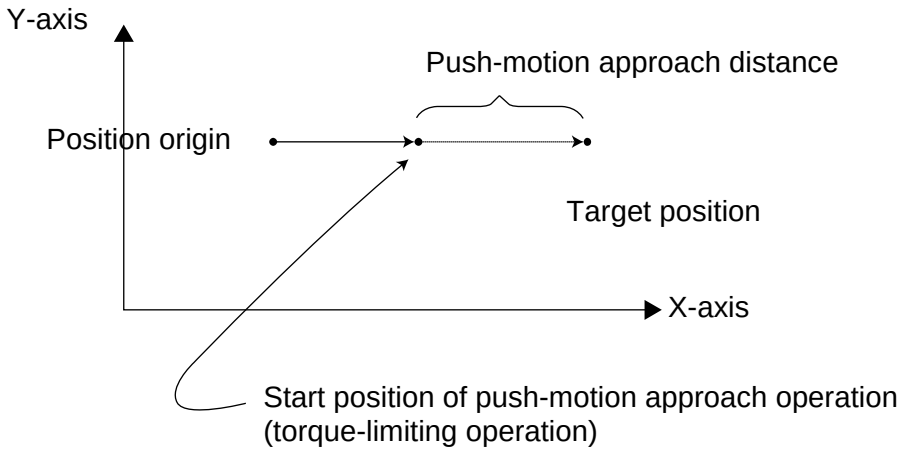
[Example]

```
VEL   100        Set the speed to 100 mm/s.
PSPL  100  120   Continuously move from position Nos. 100 to 120 along a
                 spline-interpolation curve.
```

- PUSH (Move by push motion)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PUSH	Target position number	Prohibited	PE

[Function] Perform push-motion operation until the target position specified in operand 1 is reached. The axes move in a normal mode from the position origin to the push-motion approach start position as determined by a PAPR command, after which push-motion approach operation (toque-limiting operation) will be performed. The speed of push-motion approach operation (toque-limiting operation) is determined by the push-motion approach speed specified by a PAPR command. If the output field is specified, the output will turn ON when a contact is confirmed, and turn OFF when a missed contact is detected.



The push force can be adjusted using “Driver parameter No. 38, Push torque limit at positioning” (default value: 70%).

(Note 1) A PUSH command only moves a single axis. If multiple axes are specified, an “Error No. C91, Multiple push-axes specification error” will generate.

(Note 2) A push-motion approach speed exceeding the maximum speed permitted by the system will be clamped at the maximum speed. (The maximum system speed is not the maximum practical speed. Determine a practical speed by considering the impact upon contact, etc.)

(Note 3) Push-motion operation cannot be performed with a synchro controller.

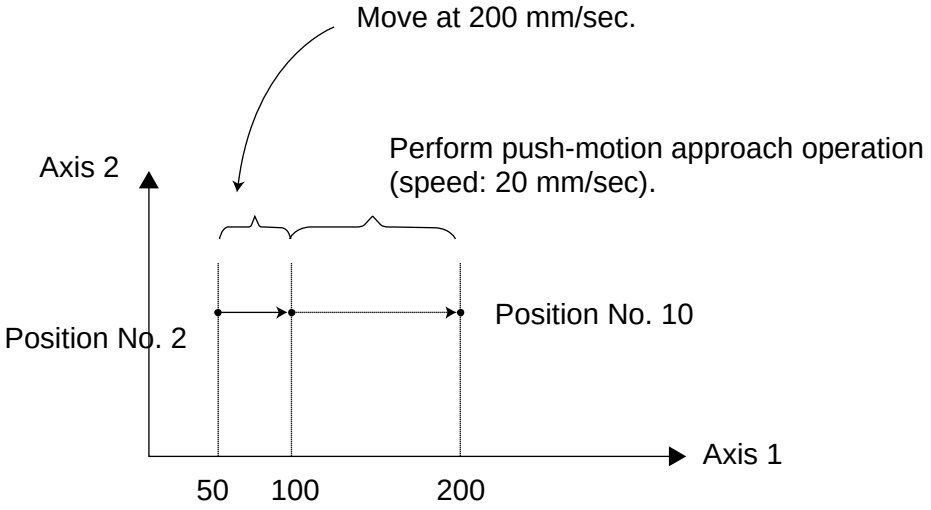
```
[Example]   PAPR    100    20
            MOVP    2
            PUSH    10
```

Set the push-motion approach distance to 100 mm and push-motion approach speed to 20 mm/sec.
Move from the current position to position No. 2.
Perform push-motion movement from position Nos. 2 to 10.

The diagram below describes a push-motion movement based on the position data shown in the table below:

Position data display in PC software

Position No.	Axis 1	Axis 2	Vel	Acc	Dcl
1					
2	50.000	100.000			
• • • •					
10	200.000		200	0.30	0.30
• •					



- PTRQ (Change push torque limit parameter)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PTRQ	Axis pattern	Ratio	CC

[Function] Change the push torque limit parameter of the axis pattern specified in operand 1 to the value in operand 2. Operand 2 is set as an integer (unit: %).
A PTRQ command temporarily rewrites “Driver parameter No. 38: Push torque limit at positioning.”

(Note 1) If a push torque limit is not set by a PTRQ command, the value set in “Driver parameter No. 38: Push torque limit at positioning” will be used.
(Note 2) The new push torque limit will remain effective even after the program ends. Therefore, when building a system using the PTRQ command, in every program explicitly specify a push torque limit using a PTRQ command before each push-motion operation. Assuming that the push torque limit will be reset to the original value when push-motion operation ends in one program can cause an unexpected problem in another program, because a different push torque limit will be used if the program is aborted due to an error, etc.
(Note 3) The new value set by a PTRQ command will become ineffective after a power-on reset or software reset.
(Note 4) A PTRQ command does not rewrite “Driver parameter No. 38: Push torque limit at positioning” (main CPU flash memory (non-volatile memory)).

[Example]

```
PTRQ   1     50    Change the push torque limit parameter for axis 1 to 50%.
PAPR   100   20    Set the push-motion approach distance to 100 mm and the
                   push-motion approach speed to 20 mm/sec.
MOVP   2           Move to position No. 2.
PUSH   10          Move by push motion from position No. 2 to position No. 10.
```

- CIR2 (Move along circle 2 (arc interpolation))

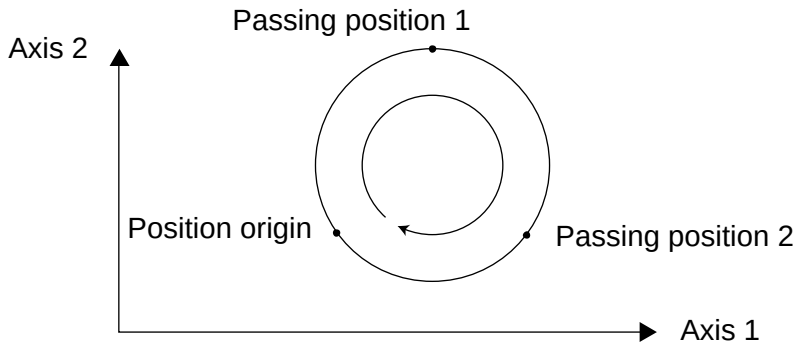
Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	CIR2	Passing position 1 number	Passing position 2 number	PE

[Function] Move along a circle originating from the current position and passing positions 1 and 2, via arc interpolation.
The rotating direction of the circle is determined by the given position data.
The diagram below describes a CW (clockwise) movement. Reversing passing positions 1 and 2 will change the direction of movement to CCW (counterclockwise).

The speed and acceleration will take valid values based on the following priorities:

Priority	Speed	Acceleration (deceleration)
1	Setting in the position data specified in operand 1	Setting in the position data specified in operand 1
2	Setting by VEL command	Setting by ACC (DCL) command
3		Default acceleration in all-axis parameter No. 11 (Default deceleration in all-axis parameter No. 12)

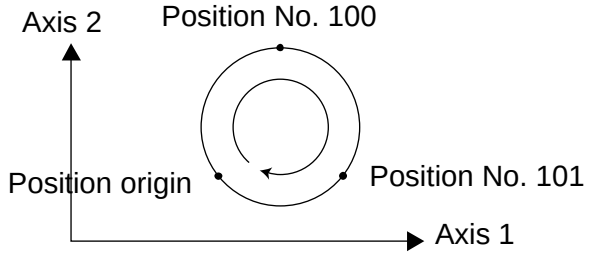
If speed is not set, a “C88 speed specification error” will generate.
If acceleration/deceleration is not valid, a “C89 acceleration/deceleration specification error” will generate.



(Note) This command is valid on arbitrary orthogonal planes. (Axis 2 may be selected automatically prior to axis 1 in accordance with the position data.)

[Example]

```
VEL   100        Set the speed to 100 mm/s.
CIR2  100  101   Move along a circle (circular interpolation) passing position
                 Nos. 100 and 101.
```



- ARC2 (Move along circle 2 (arc interpolation))

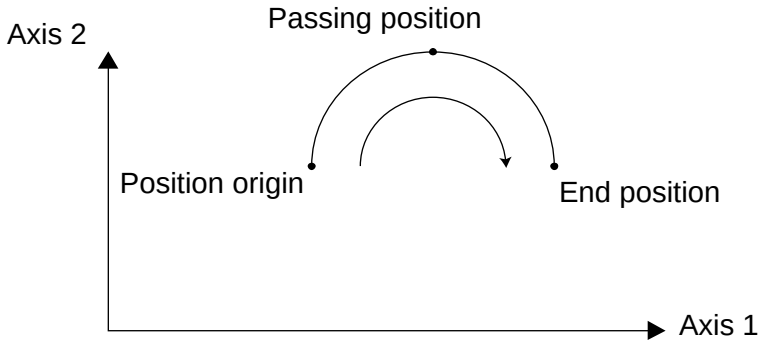
Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	ARC2	Passing position number	End position number	PE

[Function] Move along an arc originating from the current position, passing the specified position and terminating at the end position, via arc interpolation.

The speed and acceleration will take valid values based on the following priorities:

Priority	Speed	Acceleration (deceleration)
1	Setting in the position data specified in operand 1	Setting in the position data specified in operand 1
2	Setting by VEL command	Setting by ACC (DCL) command
3		Default acceleration in all-axis parameter No. 11 (Default deceleration in all-axis parameter No. 12)

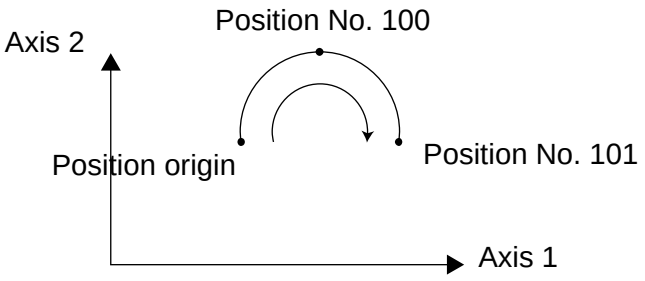
If speed is not set, a “C88 speed specification error” will generate.
If acceleration/deceleration is not valid, a “C89 acceleration/deceleration specification error” will generate.



(Note) This command is valid on arbitrary orthogonal planes. (Axis 2 may be selected automatically prior to axis 1 in accordance with the position data.)

[Example]

```
VEL   100        Set the speed to 100 mm/s.
ARC2  100  101   Move along an arc (circular interpolation) from the current
                 position to position No. 101 by passing position No. 100.
```



- CHVL (Change speed)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	CHVL	Axis pattern	Speed	CP

[Function] Change the speed of the axes operating in other task.
When a CHVL command is executed, the speed of the axes specified in operand 1 will change to the value specified in operand 2.

(Note 1) This command is not valid on an axis operated by a CIR, ARC, PSPL, PUSH, or ARCH command.
(Note 2) Executing a CHVL command for an axis operating in sigmoid motion (SCRV command) will generate an "Error No. CC1, Speed-change condition error."
(Note 3) This is a temporary speed-change command issued from other task to the active packet (point). It is not affected by the data declared by VEL.

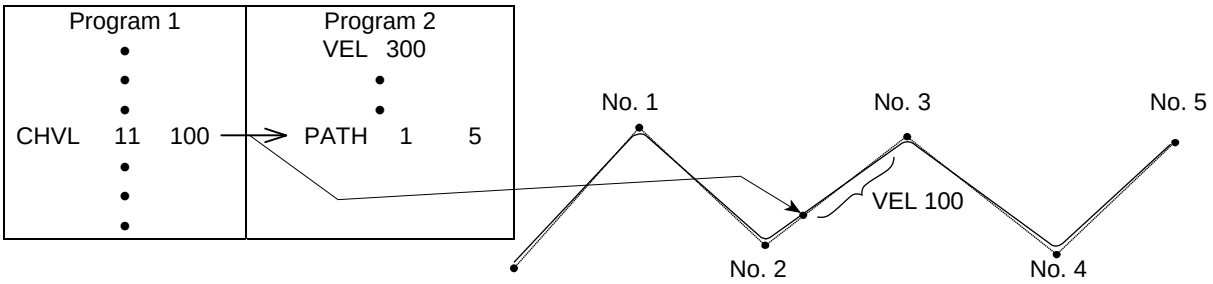
Program 1	Program 2
	VEL 300
	•
	•
	MOVP 1
CHVL 11 100	MOVP 2
	MOVP 3
	•
	•

If CHVL is executed in program 1 while MOVP 2 is executed in program 2, the travel speed of MOVP 2 will become 100 mm/sec.
The speeds of other move commands will remain 300 mm/sec.

The axis pattern can be specified indirectly using a variable.
When program 1 is rephrased based on indirect specification using a variable:
11 (binary) → 3 (decimal)

```
LET     1     3      Assign 3 to variable 1.
CHVL    *1    100
```

(Note 4) Since this command is valid only for the packet that is active at the time of execution of the command for an axis subject to continuous motion in a PATH command, etc., caution must be exercised against the timing shift. The packet handling will be put on hold during speed-change processing, so caution must also be exercised against the locus shift.



If CHVL is executed in program 1 while PATH is executed in program 2, or specifically during the PATH movement from point No. 2 to point No. 3, the speed specified by CHVL (100 mm/sec in the above example) will become valid only during the PATH movement to point No. 3. Other travel speeds will remain at the speed specified by VEL (300 mm/sec in the above example).
(Note 5) Override of the CHVL call task will be applied, so caution must be exercised.
(Note 6) The maximum speed of the specified axis completing home return will be clamped by the minimum value set in "Axis-specific parameter No. 28, Maximum operating speed of each axis" or "Axis-specific parameter No. 27, Maximum speed limited by maximum motor speed" with respect to the specified axis and related interpolation axes currently operating. To prevent the maximum speed from being limited due to the effect of other axis whose maximum speed is lower than the speed specified in the CHVL command, issue a CHVL command in multiple steps corresponding to the respective axes having different maximum speeds. In particular, specification of a CHVL command in a separate step is recommended for a rotating axis.

[Example]

```
CHVL  11  500  ⇒  CHVL  1   500
                  CHVL  10  500
```

- ARCD (Move along arc via specification of end position and center angle (arc interpolation))

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	ARCD	End position number	Center angle	PE

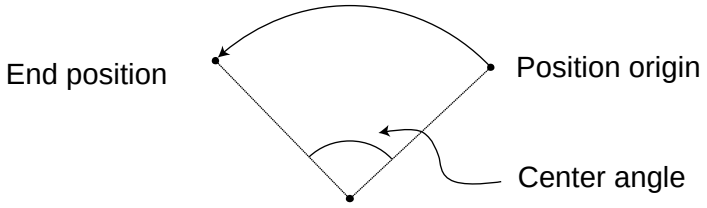
[Function] Move along an arc originating from the current position and terminating at the end position, via arc interpolation.
Specify the end position of movement in operand 1, and the center angle formed by the position origin and end position in operand 2. The center angle is set in a range from –359.999 to –0.001 or from 0.001 to 359.999. A positive value indicates CCW (counterclockwise) movement, while a negative value indicates CW (clockwise) movement.

(Note) The rotating direction of the actual operation locus may vary from the specified direction depending on how each axis is installed, how the two axes are combined, and so on. Perform test operation to check the rotating direction.
The center angle is set in degrees and may include up to three decimal places.

The speed and acceleration will take valid values based on the following priorities:

Priority	Speed	Acceleration (deceleration)
1	Setting in the position data specified in operand 1	Setting in the position data specified in operand 1
2	Setting by VEL command	Setting by ACC (DCL) command
3		Default acceleration in all-axis parameter No. 11 (Default deceleration in all-axis parameter No. 12)

If speed is not set, a “C88 speed specification error” will generate.
If acceleration/deceleration is not valid, a “C89 acceleration/deceleration specification error” will generate.



(Note) This command is valid on arbitrary orthogonal planes. (Axis 2 may be selected automatically prior to axis 1 in accordance with the position data.)

[Example]

```
VEL   100        Set the speed to 100 mm/s.
ARCD  100  120   Move along an arc from the position origin to position No.
                 100 for a center angle of 120 degrees (CCW direction).
```

- ARCC (Move along arc via specification of center position and center angle (arc interpolation))

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	ARCC	Center position number	Center angle	PE

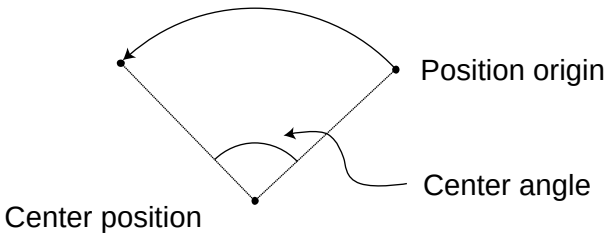
[Function] Move along an arc originating from the current position by keeping a specified radius from the center position, via arc interpolation.
Specify the center position in operand 1, and the center angle formed by the position origin and end position in operand 2. The center angle is set in a range from –3600 to 3600 degrees (±10 revolutions). A positive value indicates CCW (counterclockwise-direction) movement, while a negative value indicates CW (clockwise-direction) movement (setting unit: degree).

(Note) The rotating direction of the actual operation locus may vary from the specified direction depending on how each axis is installed, how the two axes are combined, and so on. Perform test operation to check the rotating direction.
The center angle is set in degrees and may include up to three decimal places.

The speed and acceleration will take valid values based on the following priorities:

Priority	Speed	Acceleration (deceleration)
1	Setting in the position data specified in operand 1	Setting in the position data specified in operand 1
2	Setting by VEL command	Setting by ACC (DCL) command
3		Default acceleration in all-axis parameter No. 11 (Default deceleration in all-axis parameter No. 12)

If speed is not set, a “C88 speed specification error” will generate.
If acceleration/deceleration is not valid, a “C89 acceleration/deceleration specification error” will generate.



(Note) This command is valid on arbitrary orthogonal planes. (Axis 2 may be selected automatically prior to axis 1 in accordance with the position data.)

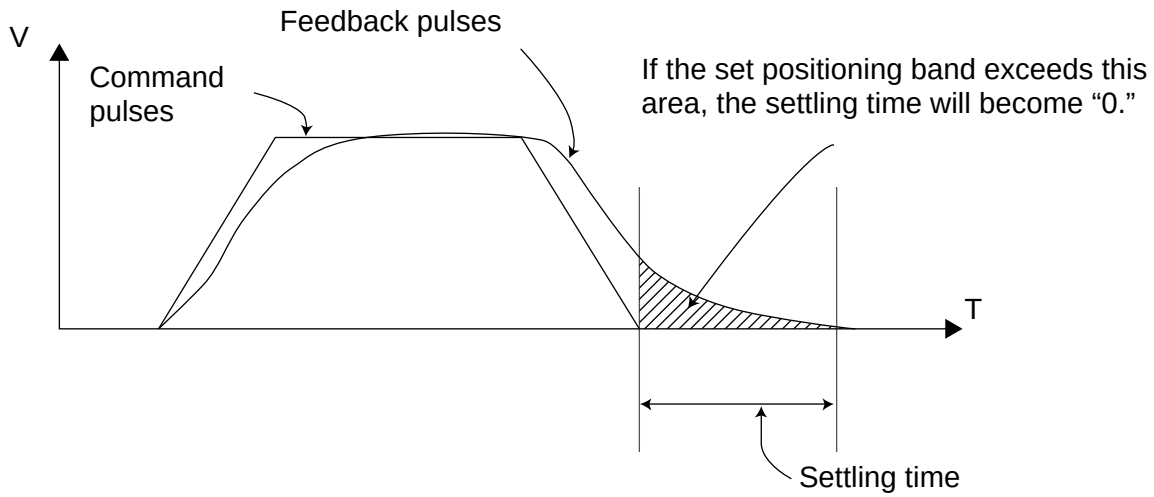
[Example]

```
VEL   100        Set the speed to 100 mm/s.
ARCC  100  120   Move along an arc from the position origin for a center angle
                 of 120 degrees around position No. 100 being the center
                 (CCW direction).
```

- PBND (Set positioning band)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PBND	Axis pattern	Distance	CP

[Function] Set the position complete width for the axes in the axis pattern specified in operand 1. The distance in operand 2 is set in mm.
As a rule, positioning is deemed complete when all command pulses have been output and the current position is inside the positioning band. Therefore, this command is effective if you wish to reduce the tact time by shortening the approximate positioning settling time. (Normally a setting of approx. 3 to 5 mm will have effect, but the effect must be confirmed on the actual machine.)
(This command can be combined with a QRTN command for special purposes. Refer to the section on QRTN command for details.)



(Note 1) If positioning band is not set with a PBND command, the value set in “Axis-specific parameter No. 58, Positioning band” will be used.
(Note 2) If the positioning band is changed, the new setting will remain valid even after the program ends. Therefore, to build a system using PBND commands, a positioning band must be expressly specified with a PBND command before operation of each program. An assumption that the positioning band will be reset to the original value when the operation ends in other program may lead to an unexpected problem, because the positioning band will become different from what is anticipated in case the applicable program is aborted due to error, etc.
(Note 3) The value set in “Axis-specific parameter No. 58, Positioning band” will not be written by a PBND command.

[Example 1]

```
PBND   11   5
```

Set the positioning band for axes 1 and 2 to 5 mm after this command.

[Example 2] The axis pattern can be specified indirectly using a variable.
When the command in [Example 1] is rephrased based on indirect specification using a variable:
11 (binary) → 3 (decimal)

```
LET    1    3     Assign 3 to variable 1.
PBND   *1   5
```

- CIR (Move along circle)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	CIR	Passing position 1 number	Passing position 2 number	PE

[Function] Move along a circle originating from the current position and passing the positions specified in operands 1 and 2.
Therefore, reversing the settings of operands 1 and 2 will implement a circular movement in the reverse direction.
The output will turn OFF at the start of circular movement, and turn ON when the movement is complete.
Difference from CIR2:
CIR processing resembles moving along a polygon with a PATH command, while CIR2 actually performs arc interpolation.
Select an applicable command by considering the characteristics of each command. (Normally CIR2 is used.)

(Note 1) If the division angle is set to “0” with a DEG command (division angle is calculated automatically based on priority speed setting), the speed set in the data at passing position 1 or speed set by a VEL command will be used (former is given priority). The speed set in the data at passing position 2 will have no meaning.

(Note 2) If the division angle is set to a value other than “0” with a DEG command (normal division angle), the speed specified in the target position data will be used. (The speed set by a VEL command will become valid if position data is not specified.)
In the case of circular movement, the axes will return from passing position 2 to the start position at the speed declared by a VEL command. Therefore, a VEL command must always be used with a CIR command.

(Note 3) The acceleration is selected in the order of the acceleration in the data at passing position 1, followed by the value in “All-axis parameter No. 11, Default acceleration.”
The deceleration will become the same value as the valid acceleration selected above.
Therefore, the deceleration in the data at passing position 1 and the acceleration/deceleration in the data at passing position 2 will not have any meaning.

(Note 4) This command is valid on arbitrary orthogonal planes. (Axis 2 may be selected automatically prior to axis 1 in accordance with the position data.)

[Example 1]

```
VEL   100        Set the speed to 100 mm/s.
CIR   100  101   Move along a circle from the current position by passing
                 positions 100 and 101 sequentially.
```

[Example 2]

```
VEL   100        Set the speed to 100 mm/s.
LET   1    5     Assign 5 to variable 1.
LET   2    6     Assign 6 to variable 2.
CIR   *1   *2    Move along a circle from the current position by passing the
                 contents of variables 1 and 2 (positions 5 and 6)
                 sequentially.
```

- ARC (Move along arc)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	ARC	Passing position number	End position number	PE

[Function] Move along an arc from the current position to the position specified in operand 2, by passing the position specified in operand 1.
The output will turn OFF at the start of arc movement, and turn ON when the movement is complete.
Difference from ARC2:
ARC processing resembles moving along a polygon with a PATH command, while ARC2 actually performs arc interpolation.
Select an applicable command by considering the characteristics of each command. (Normally ARC2 is used.)

(Note 1) If the division angle is set to “0” with a DEG command (division angle is calculated automatically based on priority speed setting), the speed set in the data at passing position 1 or speed set by a VEL command will be used (former is given priority). The speed set in the data at passing position 2 will have no meaning.
(Note 2) If the division angle is set to a value other than “0” with a DEG command (normal division angle), the speed specified in the target position data will be used. (The speed set by a VEL command will become valid if position data is not specified.)
(Note 3) The acceleration is selected in the order of the acceleration in the data at passing position 1, followed by the value in “All-axis parameter No. 11, Default acceleration.”
The deceleration will become the same value as the valid acceleration selected above. Therefore, the deceleration in the data at passing position 1 and the acceleration/deceleration in the data at passing position 2 will not have any meaning.
(Note 4) This command is valid on arbitrary orthogonal planes. (Axis 2 may be selected automatically prior to axis 1 in accordance with the position data.).

[Example 1]

```
VEL   100
ARC   100   101
```

Set the speed to 100 mm/s.
Move along an arc from the current position to position 101 by passing position 100.

[Example 2]

```
VEL   100
LET   1     5
LET   2     6
ARC   *1    *2
```

Set the speed to 100 mm/s.
Assign 5 to variable 1.
Assign 6 to variable 2.
Move along an arc from the current position to the content of variable 2 (position 6) by passing the content of variable 1 (position 5).

1.13 Structural IF

- IF□□ (Structural IF)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	IF□□	Variable number	Data	CP

[Function] Compare the content of the variable specified in operand 1 with the value specified in operand 2, and proceed to the next step if the condition is satisfied.
If the condition is not satisfied, the program will proceed to the step next to the corresponding ELSE command, if any, or to the step next to the corresponding EDIF command.
If the input condition is not satisfied and the IF□□ command is not executed, the program will proceed to the step next to the corresponding EDIF.
A maximum of 15 nests are supported when IS□□ and DW□□ are combined.

IF□□
- EQ ..... Operand 1 = Operand 2
- NE ..... Operand 1 ≠ Operand 2
- GT ..... Operand 1 > Operand 2
- GE ..... Operand 1 ≥ Operand 2
- LT ..... Operand 1 < Operand 2
- LE ..... Operand 1 ≤ Operand 2

[Example 1]

```
600   IFEQ   1   1   Select an axis.
      IFGE   2   0   Select a moving direction.
      JFWN   01  5   Move axis 1 forward.
      ELSE
      JBWN   01  5   Move axis 1 backward.
      EDIF
      ELSE
      IFLT   2   0   Select a moving direction.
      JBWN   10  5   Move axis 2 backward.
      ELSE
      JFWN   10  5   Move axis 2 forward.
      EDIF
      EDIF
```

Jog by selecting axis 1/axis 2 by variable 1 and forward/backward (+/–) by variable 2.
Nothing will happen if flag 600 is OFF, in which case the program will proceed to the step next to the last EDIF.

(Note) Using a GOTO command to branch out of or into an IF□□-EDIF syntax is prohibited.

- IS□□ (Compare strings)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	IS□□	Column number	Column number, character literal	CP

[Function] Compare the character strings in the columns specified in operands 1 and 2, and proceed to the next step if the condition is satisfied.
If the condition is not satisfied, the program will proceed to the step next to the corresponding ELSE command, if any, or to the step next to the corresponding EDIF command.
Comparison will be performed for the length set by a SLEN command.
If a character literal is specified in operand 2, comparison will be performed for the entire length of the literal.
If the input condition is not satisfied and the IS□□ command is not executed, the program will proceed to the step next to the EDIF.
A maximum of 15 nests are supported when IF□□ and DW□□ are combined.

IS□□
- EQ ..... Operand 1 = Operand 2
- NE ..... Operand 1 ≠ Operand 2

[Example 1]

```
        SCPY  10   'GOFD' (Move forward)
        SCPY  14   'GOBK' (Move backward)
        LET   1    5
        LET   2    14
        SLEN  4                Set the number of comparing characters to 4.
600     ISEQ  1    '1AXS' (Axis 1)  Select an axis.
        ISEQ  5    10          Select a moving direction.
        JFWN  01   5           Move axis 1 forward.
        ELSE
        JBWN  01   5           Move axis 1 backward.
        EDIF
        ELSE
        ISNE  *1   *2          Select a moving direction.
        JFWN  10   5           Move axis 2 backward.
        ELSE
        JBWN  10   5           Move axis 2 forward.
        EDIF
        EDIF
```

Jog by selecting axis 1/axis 2 by columns 1 to 4 and forward/backward by columns 5 to 8.
Nothing will happen if flag 600 is OFF, in which case the program will proceed to the step next to the last EDIF.
If columns 1 to 8 contain the following data, axis 1 will be moved forward.

1 2	3 4	5 6	7 8	
1A	XS	GO	FD	

(Note) Using a GOTO command to branch out of or into an IS□□-EDIF syntax is prohibited.

- ELSE (Else)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Prohibited	Prohibited	ELSE	Prohibited	Prohibited	CP

[Function] An ELSE command is used arbitrarily in conjunction with an IF□□ or IS□□ command to declare the command part to be executed when the condition is not satisfied.

[Example 1] Refer to the sections on IF□□ and IS□□.

- EDIF (End IF□□)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Prohibited	Prohibited	EDIF	Prohibited	Prohibited	CP

[Function] Declare the end of an IF□□ or IS□□ command.

[Example 1] Refer to the sections on IF□□ and IS□□.

1.14 Structural DO

- DW□□ (DO WHILE)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	DW□□	Variable number	Data	CP

[Function] Compare the content of the variable specified in operand 1 with the value specified in operand 2, and execute the subsequent commands up to EDDO while the condition is satisfied.
The program will proceed to the step next to the corresponding EDDO if the condition is no longer satisfied.
A LEAV command can be used to forcibly end a loop.
If the input condition is not satisfied and the DW□□ command is not executed, the program will proceed to the step next to the corresponding EDDO.
A maximum of 15 nests are supported when IF□□ and IS□□ are combined.

DW□□
- EQ ..... Operand 1 = Operand 2
- NE ..... Operand 1 ≠ Operand 2
- GT ..... Operand 1 > Operand 2
- GE ..... Operand 1 ≥ Operand 2
- LT ..... Operand 1 < Operand 2
- LE ..... Operand 1 ≤ Operand 2

[Example 1]

```
008   DWEQ   1   0
        :
        :
      EDDO
```

Repeat the command up to an EDDO command while variable 1 contains “0.”

If DW□□ is specified at the start and input 8 is OFF, nothing will occur and the program will proceed to the step next to EDDO.

(Note) Using a GOTO command to branch out of or into a DW□□-EDDO syntax is prohibited.

- LEAV (Pull out of DO WHILE)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	LEAV	Prohibited	Prohibited	CP

[Function] Pull out of a DO□□ loop and proceed to the step next to EDDO.

[Example 1]

```
      DWEQ   1   0
        :
600   LEAV
        :
      EDDO
```

Repeat the commands up to an EDDO command while variable 1 contains ‘0.”

Forcibly end the loop if flag 600 is ON and proceed to the step next to an EDDO command.

- ITER (Repeat)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	ITER	Prohibited	Prohibited	CP

[Function] Forcibly switch the control to EDDO while in a DO□□ loop.

[Example 1]

```
        DWEQ    1    0
          :
600     ITER
          :
        EDDO
```

Repeat the commands up to an EDDO command while variable 1 contains “0.”

Forcibly switch the control to an EDDO command and perform end judgment, if flag 600 is ON.

- EDDO (End DO WHILE)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Prohibited	Prohibited	EDDO	Prohibited	Prohibited	CP

[Function] Declare the end of a loop that began with DW□□.
If the DW□□ condition is not satisfied, the program will proceed to the step next to this command.

[Example 1] Refer to the section on DW□□.

1.15 Multi-Branching

- SLCT (Start selected group)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	SLCT	Prohibited	Prohibited	CP

[Function] Branch to the step next to any WH□□ or WS□□ command that exists before an EDSL command and whose condition is satisfied, or to the step next to an OTHE command if none of the conditions are satisfied.
A SLCT command must be followed by a WH□□, WS□□ or EDSL command.
A maximum of 15 nests are supported.

(Note) Using a GOTO command to branch out of or into a SLCT-EDSL syntax is prohibited.

[Example 1]

```
        SCPY   1   'Right'   Assign 'right' to columns 1 and 2.
          :
600     SLCT                 Jump to a W□□□ whose condition is satisfied.
        WSEQ   1   'Right'   If 'right' is stored in columns 1 and 2, this command will
          :                  be executed.
        WSEQ   1   'Left'    If 'left' is stored, this command will be executed.
          :
        OTHE                 If the content of columns 1 and 2 is neither of the above,
          :                  this command will be executed.
        EDSL                 If flag 600 is OFF, the processing will move here upon
                             execution of any of the conditions.
```

- WH□□ (Select if true; variable)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Prohibited	Prohibited	WH□□	Variable number	Data	CP

[Function] This command is used between SLCT and EDSL commands to execute the subsequent commands up to the next W□□□ command or an OTHE or EDSL command when the comparison result of the content of the variable specified in operand 1 with the value specified in operand 2 satisfies the condition.

WH□□
- EQ ..... Operand 1 = Operand 2
- NE ..... Operand 1 ≠ Operand 2
- GT ..... Operand 1 > Operand 2
- GE ..... Operand 1 ≥ Operand 2
- LT ..... Operand 1 < Operand 2
- LE ..... Operand 1 ≤ Operand 2

[Example 1]

LET	1	20	Assign 20 to variable 1.
LET	2	10	Assign 10 to variable 2.
:			
SLCT			Execute multi-branching.
WHEQ	1	10	(1) will be executed if the content of variable 1 is 10.
:			Since variable 1 contains 20, however, the next
(1)			condition will be referenced.
:			
WHGT	1	*2	This command will be executed if the content of variable
:			1 is greater than the content of variable 2.
(2)			Since variable 1 (= 20) > variable 2 (=10), (2) will be executed.
OTHE			This command will be executed if none of the conditions
:			are satisfied. In this example, since (2) was executed,
(3)			(3) will not be executed.
:			
EDSL			The processing will move here if any of the conditions
:			were satisfied and the applicable command executed. In
(4)			this example, (2) and (4) will be executed.
:			

* If multiple conditions are likely to be satisfied, remember that the first W□□□ will become valid and any subsequent commands will not be executed. Therefore, state from the command with the most difficult condition or highest priority.

- WS□□ (Select if true; character)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Prohibited	Prohibited	WS□□	Column number	Column number, character literal	CP

[Function] This command is used between SLCT and EDSL commands to execute the subsequent commands up to the next W□□□ command or an OTHE or EDSL command when the comparison result of the character strings in the columns specified in operands 1 and 2 satisfies the condition.
Comparison will be performed for the length set by a SLEN command.
If a character literal is specified in operand 2, comparison will be performed for the entire length of the literal.

WS□□
- EQ ..... Operand 1 = Operand 2
- NE ..... Operand 1 ≠ Operand 2

[Example 1]

```
SLEN   3          Set the number of comparing characters to 3.
SCPY   1   'ABC'  Assign 'ABC' to column 1.
LET    1   2      Assign 2 to variable 1.
  :
SLCT              Execute multi-branching.
WSEQ   1   'XYZ'  (1) will be executed if columns 1 to 3 contain 'XYZ.'
  :               Since columns 1 to 3 contain 'ABC,' however, this
(1)               command will not be executed.
  :
WSEQ   2   *1     (2) will be executed if the content of the number of
  :               characters specified by SLEN after column 2 is the
(2)               same as the content of the column specified in variable
  :               1.
OTHE              This command will be executed if none of the conditions
  :               are satisfied. In this example, since (2) was executed,
(3)               (3) will not be executed.
  :
EDSL              The processing will move here if any of the conditions
  :               were satisfied and the applicable command executed. In
(4)               this example, (2) and (4) will be executed.
  :
```

* If multiple conditions are likely to be satisfied, remember that the first W□□□ will become valid and any subsequent commands will not be executed. Therefore, state from the command with the most difficult condition or highest priority.

- OTHE (Select other)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Prohibited	Prohibited	OTHE	Prohibited	Prohibited	CP

[Function] This command is used between SLCT and EDSL commands to declare the command to be executed when none of the conditions are satisfied.

[Example 1] Refer to the sections on SLCT, WH□□ and WS□□.

- EDSL (End selected group)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Prohibited	Prohibited	EDSL	Prohibited	Prohibited	CP

[Function] Declare the end of a SLCT command.

[Example 1] Refer to the sections on SLCT, WH□□ and WS□□.

1.16 System Information Acquisition

- AXST (Get axis status)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	AXST	Variable number	Axis number	CP

[Function] Store in the variable specified in operand 1 the status (axis error number) of the axis specified in operand 2.

(Note 1) If the obtained result is “0,” it means no axis error is present.
(Note 2) Since the error lists are written in hexadecimals, they must be converted to decimals.

[Example] AXST 1 2 Read the error number for axis 2 to variable 1.

If 3188 (decimal) is stored in variable 1 after the execution of this command:

3188 ÷ 16 = 199 ,,,4
199 ÷ 16 = 12 (= C) ,,,7

$3188 = 12\ (= C) \times 16^2 + 7 \times 16^2 + 4$
= C74 (HEX) (Hexadecimal number)

Therefore, an “Error No. C74, Actual-position soft limit over error” is present.

- PGST (Get program status)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PGST	Variable number	Program number	CP

[Function] Store in the variable specified in operand 1 the status (program error number) of the program specified in operand 2.

(Note 1) If the obtained result is “0,” it means no program error is present.
(Note 2) Although the error lists are written in hexadecimals, the status to be stored (program error number) is a decimal. Therefore, the decimal program error numbers must be converted to hexadecimals.

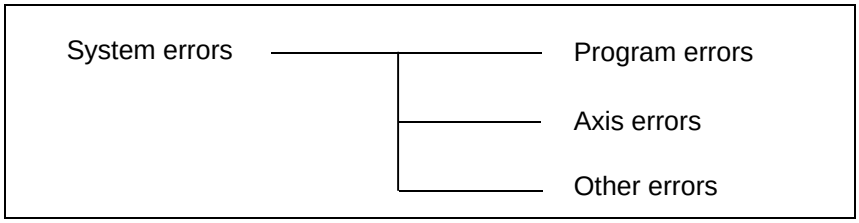
[Example] PGST 1 2 Read the error number for program No. 2 to variable 1.

- SYST (Get system status)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	SYST	Variable number	Prohibited	CP

[Function] Store the system status (top-priority system error number) in the variable specified in operand 1.

(Note 1) If the obtained result is “0,” it means no system error is present.
(Note 2) Since the error lists are written in hexadecimals, they must be converted to decimals.
(Note 3) Relationship of error statuses



* An axis error that generates during operation with a program command will be registered both as a program error and an axis error.

[Example] SYST 1 Read the system error number to variable 1.

1.17 Zone

- WZNA (Wait for zone ON, with AND)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	WZNA	Zone number	Axis pattern	CP

[Function] Wait for the zone status of all axes (AND) specified by the axis pattern in operand 2 to become ON (inside zone) with respect to the zone specified in operand 1.

(Note 1) The zone status of axes not yet completing home return will remain OFF (outside zone).
(Note 2) A maximum of four areas can be set as zones for each axis ("Axis-specific parameter Nos. 86 to 97").
(Note 3) Zone output can be specified using "Axis-specific parameter Nos. 88, 91, 94 and 97" irrespective of this command.

[Example 1]

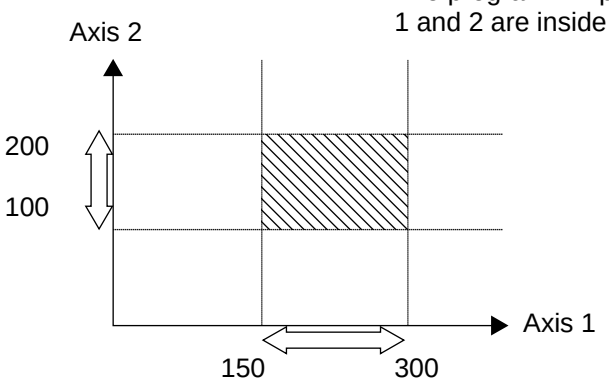
WZNA	1	11	If the parameters are set as follows, the program will wait until the zone status of axes 1 and 2 becomes ON (inside the shaded area shown in the diagram below).

[Example 2] The axis pattern can be specified indirectly using a variable.
When the command in [Example 1] is rephrased based on indirect specification using a variable:
11 (binary) → 3 (decimal)

LET	5	3	Assign 3 to variable 5.
WZNA	1	*5	

	Axis 1	Axis 2
"Axis-specific parameter No. 86, Zone 1 max." (Value is set in units of 0.001 mm)	300000	200000
"Axis-specific parameter No. 87, Zone 1 min." (Value is set in units of 0.001 mm)	150000	100000

The program will proceed to the next step if both axes 1 and 2 are inside the shaded area.



- WZNO (Wait for zone ON, with OR)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	WZNO	Zone number	Axis pattern	CP

[Function] Wait for the zone status of any of the axes (OR) specified by the axis pattern in operand 2 to become ON (inside zone) with respect to the zone specified in operand 1.

(Note 1) The zone status of axes not yet completing home return will remain OFF (outside zone).
(Note 2) A maximum of four areas can be set as zones for each axis ("Axis-specific parameter Nos. 86 to 97").
(Note 3) Zone output can be specified using "Axis-specific parameter Nos. 88, 91, 94 and 97" irrespective of this command.

[Example 1]

```
WZNO   1   11
```

If the parameters are set as follows, the program will wait until the zone status of axes 1 or 2 becomes ON (inside the shaded area shown in the diagram below).

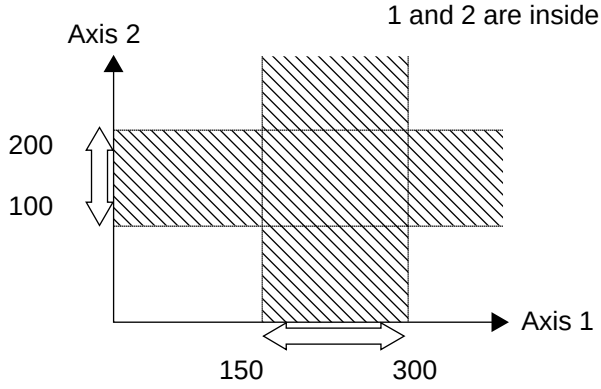
[Example 2] The axis pattern can be specified indirectly using a variable.
When the command in [Example 1] is rephrased based on indirect specification using a variable:
11 (binary) → 3 (decimal)

```
LET    5   3
WZNO   1   *5
```

Assign 3 to variable 5.

	Axis 1	Axis 2
"Axis-specific parameter No. 86, Zone 1 max." (Value is set in units of 0.001 mm)	300000	200000
"Axis-specific parameter No. 87, Zone 1 min." (Value is set in units of 0.001 mm)	150000	100000

The program will proceed to the next step if both axes 1 and 2 are inside the shaded area.



- WZFA (Wait for zone OFF, with AND)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	WZFA	Zone number	Axis pattern	CP

[Function] Wait for the zone status of all axes (AND) specified by the axis pattern in operand 2 to become OFF (outside zone) with respect to the zone specified in operand 1.

(Note 1) The zone status of axes not yet completing home return will remain OFF (outside zone).
(Note 2) A maximum of four areas can be set as zones for each axis (“Axis-specific parameter Nos. 86 to 97”).
(Note 3) Zone output can be specified using “Axis-specific parameter Nos. 88, 91, 94 and 97” irrespective of this command.

[Example]

```
WZFA    1    11
```

If the parameters are set as follows, the program will wait until the zone status of axes 1 and 2 becomes OFF (inside the shaded area shown in the diagram below)

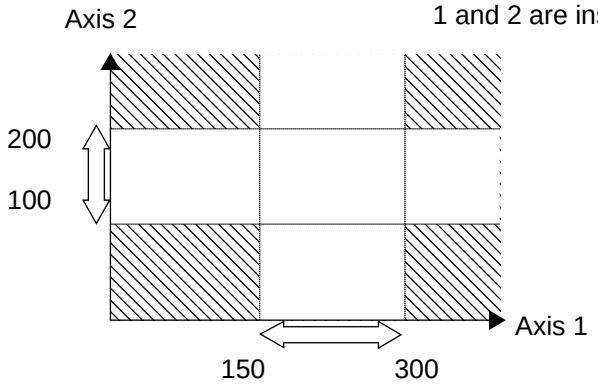
[Example 2] The axis pattern can be specified indirectly using a variable.
When the command in [Example 1] is rephrased based on indirect specification using a variable:
11 (binary) → 3 (decimal)

```
LET     5    3
WZFA    1    *5
```

Assign 3 to variable 5.

	Axis 1	Axis 2
“Axis-specific parameter No. 86, Zone 1 max.” (Value is set in units of 0.001 mm)	300000	200000
“Axis-specific parameter No. 87, Zone 1 min.” (Value is set in units of 0.001 mm)	150000	100000

The program will proceed to the next step if both axes 1 and 2 are inside the shaded area.



- WZFO (Wait for zone OFF, with OR)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	WZFO	Zone number	Axis pattern	CP

[Function] Wait for the zone status of any of the axes (OR) specified by the axis pattern in operand 2 to become OFF (outside zone) with respect to the zone specified in operand 1.

(Note 1) The zone status of axes not yet completing home return will remain OFF (outside zone).
(Note 2) A maximum of four areas can be set as zones for each axis (“Axis-specific parameter Nos. 86 to 97”).
(Note 3) Zone output can be specified using “Axis-specific parameter Nos. 88, 91, 94 and 97” irrespective of this command.

[Example 1]

```
WZFO   1   11
```

If the parameters are set as follows, the program will wait until the zone status of axes 1 or 2 becomes OFF (inside the shaded area shown in the diagram below).

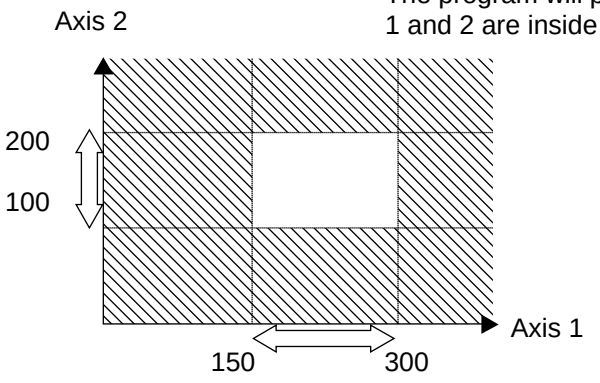
[Example 2] The axis pattern can be specified indirectly using a variable.
When the command in [Example 1] is rephrased based on indirect specification using a variable:
11 (binary) → 3 (decimal)

```
LET    5   3
WZFO   1   *5
```

Assign 3 to variable 5.

	Axis 1	Axis 2
“Axis-specific parameter No. 86, Zone 1 max.” (Value is set in units of 0.001 mm)	300000	200000
“Axis-specific parameter No. 87, Zone 1 min.” (Value is set in units of 0.001 mm)	150000	100000

The program will proceed to the next step if both axes 1 and 2 are inside the shaded area.



1.18 Communication

- OPEN (Open channel)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	OPEN	Channel number	Prohibited	CP

[Function] Open the channel specified in operand 1.
The specified channel will be enabled to send/receive hereafter.
Prior to executing this command, a SCHA command must be used to set an end character.

[Example]

```
SCHA    10
OPEN    0
```

Specify 10 (= LF) as the end character.
Open channel 0.

> Note: If “OPEN 0” is executed, communication with the teaching pendant or PC software will be cut off.

- CLOS (Close channel)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	CLOS	Channel number	Prohibited	CP

[Function] Close the channel specified in operand 1.
The specified channel will be disabled to send/receive hereafter.

[Example]

```
CLOS    0
```

Close channel 0.

```
LET     1       0
CLOS    *1
```

Assign 0 to variable 1.
Close the content of variable 1 (channel 0).

- READ (Read)

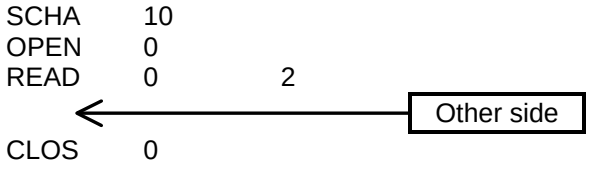
Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	READ	Channel number	Column number	CC

[Function] Read a character string from the channel specified in operand 1 to the column specified in operand 2.
Read will end when the character specified by a SCHA command is received.
Either a local or global column may be specified.
A return code will be stored in a local variable (variable 99 under the factory setting) immediately after this command is executed.
Whether or not the command has been executed successfully can be checked based on this return code. Define appropriate processing to handle situations where the command execution failed due to an error.
Setting “0” in operand 2 will specify a dummy read (receive buffer cleared and receive disabled) (the return code will indicate that the command was successfully executed).

[Example]

```
SCHA    10          Set LF (= 10) as the end character.
OPEN    0           Open channel 0.
READ    0     2     Read a character string from channel 0 to column
                    2 until LF is received.
TRAN    1     99    Assign the return code (variable 99) to variable 1.
CLOS    0           Close the channel.
SLCT                The processing flow branches out in accordance
                    with each return code.
                    (Note) Using a GOTO command to branch out of
                           a BGPA-EDPA syntax or to other branch
                           processing within the syntax is prohibited.
WHEQ    1     0     If the content of variable 1 is “0” (Completed
  :                 successfully), (1) will be executed. In (1), define
  (1)               the processing that should take place upon
  :                 successful command execution.
WHEQ    1     1     If the content of variable 1 is “1” (Timeout), (2) will
  :                 be executed. In (2), define appropriate processing
  (2)               to handle this situation, if necessary.
  :
WHEQ    1     2     If the content of variable 1 is “2” (Timer cancelled),
  :                 (3) will be executed. In (3), define appropriate
  (3)               processing to handle this situation, if necessary.
  :
OTHE                If the content of variable 1 is not “0,” “1” or “2,” (4)
  :                 will be executed. In (4), define appropriate error
  (4)               handling, if necessary.
  :
EDSL                Once one of the specified conditions was met and
                    the corresponding command has been executed,
                    the processing will move here.
```

(Note) A READ command must be executed before the other side sends the end character.



- Return code of the READ command
  The return code is stored in a local variable. The variable number can be set by “Other parameter No. 24.” The default variable number is 99.
  0: READ completed successfully (Receive complete)
  1: READ timeout (the timeout value is set by a TMRD command) (Continue to receive)
  2: READ timer cancelled (the wait status is cancelled by a TIMC command) (Continue to receive)
  3: READ SCIF overrun error (Receive disabled)
  4: READ SCIF receive error (framing error or parity error) (Receive disabled)
  5: READ factor error (program abort error) (Receive disabled)
  (Cannot be recognized by SEL commands)
  6: READ task ended (program end request, etc.) (Receive disabled)
  (Cannot be recognized by SEL commands)
  7: READ SCIF receive error due to other factor (Receive disabled)
  8: READ SIO overrun error (Receive disabled)
  9: READ SIO parity error (Receive disabled)
  10: READ SIO framing error (Receive disabled)
  11: READ SIO buffer overflow error (Receive disabled)
  12: READ SIO receive error due to other factor (Receive disabled)
  13 ~ 20: Used only in Ethernet (optional)
  21: READ SIO receive temporary queue overflow error (Receive disabled)
  22: READ SIO slave receive queue overflow error (Receive disabled)

- TMRW (Set READ/WRIT timeout value)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	TMRW	Read timer setting	(Write timer setting)	CP

[Function] Set the timeout to be applied to a READ/WRIT command.
With the PSEL controller, a write timer setting cannot be specified.
The timer setting specified in operand 1 will set the maximum time the program will wait for the character string read to end when a READ command is executed.
If the end character could not be read before the timer is up during the execution of the READ command, a timeout will occur and the program will move to the next step.
(Whether or not a timeout has occurred can be checked from the return code that will be stored in variable 99 (factory setting) immediately after the READ command is executed. If necessary, define appropriate processing to handle a timeout.)
Setting the timer to “0” will allow the READ command to wait infinitely, without timeout, until the end character is read.
The timer setting is input in seconds (setting range: 0 to 99.00 seconds) including up to two decimal places.
A variable can be specified indirectly in operand 1.

(Note) TMRW is set to “0” in the default condition before TMRW setting is performed.

[Example]

```
SCHA   10                 Set LF (=10) as the end character.
TMRW   30                 Set the READ timeout value to 30 seconds.
OPEN   0                  Open channel 0.
READ   0      2           Read the character string from channel 0 to
                          column 2 until LF is read.
TRAN   1      99          Assign the return code to variable 1.
CLOS   0                  Close the channel.
SLCT                      The processing flow branches out in accordance
                          with each return code.
                          (Note) Using a GOTO command to branch out of
                                 a BGPA-EDPA syntax or to other branch
                                 processing within the syntax is prohibited.
WHEQ   1      0           If the content of variable 1 is “0” (Completed
  :                       successfully), (1) will be executed. In (1), define
  (1)                     the processing that should take place upon
  :                       successful command execution.
WHEQ   1      1           If the content of variable 1 is “1” (Timeout), (2) will
  :                       be executed. In (2), define appropriate processing
  (2)                     to handle this situation, if necessary.
  :
WHEQ   1      2           If the content of variable 1 is “2” (Timer cancelled),
  :                       (3) will be executed. In (3), define appropriate
  (3)                     processing to handle this situation, if necessary.
  :
OTHE                      If the content of variable 1 is not “0,” “1” or “2,” (4)
  :                       will be executed. In (4), define appropriate error
  (4)                     handling, if necessary.
  :
EDSL                      Once one of the specified conditions was met and
                          the corresponding command has been executed,
                          the processing will move here.
```

Read completes successfully within 30 seconds → Variable No. 1 = 0
Timeout occurs → Variable No. 1 = 1

* The return code of READ command may not be limited to 0 or 1. The variable to store the return code can be set in “Other parameter No. 24.” Refer to the explanation of READ command for details.

- WRIT (Write)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	WRIT	Channel number	Column number	CC (Note 1)

[Function] Write the character string in the column specified in operand 2 to the channel specified in operand 1.
The operation will end when the character specified by a SCHA command is written.
Either a local or global column can be specified.

[Example]

SCHA	10		Set LF (= 10) as the end character.
OPEN	0		Open channel 0.
WRIT	0	2	Write the character string in column 2 to channel 0 until LF is written.
CLOS	0		Close the channel.

Once the channel has been opened, a WRIT command can be executed (data can be sent) for other tasks besides the one that opened the channel. Accordingly, if a READ command is executed for a channel-opening task and then a WRIT command is executed for other task, the response from the other side can be received without delay after the applicable data is sent from the PSEL.

The return code is stored in a local variable. The variable number can be set by “Other parameter No. 24.” The default variable number is 99.
0: WRIT completed successfully
1: WRIT timeout (the timeout value is set by a TMRW command)
2: WRIT timer cancelled (the wait status is cancelled by a TIMC command)
3 ~ 4: For future expansion
5: WRIT factor error (program abort error) (Cannot be recognized by SEL commands)
6: WRIT task ended (program end request, etc.) (Cannot be recognized by SEL commands)

- SCHA (Set end character)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	SCHA	Character code	Prohibited	CP

[Function] Set the end character to be used by a READ or WRIT command.
Any character from 0 to 255 (character code used in BASIC, etc.) can be specified.

[Example] Refer to the sections on READ and WRIT commands.

1.19 String Operation

- SCPY (Copy character string)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	SCPY	Column number	Column number, character literal	CC

[Function] Copy the character string in the column specified in operand 2 to the column specified in operand 1.
Copy will be performed for the length set by a SLEN command.
If a character literal is specified in operand 2, copy will be performed for the entire length of the literal.

[Example]

```
SCPY   1     'ABC'   Copy 'ABC' to column 1.

SLEN   10            Set the copying length to 10 bytes.
SCPY   100   200     Copy 10 bytes from column 200 to column 100.
```

- SCMP (Compare character strings)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	SCMP	Column number	Column number, character literal	EQ

[Function] Compare the column specified in operand 1 with the column specified in operand 2.
Comparison will be performed for the length set by a SLEN command.
If a character literal is specified in operand 2, comparison will be performed for the entire length of the literal.

[Example]

SCMP	1	'ABC'	600	Flag 600 will turn ON if columns 1 to 3 contain 'ABC.'
SLEN	5			Set the comparing length to five bytes.
SCMP	10	30	999	Turn ON flag 999 if five bytes from columns 30 and 10 match.

- SGET (Get character)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	SGET	Variable number	Column number, character literal	CP

[Function] Assign one character from the column specified in operand 2 to the variable specified in operand 1.
If a character-string literal is specified in operand 2, the first character will be assigned.

[Example]

```
SGET    1    100
```

Assign one byte from column 100 to variable 1.

```
LET     1    3     Assign 3 to variable 1.
LET     2    1     Assign 1 to variable 2.
SCPY    1    'A'   Copy 'A' to column 1.
SGET    *1   *2    Assign 'A' from the content of variable 2 (column 1) to the
                   content of variable 1 (variable 3).
```

- SPUT (Set character)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	SPUT	Column number	Data	CP

[Function] Set the data specified in operand 2 in the column specified in operand 1.

[Example]

SPUT	5	10	Set 10 (LF) in column 5.
LET	1	100	Assign 100 to variable 1.
LET	2	50	Assign 50 to variable 2.
SPUT	*1	*2	Set the content of variable 2 (50 (‘2’)) in the content of variable 1 (column 100).

- STR (Convert character string; decimal)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	STR	Column number	Data	CC

[Function] Copy to the column specified in operand 1 a decimal character string converted from the data specified in operand 2.
The data will be adjusted to the length set by a SLEN command.
If the data exceeds the specified length, it will be cut off at the length set by a SLEN command.
If the entire data has been converted within the length set by a SLEN command, the output will turn ON.

(Note) If the data specified in operand 2 is a 10-digit integer including eight or more valid digits, conversion of the values in the eighth and subsequent digits will not be guaranteed (the values through the seventh digits will be converted properly.)

[Example]

SLEN	5.3		Set a length consisting of five integer digits and three decimal digits.
STR	1	123	The following values will be set in columns 1 to 9:

1	2	3	4	5	6	7	8	9
		1	2	3	.	0	0	0

LET	1	10	Assign 10 to variable 1.
LET	102	987.6543	Assign 987.6543 to variable 102.
SLEN	2.3		Set a length consisting of two integer digits and three decimal digits.
STR	*1	*102	The following values will be set in columns 10 to 15:

10	11	12	13	14	15
8	7	.	6	5	4

Since the data exceeds the specified length, “9” in the 100’s place and “3” in the fourth decimal place will be cut off.

- STRH (Convert character string; hexadecimal)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	STRH	Column number	Data	CC

[Function] Copy to the column specified in operand 1 a hexadecimal character string converted from the data specified in operand 2.
Only the integer part will be adjusted to the length set by a SLEN command.
If the data exceeds the specified length, it will be cut off at the length set by a SLEN command.
If the entire data has been converted within the length set by a SLEN command, the output will turn ON.

(Note) If the data specified in operand 2 is a negative value, eight columns will be required to covert the entire data.

[Example]

```
SLEN   5          Set a format consisting of five integer digits.
STRH   1    255   The following values will be set in columns 1 to 5:
```

1	2	3	4	5
			F	F

```
LET    1    10        Assign 10 to variable 1.
LET    102  987.6543  Assign 987.6543 to variable 102.
SLEN   2.3            Set a length consisting of two integer digits and three
                      decimal digits.
STRH   *1   *102      The following values will be set in columns 10 and 11:
```

10	11
D	B

".3" in the SLEN command and ".6543" in variable 102, which are the decimal part, will be ignored.
The integer part is expressed as '3DB' in hexadecimal. Since the length is two digits, however, "3" in the third digit will be cut off.

- VAL (Convert character string data; decimal)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	VAL	Variable number	Column number, character literal	CC

[Function] Convert the decimal data in the column specified in operand 2 to a binary and assign the result to the variable specified in operand 1.
Conversion will be performed for the length set by a SLEN command.
If a character-string literal is specified in operand 2, conversion will be performed for the entire length of the literal.

(Note) Keep the converting length to 18 characters or less.

[Example]

SCPY	10	‘1234’	Set ‘1234’ in column 10.
SLEN	4		Set the converting length to four bytes.
VAL	1	10	Assign 1234, which is a binary converted from ‘1234’ in column 10, to variable 1.

LET	1	100	Assign 100 to variable 1.
LET	2	20	Assign 20 to variable 2.
SCPY	20	‘1234’	Copy ‘1234’ to column 20.
SCPY	24	‘.567’	Copy ‘.567’ to column 24.
SLEN	8		Set the converting length to eight bytes.
VAL	*1	*2	Assign 1234.567, which is a binary converted from ‘1234.567’ in the content of variable 2 (column 20) to the content of variable 1 (variable 100).

- VALH (Convert character string data; hexadecimal)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	VALH	Variable number	Column number, character literal	CC

[Function] Convert the hexadecimal data in the column specified in operand 2 to a binary and assign the result to the variable specified in operand 1.
Conversion will be performed for the length set by a SLEN command.
Only the integer part will be converted, with the decimal part being ignored.
If a character-string literal is specified in operand 2, conversion will be performed for the entire length of the literal.

(Note) Keep the converting length to 8 characters or less.

[Example]

```
SCPY   10   '1234'   Set '1234' in column 10.
SLEN   4             Set the converting length to four bytes.
VALH   1    10       Assign 4660, which is a binary converted from hexadecimal
                     '1234' in column 10, to variable 1.

LET    1    100      Assign 100 to variable 1.
LET    2    20       Assign 20 to variable 2.
SCPY   20   'ABCD'   Copy 'ABCD' to column 20.
SLEN   4             Set the converting length to four bytes.
VALH   *1   *2       Assign 43981, which is a binary converted from
                     hexadecimal 'ABCD' in the content of variable 2 (column
                     20) to the content of variable 1 (variable 100).
```

- SLEN (Set length)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	SLEN	Character string length	Prohibited	CP

[Function] Set the length to be processed by a string command.
This must always be set before using the following commands:

SCMP ..... Decimal part is invalid.
SCPY ..... Decimal part is invalid.
ISXX ..... Decimal part is invalid.
WSXX ..... Decimal part is invalid.
STRH ..... Decimal part is invalid.
VAL, VALH ..... Decimal part is invalid.
STR ..... Decimal part is valid.

[Example] Refer to the examples of the above commands:

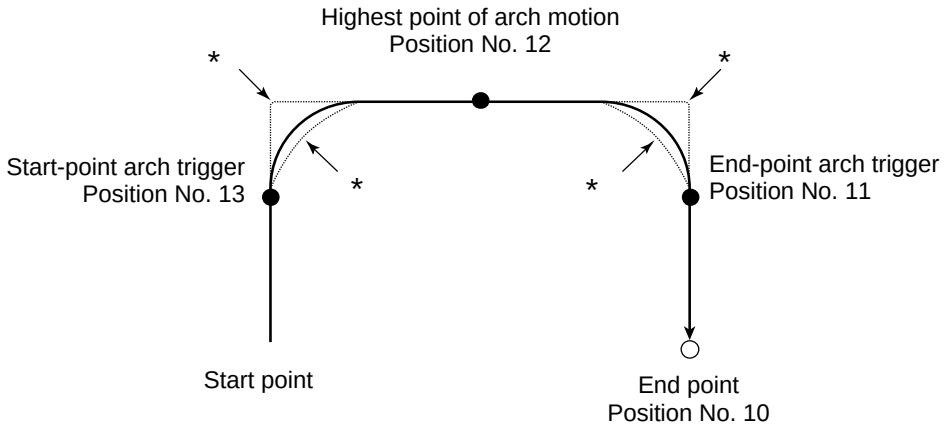
1.20 Arch-Motion-Related

- ARCH (Arch motion)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	ARCH	Position number	Position number	PE

Perform arch motion from the current point and move to the specified points.

- Move to the points specified in operand 1, via arch motion.
- Movements in directions other than the arch-motion Z-axis direction will begin after rising from the current point to the start-point arch trigger. After the Z point specified in operand 2 (as the highest point) is passed and movements in directions other than the arch-motion Z-axis direction are complete, the axes will come down to the end-point arch trigger and reach the specified point.
- Palletizing arch triggers must be set using an ATRG command.

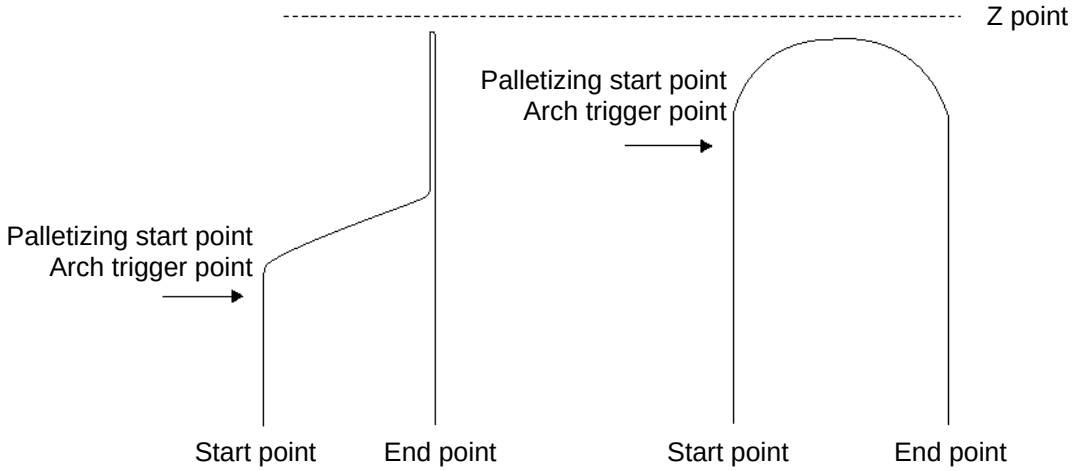


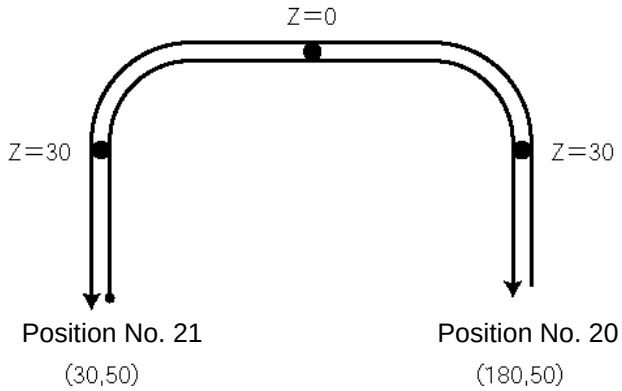

```
ACHZ   2
ATRG   13     11
  |
  |
  |
ARCH   10     12
```

* When the operation is resumed after a pause, depending on the position where the operation is resumed the locus may follow the lines (dotted lines) indicated by asterisks in the diagram for the composite section from ascent to horizontal movement or from horizontal movement to descent. Be careful not to cause interference.

- The arch-motion Z-axis coordinate of the end point will become the arch-motion Z-axis component of the point data specified in operand 1, if any, plus the arch-motion Z-axis offset. If there is no arch-motion Z component, the arch-motion Z-axis coordinate of the end point will become the arch-motion Z-axis coordinate of the start point plus the arch-motion Z-axis offset. (Normally the offset is added to all arch-motion positions, such as the arch triggers and Z point.)
- An error will generate if the start-point arch trigger is set below the start point or the end-point arch trigger is set below the end point. (Note: Up/down has nothing to do with +/– on the coordinate system.)
- The arch-motion Z-axis up direction refers to the direction toward the Z point from the start point (the down direction refers to the opposite direction), and has nothing to do with the size of coordinate value. Therefore, be sure to confirm the actual operating direction when using this command.

- The arch-motion Z-axis will come down after a rise-process command value is output. Therefore, one of the following operations will be performed depending on how the arch-trigger point and Z point are set.
  If the resulting operation is undesirable, change the arch trigger and/or Z point to improve the efficiency of movement.





The table below shows a program and data to cause the actuator to perform arch-motion operation by moving back and forth along the above path.

No.	B	E	N	Cnd	Cmnd	Operand 1	Operand 2	Pst	Comment
1					VEL	200			Speed 200mm/sec
2					ACHZ	2			Arch Motion
3					ATRG	22	22		Arch trigger position 22
4					MOVP	20			Move to pick position 20
5					TAG	1			
6					ARCH	21	23		Arch motion position 21
7					ARCH	20	23		Arch motion position 20
8					GOTO	1			

No.	Axis1	Axis2	Vel	Acc	Dcl
20	180.000	50.000			
21	30.000	50.000			
22		30.000			
23		0.000			

- ACHZ (Declare arch-motion Z-axis)

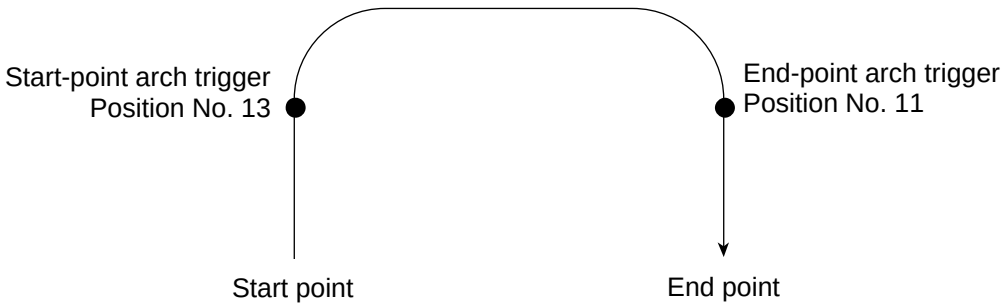
Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	ACHZ	Axis number	Prohibited	CP

Specify the axis number representing the arch-motion Z direction.
The axis number specified in operand 1 will be set as the axis number representing the arch-motion Z direction.
If the output field is specified, the output will turn ON after this command is executed.

- ATRG (Set arch triggers)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	ATRG	Position number	Position number	CP

Set the arch triggers used for arch motion.
(This setting becomes valid when an ARCH command is executed.)
Set the arch-motion Z-axis position data in the point data specified in operand 1 as the start-point arch trigger, and set the arch-motion Z-axis position data in the point data specified in operand 2 as the end-point arch trigger.



ATRG 13 11

(Refer to “Palletizing Setting” – “Arch triggers” under “How to Use.”)
For an arch-motion operation, set it so that a horizontal movement will begin when the start-point arch trigger is reached during ascent from the start point, and that the end-point arch trigger will be reached after a horizontal movement is completed during descent.
If the output field is specified, the output will turn ON after this command is executed.

- OFAZ (Set arch-motion Z-axis offset)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	OFAZ	Offset value	Prohibited	CP

Set the offset in the arch-motion Z-axis direction.
The value specified in operand 1 will be set as the offset in the arch-motion Z-axis direction.
The offset amount is set in mm and the effective resolution is 0.001 mm.
A negative value can also be specified as the offset, as long as the operation range will not be exceeded.
This offset is valid only at the end point of ARCH (arch motion) operation.
If the output field is specified, the output will turn ON after this command is executed.

1.21 Palletizing-Related

- BGPA (Declare start of palletizing setting)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	BGPA	Palletizing number	Prohibited	CP

Declare the start of a palletizing setting.
Once this command is executed, palletizing setting for the palletizing number specified in operand 1 will be enabled.
(In the case of an ACHZ, AEXT, OFAZ or ATRG command, setting is enabled without declaring BGPA.)
The input range of palletizing number is from 1 to 10.
When the palletizing setting is complete, execute EDPA.
Nested BGPAs are not supported. To declare start of another palletizing setting, execute an EDPA command and then execute a BGPA command again.
If the output field is specified, the output will turn ON after this command is executed.
Palletizing numbers are in the local range. Therefore, a given palletizing setting is valid only within the program in which it is set.

(Note) Using a GOTO command to branch out of or into a BGPA-EDPA syntax is prohibited.

- EDPA (Declare end of palletizing setting)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Prohibited	Prohibited	EDPA	Prohibited	Prohibited	CP

Declare the end of a palletizing setting.
If a palletizing-setting command (excluding BGPA, ACHZ, ATRG, AEXT and OFAZ) is executed before another BGPA is declared following an execution of this command (= while palletizing setting is not enabled), an error will generate.
If the output field is specified, the output will turn ON after this command is executed.

- PAPI (Set palletizing counts)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PAPI	Count	Count	CP

Set counts in the palletizing-axis directions.
The count specified in operand 1 will apply to the preferential-axis (PX-axis) direction, while the count specified in operand 2 will apply to the PY-axis direction.
If this command is executed before BGPA is declared (= while palletizing setting is not enabled), an error will generate.
If the output field is specified, the output will turn ON after this command is executed.

- PAPN (Set palletizing pattern)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PAPN	Pattern number	Prohibited	CP

Set a palletizing pattern.
The palletizing pattern specified in operand 1 will be set (1 = Pattern 1, 2 = Pattern 2).
If this command is not declared, pattern 1 will be used.
If this command is executed before BGPA is declared (= while palletizing setting is not enabled), an error will generate.
If the output field is specified, the output will turn ON after this command is executed.

- PASE (Declare palletizing axes)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PASE	Axis number	Axis number	CP

Set the two axes to be used in palletizing (PX and PY-axes).
The axis specified in operand 1 will be set as the preferential axis (PX-axis).
The axis specified in operand 2 will be set as the PY-axis.
This command is used in conjunction with PAPT and PAST.
It cannot be used together with a 3-point teaching (PAPS) command. Whichever is set later will be given priority.

> It is recommended to use a 3 or 4-points teaching (PAPS) command if the palletizing requires high accuracy.

If this command is executed before BGPA is declared (= while palletizing setting is not enabled), an error will generate.
If the output field is specified, the output will turn ON after this command is executed.

- PAPT (Set palletizing pitches)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PAPT	Pitch	Pitch	CP

Set palletizing pitches.
The value specified in operand 1 will be set as the pitch for the preferential axis (PX-axis), while the value specified in operand 2 will be set as the pitch for the PY-axis.
This command is used in conjunction with PASE and PAST.
If this command is executed before BGPA is declared (= while palletizing setting is not enabled), an error will generate.
If the output field is specified, the output will turn ON after this command is executed.

- PAST (Set palletizing reference point)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PAST	(Position number)	Prohibited	CP

Set the reference point used in palletizing.
If a value is set in operand 1, that position number specified in operand 1 will be used to store the reference point data.
If no value is set in operand 1, the position-number setting for storing reference point data will become invalid.
This command is used in conjunction with PASE and PAPT.
If this command is not set, coordinates (0, 0) are used as the reference point. If this command is set, the set coordinates are used as the reference point in calculating the position coordinates of palletizing points.
Coordinates in both the PX and PY-axis directions must always be set as the reference-point coordinates.
If this command is executed before BGPA is declared (= while palletizing setting is not enabled), an error will generate.
If the output field is specified, the output will turn ON after this command is executed.

- PAPS (Set palletizing points) For 3-point teaching

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PAPS	Position number	(Palletizing position setting type)	CP

Set palletizing positions in 3-point teaching.

It can also be used to set palletizing positions in 4-point teaching, in which case the pallet plane can be set to any quadrilateral other than a square, rectangle or parallelogram.

In operand 1, set the position number of the start point needed to set palletizing positions in 3-point teaching. If “n” is set as the position number for the start point, position data for the end point in the PX-axis direction will be stored in position No. n+1, while position data for the end point in the PY-axis direction will be stored in position No. n+2.

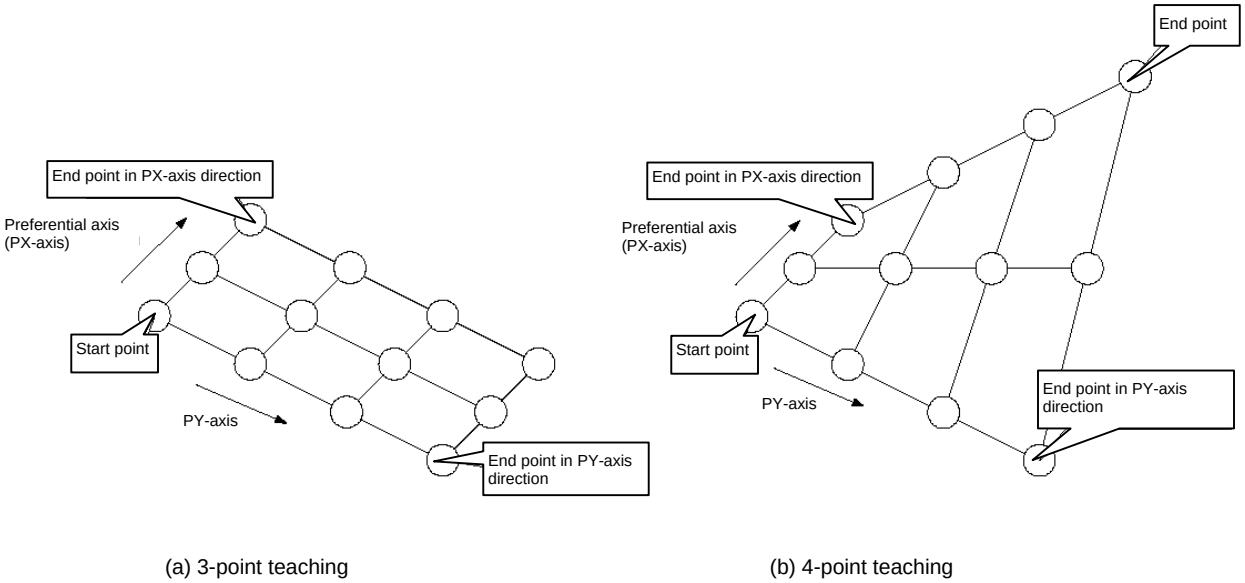
In the case of 4-point teaching, position data for the end point should be stored in position No. n+3.

In operand 2, specify the applicable palletizing position setting type.

[Palletizing position setting type]
If operand 2 is “0” or blank, 3-point teaching will be specified.
As shown in Fig. 1 (a), palletizing positions will be set on the quadrilateral pallet plane determined by the three points including the start point, end point in the PX-axis direction and end point in the PY-axis direction.

If operand 2 is “2,” 4-point teaching will be specified.
As shown in Fig. 1 (b), palletizing positions will be set on the quadrilateral pallet plane determined by the four points including the start point, end point in the PX-axis direction, end point in the PY-axis direction, and end point.

Fig. 1 shows two different arrangements of palletizing positions.



(a) 3-point teaching (b) 4-point teaching

Fig. 1 Layout of Palletizing Positions

- If the valid axis pattern does not match the point data for 3-point teaching or 4-point teaching, an error “CB0, Mismatched valid axes for palletizing 3-point teaching data” will generate. If a PAPS command is executed after specifying the applicable axes using a GRP command, only the point data corresponding to the specified axes, among all axes whose point data is valid, will be used as palletizing point data. Executing a GRP command thereafter with a different setting will have no effect.

- If there are not enough valid axes, an error “CAE, Insufficient valid axes for palletizing 3-point teaching data” will generate.

- This command cannot be used with a PASE (set palletizing axes) command. Whichever was set later will be given priority. (A single PAPS command can substitute a set of PASE, PAPT and PAST commands.)

- If this command is executed before BGPA is declared (= while palletizing setting is not enabled), an error, “CB5, BGPA not declared at palletizing setting” will generate.

- If the output field is specified, the output will turn ON after this command is executed.

- PSLI (Set zigzag)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PSLI	Offset amount	(Count)	CP

Set a zigzag palletizing.
The value specified in operand 1 will be set as the offset amount for even-numbered rows.
The count specified in operand 2 will be set as the count for even-numbered rows.
(Refer to “Palletizing Setting” – “Zigzag setting” under "How to Use.")
If operand 2 is not specified, the count for even-numbered rows will become the same as the count for odd-numbered rows.
If a setting is performed by 3-point teaching with PAPS (set palletizing points), the PX and PY-axes need not be parallel with the physical axes. In this case, the offset will apply in parallel with the PX-axis. If the offset is a positive value, the absolute value of offset will be applied toward the end-point direction of the PX-axis. If the offset is a negative value, the absolute value will be applied toward the start-point direction.
If this command is executed before a BGPA is declared (= while palletizing setting is not enabled), an error will generate.
If the output field is specified, the output will turn ON after this command is executed.

1.22 Palletizing Calculation Command

- PTNG (Get palletizing position number)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PTNG	Palletizing number	Variable number	CP

Assign the palletizing position number for the palletizing number specified in operand 1 to the variable specified in operand 2.
If the output field is specified, the output will turn ON after this command is executed.

- PINC (Increment palletizing position number by 1)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PINC	Palletizing number	Prohibited	CC

Increment by 1 the palletizing position number for the palletizing number specified in operand 1.
If the incremented value is considered normal as a palletizing position number calculated under the current palletizing setting, the value will be updated. If not, the value will not be updated.
If the output field is specified, the output will turn ON when the value was successfully incremented, and turn OFF if the increment failed.

- PDEC (Decrement palletizing position number by 1)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PDEC	Palletizing number	Prohibited	CC

Decrement by 1 the palletizing position number for the palletizing number specified in operand 1.
If the decremented value is considered normal as a palletizing position calculated under the current palletizing setting, the value will be updated. If not, the value will not be updated.
If the output field is specified, the output will turn ON when the value was successfully decremented, and turn OFF if the decrement failed.

- PSET (Set palletizing position number directly)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PSET	Palletizing number	Data	CC

Set the value specified in operand 2 as the palletizing position number for the palletizing number specified in operand 1.
If the specified value is considered normal as a palletizing position calculated under the current palletizing setting, the value will be set. If not, the value will not be set.
If the output field is specified, the output will turn ON when the palletizing position number was successfully updated, and turn OFF if the update failed.

- PARG (Get palletizing angle)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PARG	Palletizing number	Axis number	CP

Obtain the palletizing angle.
Calculate the palletizing angle (degrees) from the physical axis specified in operand 2 for the palletizing number specified in operand 1, and store the result in variable 199.

This command need not be executed, if not necessary.
If this command is executed after PAPS (set 3 palletizing points for teaching) is executed, the angle formed by the preferential axis and the specified physical axis will be calculated automatically. If this command is executed before PAPS is executed, or after both PAPS and PASE are executed in this order, an error will generate.
If point data is not available for two valid axes, an error “CAE, Insufficient valid axes for palletizing 3-point teaching data” will generate.
If the axis corresponding to the axis number in operand 2 does not specify one of the two valid axes associated with the point data, an error “CBA, Reference-axis/PX/PY-axis mismatch error at palletizing angle acquisition” will generate.
If the reference point data is the same as the point data at the PX-axis end point in 3-point teaching, an error “Reference-point/PX-axis end point duplication error at palletizing angle acquisition” will generate, and angle calculation will be disabled.
The actual operating direction may have been reversed depending on the mechanism of the rotating axis and the setting of axis-specific parameter No. 6, “Operating-direction reversing selection.” To use the value obtained by this command, be sure to confirm the actual operating direction.
If the output field is specified, the output will turn ON after this command is executed.

- PAPG (Get palletizing calculation data)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PAPG	Palletizing number	Position number	CP

Store the position coordinate data of the palletizing points for the palletizing number specified in operand 1, in the position number specified in operand 2.
If the output field is specified, the output will turn ON after this command is executed.

1.23 Palletizing Movement Command

- PMVP (Move to palletizing points via PTP)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PMVP	Palletizing number	Prohibited	PE

Move to the calculated palletizing points via PTP.
The axes will move to the palletizing points specified in operand 1, via PTP.
Executing this command will not increment the palletizing position number by 1.

- PMVL (Move to palletizing points via interpolation)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PMVL	Palletizing number	Prohibited	PE

Move to the calculated palletizing points via interpolation.
The axes will move to the palletizing points specified in operand 1, via interpolation.
Executing this command will not increment the palletizing position number by 1.

1.24 Building of Pseudo-Ladder Task

- CHPR (Change task level)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	CHPR	0 or 1	Prohibited	CP

[Function] Specify “1” (User HIGH) if you wish the target task to be processed before other tasks.
This command can also be used with non-ladder tasks.
Task level change (0: User NORMAL, 1: User HIGH) is not a required component, but specifying User HIGH will require a TSLP command explained below. (Without TSLP, tasks of the User NORMAL level will not be processed.)

- TPCD (Specify processing to be performed when input condition is not specified)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Prohibited	Prohibited	TPCD	0 or 1	Prohibited	CP

[Function] Specify the processing to be performed when input condition is not specified.
(0: Execute, 1: Follow the input condition in the last executed step)
In a ladder task, always input “1” (Follow the input condition in the last executed step) in operand 1.
In a non-ladder task, always input “0” (Execute). (The default value is “0.”)

- TSLP (Task sleep)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Prohibited	Prohibited	TSLP	Time	Prohibited	CP

[Function] Set the time during which the applicable task will sleep, in order to distribute the processing time to other tasks.
If the task level is set to User HIGH, this command must always be specified.
The applicable task will sleep during the set time.
The time in operand 1 is set in msec.
An appropriate time setting must be examined on the actual system. (Normally approx. 1 to 3 is set.)
(If the ladder statement becomes long, state this command multiple times between steps, as necessary.)
This command can also be used with non-ladder tasks.

1.25 Extended Command

- ECMD1 (Get motor current value (as percentage of rated current))

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	ECMD	1	Axis number	CC

[Function] Store the motor current value (percentage of the rated current) corresponding to the “axis number” specified in operand 2, in variable 99.

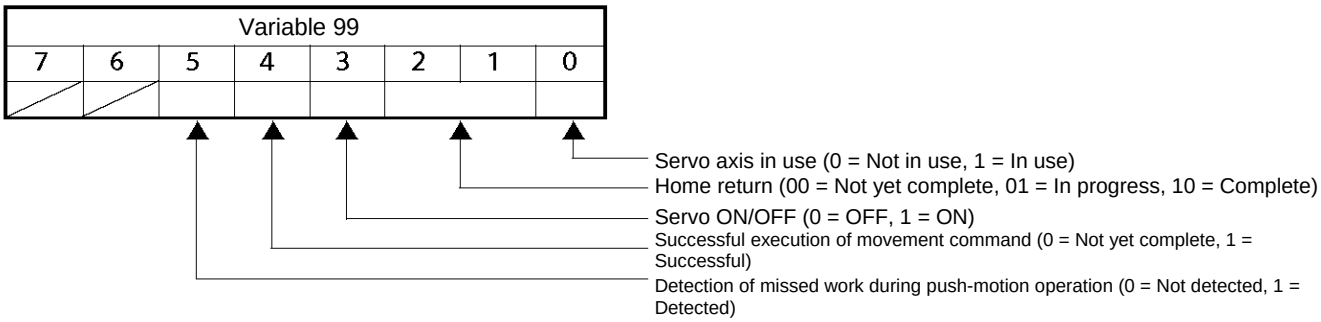
Note: The current value data (percentage of the rated current) obtained by this command has been processed by feedback current filtering and includes analog error.

[Example] ECMD 1 2 Extended command 1
Store the motor current value (percentage of the rated current) of axis 2, in variable 99.

- ECMD5 (Get axis operation status)

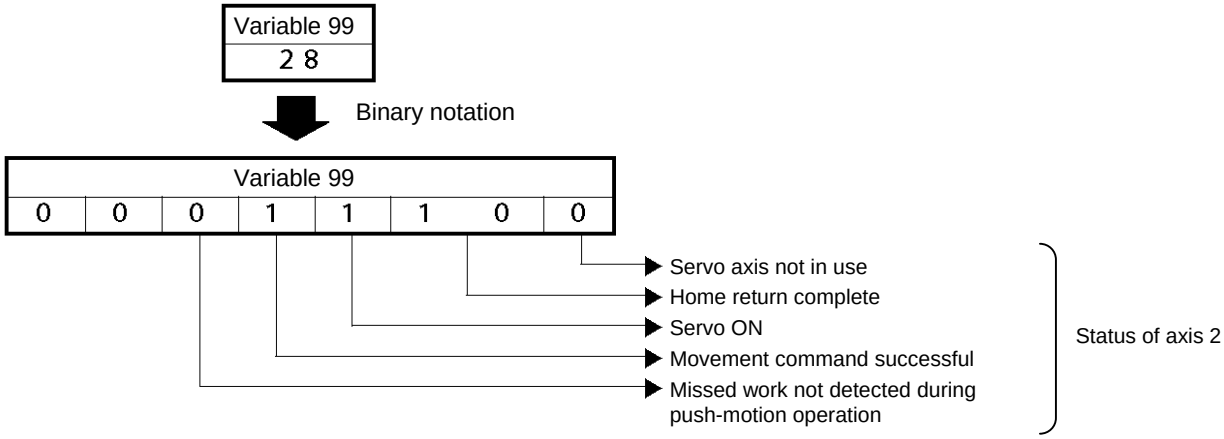
Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	ECMD	5	Axis number	CC

[Function] Store the status of the axis specified in operand 2, in variable 99.
The axis status is indicated by the ON/OFF level of each bit, as shown below. Accordingly, the obtained value must be converted to a binary value for interpretation.



(Note) If an invalid axis number is specified in operand 2, “C44, SEL data error” will generate.

[Example] ECMD 5 2 Store the status of axis 2 in variable 99. If 28 (decimal value) was stored in variable 99 after the command was executed, the status of axis 2 is interpreted as follows.



- ECMD20 (Get parameter value)

Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	ECMD	20	Variable number	CC

[Function] Store the value of the specified parameter in variable 99, using the data stored in the three consecutive variables starting from the one corresponding to the variable number specified in operand 2.
If variable No. n is set in operand 2, the data in variable No. n will indicate the parameter type, data in variable No. n+1 will indicate the device number (or axis number), and data in variable No. n+2 will indicate the parameter number, respectively. The ranges of parameter type, device number (or axis number) and parameter number are specified below. If an out-of-range value is specified, “C44, SEL data error” will generate.

	I/O	Common to all axes	Axis-specific	Driver	Encoder	I/O device	Other
Parameter type	0	1	2	3	4	5	7
Device number/axis number	0	0	1 ~ 2	1 ~ 2	1 ~ 2	0 ~ 7	0
Parameter number	1 ~ 300	1 ~ 120	1 ~ 200	1 ~ 97	1 ~ 30	1 ~ 82	1 ~ 100

Specify an integer variable in operand 2 (integer variables 98, 99, 298, 299, 1098, 1099, 1298 and 1299 cannot be specified, because three consecutive integer variables cannot be allocated if any of these integer variables is specified). If a variable of non-integer type is specified, “C3C, Variable number error” will generate.

(Note) If an invalid axis number is specified in operand 2, “C44, SEL data error” will generate.

[Example 1]

```
LET   10   2    Variable No. 10 = Parameter type (Axis-specific)
LET   11   2    Variable No. 11 = Axis number (Axis 2)
LET   12   42   Variable No. 12 = Parameter number (No. 42)
ECMD  20   10   Extended command 20 (Use variable Nos. 10 through 12)
                Store the value of axis-specific parameter No. 42 (axis 2),
                “Encoder resolution,” in variable 99.
```

[Example 2]

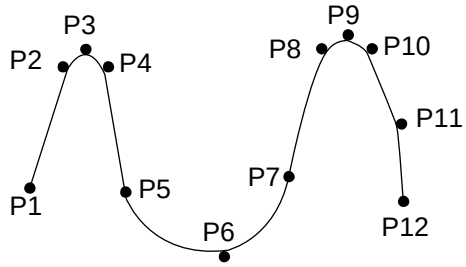
```
LET   1250   0    Variable No. 1250 = Parameter type (I/O)
LET   1251   0    Variable No. 1251 = Device number (0, in the case of I/O
                  parameter)
LET   1252   30   Variable No. 1252 = Parameter number (No. 30)
ECMD  20     1250 Extended command 20 (Use variable Nos. 1250 through
                  1252)
                  Store the value of I/O parameter No. 30, “Input function
                  selection 000,” in variable 99.
```

Chapter 4 Key Characteristics of Actuator Control Commands and Points to Note

1. Continuous Movement Commands [PATH, CIR, ARC, PSPL, CIR2, ARC2, ARCD, ARCC]

[1] By running a program with continuous movement commands input in a series of continuous program steps, you can allow the actuators to perform operations continuously without stopping between steps.

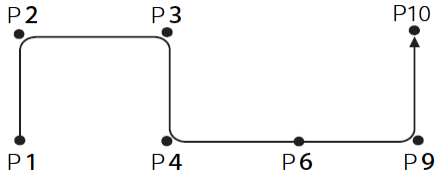
PATH	1	5
ARC2	6	7
PATH	8	12



[2] Continuous operation is possible even when all positions are not continuous.
Specify the position number of the discontinuous position for both the start position number and end position number in the PATH command. In this example, position No. 6 is a skip point.

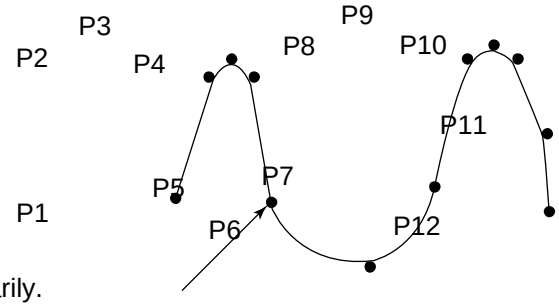
The actuator moves continuously in the sequence of position Nos. 1 → 2 → 3 → 4 → 6 → 9 → 10.

PATH	1	4	
ARC2	6	6	(Discontinuous position)
PATH	9	10	



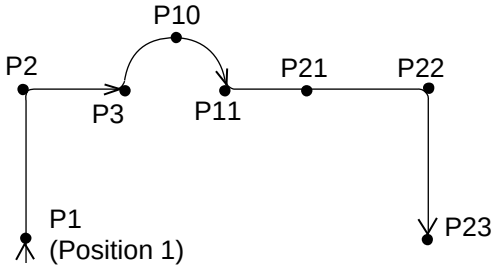
[3] Continuous movement will not be achieved if an input condition is specified for any continuous movement command.

	PATH	1	5
20	ARC2	6	7
	PATH	8	12



Stops momentarily.

[4] The output field of each command will turn ON as the end position of that command approaches. Only with the last command in a series of continuous movement commands, the output will turn ON upon completion of operation (if there is no input condition).



[Example 1]

```
(POTP = 1)
POTP    1
  |
  |
  |
PATH    1     3     600
ARC2    10    11    603
PATH    21    23    604
  |
  |
```

Output field	Timing
600	Turn ON as P1 approaches.
601	Turn ON as P2 approaches.
602	Turn ON as P3 approaches.
603	Turn ON as P11 approaches.
604	Turn ON as P21 approaches.
605	Turn ON as P22 approaches.
606	Turn ON when P23 operation is complete.

[Example 2]

```
(POTP = 0)
PATH    1     3     600
ARC2    10    11    603
PATH    21    23    604
```

Output field	Timing
600	Turn ON as P3 approaches.
603	Turn ON as P11 approaches.
604	Turn ON when P23 operation is complete.

[Example 3] If an input condition is specified, the output will turn ON upon completion of operation in the step before the one in which the input condition is specified.

```
POTP    1
  |
  |
  |
PATH    1     3     600
ARC2    10    11    603
PATH    21    23    604
```

Output field	Timing
600	Turn ON as P1 approaches.
601	Turn ON as P2 approaches.
602	Turn ON when P3 operation is complete.
603	Turn ON as P11 approaches.
604	Turn ON as P21 approaches.
605	Turn ON as P22 approaches.
606	Turn ON when P23 operation is complete.

[5] When executing continuous movement commands sequentially, the controller is calculating approx. 100 positions ahead. This is why the steps are displayed continuously on the PC screen or teaching-pendant screen, regardless of the actual operation. The last step in the continuous operation section executed by continuous movement commands will wait for the applicable operation to complete.

```
  |
  |
  |
PATH    1     5      <- Actuator operation
ARC     6     7
PATH    8     12     <- Step displayed on the PC software or teaching pendant
BTON    310
  |
  |
  |
```

[6] Do not allow the output fields to duplicate in the continuous operation section executed by continuous movement commands.
Duplicating output fields in the continuous operation section will not achieve the expected result.
The output field will turn OFF at the start of processing of each command.

```
POTP    1
PATH    1     5     605   }
  |                       } Continuous operation section executed by continuous
  |                       } movement commands
  |
PATH    11    15    604   }
```

Do not let outputs 605 and 604 to duplicate, as in the example shown at left.

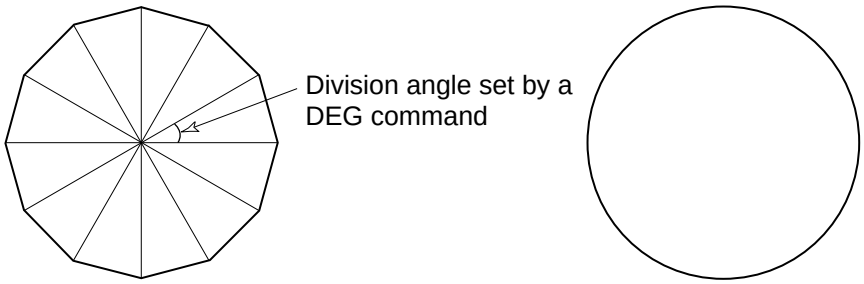
The final output status of duplicate 605 and 604 is indeterminable, because it is affected by the positioning calculation time and the relationship of durations of actual operations.

2. PATH/PSPL Commands

When executing a PATH or PSPL command, pay attention to the locus because it will change if the acceleration/deceleration is different between points.
The locus can be fine-tuned by changing the acceleration/deceleration, but different acceleration/deceleration settings between points will prevent smooth transition of speeds when moving from one position to another.
If there is a large difference in deceleration/acceleration between points and the positioning distance is small, the speed may drop. Exercise caution.

3. CIR/ARC Commands

The processing by a CIR or ARC command resembles moving along a polygon with a PATH command.
A small division angle may cause the speed to drop.
CIR2, ARC2, ARCD and ARCC commands actually perform arc interpolation.



CIR CIR2

4. CIR2/ARC2/ARCD/ARCC Commands

With a CIR2, ARC2, ARCD or ARCC command, the speed can be changed (only in the arc interpolation section) by inputting a speed for the point specified in operand 1. These commands are effective when you must lower the speed partially because the radius is small and the arc locus cannot be maintained inside the allowable range.

The speed and acceleration will take valid values based on the following priorities:

Priority	Speed	Acceleration (deceleration)
1	Setting in the position data specified in operand 1	Setting in the position data specified in operand 1
2	Setting by VEL command	Setting by ACC (DCL) command
3		Default acceleration in all-axis parameter No. 11 (Default deceleration in all-axis parameter No. 12)

Chapter 5 Palletizing Function (2-axis Specification)

The SEL language used by the PSEL Controller provides palletizing commands that support palletizing operation. These commands allow simple specification of various palletizing settings and enable arch motion ideal for palletizing.

1. How to Use

Use palletizing commands in the following steps:

(1) Palletizing setting
Set palletizing positions, arch motion, etc., using palletizing setting commands.

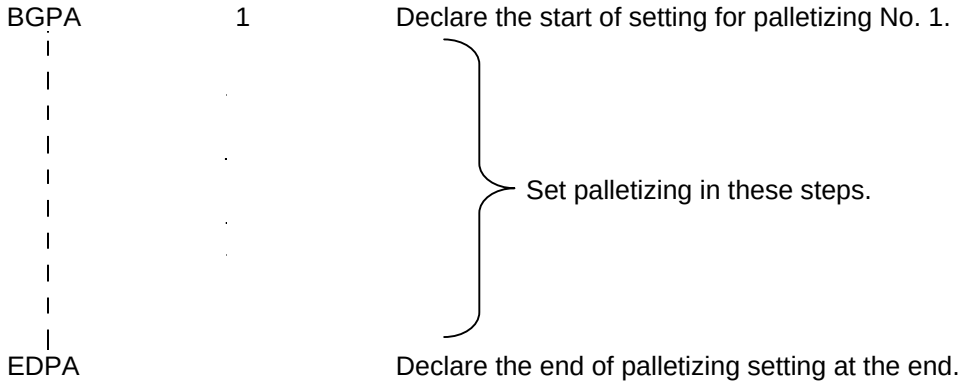
(2) Palletizing calculation
Specify palletizing positions using palletizing calculation commands.

(3) Palletizing movement
Execute motion using palletizing movement commands.

2. Palletizing Setting

Use the palletizing setting commands to set items necessary for palletizing operation. The setting items include the following:

(1) Palletizing number setting --- Command: BGPA
At the beginning of a palletizing setting, determine a palletizing number using a BGPA command to declare the start of palletizing setting.
At the end, declare the end of palletizing setting using an EDPA command.



A maximum of 10 sets (palletizing Nos. 1 to 10) of palletizing setting can be specified for each program.

(2) Palletizing pattern --- Command: PAPN
Select a pattern indicating the palletizing order.
The two patterns illustrated below are available.
The encircled numbers indicate the order of palletizing and are called “palletizing position numbers.”

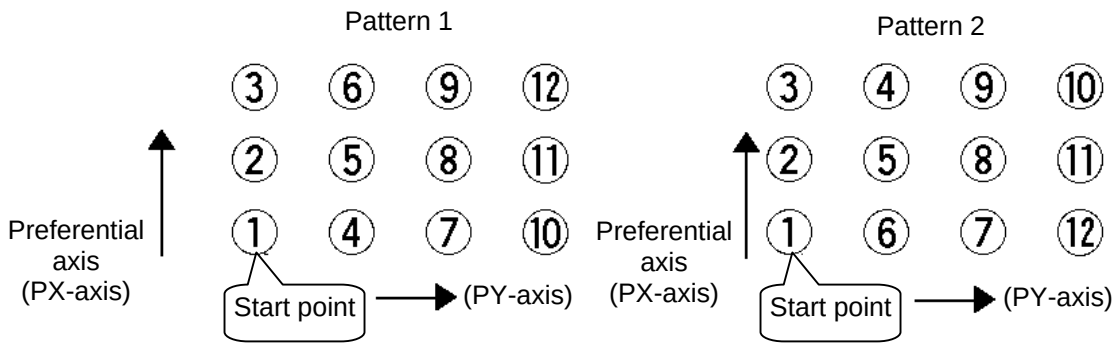


Fig. 1

PAPN	2	When pattern 2 is selected (Setting is not necessary if pattern 1 is selected.)

The row from 1 to 3 to be placed first is called the “preferential axis (PX-axis),” while the other direction comprising the palletizing plane is called the “PY-axis.”

(3) Palletizing counts --- Command: PAPI
Set the palletizing counts.

PAPI	3	4	Count for preferential axis (PX-axis): 3, Count for PY-axis: 4

(4) Palletizing position setting
Palletizing position setting is performed mainly by method A or B, as explained below. Set the palletizing positions for each palletizing setting based on method A or B.

	Setting method	Commands
A	3-point teaching method Set three position-data points specifying the palletizing positions.	PAPS
B	Method to set palletizing positions in parallel with the actuators Set from the palletizing axes, palletizing reference point and palletizing pitches.	PASE, PAST, PAPT

A. 3-point teaching method

To set the palletizing positions by 3-point teaching, store desired positions in position data fields as three continuous position data and then specify the first position number using a PAPS command. This method allows you to set the PX-axis and PY-axis as three-dimensional axes not parallel with the actuators and not crossing with each other.
In the example shown below, position data ①, ③ and ⑩ are stored in three continuous position data fields.

When three points are taught from position No. 11

Position No. 11	[1]:	Start point (First palletizing position)
Position No. 12	[3]:	Palletizing position corresponding to the end point in the PX-axis direction
Position No. 13	[10]:	Palletizing position corresponding to the end point in the PY-axis direction

The encircled numbers indicate palletizing position numbers (palletizing order).

Use a PAPS command to specify the position number corresponding to the start point.

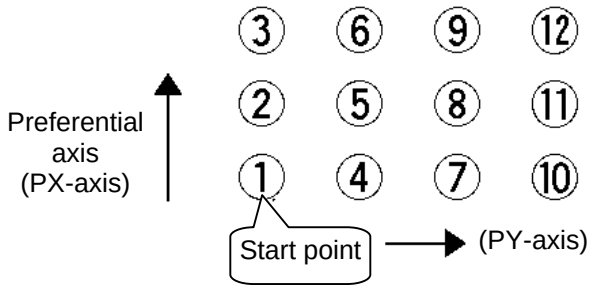


Fig. 1

```
PAPS        11
```

The pitches are calculated automatically from the count set for each axis.
When setting data for 3-point teaching, specify position data for two axes.

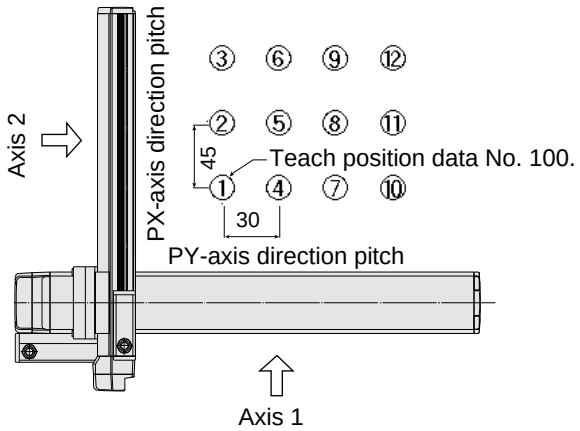
B. Method to set palletizing positions in parallel with the actuators

Palletizing reference point: Store the position data of the start point (palletizing position No. 1) in a position data field and specify the applicable position number using a PAST command, as shown below.

Palletizing pitches: Use a PAPT command to specify the pitches in the PX-axis and PY-axis directions.

Palletizing axes: Use a PASE command to specify the two axes, one representing the PX-axis direction and the other representing the PY-axis direction, to be used in palletizing.

(An actuator axis number parallel with the preferential axis (PX-axis) and another perpendicular to the preferential axis)



PAST	100		Teach position data No. 100 as the start point.
PAPT	45	30	The PX-axis direction pitch is 45 mm and the PY-axis direction pitch is 30 mm.
PASE	2	1	Set the PX-axis as axis 2 and PY-axis as axis 1.

(Note) When the above palletizing axes, palletizing pitches and palletizing reference point are used, the PX-axis and PY-axis must be parallel with the actuators and crossing with each other.

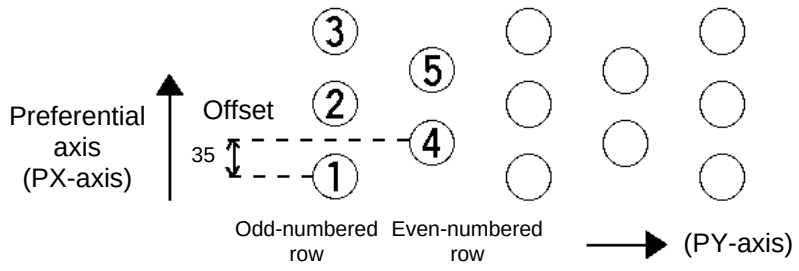
Select either method A or B for each palletizing setting.

(5) Zigzag setting --- Command: PSLI

Use a PSLI command to set a zigzag layout as shown below.

Zigzag offset: Offset amount in the preferential-axis direction, which will be applied when even-numbered rows are placed.
"Even-numbered rows" refer to the rows occurring at the even numbers based on the row placed first representing the first row.

Zigzag count: Number in the even-numbered rows. Two in the diagram below.



PSLI 35 2

3. Palletizing Calculation

The items that can be operated or obtained using palletizing calculation commands are shown below:

(1) Palletizing position number Commands --- PSET, PINC, PDEC, PTNG
Number showing the ordinal number of a palletizing point.
(In Fig. 1 given in the explanation of palletizing pattern, the encircled numbers are palletizing position numbers.)

Always set this command before executing a palletizing movement command --- PSET

For example, executing a palletizing movement command by setting 1 as the palletizing position number will move the axes to the start point. Executing a palletizing movement command by setting 2 as the palletizing position number will move the axes to the point immediately next to the start point in the PX-axis direction.

(2) Palletizing angle Command --- PARG
This is the angle formed by the physical axis (actuator) and the preferential palletizing axis (PX-axis) (θ in the figure below).
In the figure below, θ will become a negative value if axis 1 is used as the reference for angle calculation.

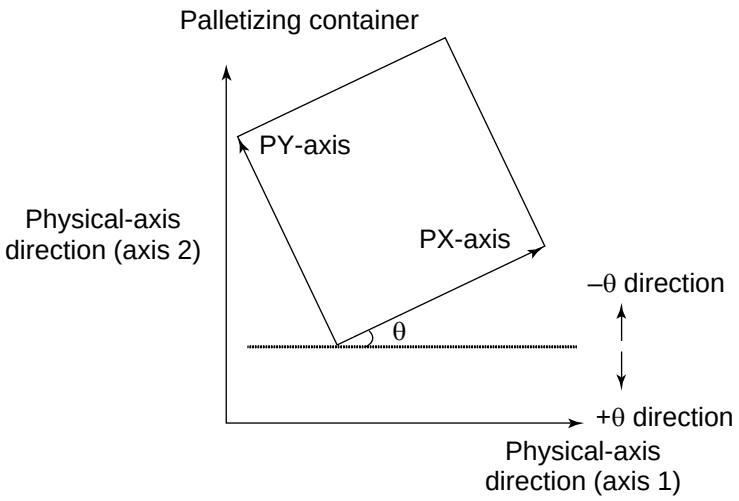


Fig. 4

With PSEL commands, executing a “get palletizing angle” command following a palletizing setting via 3-point teaching will automatically obtain the palletizing angle.

(3) Palletizing calculation data Command --- PAPG
When a palletizing position number is set, this data refers to the position coordinate data of the palletizing point corresponding to that palletizing position number.
Note that this position coordinate data does not reflect normal offset or palletizing Z-axis offset.

4. Palletizing Movement

Palletizing movement commands are used to move the actuator to palletizing points.

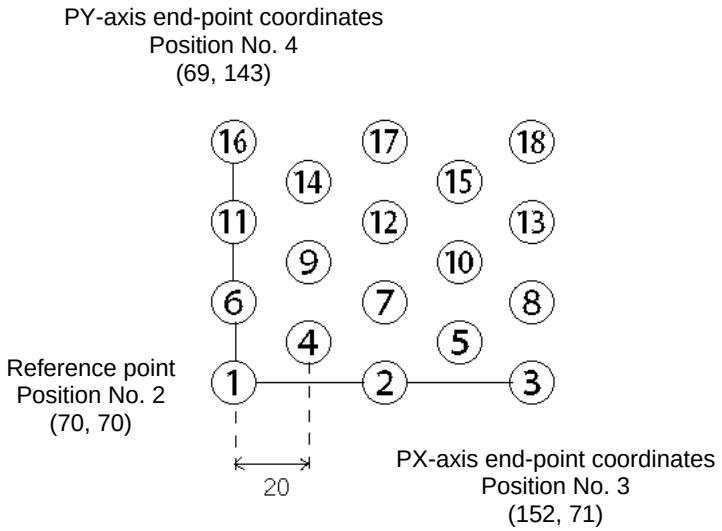
(1) Movement commands to palletizing point --- PMVP, PMVL
Position coordinates of a two-dimensionally placed palletizing point are calculated and movement is performed using the calculated point as the end point. (The axes will move to the palletizing point of the palletizing position number specified in the executed command.)
Two actuator axes will be required to comprise a two-dimensional plane.
PMVP: Move from the current position to a palletizing point via PTP.
PMVL: Move from the current position to a palletizing point via interpolation.

5. Program Examples

(1) Simple program example (two-axis specification) using PAPS (set by 3-point teaching)
The example below specifies movement only and does not cover picking operation.

No.	B	E	N	Cnd	Cmnd	Operand 1	Operand 2	Pst	Comment
1					BGPA	1			Start setting palletizing number 1
2					PAPI	3	7		Palletizing counts 3x7
3					PAPS	2			Set 3 point for teaching
4					PSLI	20	3		Zigzag offset = 20 mm
5					EDPA				End palletizing number 1 setting
6									
7					VEL	200			Speed 20 mm/sec.
8					MOVP	1			Move to pick position
9					PSET	1	1		Set palletizing position number to 1
10					TAG	1			
11					PMVP	1			Move to palletizing points via PTP
12					MOVP	1			Move to pick position via PTP
13					PINC	1		600	Palletizing position number by +1
14				600	GOTO	1			Loop begging when PINC successful
15					EXIT				
16									

No.	Axis1	Axis2	Vel	Acc	Dcl
1	10.000	10.000			
2	70.000	70.000			
3	152.000	71.000			
4	69.000	143.000			
5					

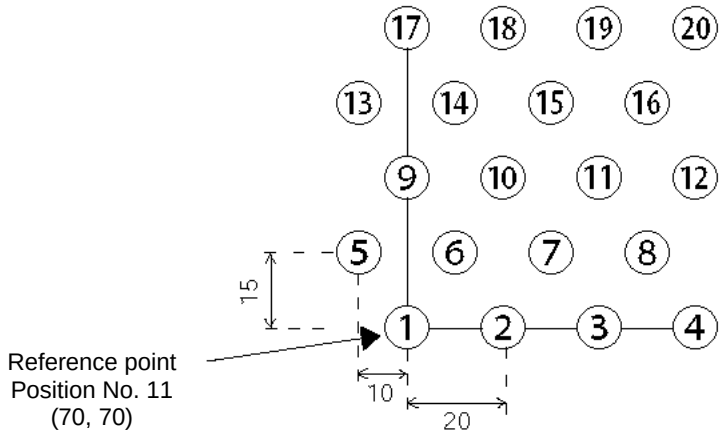


● Picking position
Position No. 1

(2) Simple program example (two-axis specification) using PAPS, PAPT and PAST
The example below specifies movement only and does not cover picking operation.

No.	B	E	N	Cnd	Cmnd	Operand 1	Operand 2	Pst	Comment
1					BGPA	2			Start setting palletizing number 2
2					PAPI	4	5		Palletizing counts 4x5
3					PASE	1	2		PX axis = 1, PY axis = 2
4					PAPT	20	15		Pitch X = 20, Y = 15
5					PAST	11			Position number 11 reference point
6					PSLI	-10	4		Zigzag offset = 10 mm
7					EDPA				End palletizing number 2 setting
8									
9					VEL	200			Speed 20 mm/sec.
10					MOVP	10			Move to pick position
11					PSET	2	1		Set palletizing position number to 1
12					TAG	1			
13					PMVP	2			Move to palletizing points via PTP
14					MOVP	10			Move to pick position via PTP
15					PINC	2		600	Palletizing position number by +1
16				600	GOTO	1			Loop begging when PINC successful
17					EXIT				
18									

No.	Axis1	Axis2	Vel	Acc	Dcl
10	10.000	10.000			
11	70.000	70.000			
12					



● Picking position
Position No. 10

Chapter 6 Pseudo-Ladder Task

With the PSEL Controller, a pseudo-ladder task function can be used depending on the command and extension condition.
The input format is shown below. Note that this function must be used by expert engineers with a full knowledge of PLC software design.

1. Basic Frame

Extension condition E	N	Input condition Cnd	Command Cmnd	Operand 1	Operand 2	Output Pst	
LD		7001	CHPR	1			
			TPCD	1			
			TAG	1			
⎸ ⎸ ⎸ ⎸ ⎸ ⎸		⎸ ⎸ ⎸ ⎸ ⎸ ⎸	⎸ ⎸ ⎸ ⎸ ⎸ ⎸	⎸ ⎸ ⎸ ⎸ ⎸ ⎸			Ladder statement field
LD		7001	TSLP	1 ~ 100			
⎸ ⎸ ⎸ ⎸ ⎸ ⎸		⎸ ⎸ ⎸ ⎸ ⎸ ⎸	⎸ ⎸ ⎸ ⎸ ⎸ ⎸	⎸ ⎸ ⎸ ⎸ ⎸ ⎸			Ladder statement field
LD		7001	TSLP	1 ~ 100			
LD		7001	GOTO	1			
LD		7001	EXIT				

* * Virtual input 7001: “Normally ON” contact

2. Ladder Statement Field

[1] Extension conditions

LD ..... LOAD
A ..... AND
O ..... OR
AB ..... AND BLOCK
OB ..... OR BLOCK

All of the above extension conditions can be used in non-ladder tasks.

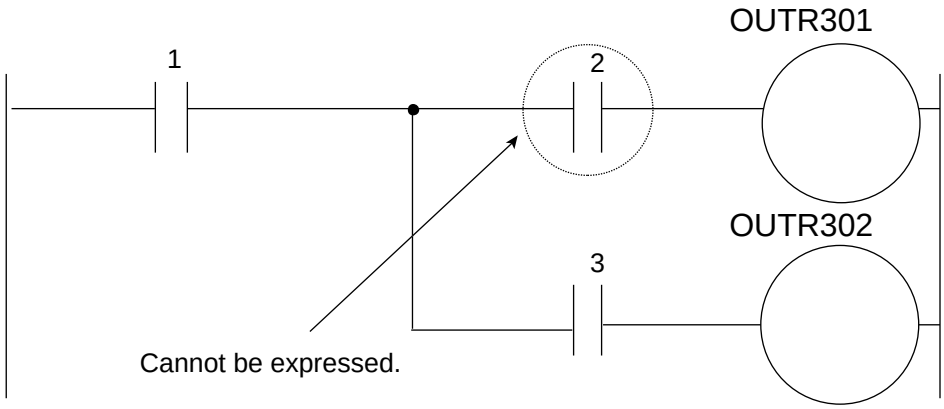
[2] Ladder commands

OUTR ..... Ladder output relay (Operand 1 = Output, flag number)
TIMR ..... Ladder timer relay (Operand 1 = Local flag number, Operand 2 = Timer setting (sec))

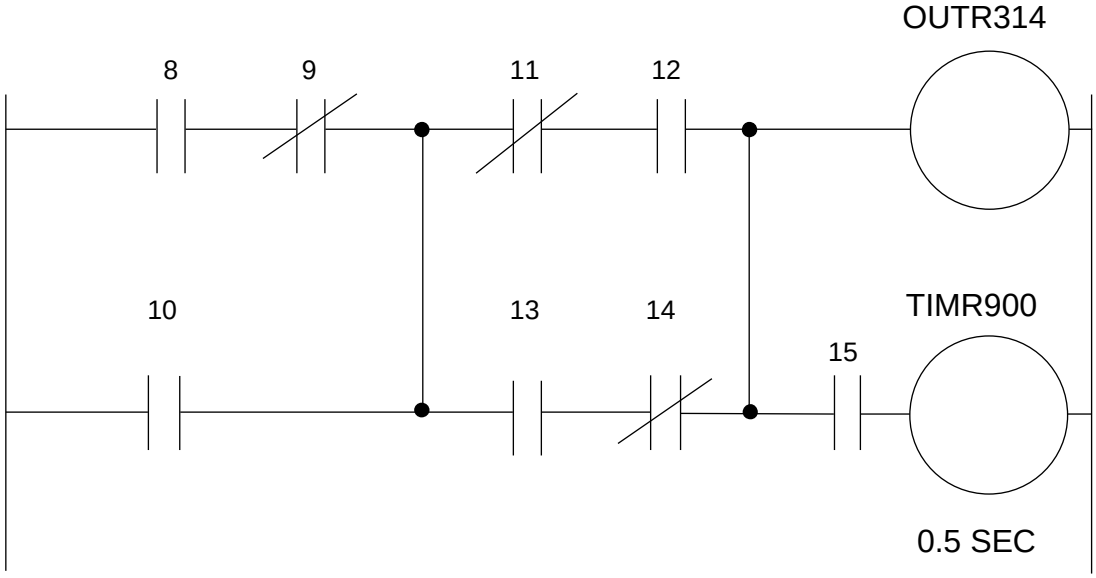
3. Points to Note

- This system only processes software ladders using an interpreter. Therefore, the processing time is much longer than that of a dedicated commercial sequencer.
  (This system is not suitable for large-scale ladder processing.)
- If an extension condition is not specified for steps in which an input condition is specified, the steps will be treated as LD (LOAD).
- Always specify a “normally ON” contact for those steps that must be processed without fail, such as CHPR, TSLP and GOTO. (LD 7001)
  Virtual input 7001: “Normally ON” contact

- The following circuit cannot be expressed. Create an equivalent circuit.



4. Program Example



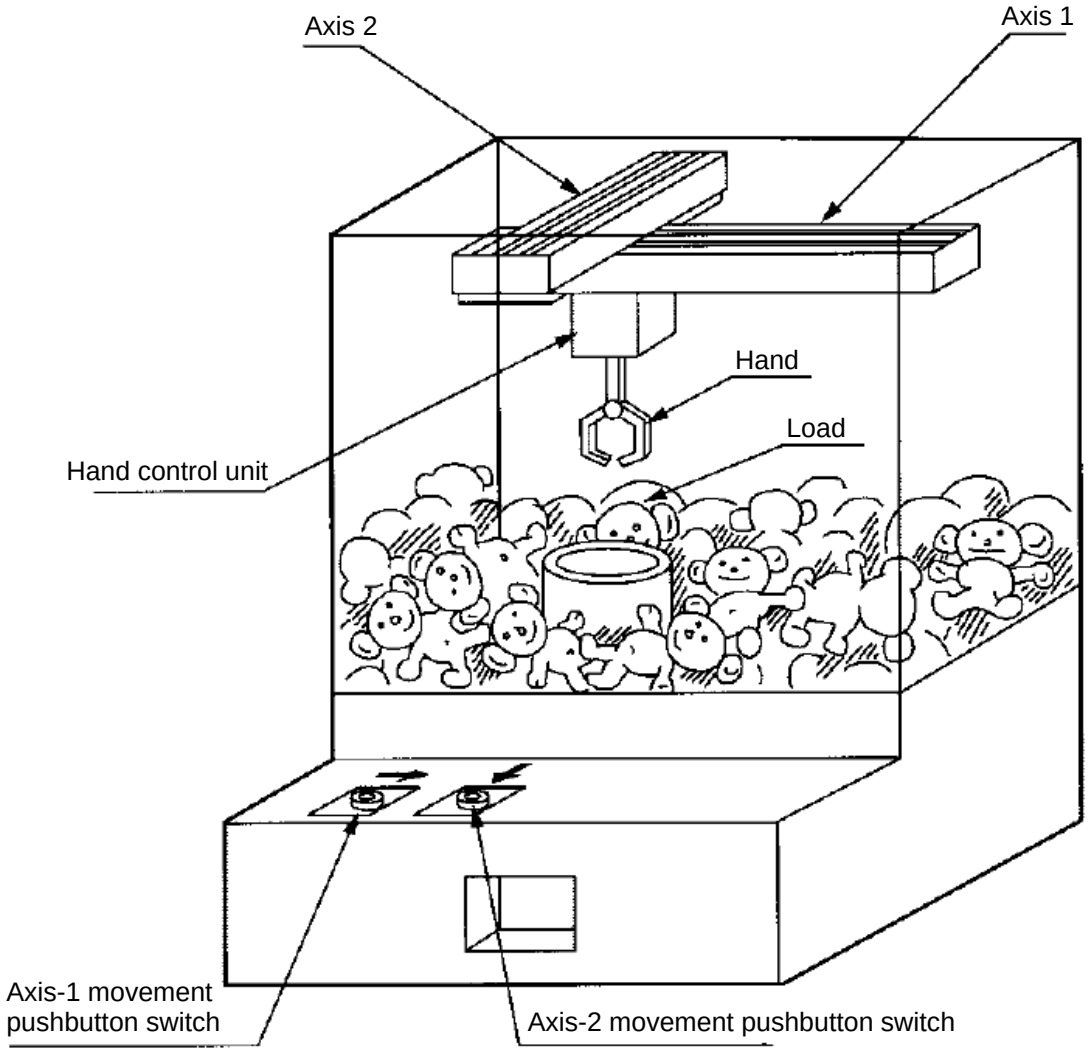
Extension condition E	N	Input condition Cnd	Command Cmnd	Operand 1	Operand 2	Output Pst
LD		7001	CHPR	1		
			TPCD	1		
			TAG	1		
LD		8				
A	N	9				
O		10				
LD	N	11				
A		12				
LD		13				
A	N	14				
OB						
AB			OUTR	314		
A		15	TIMR	900	0.5	
LD		7001	TSLP	3		
LD		7001	GOTO	1		
LD		7001	EXIT			

Chapter 7 Application Program Examples

1. Operation by Jog Command [Doll-Picking Game Machine]

(1) Overview of the system

This system is a doll-picking game machine consisting of axis-1 and axis-2 actuators. Pushbutton switches corresponding to the two axes are provided on an external operation switch box, and these switches are used to move the actuators to a desired position to grab and pick up dolls inside the case.

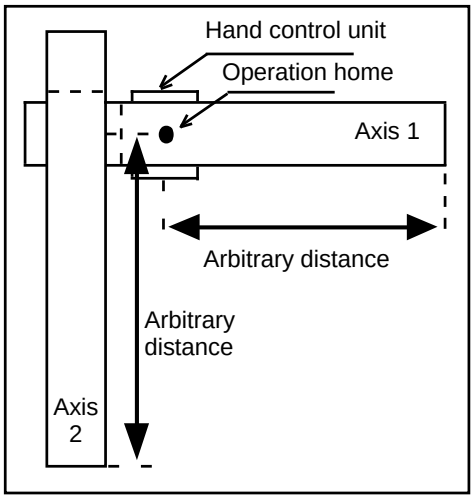


(2) Explanation of the operation

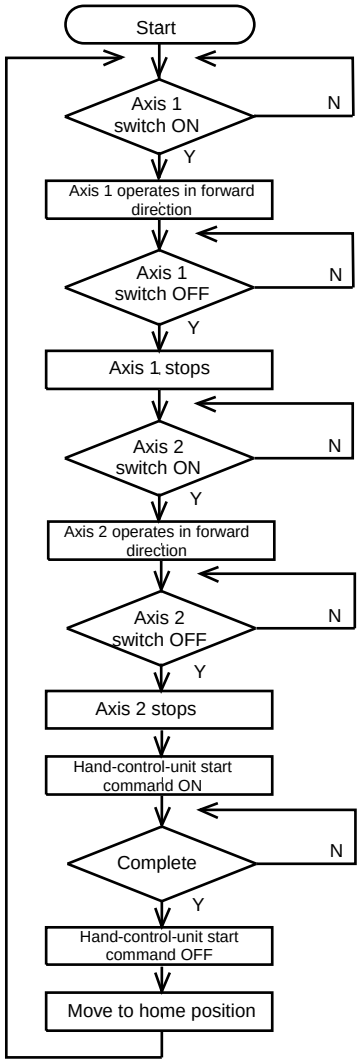
[1] Wait for the axis-1 movement pushbutton switch to turn ON.
[2] The X-axis moves while the pushbutton switch is ON, and stops when the switch turns OFF.
[3] Wait for the axis-2 movement pushbutton switch to turn ON.
[4] The Y-axis moves while the pushbutton switch is ON, and stops when the switch turns OFF.
[5] Output a start command to the hand control unit.
[6] Wait for an operation completion input from the hand control unit.
[7] Move to the home after the input is received.

The above operation will be repeated. The operation position, external I/O assignments and operation flow chart of this operation are shown below:

Operation Position



Operation Flow Chart



I/O Assignments

Category		I/O No.	Signal name	Specification
PSEL	Input	16	Axis-1 movement command	Pushbutton switch
		17	Axis-2 movement command	Pushbutton switch
		18	Hand operation completion	External control unit
	Output	307	Hand start command	24 VDC
	* Flag is not used.			

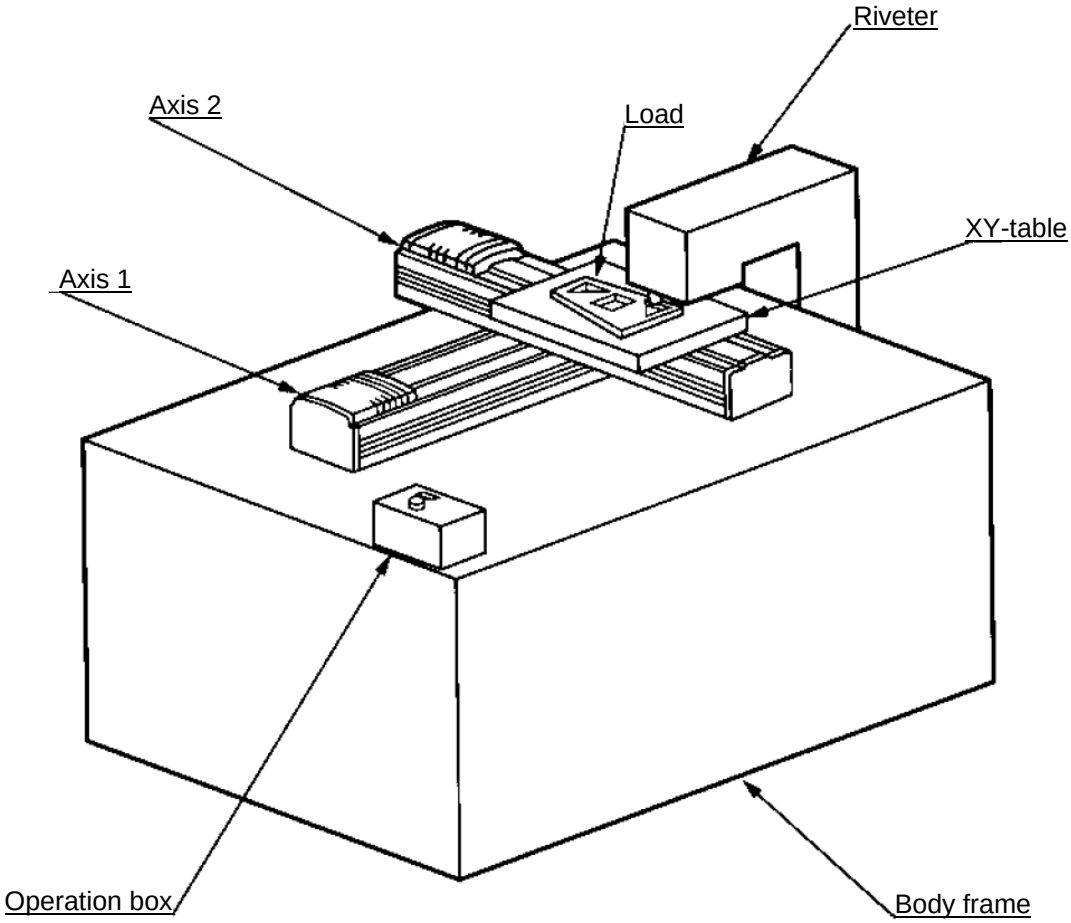
(3) PSEL Controller application program

Step	E	N	Cnd	Cmnd	Operand 1	Operand 2	Pst	Comment
1				HOME	11			Axes 1 and 2 return to home (servo ON).
2				VEL	400			Set speed to 400 mm/s.
3				TAG	1			
4				WTON	16			Wait for input from axis-1 movement switch.
5				JFWN	1	16		Move forward while axis-1 movement switch is ON.
6				WTON	17			Wait for input from axis-2 movement switch.
7				JFWN	10	17		Move forward while axis-2 movement switch is ON.
8				BTON	307			Start command for external control unit turns ON.
9				WTON	18			Wait for external control unit to complete operation.
10				BTOF	307			Start command for external control unit turns OFF.
11				JBWF	11	18		Axes 1 and 2 move backward while 18 is ON.
12				GOTO	1			Jump to TAG1.
13								
14								
15								
16								
17								
18								
19								
20								
21								
22								
23								
24								
25								
26								
27								
28								
29								
30								
31								
32								

2. Operation by Point Movement Command [Riveting System]

(1) Overview of the system

This system is a riveting system consisting of an XY-table operated by axis-1 and axis-2 actuators and a riveter. By setting a load on the XY-table at the operation home and turning on the start switch, rivets will be driven at the three points specified on the load.

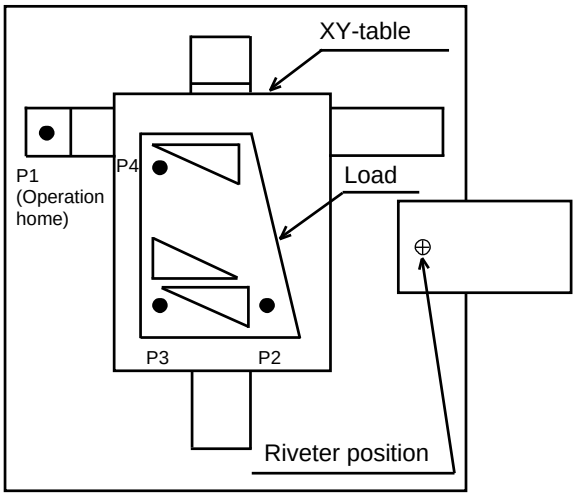


(2) Explanation of the operation

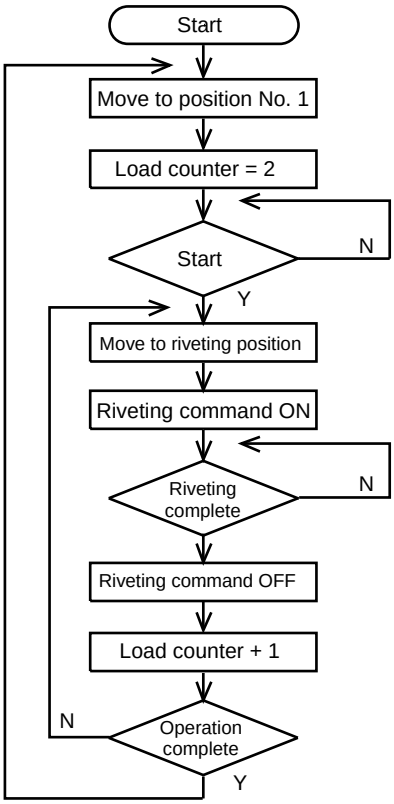
[1] The XY-table moves to the operation home (P1) and waits.
[2] The operator sets a load on the XY-table and turns on the start switch.
[3] The XY-table moves to riveting position No. 1 (P2) on the load and a riveting command is output to the riveter.
[4] When the riveter completes the riveting operation and a completion signal is input, the table will move to riveting position No. 2 (P3) and then to No. 3 (P4), in the same manner.
[5] When all three points have been riveted, the table will return to the operation home (P1).

The above operation will be repeated. The operation position, external I/O assignments and operation flow chart of this operation are shown below:

Operation Position



Operation Flow Chart



I/O Assignments

Category		I/O No.	Signal name	Specification
PSEL	Input	16	Start command	Pushbutton switch
		17	Riveting completion	Contact signal
	Output	307	Riveting command	24 VDC
	* Flag is used from 600.			

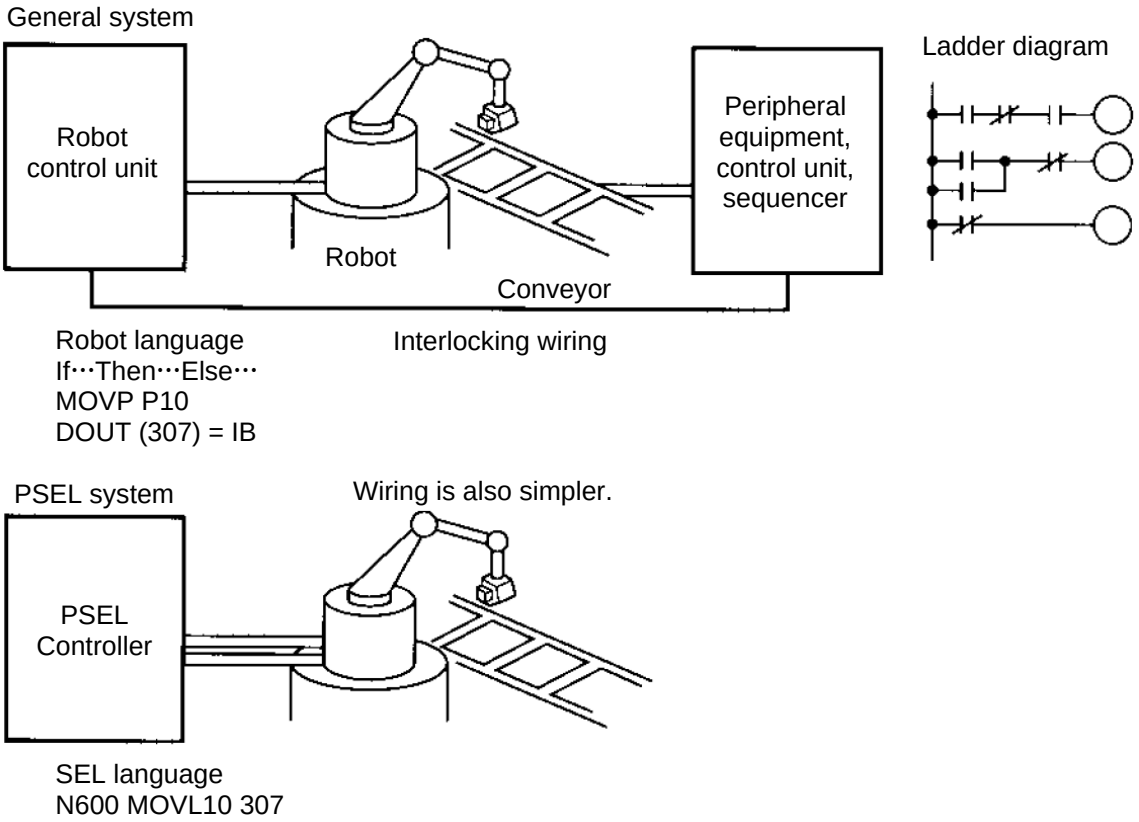
(3) PSEL Controller application program

Step	E	N	Cnd	Cmnd	Operand 1	Operand 2	Pst	Comment
1				HOME	11			XY-table returns to home (servo ON).
2				VEL	400			Set speed to 400 mm/s.
3				TAG	1			
4				MOVL	1			Move to position No. 1 (operation home)
5				LET	1	2		Set 2 in load counter.
6				BTOF	600			Clear completion flag.
7				WTON	16			Wait for start command.
8				TAG	2			
9				MOVL	*1			Move to load counter position.
10				BTON	307			Riveting command turns ON.
11				WTON	17			Wait for riveting to complete.
12				BTOF	307			Riveting command turns OFF.
13				ADD	1	1		Increment load counter by 1.
14				CPEQ	1	5	600	Turns ON flag if operation is complete.
15		N	600	GOTO	2			Jump to TAG2 if not complete.
16				GOTO	1			Jump to TAG1 if complete.
17								
18								
19								
20								
21								
22								
23								
24								
25								
26								
27								
28								
29								
30								
31								
32								

Chapter 8 Real-Time Multi-Tasking

1. SEL Language

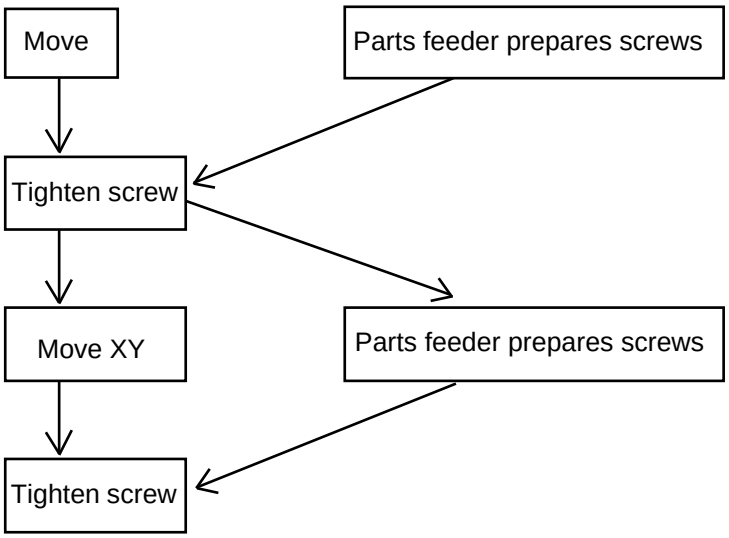
The PSEL Controller allows integrated control of actuators and peripherals with a single controller using its 32-bit RISC CPU and high-speed real-time operating system. There is no need to learn various languages for different units, such as robot language for robots and sequencer language for peripherals. Since SEL language is the only language used, an efficient system can be designed.
The current version of SEL language represents a pioneering evolution of the widely proven programming language, evidenced by higher-performance features and advanced functions. The latest version is also easier to use compared with the conventional SEL language.



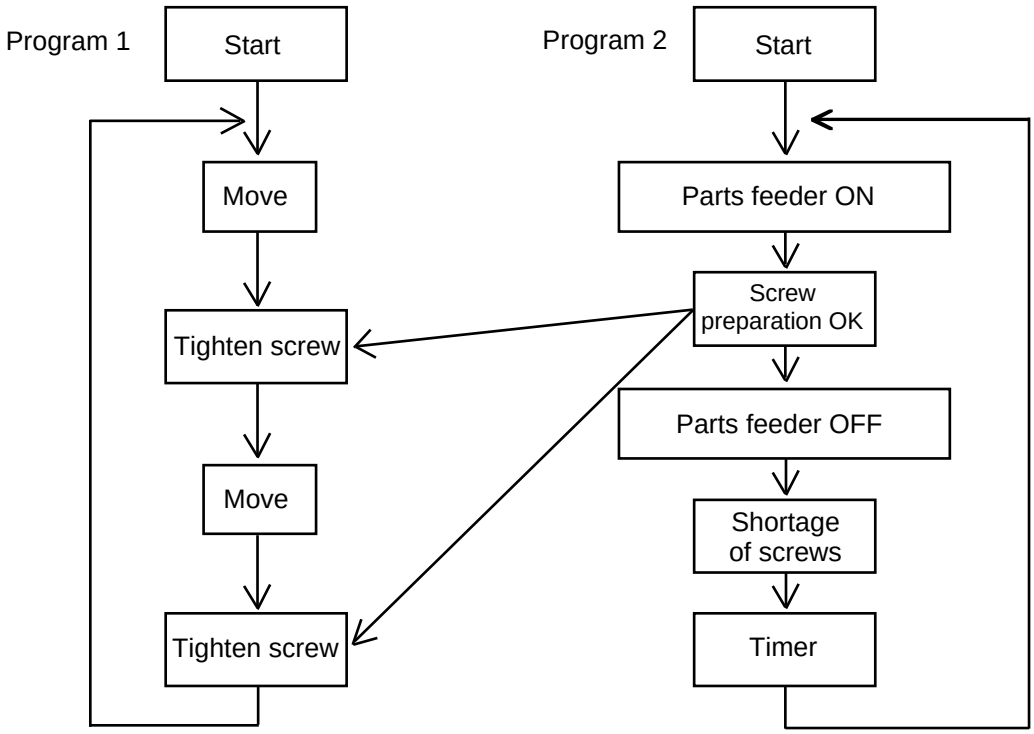
2. Multi-Tasking

“Multi-tasking” operation may not be a familiar term, but it is widely used in computer programming to refer to parallel processing. Simply put, multi-tasking means running several programs in parallel.
Take a screw-tightening robot, for example.
In general, a screw-tightening robot consists of axis-1 and axis-2 actuators and a screw-tightening machine (up/down air cylinder, etc.).

Operation Flow



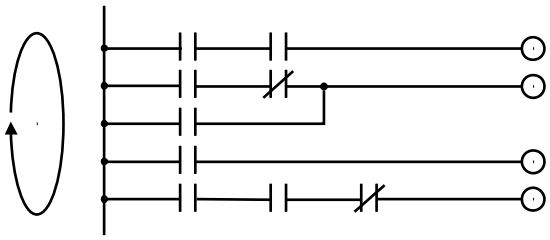
Although the flow chart is simple, the movement of axis-1 and axis 2 actuators and the operation of the parts feeder must take place simultaneously. This requires “multi-tasking” operation.



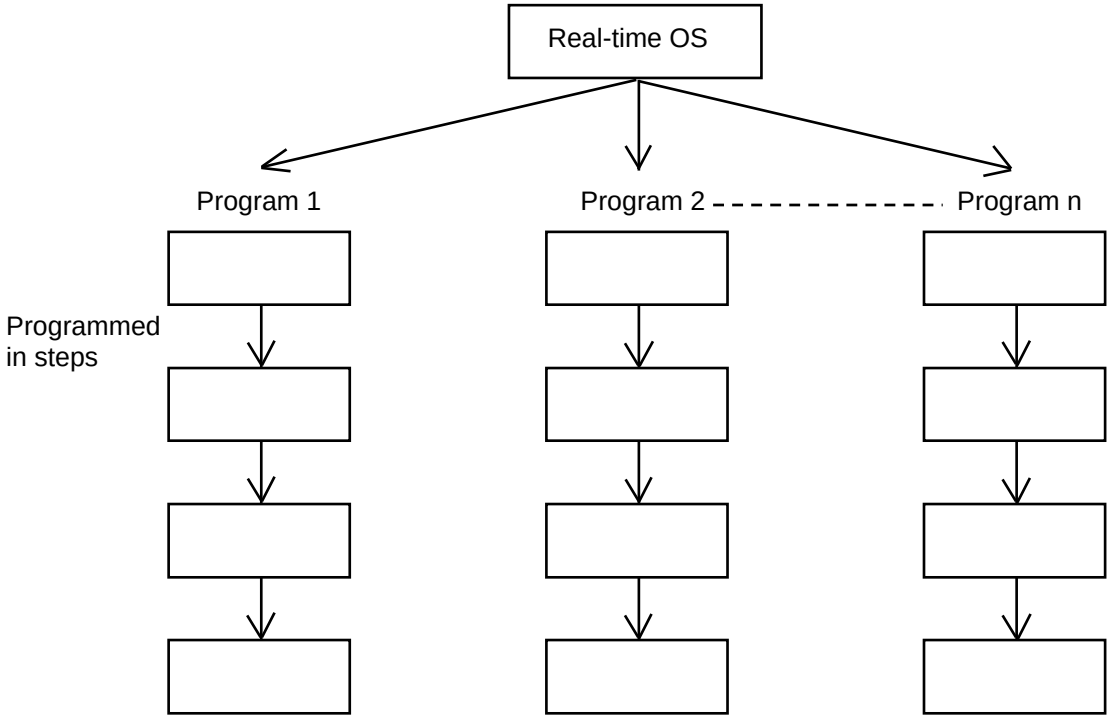
3. Difference from a Sequencer

The parallel processing method has evolved from the traditional method of using a sequence control circuit consisting of relays to a more recent one using a sequencer equipped with a microcomputer. Since a microcomputer basically allows one process for each clock, a sequence control circuit with a microcomputer must scan the entire program to achieve apparent parallel processing. For this reason, a scan time is required, which adds to overhead (dead time).

The microcomputer scans the enter program and outputs only where the condition is satisfied.



On the other hand, a system consisting of a microcomputer and a real-time operating system no longer uses parallel processing scan (by always scanning the entire program), but adopts an event-driven method instead (whereby the system operates only when an event occurs, such as upon receipt of an input signal). Since no extra scan is necessary, the system can operate at high speed. In addition, each program to be processed in parallel is programmed in steps, so the program is easy to understand and maintain.



The programmer need not worry about running all programs in parallel, which is controlled by the real-time operating system.

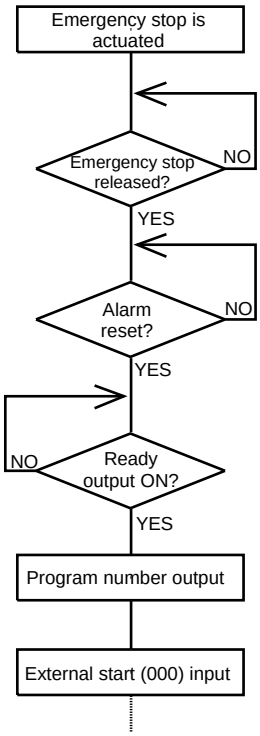
4. Release of Emergency Stop

Default factory settings of parameters

(“Other parameter No. 10, Emergency-stop recovery type” = 0
“Other parameter No. 11, Safety-gate open recovery type” = 0
“Other parameter No. 12, Recognition type during automatic operation” = 0)

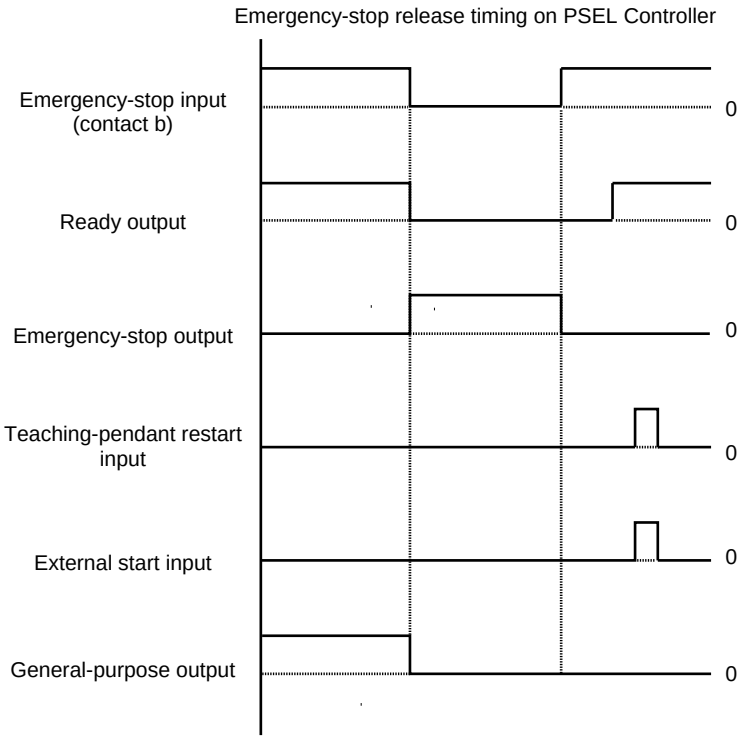
An emergency stop is actuated by turning the emergency-stop contact b input to OFF, and released by turning the input to ON.

[1] Flow chart



The selected program is executed from step 1.

[2] Timing chart



⊙ The internal conditions of the controller during an emergency stop are as follows:

- Programs ..... Aborted (excluding “I/O processing programs operation when program is aborted”)
- Output ports, local flags, local variables ..... Cleared
- Global flags, global variables ..... Retained

If the peripherals are to be controlled by program, create a management program beforehand and use the program to control the peripherals. Alternatively, start (EXPG) or abort (ABPG) other programs in accordance with the status of each general-purpose input.

5. Program Switching

Various methods are available to switch between programs, depending on the purpose of programs. The representative methods are explained below.

- Program switching
  - External start
  - Program
    - Single-tasking ..... EXIT command
    - Multi-tasking ..... EXPG command

First, the program switching methods are largely divided into switching by external start and switching by application program.

(1) External start method ..... Refer to Chapter 4, 2.2, “Standing via External Signal Selection” in Part 1.

(2) Program method

○ Single-tasking
Executing an EXIT command (end program) at the end of each program will end the program and cause the system to return to the condition immediately after the power is turned on. However, since the home position is retained, another program can be started by an external start input with the corresponding program number specified.

○ Multi-tasking
Creating a management program and executing EXPG commands (start other program) will allow a series of programs to be run in parallel.

Chapter 9 Example of Building a System

How to build hardware and software is explained in details by using a screw-tightening robot as an example.

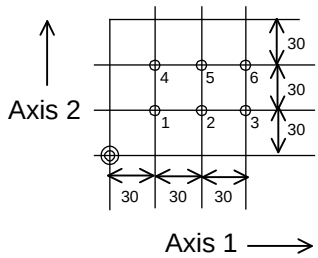
1. Equipment

Screw-tightening machine (for Z-axis)	
Actuators (for axes 1 and 2)	IAI's actuator with 300-mm stroke x 2
Controller	IAI's PSEL controller

2. Operation

(1) Tighten six screws at 30-mm pitches on axes 1 and 2.
- [1] The actuators move to a screw-tightening position.
- [2] The Z-axis air cylinder of the screw-tightening machine comes down.
- [3] The screw-tightening machine starts operating.
- [4] When the screw tightening is complete, the Z-axis air cylinder rises.
- [5] The actuators move to the next position.

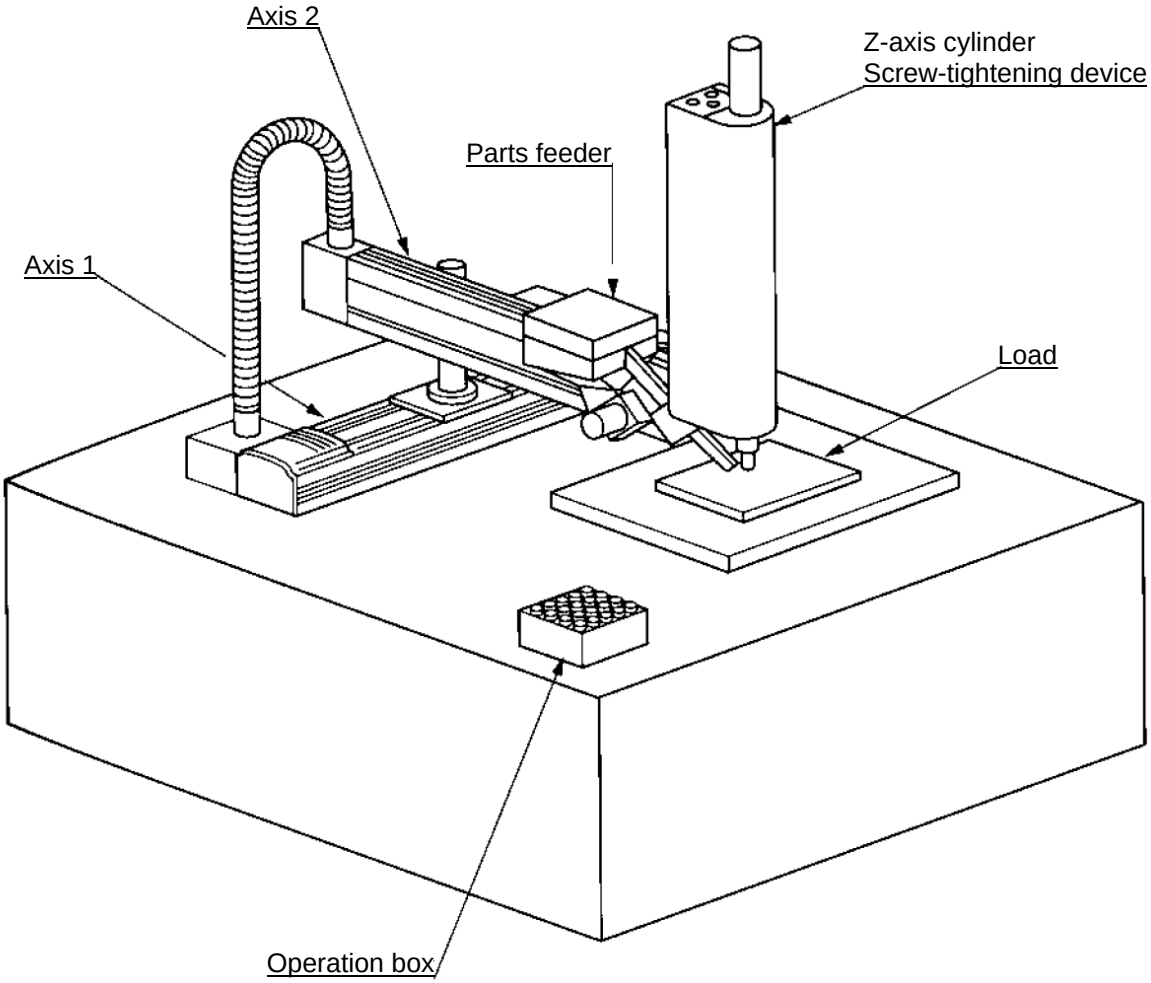
Coordinates



(2) The parts feeder operates in parallel with the above operation.
- [1] The parts feeder starts when screws are short.
- [2] The parts feeder stops when the screws are fully loaded.

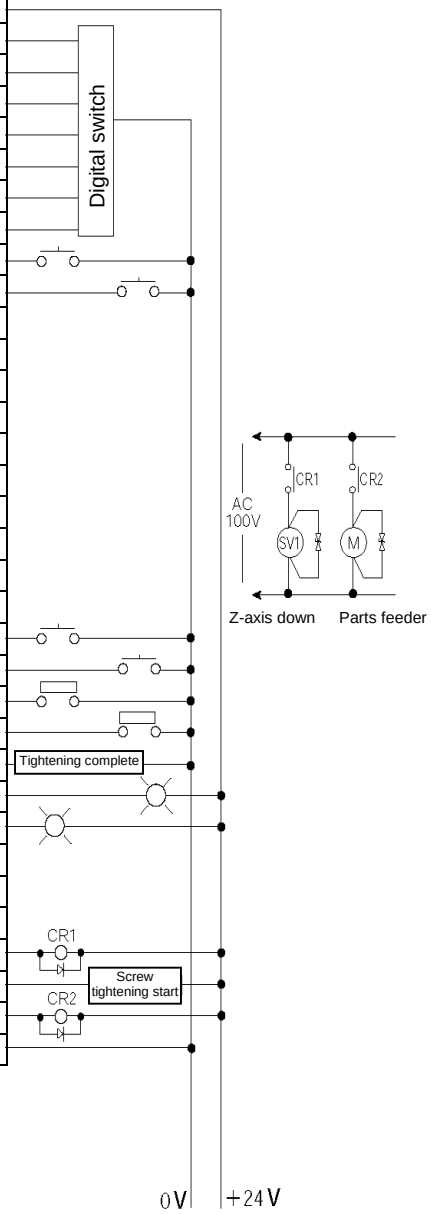
3. Overview of the Screw-Tightening System

This system consists of axis-1 and axis-2 actuators, Z-axis cylinder, screw-tightening device and parts feeder, and tightens the screws fed by the parts feeder at the specified positions on the load.



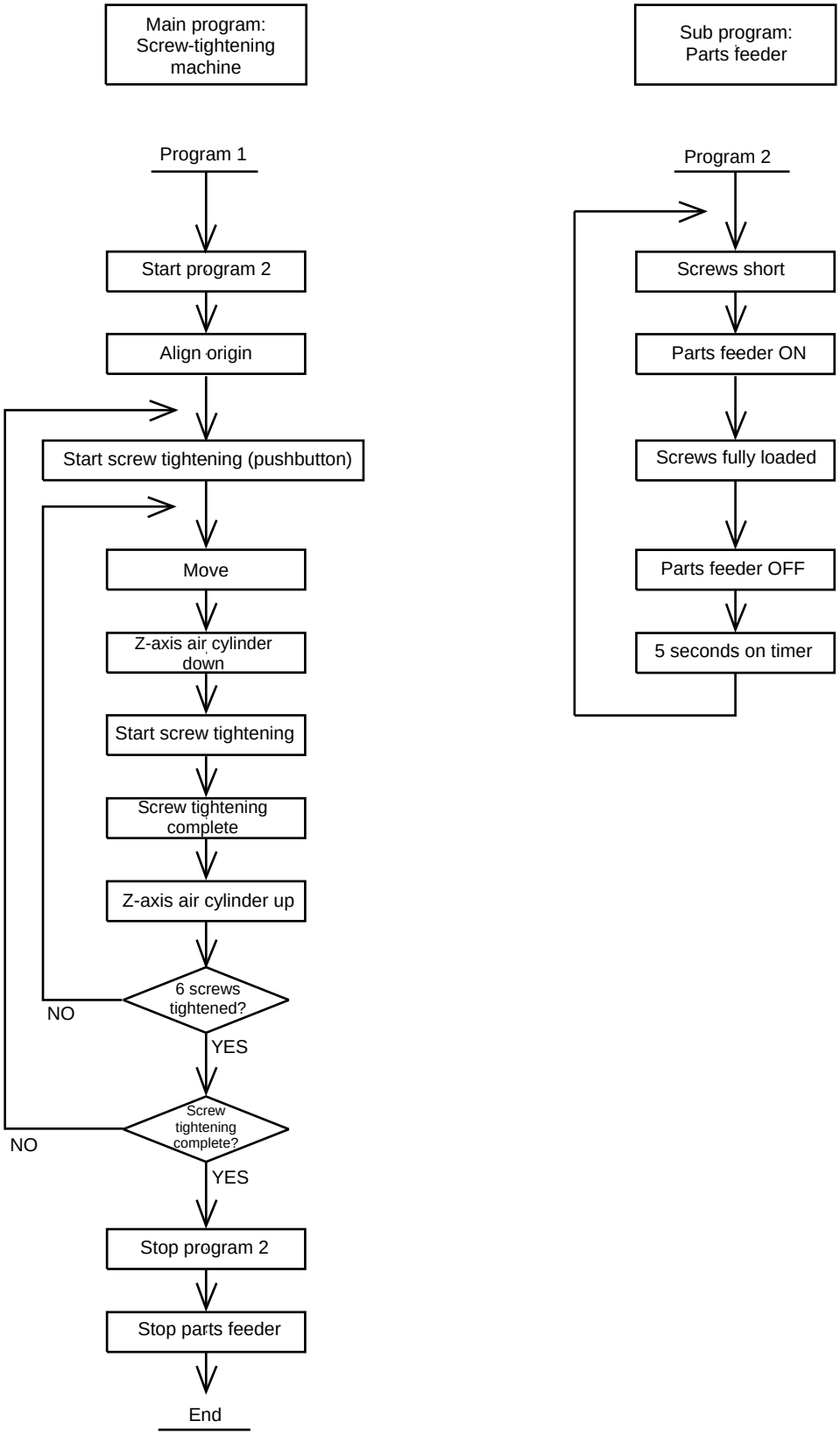
4. Hardware

Pin No.	Category	Port No.	Function
1A	P24		External power supply 24 V
1B	Input	016	Program specification (PRG No. 1)
2A		017	Program specification (PRG No. 2)
2B		018	Program specification (PRG No. 4)
3A		019	Program specification (PRG No. 8)
3B		020	Program specification (PRG No. 10)
4A		021	Program specification (PRG No. 20)
4B		022	Program specification (PRG No. 40)
5A		023	Software reset (restart)
5B		000	Program start
6A		001	General-purpose input
6B		002	General-purpose input
7A		003	General-purpose input
7B		004	General-purpose input
8A		005	General-purpose input
8B		006	General-purpose input
9A		007	General-purpose input
9B		008	General-purpose input
10A		009	General-purpose input
10B		010	General-purpose input
11A		011	General-purpose input (Screw tightening start)
11B		012	General-purpose input (Screw tightening end)
12A		013	General-purpose input (Z-axis air cylinder top)
12B		014	General-purpose input (Parts-feeder all screws tightened)
13A		015	General-purpose input (Screw tightening complete)
13B	Output	300	Alarm output
14A		301	Ready output
14B		302	General-purpose output
15A		303	General-purpose output
15B		304	General-purpose output
16A		305	General-purpose input (Z-axis air cylinder down)
16B		306	General-purpose input (Screw tightening start)
17A		307	General-purpose input (Parts feeder start)
17B	N		External power supply 0 V



5. Software

(1) Control flow chart



(2) Main program
Screw-tightening program No. 1

Application program

Comment	Extension condition	Input condition	Command			Output condition	Comment
	AND, OR	I/O, flag	Command	Operand 1	Operand 2	Output port, flag	
1			EXPG	2			Start program 2.
2			HOME	11			Align home.
3			VEL	100			Speed: 100 mm/sec
4			ACC	0.3			Acceleration: 0.3 G
5			TAG	1			Jump destination at restart
6			WTON	11			Screw-tightening start pushbutton
7			LET	1	1		Set screw counter.
8			TAG	2			Jump destination after tightening one screw
9			MOVL	*1			Move.
10			BTON	305			Z-axis air cylinder down
11			BTON	306			Start screw tightening.
12			WTON	12			Screw tightening complete.
13			BTOF	305	306		Cylinder up, screw tightening stopped.
14			WTON	13			Check Z-axis air cylinder top position.
15			ADD	1	1		Increment screw counter by 1.
16			CPEQ	1	7	900	Compare after tightening six screws.
17		N900	GOTO	2			Go to next screw-tightening cycle after tightening one screw.
18		N17	GOTO	1			Restart screw tightening.
19			ABPG	2			Stop program 2.
20			BTOF	307			Stop parts feeder.
21			EXIT				End of program 1

Position program

No.	X	Y
1	30	30
2	60	30
3	90	30
4	30	60
5	60	60
6	90	60

(3) Sub program
Parts feeder program No. 2

Application program

Comment	Extension condition	Input condition	Command			Output condition	Comment
	AND, OR	I/O, flag	Command	Operand 1	Operand 2	Output port, flag	
1			TAG	1			Jump destination for repeating
2			WTOF	14			Screws short.
3			BTON	307			Start parts feeder.
4			WTON	14			Screws fully loaded.
5			BTOF	307			Stop parts feeder.
6			TIMW	5			5 seconds on restart timer
7			GOTO	1			Repeat.

Chapter 10 Example of Building a System

1. Position Table

Position Table

Up to 1,500 position points can be registered in the PSEL controller.
Positions are registered using the PC software or teaching pendant.

(Example of 2-axis system)

No.	Axis1	Axis2	Vel	Acc	Dcl
1	50.000	50.000			
2	100.000	30.000			
3	125.000	96.000			
4	75.000	102.000			
5	200.000	110.000			
6	150.500	116.000			
	⋮	⋮	⋮	⋮	⋮

No.: Specify a number, and the actuator will move to the position registered for the specified number in the program.

Axis 1 to Axis 2: Enter the target position of each axis for each position number.

Vel: Set a speed. The speed set in this field takes precedence over the speed specified in the program. In other words, the actuator uses the speed specified here when moving to the position specified for the corresponding position number.

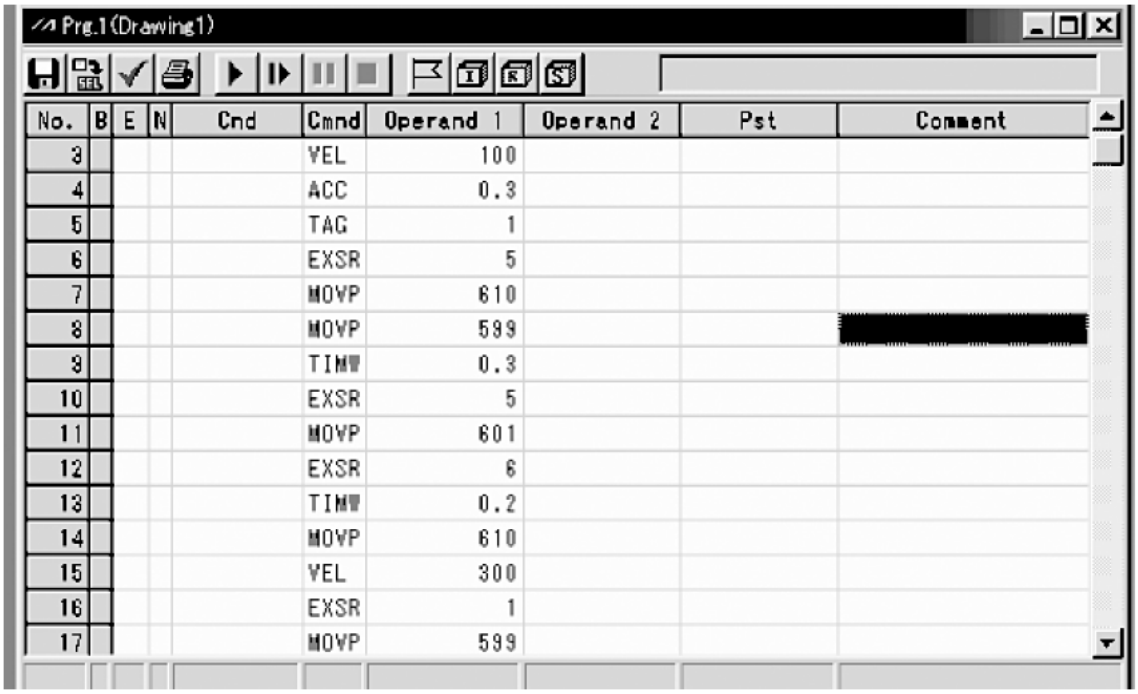
Acc: Set an acceleration. The acceleration set in this field takes precedence over the acceleration specified in the program or one set by the applicable parameter.

Dcl: Set a deceleration. The deceleration set in this field takes precedence over the deceleration specified by the program or one set by the applicable parameter.

2. Programming Format

Program Edit Screen (PC Software)

The PSEL controllers support programs consisting of up to 2,000 steps.
Programs are edited using the PC software or teaching pendant.



No.	B	E	N	Cnd	Cmnd	Operand 1	Operand 2	Pst	Comment
3					YEL	100			
4					ACC	0.3			
5					TAG	1			
6					EXSR	5			
7					MOVP	610			
8					MOVP	599			
9					TIMW	0.3			
10					EXSR	5			
11					MOVP	601			
12					EXSR	6			
13					TIMW	0.2			
14					MOVP	610			
15					VEL	300			
16					EXSR	1			
17					MOVP	599			

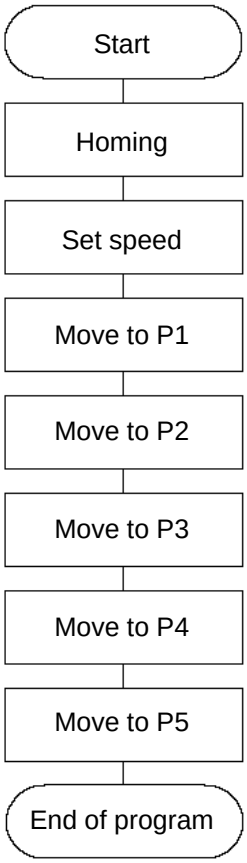
No.:	Step number
B:	Set a breakpoint (this field becomes editable during online edit). Click the “B” field in the line where you want to set a breakpoint. Once a breakpoint has been set, “B” is shown in the line. * Breakpoint --- A breakpoint is set in a step where you want to stop the program temporarily while the program is run from the PC software.
E:	Enter a desired extension condition (A, O, LD, AB or OB).
N:	Specify “N” to indicate negation of the input condition.
Cnd:	Enter an input condition.
Cmnd:	Enter a SEL command.
Operand 1:	Enter operand 1.
Operand 2:	Enter operand 2.
Pst:	Enter an output (operand 3).
Comment:	Enter a comment, if necessary (using up to 18 single-byte characters).

3. Positioning to Five Positions

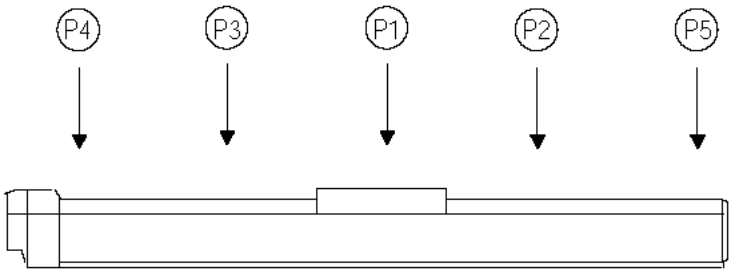
Description

Move the actuator to positions 1 through 5 at a speed of 100 mm/sec after homing.
Use of only 1 axis is assumed.

Flowchart



- Homing must be performed and a speed must be set, before the actuator can be operated.
- The actuator moves to the position data coordinates specified by the respective move commands.
- With the absolute specification, homing (HOME command) is not required.



Application program

No.	B	E	N	Cnd	Cmnd	Operand 1	Operand 2	Pst	Comment
1					HOME	1			Home Axis 1
2					VEL	100			Set velocity- mm/s
3					MOVL	1			Move to point 1
4					MOVL	2			Move to point 2
5					MOVL	3			Move to point 3
6					MOVL	4			Move to point 4
7					MOVL	5			Move to point 5
8					EXIT				End Program

Position data

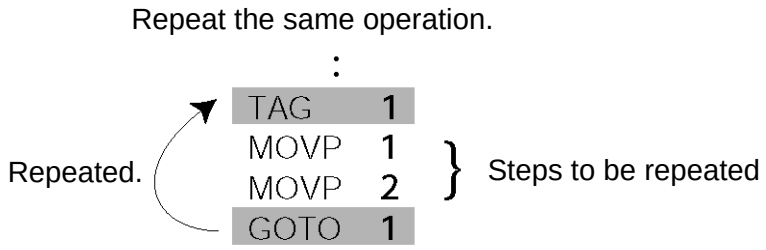
No.	Axis1
1	100.000
2	150.000
3	50.000
4	0.000
5	200.000
6	
7	
8	
9	

4. How to Use TAG and GOTO

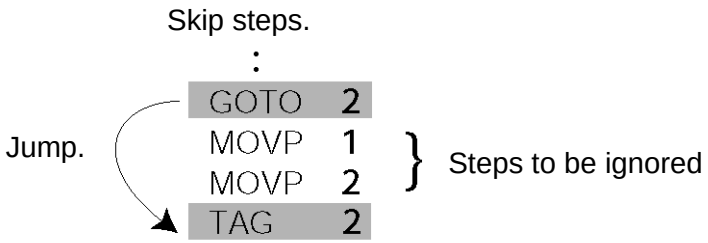
Description

Use GOTO and TAG commands to repeat the same operation within the program or to jump to a desired step if a condition is satisfied. A TAG command can be written in a step either before or after a GOTO command.

Example of Use 1



Example of Use 2

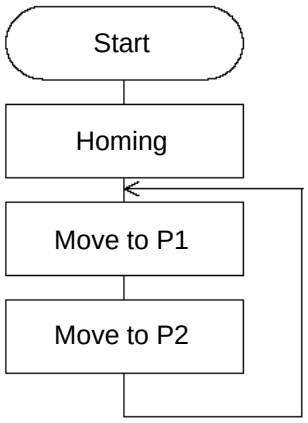


5. Moving Back and Forth between Two Points

Description

Moves back and forth between two points.

Flowchart



- The actuator moves back and forth between P1 and P2 indefinitely.
- Use of only 1 axis is assumed.
- Enter TAG in the first of the steps to be repeated, and enter GOTO in the last of the steps to be repeated.

Application program

No.	B	E	N	Cnd	Cmnd	Operand 1	Operand 2	Pst	Comment
1					HOME	1			Home Axis 1
2					VEL	100			Set velocity- mm/s
3					TAG	1			Set loop marker 1
4					MOVL	1			Move to point 1
5					MOVL	2			Move to point 2
6					GOTO	1			Loop to TAG 1
7									

Position data

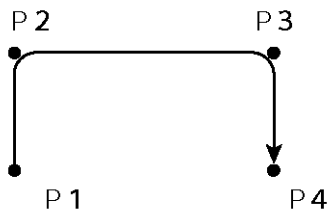
No.	Axis1
1	100.000
2	150.000
3	
4	
5	
6	

6. Path Operation

Description

Move continuously through four arbitrary points without stopping (PATH movement).

The actuator moves along the path shown at right, without stopping at P2 and P3.
Compared with MOVP and MOVL, this command does not require the actuator to position exactly at P2 and P3, and thus the movement tact time can be reduced.
Assume the following command is executed when the actuator is stopped at P1:
PATH 2 4
The actuator will move from P1 to P4 by passing points near P2 and P3. (The passing points can be brought closer to the specified positions by increasing the acceleration.)

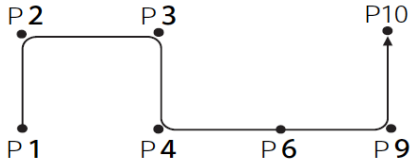


Even if “PATH 2 3” and “PATH 3 4” are input successively, the actuator will still move in the same way as when “PATH 2 4” is input.

If “PATH 4 1” is executed while the actuator is stopped at P4, the actuator will move along the same path in the opposite direction (P4 → P3 → P2 → P1).

Continuous operation is possible even when all positions are not continuous.

PATH	1	4	
PATH	6	6	(Discontinuous position)
PATH	9	10	



As shown above, specify the position number of the discontinuous position, or No. 6 in this case, for both the start position number and end position number in the PATH command. The actuator will move continuously in the sequence of position Nos. P1 → P2 → P3 → P4 → P6 → P9 → P10.

7. Output Control during Path Movement

Description

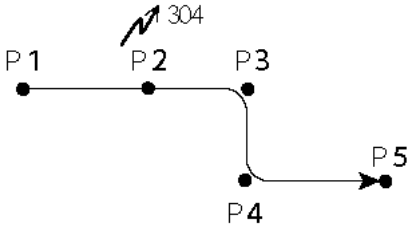
In spray operation, etc., output control may be required while the actuator is moving. The PSEL controller can output signals while the actuator is moving with a PATH command.

How to Use

Before executing a PATH command, declare a POTP command to specify signal output during movement. If a given output or global flag is specified in the output field of the PATH command, the output or flag specified in the output field will turn ON as the actuator approaches, via path movement, the position specified in the PATH command.

Example of Use 1

The actuator moves from P1 to P5 along the positions shown at right, without stopping. As the actuator approaches P2, output port 304 turns ON.



Cmnd	Operand 1	Operand 2	Pst
VEL	100		
POTP	1		
PATH	1	1	
PATH	2	2	304
PATH	3	5	

← A declaration command to specify signal output during path movement.

← 304 turns ON when the actuator approaches P2 specified in this step.

Outputs and flags can only be turned ON. The output or flag that was turned ON during path operation must be turned OFF (using a BTOF command) after the operation is completed.

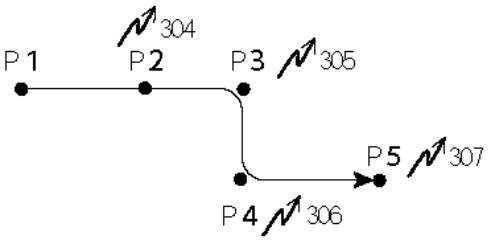
Example of Use 2

Outputs 304 to 307 can be turned ON sequentially at the respective points of P2 to P5.

Cmnd	Operand 1	Operand 2	Pst
VEL	100		
POTP	1		
PATH	1	1	
PATH	2	5	304

← A declaration command to specify signal output during path movement.

← 304 to 307 turn ON sequentially at P2 to P5 specified in this step.



8. Circle/Arc Operation

Description

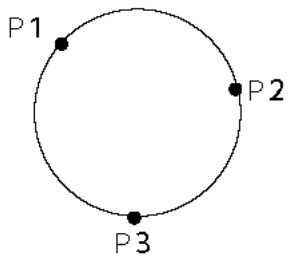
The actuator moves along a two-dimensional circle or arc.

How to Use

To specify a circle, specify three points the actuator will pass. To specify an arc, specify the starting point, passing point and end point.

Example of Use 1

Circle



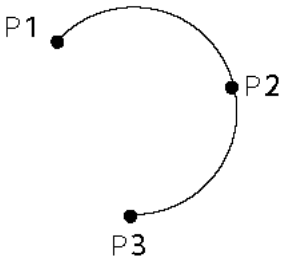
- Specify “CIR2 2 3” after the actuator has moved to P1.
- If “CIR2 2 3” is specified in the figure shown at left, the actuator will move along this circle clockwise.

E	N	Cnd	Cmnd	Operand 1	Operand 2	Pst
			VEL	100		
			MOVP	1		
			CIR2	2	3	

- To cause the actuator to move counterclockwise, specify “CIR2 3 2.”

Example of Use 2

Arc



- Specify “ARC2 2 3” after the actuator has moved to P1.

E	N	Cnd	Cmnd	Operand 1	Operand 2	Pst
			VEL	100		
			MOVP	1		
			ARC2	2	3	

9. Home Return Completion Output

Description

Output a signal to confirm completion of homing (incremental specification).
With the PSEL controller, a home return completion signal can be output using an I/O parameter.
However, the following explains how to output a home return completion signal within a program using a general-purpose output.
Once turned ON, a general-purpose output will remain ON even after the current program ends or other program is started. (It will turn OFF upon emergency stop, etc., but the ON status can be maintained using an I/O parameter (I/O parameter Nos. 70 and 71).)

Example of Use

a. Output a home return completion signal.

E	N	Cnd	Cmnd	Operand 1	Operand 2	Pst
			HOME	11		
			BTON	303		

Execute homing.
General-purpose output (arbitrary)

b. Use a home return completion signal to make sure the actuator will not perform homing if it has already been performed.

E	N	Cnd	Cmnd	Operand 1	Operand 2	Pst
	N	303	HOME	11		
			BTON	303		

Execute homing if output 303 is OFF.
Home return completion output

c. Use the output field instead of a BTON command.

E	N	Cnd	Cmnd	Operand 1	Operand 2	Pst
	N	303	HOME	11		303

Execute the same processing performed with the above two steps.

Reference

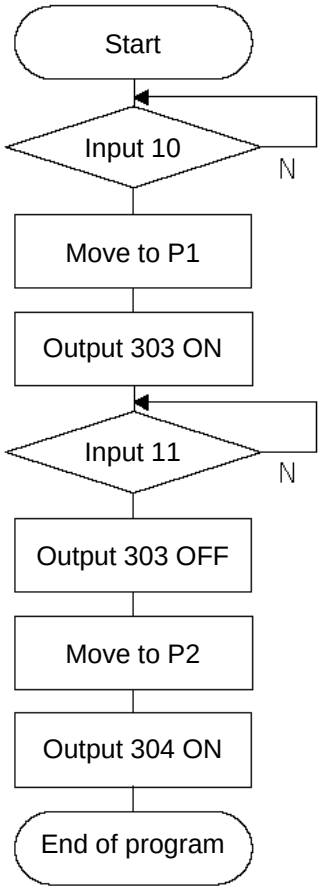
Output port No. 304 can be used as a home return completion output (dedicated output) by setting I/O parameter No. 50 to “13.”

10. Axis Movement by Input Waiting and Completion Output

Description

How to perform input waiting and output a processing completion signal is explained.

Flowchart



Example of Use

The actuator waits until input port 10 turns ON, and then moves to P1.
The actuator waits until input port 11 turns ON, and then moves to P2.
A movement completion signal is output from 303 upon reaching P1, and from 304 upon reaching P2.

Application program

E	N	Cnd	Cmnd	Operand 1	Operand 2	Pst	Comment
			VEL	100			Set velocity- mm/s
			WTON	10			Wait on input 10
			MOVP	1			Move to point 1
			BTON	310			Turn ON output 310
			WTON	11			Wait on input 11
			BTOF	310			Turn OFF outpt 310
			MOVP	2			Move to point 2
			BTON	311			Turn ON output 310
			EXIT				End Program

11. Changing the Moving Speed

Description

Change the moving speed.

How to Use

With the PSEL controller, the speed can be set using the following two methods:
a: Use a VEL command within the application program
b: Use a speed setting in the position data table

Example of Use

Application program

E	N	Cnd	Cmnd	Operand 1	Operand 2	Pst
			MOVP	1		
			VEL	1000		
			MOVP	2		
			MOVP	3		
			VEL	50		
			MOVP	4		

Position data

No.	Axis1	Vel	Acc	Dcl
1	100.000	100		
2	200.000	500		
3	300.000			
4	400.000			

Moving speeds in the above program

Position at 100 mm --- The actuator moves at 100 mm/sec.
Position at 200 mm --- The actuator moves at 500 mm/sec.
Position at 300 mm --- The actuator moves at 1000 mm/sec.
Position at 400 mm --- The actuator moves at 50 mm/sec.

If a speed is specified in the position data table, this speed takes precedence over the speed specified in the application program, as shown above. In general, speeds are set in the application program using VEL.

Vel in Point Data Table and PATH Command

The speed can be changed without stopping the actuator, by using a PATH command and Vel in the position data table. (Refer to the next page.)

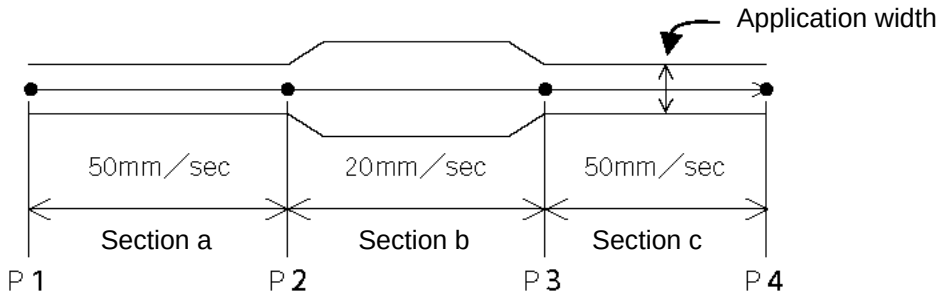
12. Changing the Speed during Operation

Description

Use a PATH command to change the speed while the actuator is moving.
For example, this command is useful in a paint dispensing application where the application volume changes in the middle.

Example of Use

The actuator moves through linear sections a, b and c at 50 mm/sec, 20 mm/sec and 50 mm/sec, respectively, without stopping (PATH movement).



Position data

No.	Axis1	Vel	Acc	Dcl
1	0.000	50		
2	100.000	50		
3	200.000	20		
4	300.000	50		

Application program

“PATH 1 4” is the only movement command required.

E	N	Cnd	Cmnd	Operand 1	Operand 2	Pst
			PATH	1	4	

Reference

The speed can also be changed from other program using a CHVL (speed change) command (in the multi-tasking mode).

13. Local/Global Variables and Flags

Description

The internal variables and flags used in the SEL language are classified into local and global types.
The data range used commonly by all programs is called the global range, while the data range used only by each program is called the local range. When multi-tasking programs are run simultaneously, the global range must be used to synchronize the programs and allow cross-referencing of variables among the programs.

Example of Use

Program handshake

Program A

Cmnd	Operand 1
MOVL	1
BTON	600
WTON	601
MOVL	3

Program B

Cmnd	Operand 1
WTON	600
MOVL	2
BTON	601

Use of global flags with the above two programs permits handshake between the programs, and the actuator moves per “MOVL 1” in program A, moves per “MOVL 2” in program B, and then move per “MOVL 3” in program A, for example.

Backup in Battery

If the PSEL controller has a built-in battery (optional), variables and flags used in the programs are retained. For both variables and flags, only those in the global area will be retained after the controller power is cut off.
The variables and flags in the local range are cleared when the program is started (the variables are reset to “0,” while the flags turn OFF).

14. How to Use Subroutines

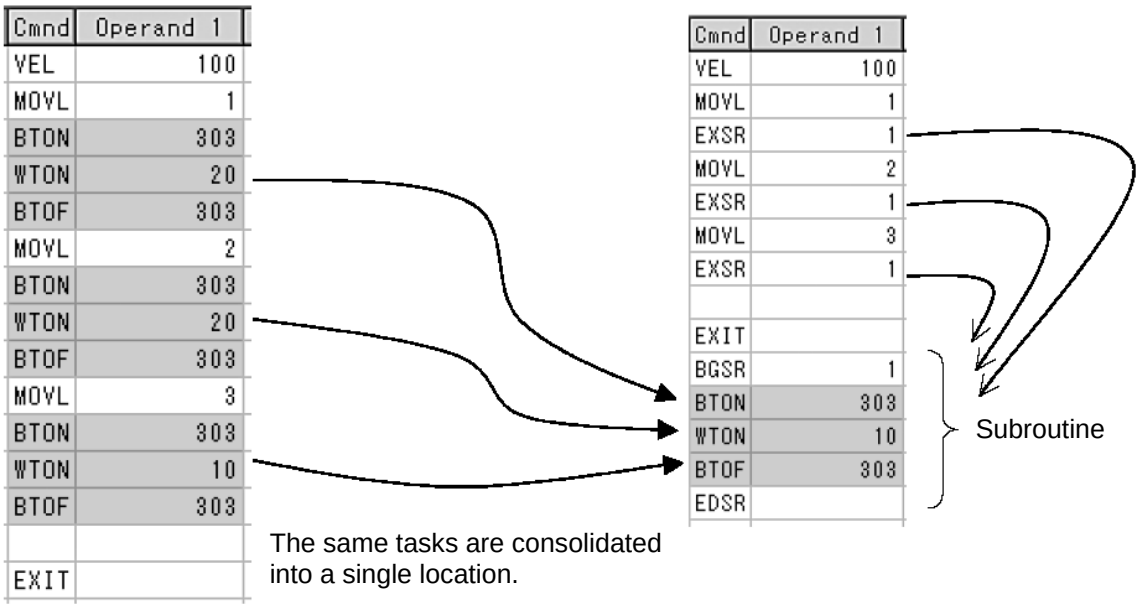
Description

A subroutine is a group of steps that are called and executed several times within a program. Subroutines are used to reduce the number of program steps and make the program easy to read. Up to 99 subroutines can be used in one program. Up to 15 subroutine calls can be nested.

How to Use

Declare/call subroutines using the following commands:
EXSR: Call a subroutine
BGSR: Declare the start of a subroutine (start of a group of steps)
EDSR: Declare the end of a subroutine (end of a group of steps)

Example of Use



Caution

Jumping from within a subroutine to a TAG position outside the subroutine using a GOTO command is prohibited.

15. Pausing the Operation

Description

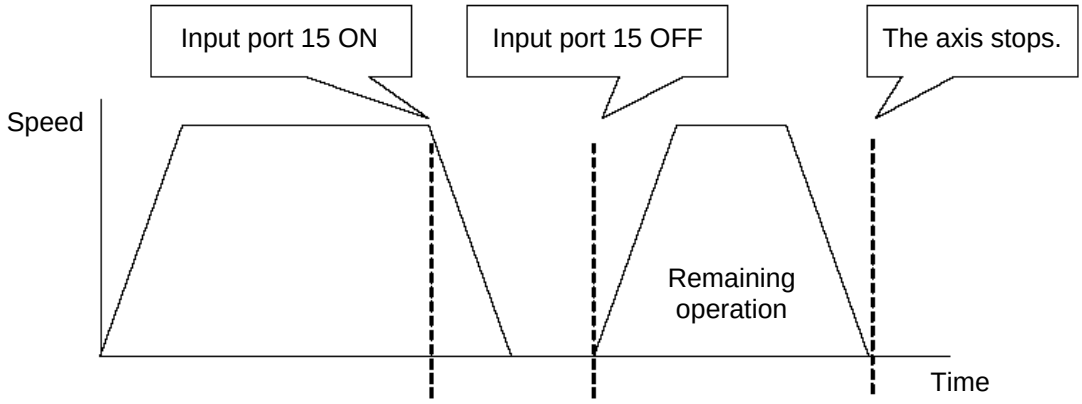
Use a declaration command HOLD to pause the moving axis temporarily via external input.

How to Use

A pause interruption operation can be executed to a moving axis (to decelerate the axis to a stop) by declaring a HOLD command within the program.
While HOLD is input, the actuator pauses (decelerates to a stop, if currently moving) against all moving commands in the same program.

Example of Use

HOLD 15 A declaration to execute pause if general-purpose input 15 turns ON.



Application

You can specify a global flag, instead of an input port, in Operand 1 of the HOLD command.
Use of a global flag allows the actuator to be paused from other program.
The input signal pattern and stop action can be selected using Operand 2.

0 = Contact a (Decelerates to a stop) ⇒ Same as when Operand 2 is not specified.
1 = Contact b (Decelerates to a stop)
2 = Contact b (Decelerates to a stop, and then servo OFF ⇒ The drive power is not cut off.)

E	N	Cnd	Cmnd	Operand 1	Operand 2	Pst	Comment
			HOLD	20	2		SVOF when input 20

Caution

If the actuator is paused during homing, it will start the homing sequence from the beginning upon restart.

16. Canceling the Operation 1 (CANC)

Description

Use a declaration command CANC to decelerate the moving axis to a stop and cancel the remaining operation.

How to Use

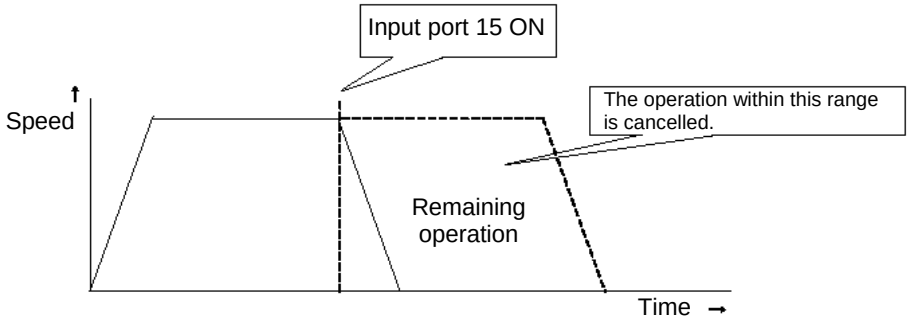
While CAN is input, all movement commands in the same program are cancelled.

Example of Use

CANC command

```
CANC   15        Cancel the movement commands if input port 15 turns ON (declaration).
  :
MOVP    1
MOVP    2
  :
WTON   14
  :
```

* Declare this command in a step before the movement commands you want to cancel.
* While CANC is input, all operation commands are cancelled sequentially, while tasks other than operation commands (such as I/O processing and calculation processing) are executed sequentially.



Caution

Since execution of this command makes it no longer possible to specify which program step is currently executed, it is recommended that a WTON command be used to create an input wait step.

Application

A desired input signal pattern can be selected for a CANC command using Operand 2.

0 = Contact a (Decelerates to a stop) ⇒ Same as when Operand 2 is not specified.
1 = Contact b (Decelerates to a stop)

E	N	Cnd	Cmnd	Operand 1	Operand 2	Pst	Comment
			CANC	20	1		Halt when input 20

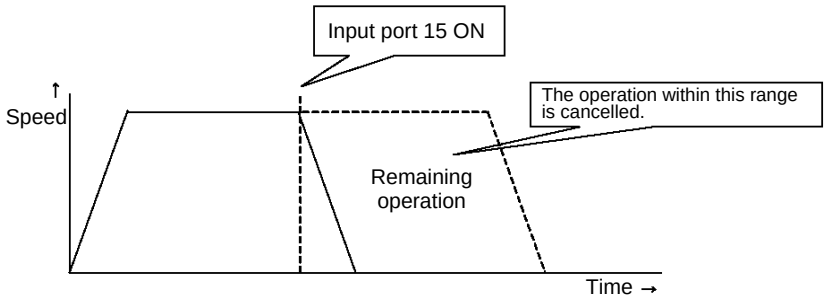
17. Canceling the Operation 2 (STOP)

Description

Decelerate the moving axis to a stop and cancel the remaining operation. (STOP)

How to Use

Execute a STOP command from other program to forcibly stop the operation (in the multi-tasking mode). Specify the axis you want to stop using an axis pattern.



Example of Use 1

STOP command

Main program

```
EXPG    n
  :
MOVL    1
MOVL    2
  :
```

The stop program starts.

Stop control program

```
WTON    15    Wait for stop input.
STOP    11    Axes 1 and 2 stop.
```

If “STOP 11” is executed while “MOVL 1” is being executed, “MOVL 1” will be cancelled and the actuator will continue its operation from “MOVL 2.”

Example of Use 2

Main program

```
EXPG    n
  :
MOVP    1
MOVP    2
  :
```

The stop program starts.

Stop control program

```
WTON    15    Wait for stop input.
STOP    10    Axis 2 stops.
```

If “STOP 10” is executed while “MOVL 1” is being executed, only the axis 2 part of “MOVL 1” will be cancelled. Both axes 1 and 2 will operate under “MOVL 2.”

Caution

If a STOP command is executed during a CP operation (interpolation operation) initiated by MOVL, etc., the operations of all axes will be cancelled regardless of the axis pattern specified in the STOP command.

18. Movement by Position Number Specification

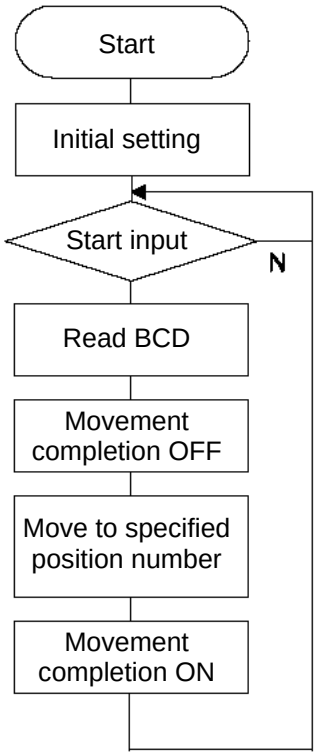
Description

Load externally input BCD codes as position numbers to execute movements.

Example of Use

Use an INB command to load a position number as a BCD code from an input port.
A position number can be specified using a value consisting of up to three digits.

Flowchart



Input assignment

Port	Description
1	Start input
2	Position specification 1
3	Position specification 2
4	Position specification 4
5	Position specification 8
6	Position specification 10
7	Position specification 20
8	Position specification 40
9	Position specification 80
10	Position specification 100
11	Position specification 200
12	Position specification 400
13	Position specification 800

Output

303	Movement completion

Application program

E	N	Cnd	Cmnd	Operand 1	Operand 2	Pst	Comment
			HOME	11			Home axis 1 & 2
			VEL	100			Set velocity- mm/s
			TAG	1			Set loop marker 1
			WTON	1			Wait on start inpt
			INB	15	3		Read position #
			BTOF	303			Mov cmplt sgnl OFF
			MOVL	*99			Move to position
			BTON	303			Move cmplt sgnl ON
			GOTO	1			Jump to marker 1 ^

19. Movement by External Position Data Input

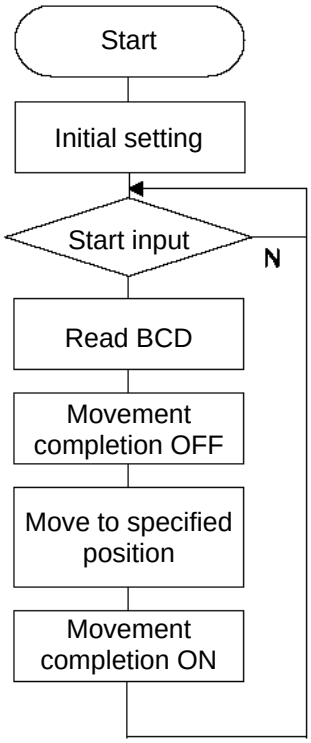
Description

Receive target position data as absolute values from a host device to execute movements.

Example of Use

Use an INB command to load position data as a BCD code from an input port.
Each BCD value should consist of four digits, with the last digit indicating a decimal place.
The moving axis is axis 1.
Example: If a BCD of “1234” is received, the axis will move to the position at 123.4 mm.
Note: When using input port Nos. 16 and 17, do so after changing them to general-purpose inputs.

Flowchart



Input assignment

Port	Description
1	Start input
2	0.1mm
3	0.2mm
4	0.4mm
5	0.8mm
6	1mm
7	2mm
8	4mm
9	8mm
10	10mm
11	20mm
12	40mm
13	80mm
14	100mm
15	200mm
16	400mm
17	800mm

Output

303	Movement completion

Application program

E	N	Cnd	Cmnd	Operand 1	Operand 2	Pst	Comment
			HOME	11			Home axis 1 & 2
			VEL	100			Set velocity- mm/s
			TAG	1			Set loop marker 1
			WTON	1			Wait on start inpt
			INB	15	4		Read position #
			LET	199	*99		Indirect refernce
			DIV	199	10		Div by 10 resolutn
			PPUT	1	1000		Put 1K for axis 1
			BTOF	303			Mov cmplt sigl OFF
			MOVL	1000			Move to entry pos.
			BTON	303			Mov cmplt signl ON
			GOTO	1			Jump to marker 1 ^

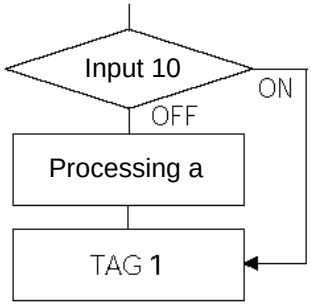
20. Conditional Jump

Description

Select the destination to jump to via GOTO using the external input, output and/or internal flag statuses as a condition.
The controller waits for multiple inputs, and performs processing according to the received input(s).

Example of Use 1

If input 10 turns ON, the actuator will jump to TAG 1. If it turns OFF, the actuator will proceed to the next processing.



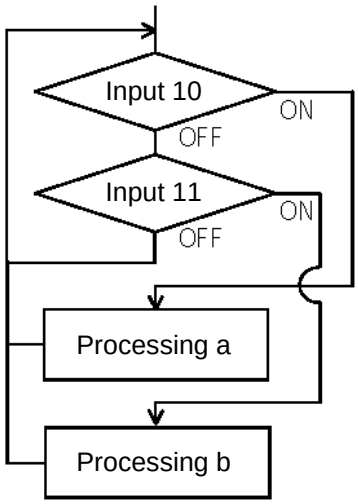
E	N	Cnd	Cmnd	Operand 1
		10	GOTO	1
Processing a				
			TAG	1
Processing b				

Execute GOTO 1 if input 10 turns ON.

* If input 10 turns ON, the actuator will skip processing a and perform processing b.
If input 10 turns OFF, the actuator will perform processing a, and then perform processing b.

Example of Use 2

The controller waits for an input signal to be received at input port 10 or 11. If an input signal is received at input 10, the actuator will perform processing a. If an input signal is received at input 11, it will perform processing b.



E	N	Cnd	Cmnd	Operand 1
			TAG	1
		10	GOTO	2
		11	GOTO	3
			GOTO	1
			TAG	2
Processing a				
			GOTO	1
			TAG	3
Processing b				
			GOTO	1

No input.
Input 10 turns ON.
Input 11 turns ON.

If both inputs 10 and 11 turn ON, the actuator will perform processing a.

21. Waiting Multiple Inputs

Description

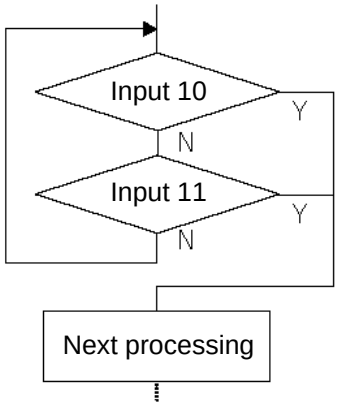
The controller waits for multiple different inputs and performs processing upon reception of any of these inputs.

Point

A WTON command permits processing only when the specified input is received. The controller cannot wait for multiple inputs.

Example of Use

Inputs 10 and 11 are monitored, and the actuator will proceed to the next step when either input is received (OR logic).



Program a

E	N	Cnd	Cmnd	Operand 1
			TAG	1
		10		
O		11	GOTO	2
			GOTO	1
			TAG	2

Next processing

Program b

E	N	Cnd	Cmnd	Operand 1
			TAG	1
	N	10		
A	N	11	GOTO	1

Next processing

* Both programs a and b perform the same processing.

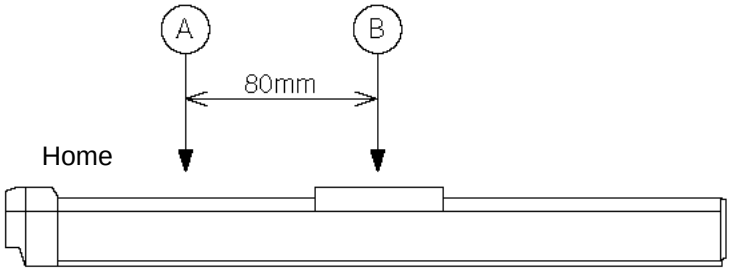
As shown in the sample, the controller waits for input without using a WTON command.
This method can also be used when multiple input conditions must be combined.

22. How to Use Offset

Description

With an OFST command, an offset can be specified for position data when you want to shift (offset) all teaching points by several millimeters because the actuator was not installed exactly in the specified position or for other reasons.
An OFST command can also be used to perform pitch feed. (Refer to 24, “Constant-pitch Feed.”)

E	N	Cnd	Cmnd	Operand 1	Operand 2	Pst	Comment
			VEL	100			Set velocity- mm/s
			MOVP	1			Move to point 1
			OFST	1	80		Offset axis 1 80mm
			MOVP	1			Move to point 1



Caution

Once an offset has been set, the offset applies to all movement commands executed thereafter. To cancel the offset, execute an offset command again by specifying “0” mm. An offset does not apply to other programs (even in the multi-tasking mode). If a given offset must be applied to all programs, it must be set for all programs individually.

23. Executing an Operation N times

Description

Execute a specific operation n times.

Example of Use

The actuator moves back and forth between P1 and P2 ten times, and then the program ends.
Use a CPEQ command to compare the number of times the movement has been actually repeated, against 10.
It is assumed that homing has been completed.

Application program

E	N	Cnd	Cmnd	Operand 1	Operand 2	Pst	Comment
			VEL	100			Set velocity- mm/s
			LET	1	0		Initlz counter 1
			TAG	1			Set loop marker 1
			MOVP	1			Move to point 1
			MOVP	2			Move to point 2
			ADD	1	1		Incrmt cntr by 1
			CPEQ	1	10	900	Repeat 10 times
	N	900	GOTO	1			Loop if not done
			EXIT				Else end program

Reference

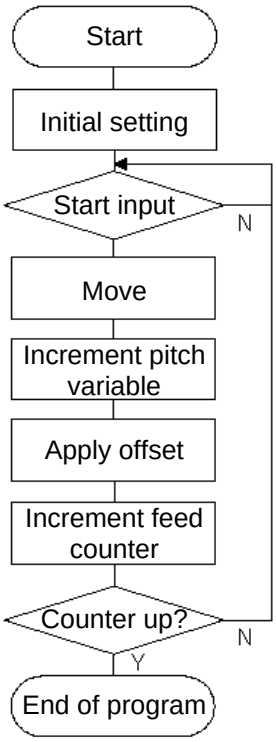
The same operation can also be performed using a DWEQ command.

24. Constant-pitch Feed

Description

Feed the actuator by a specified pitch n times from a reference point.
The pitch and number of repetitions are specified by variables in advance.

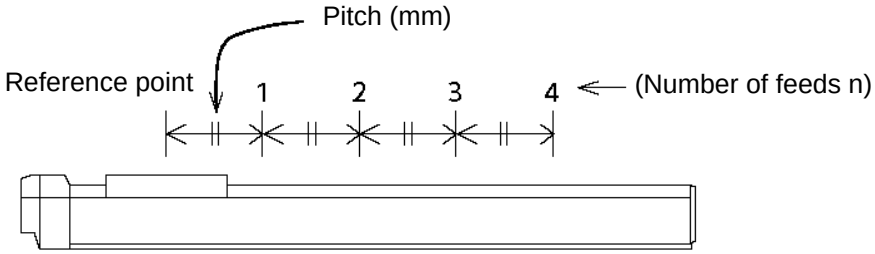
Flowchart



Example of Use

Use an OFST command to perform pitch feed.
The number of times the actuator has been fed is counted by a counter variable.
The X-axis is fed in the positive direction.

Point
An OFST command applies to movement commands.
Executing an OFST command alone does not move the axis.



Application program

E	N	Cnd	Cmnd	Operand 1	Operand 2	Pst	Comment
			LET	1	4		Variable feed #= 4
			LET	100	80		Var. pitch = 80mm
			LET	2	0		Clear counter 2
			LET	101	0		Initializ var. 101
			HOME	1			Home axis 1
			VEL	100			Set velocity- mm/s
			TAG	1			Set loop marker 1
			WTON	1			Wait on start inpt
			MOVP	1			Move to point 1
			ADD	101	*100		Add pitch to offst
			OFST	1	*101		Process x offset
			ADD	2	1		Add 1 to counter 2
			CPGT	2	*1	900	Confirm feed cmplt
	N	900	GOTO	1			Repeat if needed
			EXIT				End Program

Reference

Pitch feed can also be performed using a MVPI or MVLI command.

25. Jogging

Description

The slider moves forward or backward while an input is ON or OFF.
Instead of an input, an output or global flag can be used as a cue.
The slider will move directly to the next step if the specified input does not satisfy the condition when the command is executed.
Regardless of the input status, the slider will stop upon reaching the soft limit, and the command in the next step will be executed.

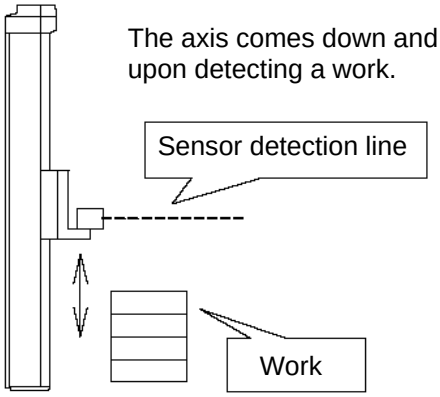
How to Use

- Explanation of commands

JFWN	1	1	Axis 1 moves forward while input 1 is ON.
JFWF	1	2	Axis 1 moves forward while input 2 is OFF.
JBWN	10	3	Axis 2 moves backward while input 3 is ON.
JBWF	10	4	Axis 2 moves backward while input 4 is OFF.

Example of Use 1

- Stop the axis when a sensor input is received.




```
    :
VEL   50        Specify a low speed.
JFWF   1   10   Move until a sensor input (10) is received.
EXIT            The program ends.
```

Example of Use 2

- Cause the actuator to jog just like in teaching pendant operation (2 axes are operated).

<u>Application program</u>

E	N	Cnd	Cmnd	Operand 1	Operand 2	Pst
			TAG	1		
			JFWN	1	1	
			JBWN	1	2	
			JFWN	10	3	
			JBWN	10	4	
	N	24	GOTO	1		
			EXIT			

Reference

HOLD, STOP and CANC commands remain valid while the actuators are jogging.

26. Switching Programs

Description

Use EXPG/ABPG commands to switch programs using a program.

Example of Use 1

Start program 2 once the processing of program 1 is completed, and then end program 1.

Program 1	Program 2
:	:
EXPG 2	:
EXIT	

Example of Use 2

Start a program via an external signal, and then end the other program.

Program 1	Program 2
ABPG 2	ABPG 1
:	:

If program 2 is started while program 1 is running, program 1 will be aborted.
If program 1 is started while program 2 is running, program 2 will be aborted.

Application

If a program number is specified in operand 2, the programs from the one corresponding to the program number in operand 1 to the other corresponding to the program number in operand 2 can be started (EXPG) or ended (ABPG) simultaneously.

Caution

- The PSEL controller supports multi-tasking. Up to 8 programs can be run at the same time. To use other programs when the controller is already running 8 programs, switch programs by closing a program or programs that are not required.
- If an ABPG command was executed to end a program while the program was executing a movement command, the actuator immediately decelerates to a stop.

27. Aborting a Program

Description

Abort a program currently running.
Execute an ABPG command (command to abort other program) from other program in the multi-tasking mode.

Caution

* If the target program was executing a movement command, the actuator immediately decelerates to a stop and the program ends.

Example of Use

Main program (Prg. 1)

```
EXPG    n     The abort control program starts.
WTON    10
MOVP    1
BTON    303
   :
   :
```

Abort control program (Prg. n)

```
WTON    10    Wait for an abort input.
ABPG    1     Prg. 1 is aborted.
EXIT          The program ends.
```

* If ABPG was executed while the actuator was moving via a MOVP command, the actuator immediately decelerates to a stop and the program ends.

Part 3 Positioner Mode

In the positioner mode, position data is input in the MANU mode and positioning operation based on input data is performed in the AUTO mode (the controller modes are switched using the AUTO/MANU switch). If the controller mode is changed to MANU while positioning is performed in the AUTO mode, the controller will maintain the servo ON or OFF status that was effective prior to the mode change. The output conditions of ready/alarm status and absolute-data/system battery error status will be retained. All other outputs will be turned OFF.
When the controller is returned to the AUTO mode in this condition, the outputs will also return to their original conditions.

Chapter 1 Modes and Signal Assignments

The positioner mode provides five sub-modes associated with different PIO (parallel I/O) patterns. Select a mode appropriate for your specific purpose.
To select a desired mode, set a number between 1 and 4 or 16 in other parameter No. 25, “Operation mode type.”

1. Feature of Each Mode

Value set in parameter No. 25	Feature of each mode
1	Standard mode Positioning to up to 1,500 positions can be performed. Push-motion operation is also supported.
2	Product switching mode Product numbers can be set in addition to position numbers. A position number can be changed for each product under the same position number. Push-motion operation is also supported.
3	2-axis independent mode Operations of two axes (start/stop) can be controlled separately.
4	Teaching mode Positions to be registered can be taught externally.
16	DS-S-C1 compatible mode This mode reflects the operation of the DS-S-C1 controller by adopting compatible pin assignments. Replacement without any modification is possible.

2. Number of Positions Supported in Each Mode

Mode	Number of positions
Standard mode	Maximum 1,500 positions
Product switching mode	Total 1,500 positions for all products (The same number of position data sets is used for each product.)
2-axis independent mode	13 input bits are divided into position-number input bits for axis 1 and position-number input bits for axis 2.
Teaching mode	Maximum 1,500 positions
DS-S-C1 compatible mode	Maximum 1,500 positions

Note) Two sets of position data are needed for push-motion operation. (Push-motion operation can be performed only in the standard mode and product switching mode.)

3. Quick Mode Function Reference Table

		Other parameter No. 25				
		1	2	3	4	16
I/O	Function	Standard mode	Product switching mode	2-axis independent mode	Teaching mode	DS-S-C1 compatible mode
Input	Push-motion operation	○	○	x	x	x
	Error reset	○	○	○	○	x
	CPU reset	x	x	x	x	○
	Home return	○	○	○	Note 1	Note 2
	Servo ON	○	○	○	○	x
	Cancellation	○	○	○	x	○
	Interpolation	○	○	x	x	○
	Jog	x	x	x	○	x
Output	Home return complete	○	○	○	○	x
	Servo ON output	○	○	○	○	x
	System battery error	○	○	x Note 3	○	○

Note 1) In the teaching mode, home return will be performed when the start signal is input after specifying a desired position number in a condition where home return is not yet complete.

Note 2) In the DS-S-C1 compatible mode, home return will be performed when the start signal is input after specifying position No. 0.

Note 3) In the 2-axis independent mode, a system-battery voltage low warning will not be output. In this mode, it is recommended not to back up the position data and error list using the battery (not to use the optional system-memory backup battery).

4. Interface List of All PIO Patterns

Pin No.	Category	Port No.	Positioner mode					Cable color
			Standard mode	Product switching mode	2-axis independent mode	Teaching mode	DS-S-C1 compatible mode	
1A	P24				24-V input			1-Brown
1B	Input	16	Position input 10	Input 10	Position input 7	Axis 1 jog-	Position No. 1000 input	1-Red
2A		17	Position input 11	Input 11	Position input 8	Axis 2 jog+	-	1-Orange
2B		18	Position input 12	Input 12	Position input 9	Axis 2 jog-	-	1-Yellow
3A		19	Position input 13	Input 13	Position input 10	Inching (0.01 mm)	-	1-Green
3B		20	-	Input 14	Position input 11	Inching (0.1 mm)	-	1-Blue
4A		21	-	Input 15	Position input 12	Inching (0.5 mm)	-	1-Purple
4B		22	-	Input 16	Position input 13	Inching (1 mm)	-	1-Gray
5A		23	Error reset	Error reset	Error reset	Error reset	CPU reset	1-White
5B		0	Start	Start	Axis 1 start	Start	Start	1-Black
6A		1	Home return	Home return	Home return	Servo ON	Pause	2-Brown
6B		2	Servo ON	Servo ON	Axis 1 servo ON	*Pause	Cancellation	2-Red
7A		3	Push motion	Push motion	*Axis 1 pause	Position input 1	Interpolation setting	2-Orange
7B		4	*Pause	*Pause	*Axis 1 cancellation	Position input 2	Position No. 1 input	2-Yellow
8A		5	*Cancellation	*Cancellation	Axis 2 start	Position input 3	Position No. 2 input	2-Green
8B		6	Interpolation	Interpolation	Axis 2 home return	Position input 4	Position No. 4 input	2-Blue
9A		7	Position input 1	Input 1	Axis 2 servo ON	Position input 5	Position No. 8 input	2-Purple
9B		8	Position input 2	Input 2	*Axis 2 pause	Position input 6	Position No. 10 input	2-Gray
10A		9	Position input 3	Input 3	*Axis 2 cancellation	Position input 7	Position No. 20 input	2-White
10B		10	Position input 4	Input 4	Position input 1	Position input 8	Position No. 40 input	2-Black
11A		11	Position input 5	Input 5	Position input 2	Position input 9	Position No. 80 input	3-Brown
11B		12	Position input 6	Input 6	Position input 3	Position input 10	Position No. 100 input	3-Red
12A		13	Position input 7	Input 7	Position input 4	Position input 11	Position No. 200 input	3-Orange
12B		14	Position input 8	Input 8	Position input 5	Teaching mode specification	Position No. 400 input	3-Yellow
13A		15	Position input 9	Input 9	Position input 6	Axis 1 jog+	Position No. 800 input	3-Green
13B	Output	300	*Alarm	*Alarm	*Alarm	*Alarm	Alarm	3-Blue
14A		301	Ready	Ready	Ready	Ready	Ready	3-Purple
14B		302	Positioning complete	Positioning complete	Axis 1 positioning complete	Positioning complete	Positioning complete	3-Gray
15A		303	Home return complete	Home return complete	Axis 1 home return complete	Home return complete	-	3-White
15B		304	Servo ON output	Servo ON output	Axis 1 servo ON	Servo ON output	-	3-Black
16A		305	Push motion complete	Push motion complete	Axis 2 positioning complete		-	4-Brown
16B		306	System battery error	System battery error	Axis 2 home return complete	System battery error	System battery error	4-Red
17A		307	-	-	Axis 2 servo ON	-	-	4-Orange
17B	N		0-V input					4-Yellow

*: Contact B (always ON)

Chapter 2 Standard Mode

The standard mode provides a PIO pattern of greatest general utility among all positioner modes accessible in the PSEL controller.

1. I/O Interface List

Pin No.	Category	Port No.	Signal name	Signal symbol	Function overview	Cable color
1A	P24		External power supply 24 V	P24		1-Brown
1B	Input	016	Position input 10	PC10	(Same as position inputs 1 through 9)	1-Red
2A		017	Position input 11	PC11		1-Orange
2B		018	Position input 12	PC12		1-Yellow
3A		019	Position input 13	PC13		1-Green
3B		020	-			1-Blue
4A		021	-			1-Purple
4B		022	-			1-Gray
5A		023	Error reset	RES	Present alarms will be reset at the leading edge of this signal.	1-White
5B		000	Start	CSTR	The actuator will start moving at the leading edge of this signal.	1-Black
6A		001	Home return	HOME	The actuator will start home-return operation at the leading edge of this signal.	2-Brown
6B		002	Servo ON	SON	The servo will remain on while this signal is ON, and remain off while this signal is OFF.	2-Red
7A		003	Push motion	PUSH	The actuator will start linear interpolation operation if the start input signal is turned ON while this signal is ON.	2-Orange
7B		004	*Pause	*STP	The actuator can be moved when this signal is ON, and will decelerate to a stop when the signal turns OFF.	2-Yellow
8A		005	*Cancellation	*CANC	The remaining travel distance will be cancelled if this signal turns OFF.	2-Green
8B		006	Interpolation	LINE	With the 2-axis specification, linear interpolation operation will start when the start input signal is turned ON while this signal is ON.	2-Blue
9A		007	Position input 1	PC1	Input the position number corresponding to the position you want to move the actuator to. Be sure to specify a position input by no later than 6 msec before the start input signal turns ON. Position numbers are input as binary codes (factory setting). The input mode can be changed to BCD by changing the setting of other parameter No. 71. (PC1 through 4 indicate the one's place, PC5 through 8 indicate ten's place, PC9 through 12 indicate the hundred's place, and PC13 indicates the thousand's place.)	2-Purple
9B		008	Position input 2	PC2		2-Gray
10A		009	Position input 3	PC3		2-White
10B		010	Position input 4	PC4		2-Black
11A		011	Position input 5	PC5		3-Brown
11B		012	Position input 6	PC6		3-Red
12A		013	Position input 7	PC7		3-Orange
12B		014	Position input 8	PC8		3-Yellow
13A		015	Position input 9	PC9		3-Green
13B	Output	300	*Alarm	*ALM	This signal remains ON if the controller is normal. It will turn OFF if an alarm occurs.	3-Blue
14A		301	Ready	RDY	This signal will turn ON when the controller becomes ready.	3-Purple
14B		302	Positioning complete	PEND	This signal will turn ON once the actuator has moved to the target position and entered the positioning band.	3-Gray
15A		303	Home return complete	HEND	This signal is OFF when the power is input, and will turn ON when home return is completed.	3-White
15B		304	Servo ON output	SVON	This signal will turn ON when the servo is turned on, and turn OFF when the servo is turned off.	3-Black
16A		305	Push motion complete	PSED	This signal will turn ON when the push-motion operation is completed successfully, and turn OFF if the work is mixed.	4-Brown
16B		306	System battery error	SSER	This signal will turn ON when the voltage of the system-memory backup battery drops to the voltage-low warning level.	4-Red
17A		307				4-Orange
17B	N		External power supply 0 V	N		4-Yellow

*: Contact B (always ON)

2. Parameters

To use the controller in the standard mode, set other parameter No. 25 to “1.”
Position numbers are specified as binary codes according to the factory setting. To change the input mode to BCD, set a value “other than 0” in other parameter No. 25.

	No.	Parameter	Function
Other	25	Operation mode type	1: Standard mode
	71	Positioner mode parameter 1	Position-number input mode specification (0: Binary, ≠ 0: BCD) * Default value: 0 (Binary)

3. Details of Each Input Signal

■ Start (CSTR)

When the OFF → ON leading edge of this signal is detected, the controller will load the target point number specified by the 13-bit binary code consisting of PC1 through PC13, and perform positioning to the target position specified by the corresponding position data.
Before movement is started, the target position, speed and other operation data must be set in the position table using a PC or teaching pendant.
If this signal is input when no single home-return operation has been performed after the power was input (= when the HEND output signal is OFF), “C6F, Home-return incomplete error” will generate.

■ Command position number (PC1 through PC13)

When a movement command is executed upon the OFF → ON edge of the start signal, the controller will load the command position number specified by the 13-bit binary code consisting of signals PC1 through PC13.
The weight of each bit is as follows: 2^0 for PC1, 2^1 for PC2, 2^2 for PC3, 2^3 for PC4, 2^4 for PC5, ..., and 2^{10} for PC11. By combining these bits, any position number between 0 and 1500 (maximum) can be specified.
The input mode can be changed to BCD by changing the setting of other parameter No. 71, as follows:
Other parameter No. 71 = 1 (BCD input)
(Default setting of other parameter No. 71 = 0 (Binary input))
In the BCD input mode, PC1 through 4 indicate the one’s place, PC5 through 8 indicate ten’s place, PC9 through 12 indicate the hundred’s place, and PC13 indicates the thousand’s place.

■ Pause (*STP)

If this signal turns OFF while the actuator is moving, the controller will cause the actuator to decelerate to a stop.
The remaining travel distance will be held, which means that when the signal turns ON again, the actuator will resume movement of the remaining travel distance.
To cancel the movement command altogether after turning OFF the pause signal, turn OFF the cancellation signal while this signal is OFF to cancel the remaining travel distance.
The pause signal can be used for the following purposes:

[1] As a sensor to detect entry into a specified area around the system or for other lower-level safety measures to stop the axis while the servo is on
[2] To prevent contact with other equipment
[3] For positioning based on sensor or LS signal detection

(Note) When this signal is input during home return, the movement command will be held if the actuator has not yet contacted the mechanical end. If the signal is input after the actuator has reversed upon contacting the mechanical end, home return will be performed again.

■ Cancellation (*CANC)

If this signal turns OFF while the actuator is moving, the controller will cause the actuator to decelerate to a stop. The remaining travel distance will be cancelled and the movement will not resume even when the signal turns ON thereafter.

■ Home return (HOME)

The actuator will start home-return operation upon detection of the OFF → ON edge of this signal.
Once the home return is complete, the HEND signal will be output. This signal can be input as many times as desired after completion of the initial home return.
(Note) An actuator of incremental specification must always perform home return after the power is turned on.

■ Servo ON (SON)

The servo remains on while this signal is ON.
To operate the actuator using the start input/home return input, the servo ON input signal must be ON. If the servo ON input signal is OFF, these operation commands will not be accepted. (Only the commands will be ignored, and no error will generate.)
(Note) When this signal turns OFF while the actuator is moving, the actuator will not decelerate to a stop. It will complete the movement to the target position, after which the servo will turn off.

■ Error reset (RES)

This signal is used to reset the alarm output signal (*ALM) that has been generated due to an error.
If an error occurred, check the content of the error and then turn this signal ON.
The error will be reset upon detection of the leading edge of the signal.
(Note) Errors of cold start and higher level cannot be reset using this signal. The power must be reconnected to reset these errors. For details, refer to Appendix, “Error Level Management.”

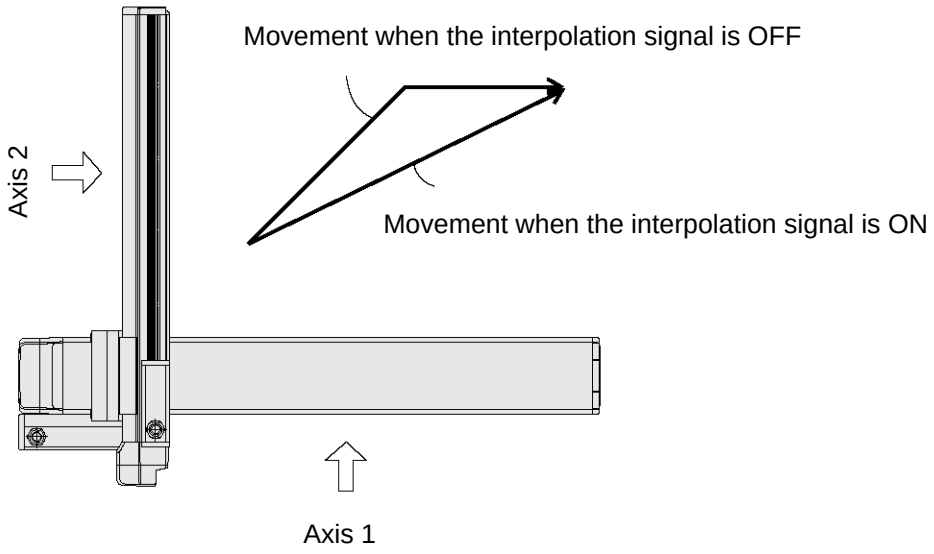
■ Push motion (PUSH)

The actuator will perform push-motion operation if the position signal and start signal are input while this signal is ON. To perform push-motion operation, turn ON the push-motion input signal before turning the start input signal ON.
A push-motion operation command is specified using two successive position data points.
If the “start” input signal is turned ON while the “push-motion” input signal is ON for position No. n, the position data corresponding to position No. n and position No. n+1 will indicate the following items:
The position data for position No. n indicates the target position.
The position data for position No. n+1 indicates the push width.
The acceleration data for position No. n+1, multiplied by 100, indicates the current-limiting value during push-motion operation.
The speed data for position No. n+1 indicates the push speed.
Example: The position data for position No. 1, as specified in the table below, is used for push-motion operation.
Target position: 100 mm, Push width: 30 mm, Current-limiting value: 50%
Acceleration/deceleration until the push width before the target position: 0.2 G
Push speed: 25 mm/sec

No	Axis1	Axis2	Vel	Acc	Dcl
1	100.000		100	0.20	0.20
2	30.000		25	0.50	

■ Interpolation (LINE)

With the 2-axis specification, input of the position signal and start signal while this signal is ON will cause the two axes to perform interpolation operation (the two axes will start simultaneously and arrive at the target position simultaneously).
To perform interpolation operation, turn ON the interpolation input signal before turning ON the start input signal.



4. Details of Each Output Signal

■ Positioning complete (PEND)

This signal indicates that the actuator reached the target position and the positioning has completed.
After the power was input and the servo has turned on, this signal will turn ON if the position deviation is within the in-position band when the controller becomes ready.
Thereafter, this signal will turn OFF when the start signal is turned ON to execute a movement command.
The signal will turn ON if the position deviation from the target position is within the in-position band after the start signal has turned OFF.
Once this signal turns ON, it will not turn OFF even after the position deviation subsequently exceeds the in-position band.

(Note) If the start signal is ON, this signal will not turn ON even when the position deviation from the target position falls within the in-position band. The signal will turn ON after the start signal turns OFF.
Even if the motor is stopped, this signal will remain OFF if a pause signal is input or the servo is off.

■ Home return complete (HEND)

This signal is OFF when the power is input, and will turn ON when the home-return operation initiated by input of the home-return signal is completed.
Once this signal turns ON, it will not turn OFF until the input power is cut off or the home-return signal is input again.

■ Alarm (*ALM)

This signal remains ON while the controller is normal, and will turn OFF if an alarm occurs.
This signal will turn OFF when an error of operation-cancellation level or higher generates.
Program the PLC so that it will monitor this signal and implement appropriate safety measures to protect the entire system when the signal turns OFF.
For details on alarms, refer to Appendix “⊙ Error Level Management” and “⊙ Error List.”

■ Ready (RDY)

This signal will turn ON when the initialization has completed successfully after the main power was input, and the controller enters the mode where it can control the actuator.
This signal will turn OFF when an error of cold level or higher generates.
Use this signal as a condition to start control on the PLC side.

■ Servo ON output (SVON)

This signal will turn ON when the servo turns on. Issue a movement command after the servo ON output signal has turned ON.

■ System battery error

This signal will turn ON when the voltage of the optional system-memory backup battery drops to a specified level.

5. Timing Chart

5.1 Recognition of I/O Signals

An input time constant is set for the input signals of this controller to prevent malfunction due to chattering, noise, etc.
Except for certain signals, the input signal will switch if the new signal level has remained for at least 6 [msec].
For example, when an input signal is turned ON, the controller will recognize that the signal is ON after elapse of 6 [msec]. The same applies when the signal is turned OFF. (Fig. 1)

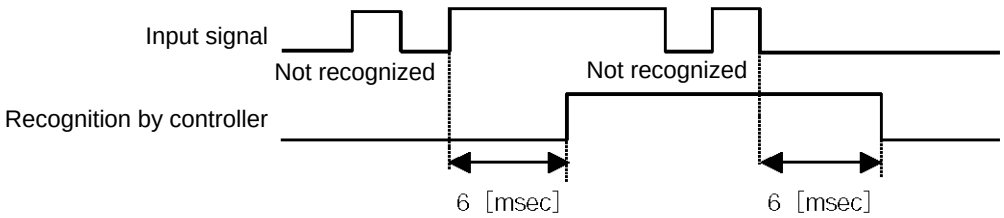
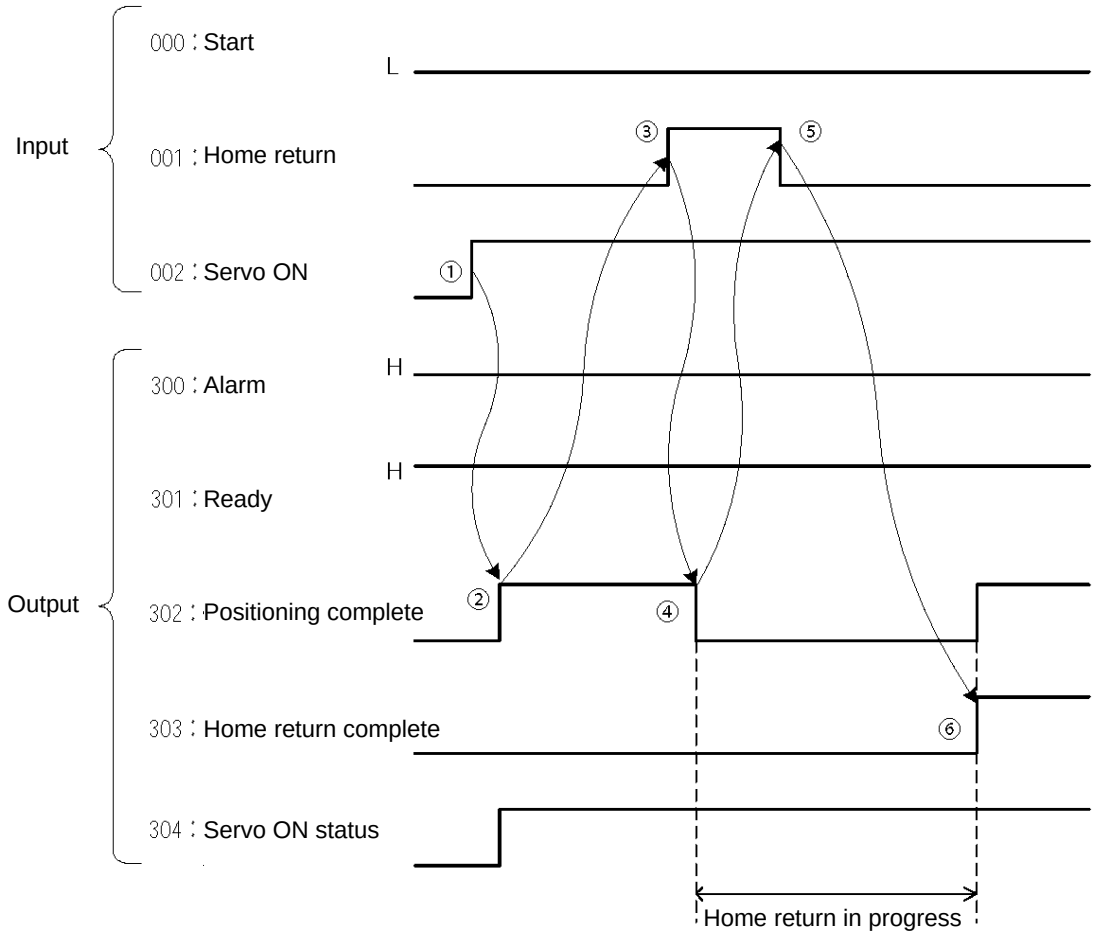


Fig. 1 Recognition of Input Signal

5.2 Home Return

Timings associated with home-return operation are illustrated below.



Timing Chart of Home-return Operation (Standard Positioner Mode)

Perform home-return operation by following the procedure explained below.
* Before commencing the procedure, confirm that the ready output signal and alarm output signal are ON.

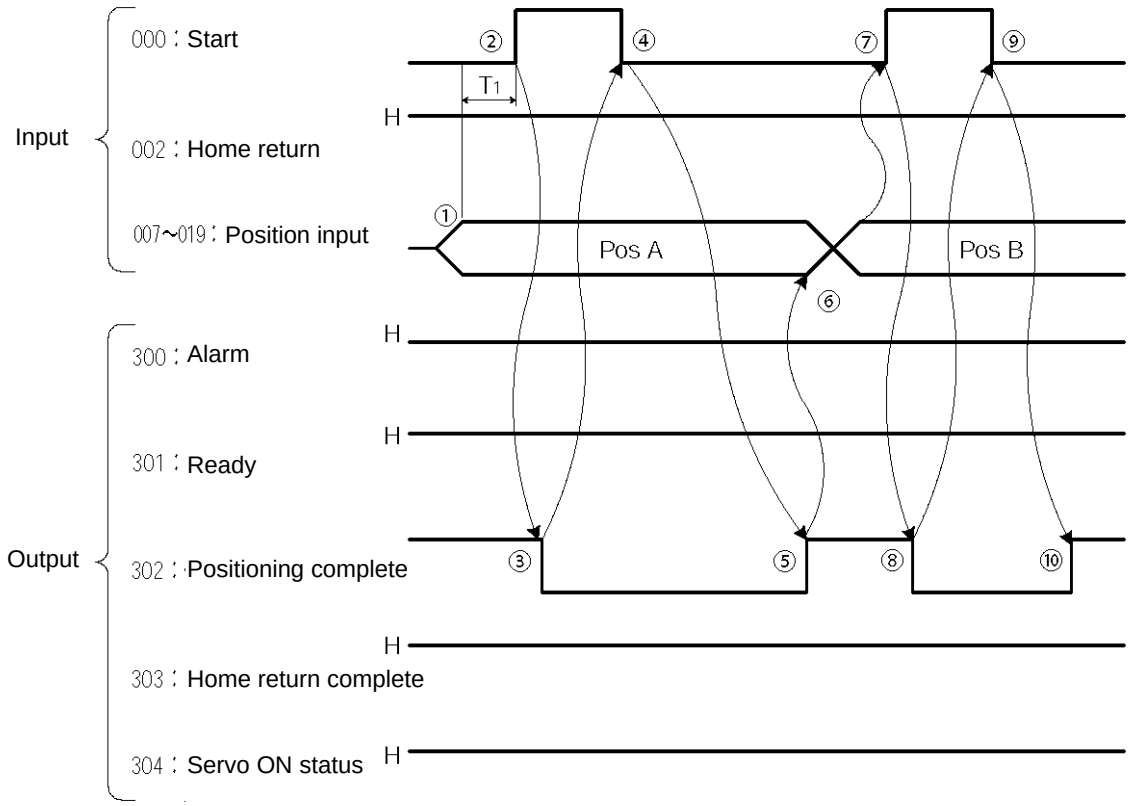
[1] Turn ON the servo ON input signal.
[2] Confirm that the servo-ON status output signal is ON.
[3] Turn ON the home-return input signal.
[4] Confirm that the positioning complete output signal is OFF.
[5] Turn OFF the home-return input signal.
[6] Confirm that the home-return complete output signal is ON. Home return is now completed.

*Pause and *cancellation inputs are contact-B input signals (always ON), so keep these signals ON while home return is in progress.
To initiate home return using the home-return signal input, the servo ON input signal must be ON. These operation commands will not be accepted if the servo ON input signal is OFF. Note, however, that only the commands will be ignored and no error will generate.

With the 2-axis specification, the controller has been configured at the factory so that the two axes will start home return simultaneously.
You can cause either axis to start home return earlier than the other axis by changing the applicable parameter setting.
Specifically, change the setting in axis-specific parameter No. 13, “SIO/PIO home-return order” so that the parameter value for the axis number corresponding to the axis for which you want to complete home return first, will become smaller than the parameter value for the other axis number.
Example) Cause axis 1 to perform home return after axis 2 has completed home return, set “1” for axis 1 and “0” for axis 2 in axis-specific parameter No. 13.

5.3 Movements through Positions

Timings of how the actuator moves through positions are illustrated below.



Timing Chart of Movement through Positions (Standard Positioner Mode)
Ti: At least 6 msec

Operate the actuator to move through positions by following the procedure explained below.

* Confirm beforehand that the positioning complete output signal, home-return complete output signal and servo-ON status output signal are all ON.
  [1] Change the previous position number input to a different position number.
  [2] Turn ON the start input signal.
  [3] Confirm that the positioning complete output signal is OFF.
  [4] Turn OFF the start input signal.
  [5] Confirm that the positioning complete output signal is ON.
  Repeat steps [1] through [5] sequentially.
* Pause and *cancellation inputs are contact-B input signals (always ON), so keep these signals ON while the actuator are moving through the specified positions.

* To perform push-motion or interpolation operation, turn ON the applicable input signal before turning ON the start input signal. Turn the operation signal OFF after the start input signal has turned OFF.
* While the actuator is moving to the target position, only the pause or cancellation input is accepted. The servo cannot be turned off even if the servo ON input signal is turned OFF. (The servo can be turned off only when the positioning complete output signal is ON.)
* While the start input signal is ON, the positioning complete output signal will not turn ON even after the actuator physically completes moving to the target position. Therefore, always turn OFF the start input signal ([4]) to detect the completion of positioning.

* As for the positioning complete output signal and push-motion complete output signal, they will not be output until the start signal turns OFF (based on the I/O control handshake rules).

* For the actuator to operate upon start signal input, the servo ON input signal must be ON. If the servo ON input signal is OFF, these operation commands will not be accepted. Note, however, that only the commands will be ignored and no error will generate.

Chapter 3 Product Switching Mode

In addition to position numbers, product numbers can also be specified in this mode. Sixteen bits of inputs 1 through 16 are divided into position number inputs and product number inputs.
In other words, the actuator can be moved to different positions for different products by specifying the same position number.

1. I/O Interface List

Pin No.	Category	Port No.	Signal name	Signal symbol	Function overview	Cable color
1A	P24		External power supply 24 V	P24		1-Brown
1B	Input	016	Input 10	PC10	(Same as inputs 1 through 9)	1-Red
2A		017	Input 11	PC11		1-Orange
2B		018	Input 12	PC12		1-Yellow
3A		019	Input 13	PC13		1-Green
3B		020	Input 14	PC14		1-Blue
4A		021	Input 15	PC15		1-Purple
4B		022	Input 16	PC16		1-Gray
5A		023	Error reset	RES	Present alarms will be reset at the leading edge of this signal.	1-White
5B		000	Start	CSTR	The actuator will start moving at the leading edge of this signal.	1-Black
6A		001	Home return	HOME	The actuator will start home-return operation at the leading edge of this signal.	2-Brown
6B		002	Servo ON	SON	The servo will remain on while this signal is ON, and remain off while this signal is OFF.	2-Red
7A		003	Push motion	PUSH	The actuator will start linear interpolation operation if the start input signal is turned ON while this signal is ON.	2-Orange
7B		004	*Pause	*STP	The actuator can be moved when this signal is ON, and will decelerate to a stop when the signal turns OFF.	2-Yellow
8A		005	*Cancellation	*CANC	The remaining travel distance will be cancelled if this signal turns OFF.	2-Green
8B		006	Interpolation	LINE	With the 2-axis specification, linear interpolation operation will start when the start input signal is turned ON while this signal is ON.	2-Blue
9A		007	Input 1	PC1	These input signals specify position numbers and product numbers. Sixteen bits of inputs 1 through 16 are divided into position number inputs and product number inputs. Be sure to specify an input by no later than 6 msec before the start signal turns ON. Position numbers and product numbers are input as binary codes (factory setting). The input mode can be changed to BCD by changing the setting of other parameter No. 71.	2-Purple
9B		008	Input 2	PC2		2-Gray
10A		009	Input 3	PC3		2-White
10B		010	Input 4	PC4		2-Black
11A		011	Input 5	PC5		3-Brown
11B		012	Input 6	PC6		3-Red
12A		013	Input 7	PC7		3-Orange
12B		014	Input 8	PC8		3-Yellow
13A		015	Input 9	PC9		3-Green
13B	Output	300	Alarm	*ALM	This signal remains ON if the controller is normal. It will turn OFF if an alarm occurs.	3-Blue
14A		301	Ready	RDY	This signal will turn ON when the controller becomes ready.	3-Purple
14B		302	Positioning complete	PEND	This signal will turn ON once the actuator has moved to the target position and entered the positioning band.	3-Gray
15A		303	Home return complete	HEND	This signal is OFF when the power is input, and will turn ON when home return is completed.	3-White
15B		304	Servo ON output	SVON	This signal will turn ON when the servo is turned on, and turn OFF when the servo is turned off.	3-Black
16A		305	Push motion complete	PSED	This signal will turn ON when the push-motion operation is completed successfully, and turn OFF if the work is mixed.	4-Brown
16B		306	System battery error	SSER	This signal will turn ON when the voltage of the system-memory backup battery drops to the voltage-low warning level.	4-Red
17A		307				4-Orange
17B	N		External power supply 0 V	N		4-Yellow

*: Contact B (always ON)

2. Parameters

The following parameters must be set in the product switching mode.

Table: Parameter Settings in Product Switching Mode

Type	No.	Parameter	Function
Other	25	Operation mode type	2: Product switching mode
	71	Positioner mode parameter 1	Position-number input mode specification (0: Binary, ≠ 0: BCD) * Default value: 0 (Binary)
	72	Positioner mode parameter 2	Number of position-number input bits Binary: Number of bits – 1 through 15 bits BCD: Number of BCD digits – 1 through 3 digits
	73	Positioner mode parameter 3	Number of positions per product

When the above parameters are set, the actual position movement commands will apply based on the following formula:
“(Product number input – 1) x Number of positions per product + Position number input”

For example, assume that the parameters are set as follows:
Other parameter No. 71 = 0 (Binary) “Position-number input mode specification”
Other parameter No. 72 = 6 “Number of position-number input bits”
Other parameter No. 73 = 50 “Number of positions per product”
Each position number is assigned to six bits of inputs 1 through 6 (007 through 012), as a binary code, and position Nos. 1 through 63 can be specified.
Each product number is assigned to 10 bits of inputs 7 through 6 (013 through 022), as a binary code, and 30 types can be specified (the number of types is limited to 30, because the maximum number of position data is 1500). If any greater value is set that brings the number of position data to more than 1500, a “point number error” will generate.

* If the value of position number input exceeds the number of positions per product, the controller will recognize that “1” has been set as the position number.

(Note) The result of “Number of position-number input bits” + “Number of product-number input bits” must not exceed 16 (bits).

3. Details of Each Input Signal

■ Start (CSTR)

Movement to the position corresponding to the position data of the specified product will start upon detection of the OFF → ON leading edge of this signal. Product numbers and position numbers are specified by the 16-bit binary code consisting of inputs 1 through 16.
Before movement is started, the target position, speed and acceleration/deceleration must be set as position data. Use a PC (software) or teaching pendant to set position data.
If this signal is input when no single home-return operation has been performed after the power was input (= when the HEND output signal is OFF), “C6F, Home-return incomplete error” will generate.

■ Inputs 1 through 16 (PC1 through 16)

Sixteen bits of inputs 1 through 16 are divided into position-number input bits and product-number input bits.
Example) Assume that the parameters are set as follows:
Other parameter No. 71 = 0 (Binary) “Position-number input mode specification”
Other parameter No. 72 = 6 “Number of position-number input bits”
Other parameter No. 73 = 50 “Number of positions per product”
Each position number input is assigned to six bits of inputs 1 through 6 (007 through 012), as a binary code.
Each product number input is assigned to 10 bits of inputs 7 through 16 (013 through 022), as a binary code
Position numbers and product numbers are specified as shown in the table below, based on the ON/OFF levels of inputs 1 through 16.

		Product				Position number input					
		Product 1	Product 2	Product 3	Product 4	Input 6	Input 5	Input 4	Input 3	Input 2	Input 1
Position number (when set)		1	51	101	151	0	0	0	0	0	0
		2	52	102	152	0	0	0	0	0	0
		3	53	103	153	0	0	0	0	0	0
		4	54	104	154	0	0	0	0	0	0
		:	:	:	:	:	:	:	:	:	:
		:	:	:	:	:	:	:	:	:	:
		49	59	149	199	1	1	0	0	0	1
		50	100	150	200	1	1	0	0	1	0
Product number input	Input 7	1	0	1	0						
	Input 8	0	1	1	0						
	Input 9	0	0	0	1						
	Input 10	0	0	0	0						
	Input 11	0	0	0	0						
	Input 12	0	0	0	0						
	Input 13	0	0	0	0						
	Input 14	0	0	0	0						
	Input 15	0	0	0	0						
	Input 16	0	0	0	0						

Fifty position numbers (Nos. 1 through 50) can be specified for each product.
Position No. 49 for product 2 (set as No. 99 within the entire data) is specified as follows.

Input 16	Input 15	Input 14	Input 13	Input 12	Input 11	Input 10	Input 9	Input 8	Input 7	Input 6	Input 5	Input 4	Input 3	Input 2	Input 1
0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	1

The input mode can be changed to BCD by changing the setting of other parameter No. 71.
Assume the following settings:

Other parameter No. 71, “Position-number input method specification” = 1 (BCD)
Other parameter No. 72, “Number of position-number input bits” = 8
(In the BCD input mode, one digit consists of four bits. In other words, bits are input in units of four.)
Other parameter No. 73, “Number of positions per product” = 50
Each position number is assigned to eight bits of inputs 1 through 8 (007 through 014), as a two-digit BCD code.

Each product number is assigned to eight bits of inputs 9 through 16 (015 through 022), as a two-digit BCD code.
As for the position number, specify the one’s place in inputs 1 through 4, and ten’s place in inputs 5 through 8.
As for the product number, specify the one’s place in inputs 9 through 12, and ten’s place in inputs 13 through 16.

■ Pause (*STP)

If this signal turns OFF while the actuator is moving, the controller will cause the actuator to decelerate to a stop.
The remaining travel distance will be held, which means that when the signal turns ON again, the actuator will resume movement of the remaining travel distance.
To cancel the movement command altogether after turning OFF the pause signal, turn OFF the cancellation signal while this signal is OFF to cancel the remaining travel distance.
The pause signal can be used for the following purposes:

[1] As a sensor to detect entry into a specified area around the system or for other lower-level safety measures to stop the axis while the servo is on
[2] To prevent contact with other equipment
[3] For positioning based on sensor or LS signal detection

(Note) When this signal is input during home return, the movement command will be held if the actuator has not yet contacted the mechanical end. If the signal is input after the actuator has reversed upon contacting the mechanical end, home return will be performed again.

■ Cancellation (*CANC)

If this signal turns OFF while the actuator is moving, the controller will cause the actuator to decelerate to a stop. The remaining travel distance will be cancelled and the movement will not resume even when the signal turns ON thereafter.

■ Home return (HOME)

The actuator will start home-return operation upon detection of the OFF → ON edge of this signal.
Once the home return is complete, the HEND signal will be output. This signal can be input as many times as desired after completion of the initial home return.

(Note) An actuator of incremental specification must always perform home return after the power is turned on.

■ Servo ON (SON)

The servo remains on while this signal is ON.
Use this signal if servo ON/OFF control is required as part of the safety circuit for the entire system to be provided on the PLC side.
To operate the actuator using the start input/home return input, the servo ON input signal must be ON. If the servo ON input signal is OFF, these operation commands will not be accepted. (Only the commands will be ignored, and no error will generate.)

(Note) When this signal turns OFF while the actuator is moving, the actuator will not decelerate to a stop. It will complete the movement to the target position, after which the servo will turn off.

■ Error reset (RES)

This signal is used to reset the alarm output signal (*ALM) that has been generated due to an error.
If an error occurred, check the content of the error and then turn this signal ON.
The error will be reset upon detection of the leading edge of the signal.
(Note) Errors of cold start and higher level cannot be reset using this signal. The power must be reconnected to reset these errors. For details, refer to Appendix, “Error Level Management.”

■ Push motion (PUSH)

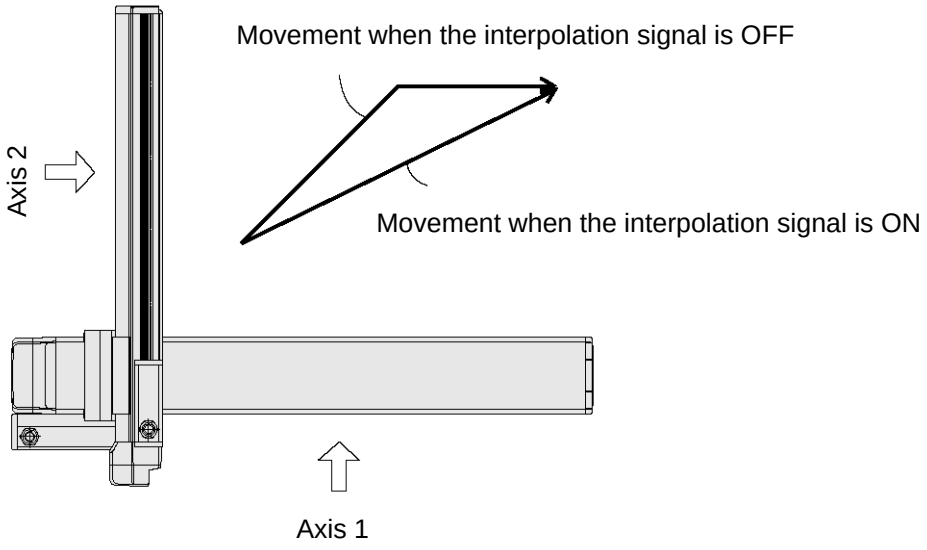
The actuator will perform push-motion operation if the position signal and start signal are input while this signal is ON. To perform push-motion operation, turn ON the push-motion input signal before turning the start input signal ON.
A push-motion operation command is specified using two successive position data points.
If the “start” input signal is turned ON while the “push-motion” input signal is ON for position No. n, the position data corresponding to position No. n and position No. n+1 will indicate the following items:
The position data for position No. n indicates the target position.
The position data for position No. n+1 indicates the push width.
The speed data for position No. n+1 indicates the push speed.
The acceleration data for position No. n+1, multiplied by 100, indicates the current-limiting value during push-motion operation.
Example: The position data for position No. 1, as specified in the table below, is used for push-motion operation.

No	Axis1	Axis2	Vel	Acc	Dcl
1	100.000		100	0.20	0.20
2	30.000		25	0.50	

The actuator moves at a speed of 100 mm/sec, acceleration of 0.2 G and deceleration of 0.2 G, until 30 mm before a target position of 100 mm. Thereafter, the actuator performs push-motion operation to the target position at a speed of 25 mm/sec and current-limiting value of 50%.

■ Interpolation (LINE)

With the 2-axis specification, input of the position signal and start signal while this signal is ON will cause the two axes to perform interpolation operation (the two axes will start simultaneously and arrive at the target position simultaneously).
To perform interpolation operation, turn ON the interpolation input signal before turning ON the start input signal.



4. Details of Each Output Signal

■ Positioning complete (PEND)

This signal indicates that the actuator reached the target position and the positioning has completed.
After the power was input and the servo has turned on, this signal will turn ON if the position deviation is within the in-position band when the controller becomes ready.
Thereafter, this signal will turn OFF when the start signal is turned ON to execute a movement command.
The signal will turn ON if the position deviation from the target position is within the in-position band after the start signal has turned OFF.
Once this signal turns ON, it will not turn OFF even after the position deviation subsequently exceeds the in-position band.

(Note) If the start signal is ON, this signal will not turn ON even when the position deviation from the target position falls within the in-position band. The signal will turn ON after the start signal turns OFF.
Even if the motor is stopped, this signal will remain OFF if a pause signal is input or the servo is off.

■ Home return complete (HEND)

This signal is OFF when the power is input, and will turn ON when the home-return operation initiated by input of the home-return signal is completed.
Once this signal turns ON, it will not turn OFF until the input power is cut off or the home-return signal is input again.

■ Alarm (*ALM)

This signal remains ON while the controller is normal, and will turn OFF if an alarm occurs.
This signal will turn OFF when an error of operation-cancellation level or higher generates.
Program the PLC so that it will monitor this signal and implement appropriate safety measures to protect the entire system when the signal turns OFF.
For details on alarms, refer to Appendix “⊙ Error Level Management” and “⊙ Error List.”

■ Ready (RDY)

This signal will turn ON when the initialization has completed successfully after the main power was input, and the controller enters the mode where it can control the actuator.
This signal will turn OFF when an error of cold level or higher generates.
Use this signal as a condition to start control on the PLC side.

■ Servo ON output (SVON)

This signal will turn ON when the servo turns on. Issue a movement command after the servo ON output signal has turned ON.

■ System battery error

This signal will turn ON when the voltage of the optional system-memory backup battery drops to a specified level.

5. Timing Chart

5.1 Recognition of I/O Signals

An input time constant is set for the input signals of this controller to prevent malfunction due to chattering, noise, etc.
Except for certain signals, the input signal will switch if the new signal level has remained for at least 6 [msec].
For example, when an input signal is turned ON, the controller will recognize that the signal is ON after elapse of 6 [msec]. The same applies when the signal is turned OFF. (Fig. 1)

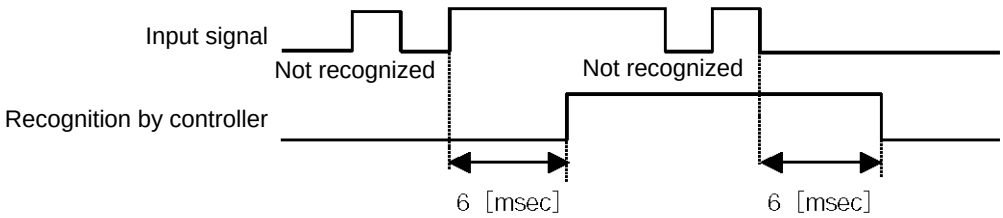
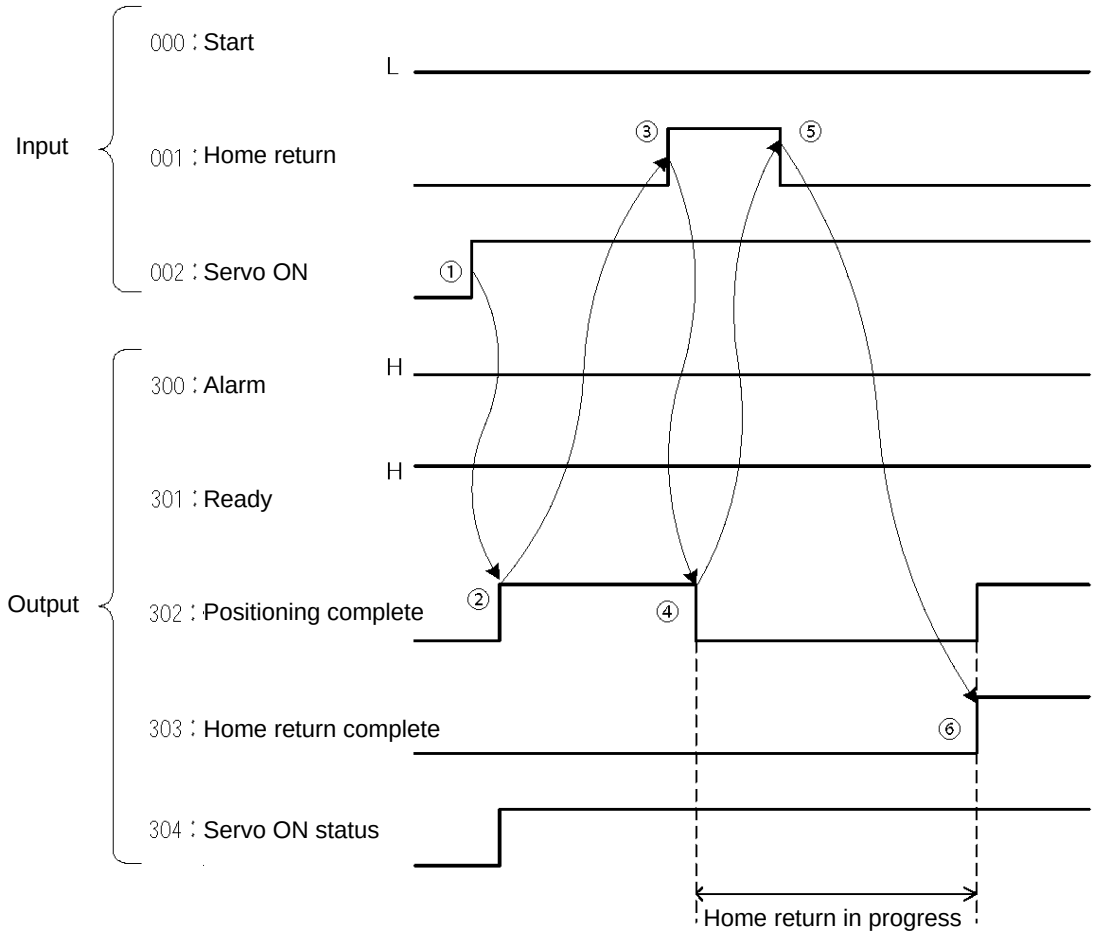


Fig. 1 Recognition of Input Signal

5.2 Home Return

Timings associated with home-return operation are illustrated below.



Timing Chart of Home-return Operation (Standard Positioner Mode)

Perform home-return operation by following the procedure explained below.
* Before commencing the procedure, confirm that the ready output signal and alarm output signal are ON.

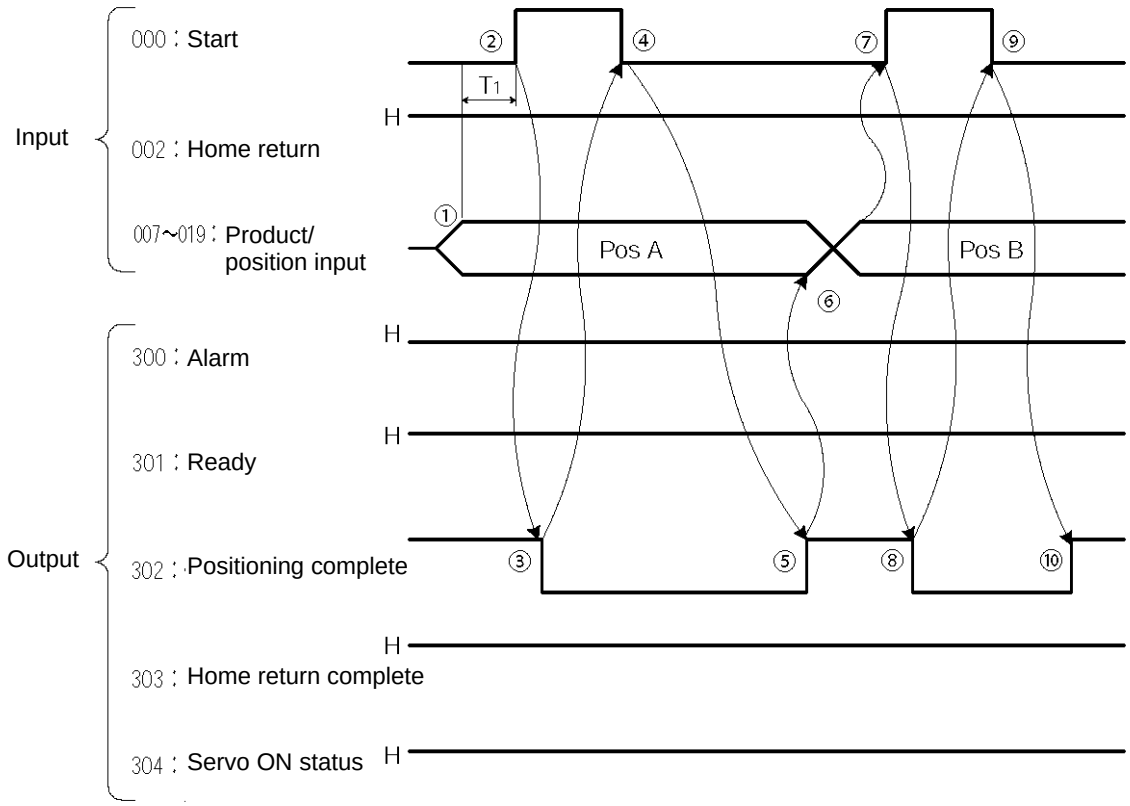
[1] Turn ON the servo ON input signal.
[2] Confirm that the servo-ON status output signal is ON.
[3] Turn ON the home-return input signal.
[4] Confirm that the positioning complete output signal is OFF.
[5] Turn OFF the home-return input signal.
[6] Confirm that the home-return complete output signal is ON. Home return is now completed.

*Pause and *cancellation inputs are contact-B input signals (always ON), so keep these signals ON while home return is in progress.
To initiate home return using the home-return signal input, the servo ON input signal must be ON. These operation commands will not be accepted if the servo ON input signal is OFF. Note, however, that only the commands will be ignored and no error will generate.

With the 2-axis specification, the controller has been configured at the factory so that the two axes will start home return simultaneously.
You can cause either axis to start home return earlier than the other axis by changing the applicable parameter setting.
Specifically, change the setting in axis-specific parameter No. 13, “SIO/PIO home-return order” so that the parameter value for the axis number corresponding to the axis for which you want to complete home return first, will become smaller than the parameter value for the other axis number.
Example) Cause axis 1 to perform home return after axis 2 has completed home return, set “1” for axis 1 and “0” for axis 2 in axis-specific parameter No. 13.

5.3 Movements through Positions

Timings of how the actuator moves through positions are illustrated below.



Timing Chart of Movement through Positions (Standard Positioner Mode)
Ti: At least 6 msec

Operate the actuator to move through positions by following the procedure explained below.

* Confirm beforehand that the positioning complete output signal, home-return complete output signal and servo-ON status output signal are all ON.
  [1] Change the previous product/position number inputs to different product/position numbers.
  [2] Turn ON the start input signal.
  [3] Confirm that the positioning complete output signal is OFF.
  [4] Turn OFF the start input signal.
  [5] Confirm that the positioning complete output signal is ON.
  Repeat steps [1] through [5] sequentially.
* Pause and *cancellation inputs are contact-B input signals (always ON), so keep these signals ON while the actuator are moving through the specified positions.

* To perform push-motion or interpolation operation, turn ON the applicable input signal before turning ON the start input signal. Turn the operation signal OFF after the start input signal has turned OFF.
* While the actuator is moving to the target position, only the pause or cancellation input is accepted. The servo cannot be turned off even if the servo ON input signal is turned OFF. (The servo can be turned off only when the positioning complete output signal is ON.)
* While the start input signal is ON, the positioning complete output signal will not turn ON even after the actuator physically completes moving to the target position. Therefore, always turn OFF the start input signal ([4]) to detect the completion of positioning.

* As for the positioning complete output signal and push-motion complete output signal, they will not be output until the start signal turns OFF (based on the I/O control handshake rules).

* For the actuator to operate upon start signal input, the servo ON input signal must be ON. If the servo ON input signal is OFF, these operation commands will not be accepted. Note, however, that only the commands will be ignored and no error will generate.

Chapter 4 2-axis Independent Mode

With the 2-axis specification, each axis can be controlled separately in this mode. A set of signals, such as the start input signal and positioning complete output signal, are provided for each axis.
Although the position number specification applies commonly to both axes, 13 bits of position inputs 1 through 13 (PC1 through 13) are divided into position-number specification bits for axis 1 and position-number specification bits for axis 2.

1. I/O Interface List

Pin No.	Category	Port No.	Signal name	Signal symbol	Function overview	Cable color
1A	P24		External power supply 24 V	P24		1-Brown
1B	Input	016	Position input 7	PC7	(Same as position inputs 1 through 6)	1-Red
2A		017	Position input 8	PC8		1-Orange
2B		018	Position input 9	PC9		1-Yellow
3A		019	Position input 10	PC10		1-Green
3B		020	Position input 11	PC11		1-Blue
4A		021	Position input 12	PC12		1-Purple
4B		022	Position input 13	PC13		1-Gray
5A		023	Error reset	RES	Present alarms will be reset at the leading edge of this signal.	1-White
5B		000	Axis 1 start	CSTR1	Axis 1 will start moving at the leading edge of this signal.	1-Black
6A		001	Axis 1 home return	HOME1	Axis 1 will start home-return operation at the leading edge of this signal.	2-Brown
6B		002	Axis 1 servo ON	SON1	The servo for axis 1 will remain on while this signal is ON, and remain off while this signal is OFF.	2-Red
7A		003	*Axis 1 pause	*STP1	Axis 1 can be moved when this signal turns ON, and will decelerate to a stop when the signal turns OFF.	2-Orange
7B		004	*Axis 1 cancellation	*CANC	The remaining travel distance of axis 1 will be cancelled if this signal turns OFF.	2-Yellow
8A		005	Axis 2 start	CSTR2	Axis 2 will start moving at the leading edge of this signal.	2-Green
8B		006	Axis 2 home return	HOME2	Axis 2 will start home-return operation at the leading edge of this signal.	2-Blue
9A		007	Axis 2 servo ON	SON2	The servo for axis 2 will remain on while this signal is ON, and remain off while this signal is OFF.	2-Purple
9B		008	*Axis 2 pause	*STP2	Axis 2 can be moved when this signal turns ON, and will decelerate to a stop when the signal turns OFF.	2-Gray
10A		009	*Axis 2 cancellation	*CANC2	The remaining travel distance of axis 2 will be cancelled if this signal turns OFF.	2-White
10B		010	Position input 1	PC1	Thirteen bits of position inputs 1 through 13 are divided into position-number specification bits for axis 1 and position-number specification bits for axis 2.	2-Black
11A		011	Position input 2	PC2		3-Brown
11B		012	Position input 3	PC3		3-Red
12A		013	Position input 4	PC4		3-Orange
12B		014	Position input 5	PC5		3-Yellow
13A		015	Position input 6	PC6		3-Green
13B	Output	300	*Alarm	*ALM	This signal remains ON if the controller is normal. It will turn OFF if an alarm occurs.	3-Blue
14A		301	Ready	RDY	This signal will turn ON when the controller becomes ready.	3-Purple
14B		302	Axis 1 positioning complete	PEND1	This signal will turn ON once axis 1 has moved to the target position and entered the positioning band.	3-Gray
15A		303	Axis 1 home-return complete	HEND1	This signal is OFF when the power to axis 1 is input, and will turn ON when home return is completed.	3-White
15B		304	Axis 1 servo ON	SVON1	This signal will turn ON when the servo for axis 1 is turned on, and turn OFF when the servo is turned off.	3-Black
16A		305	Axis 2 positioning complete	PEND2	This signal will turn ON once axis 2 has moved to the target position and entered the positioning band.	4-Brown
16B		306	Axis 2 home-return complete	HEND2	This signal is OFF when the power to axis 2 is input, and will turn ON when home return is completed.	4-Red
17A		307	Axis 2 servo ON	SVON2	This signal will turn ON when the servo for axis 2 is turned on, and turn OFF when the servo is turned off.	4-Orange
17B	N		External power supply 0 V	N		4-Yellow

*: Contact B (always ON)

2. Parameters

The following parameters must be set in the 2-axis independent mode.

Type	No.	Parameter	Function
Other	25	Operation mode type	3: 2-axis independent mode
	71	Positioner mode parameter 1	Position-number input mode specification (0: Binary, ≠ 0: BCD) * Default value: 0 (Binary)
	72	Positioner mode parameter 2	Specification of number of position-number input bits for axis 1 Binary: Number of bits – 1 through 12 bits BCD: Number of BCD digits – 1 or 2 digits

Specify the number of position-number input bits for axis 1 in other parameter No. 72, “Positioner mode parameter 2.” Specify how many bits will be assigned to axis 1, from among the 13 bits of position inputs 1 through 13. The remainder of the bits will be assigned to axis 2.
By specifying binary or BCD in the “position-number input mode specification” parameter, the setting unit of this parameter will change between bit and BCD digit.

Example) Assume that the parameters are set as follows:
Other parameter No. 71 = 0 (Binary) “Position-number input mode specification”
Other parameter No. 72 = 7 “Specification of position-number input bits for axis 1”

Each position number input for axis 1 is assigned to seven bits of inputs 1 through 7 (010 through 016), as a binary code, and position Nos. 1 through 127 can be specified.
Each position number input for axis 2 is assigned to the remaining six bits of inputs 8 through 13 (017 through 022), as a binary code, and position Nos. 1 through 63 can be specified.

3. Details of Each Input Signal

■ Position inputs 1 through 13 (PC1 through 13)

Thirteen bits of PC1 through 13 are divided into position-number specification bits for axis 1 and position-number specification bits for axis 2.
Example) Assume that the parameters are set as follows:
Other parameter No. 71 = 0 (Binary) “Position-number input mode specification”
Other parameter No. 72 = 7 “Specification of position-number input bits for axis 1”

Each position number input for axis 1 is assigned to seven bits of PC1 through 7 (010 through 016), as a binary code, and position Nos. 1 through 127 can be specified.
Each position number input for axis 2 is assigned to the remaining six bits of PC8 through 13 (017 through 022), as a binary code, and position Nos. 1 through 63 can be specified.
Position numbers for respective axes are specified as shown in the table below, based on the ON/OFF levels of PC1 through 13.

Position number specification for axis 2						Position No.	Position number specification for axis 1						
PC13	PC12	PC11	PC10	PC9	PC8		PC7	PC6	PC5	PC4	PC3	PC2	PC1
0	0	0	0	0	1	1	0	0	0	0	0	0	1
0	0	0	0	1	0	2	0	0	0	0	0	1	0
0	0	0	0	1	1	3	0	0	0	0	0	1	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	1	1	1	1	0	62	0	1	1	1	1	0	0
1	1	1	1	1	1	63	0	1	1	1	1	1	1
						⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
						126	1	1	1	1	1	1	0
						127	1	1	1	1	1	1	1

Also, the input mode can be changed to BCD by changing the setting of other parameter No. 71.
In the BCD input mode, one digit consists of four bits. Since there are 13 position input bits, the total number of digits assigned to the two axes will become 3.

Assume that the parameters are set as follows:
Other parameter No. 71 = 1 (BCD) “Position-number input mode specification”
Other parameter No. 72 = 8 “Specification of position-number input bits for axis 1”
(Bits are input in units of four.)
Each position number input for axis 1 is assigned to eight bits of PC1 through 8 (010 through 017), as a two-digit BCD code (position Nos. 1 to 99 can be specified). Specify the one’s place in PC1 through 4, and ten’s place in PC5 through 8.
Each position number input for axis 2 is assigned to five bits (actually four bits) of PC9 through 13 (011 through 022), as a one-digit BCD code (position Nos. 1 to 9 can be specified).

■ Axis 1 start (CSTR1)

Axis 1 will start moving to the position corresponding to the specified position data for axis 1 upon detection of the OFF → ON leading edge of this signal. Position numbers are specified using, among the 13 bits of PC1 through 13, the number of bits set in other parameter No. 72. Position numbers are specified as binary codes according to the factory setting.
Before movement is started, the target position, speed and acceleration/deceleration must be set as position data. Use a PC (software) or teaching pendant to set position data.
If this signal is input when no single home-return operation has been performed after the power was input (= when the HEND output signal is OFF), “C6F, Home-return incomplete error” will generate.

■ Axis 2 start (CSTR2)

Axis 2 will start moving to the position corresponding to the specified position data for axis 2 upon detection of the OFF → ON leading edge of this signal. Position numbers are specified using, among the 13 bits of PC1 through 13, the remainder of the bits excluding those used for axis 1. Other specifications are the same as those explained under “Start 1 (CSTR1).”

■ Axis 1 pause (*STP1)

If this signal turns OFF while the actuator is moving, the controller will cause the actuator to decelerate to a stop.
The remaining travel distance will be held, which means that when the signal turns ON again, the actuator will resume movement of the remaining travel distance.
To cancel the movement command altogether after turning OFF the pause signal, turn ON the cancellation signal while this signal is OFF to cancel the remaining travel distance.
The pause signal can be used for the following purposes:

[1] As a sensor to detect entry into a specified area around the system or for other lower-level safety measures to stop the axis while the servo is on
[2] To prevent contact with other equipment
[3] For positioning based on sensor or LS signal detection

(Note) When this signal is input during home return, the movement command will be held if the actuator has not yet contacted the mechanical end. If the signal is input after the actuator has reversed upon contacting the mechanical end, home return will be performed again.

■ Axis 2 pause (*STP2)

Axis 2 will decelerate to a stop if this signal turns OFF while axis 2 is moving.
Other details of this signal are the same as those explained under “Axis 1 pause (*STP1),” except that CANC2 is used to cancel the movement command.

■ Axis 1 cancellation (*CANC1)

Axis 1 will decelerate to a stop if this signal turns OFF while axis 1 is moving. The remaining travel will be cancelled and the axis movement will not resume even after the signal turns ON again.

■ Axis 2 cancellation (*CANC2)

Axis 2 will decelerate to a stop if this signal turns OFF while axis 2 is moving. The remaining travel will be cancelled and the axis movement will not resume even after the signal turns ON again.

■ Axis 1 home return (HOME1)

Axis 1 will stat home-return operation upon detection of the OFF → ON edge of this signal.
Once the home return is complete, the HEND1 signal will be output. This signal can be input as many times as desired after completion of the initial home return.

(Note) An actuator of incremental specification must always perform home return after the power is turned on.

■ Axis 2 home return (HOME 2)

Axis 2 will start home-return operation upon detection of the OFF → ON edge of this signal.
Once the home return is complete, the HEND2 signal will be output.
Other details are the same as those explained under “Axis 1 home return (HOME1).”

■ Axis 1 servo ON (SON1)

The servo for axis 1 will remain ON while this signal is ON.
To operate the actuator using the start input/home return input, the servo ON input signal must be ON. If the servo ON input signal is OFF, these operation commands will not be accepted. (Only the commands will be ignored, and no error will generate.)

(Note) When this signal turns OFF while the actuator is moving, the actuator will not decelerate to a stop. It will complete the movement to the target position, after which the servo will turn off.

■ Axis 2 servo ON (SON2)

The servo for axis 2 will remain ON while this signal is ON.
Other details are the same as those explained under “Axis 1 servo ON (SON1).”

■ Error reset (RES)

[1] This signal is used to reset the alarm output signal (*ALM) that has been generated due to an error.
If an error occurred, check the content of the error and then turn this signal ON.
The error will be reset upon detection of the leading edge of the signal.

(Note) Depending on the nature of error, some errors cannot be reset using this signal. For details, refer to 10, “Troubleshooting.”
Errors of cold start and higher level cannot be reset using this signal. The power must be reconnected to reset these errors. For details, refer to Appendix, “Error Level Management.”

4. Details of Each Output Signal

■ Axis 1 positioning complete (PEND1)

This signal indicates that axis 1 reached the target position and the positioning has completed.
Use it together with the aforementioned MOVE signal to determine the positioning completion status on the PLC side.
After the power was input and the servo has turned on, this signal will turn ON if the position deviation is within the in-position band when the controller becomes ready.
Thereafter, this signal will turn OFF when the start signal is turned ON to execute a movement command.
The signal will turn ON if the position deviation from the target position is within the in-position band after the start signal has turned OFF.
Once this signal turns ON, it will not turn OFF even after the position deviation subsequently exceeds the in-position band.

(Note) If the start signal is ON, this signal will not turn ON even when the position deviation from the target position falls within the in-position band. The signal will turn ON after the start signal turns OFF.
Even if the motor is stopped, this signal will remain OFF if a pause signal is input or the servo is off.

■ Axis 2 positioning complete (PEND2)

This signal indicates that axis 2 reached the target position and the positioning has completed.
Other details are the same as those explained under “Axis 1 positioning complete (PEND1).”

■ Axis 1 home return complete (HEND1)

This signal is OFF when the power is input.
It will turn ON in the following conditions:
[1] When the home-return operation initiated by the start signal for the first movement command has completed.
[2] When the home-return operation initiated by input of the home-return 1 signal has completed.
Once this signal turns ON, it will not turn OFF until the input power is cut off or the home-return 1 signal is input again.

■ Axis 2 home return complete (HEND2)

This signal is OFF when the power is input.
It will turn ON in the following conditions:
[1] When the home-return operation initiated by the start signal for the second movement command for axis 2 has completed.
[2] When the home-return operation of axis 2 initiated by input of the axis 2 home-return signal (HOME2) has completed.
Once this signal turns ON, it will not turn OFF until the input power is cut off or the axis 2 home-return signal (HOME2) is input again.

■ Alarm (*ALM)

This signal remains ON while the controller is normal, and will turn OFF if an alarm occurs.
This signal will turn OFF when an error of operation-cancellation level or higher generates.
Program the PLC so that it will monitor this signal and implement appropriate safety measures to protect the entire system when the signal turns OFF.
For details on alarms, refer to Appendix “⊙ Error Level Management” and “⊙ Error List.”

■ Ready (RDY)

This signal will turn ON when the initialization has completed successfully after the main power was input, and the controller enters the mode where it can control the actuator.
This signal will turn OFF when an error of cold level or higher generates.
Use this signal as a condition to start control on the PLC side.

■ Servo ON output 1 (SVON1)

This signal will turn ON when the servo for axis 1 turns on. Issue a movement command after the servo ON output signal has turned ON.

■ Servo ON output 2 (SVON2)

This signal will turn ON when the servo for axis 2 turns on. Issue a movement command after the servo ON output signal has turned ON.

5. Timing Chart

5.1 Recognition of I/O Signals

An input time constant is set for the input signals of this controller to prevent malfunction due to chattering, noise, etc.
Except for certain signals, the input signal will switch if the new signal level has remained for at least 6 [msec].
For example, when an input signal is turned ON, the controller will recognize that the signal is ON after elapse of 6 [msec]. The same applies when the signal is turned OFF. (Fig. 1)

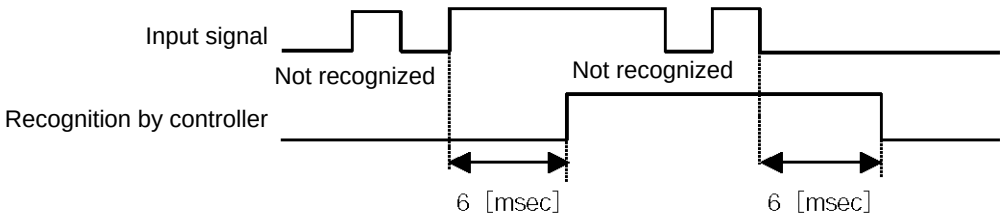
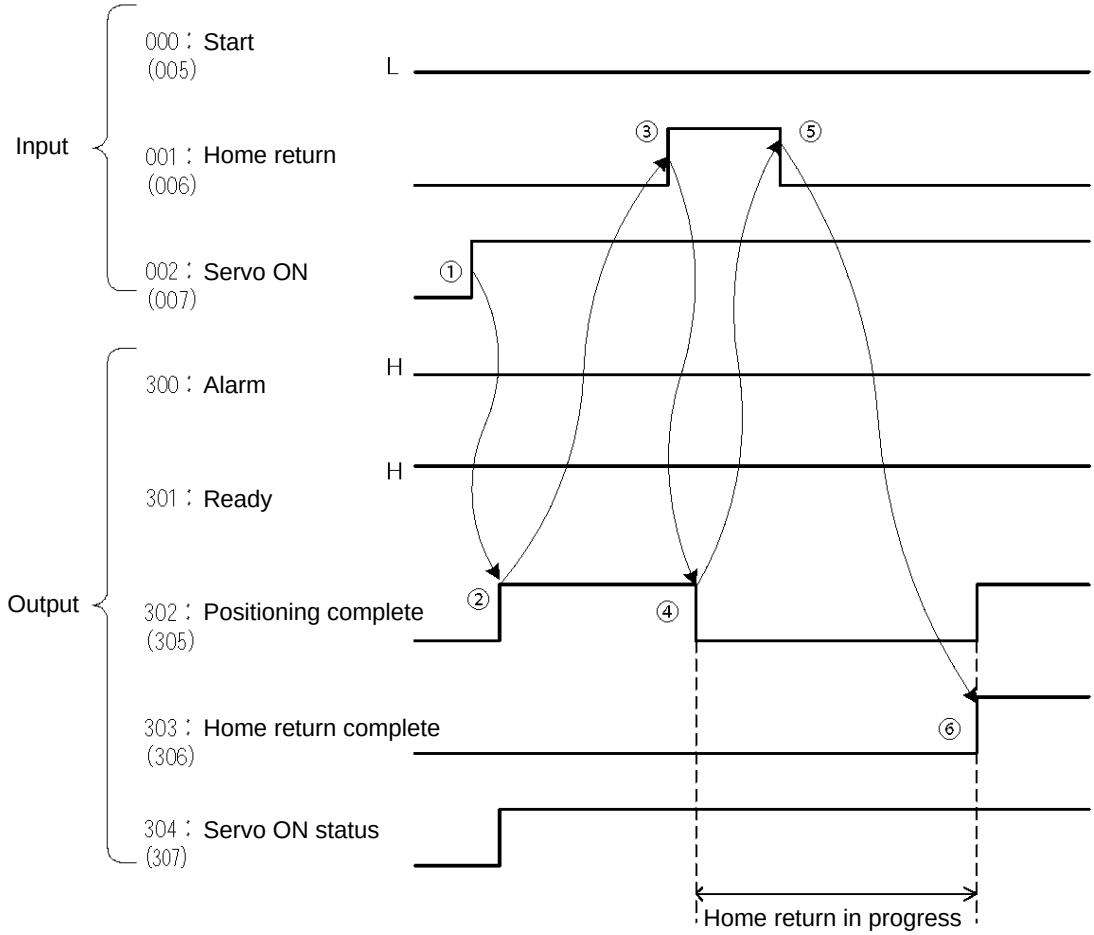


Fig. 1 Recognition of Input Signal

5.2 Home Return

Timings associated with home-return operation are illustrated below. The figures in parentheses indicate port numbers for axis 2.



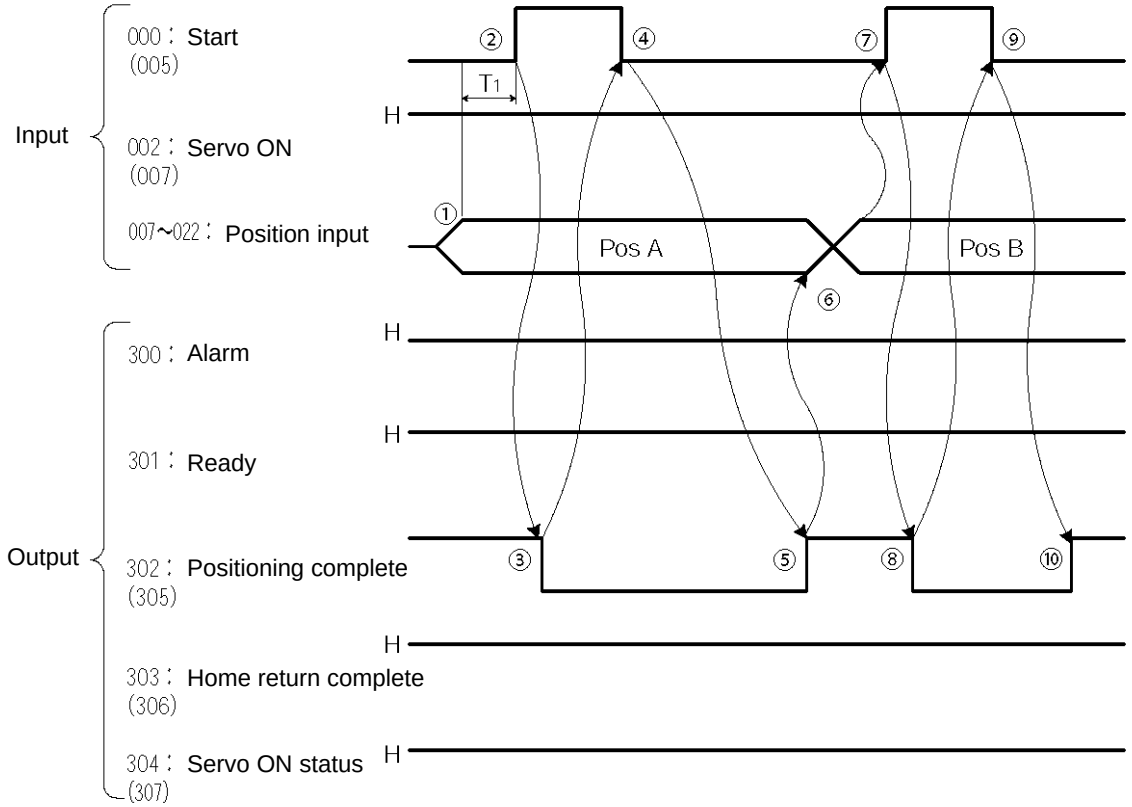
Timing Chart of Home-return Operation (Standard Positioner Mode)

Perform home-return operation by following the procedure explained below.
* Before commencing the procedure, confirm that the ready output signal and alarm output signal are OFF.
  [1] Turn ON the servo ON input signal.
  [2] Confirm that the servo-ON status output signal is ON.
  [3] Turn ON the home-return input signal.
  [4] Confirm that the positioning complete output signal is OFF.
  [5] Turn OFF the home-return input signal.
  [6] Confirm that the home-return complete output signal is ON. Home return is now completed.

*Pause and *cancellation inputs are contact-B input signals (always ON), so keep these signals ON while home return is in progress.
To initiate home return using the home-return signal input, the servo ON input signal must be ON. These operation commands will not be accepted if the servo ON input signal is OFF. Note, however, that only the commands will be ignored and no error will generate.

5.3 Movements through Positions

Timings of how the actuator moves through positions are illustrated below. The figures in parentheses indicate port numbers for axis 2.



Timing Chart of Movement through Positions (Standard Positioner Mode)
Ti: At least 6 msec

Operate the actuator to move through positions by following the procedure explained below.

* Confirm beforehand that the positioning complete output signal, home-return complete output signal and servo-ON status output signal are all ON.
  [1] Change the previous position number input (BCD input) to a different position number .
  [2] Turn ON the start input signal.
  [3] Confirm that the positioning complete output signal is OFF.
  [4] Turn OFF the start input signal.
  [5] Confirm that the positioning complete output signal is ON.
  Repeat steps [1] through [5] sequentially.
* Pause and *cancellation inputs are contact-B input signals (always ON), so keep these signals ON while the actuator are moving through the specified positions.
* While the actuator is moving to the target position, only the pause or cancellation input is accepted. The servo cannot be turned off even if the servo ON input signal is turned OFF. (The servo can be turned off only when the positioning complete output signal is ON.)
* While the start input signal is ON, the positioning complete output signal will not turn ON even after the actuator physically completes moving to the target position. Therefore, always turn OFF the start input signal ([4]) to detect the completion of positioning.
* As for the positioning complete output signal and push-motion complete output signal, they will not be output until the start signal turns OFF (based on the I/O control handshake rules).
* For the actuator to operate upon start signal input, the servo ON input signal must be ON. If the servo ON input signal is OFF, these operation commands will not be accepted. Note, however, that only the commands will be ignored and no error will generate.

Chapter 5 Teaching Mode

In addition to normal positioning operation, jogging, inching and teaching can be performed in this mode. A dedicated input is used to switch to the teaching mode, where the actuator can be moved using I/Os and the achieved position can be written to the position data table.

Caution: Position data input via teaching will be lost when the power is turned off. To retain the position data, one of the following measures must be taken:

- Install the optional system-memory backup battery to back up the position data. To do this, the setting of other parameter No. 20 must be changed to “2.”
  Note, however, that the position data may still be lost if the battery voltage drops.
  (The battery should be replaced after approx. five years.)
  If the battery is replaced as soon as a voltage-low warning generates, the data will be retained.
  Once a voltage-low error generates, the data will be lost.
  Use the host PLC, etc., to monitor for a system-memory backup error output.
- Write the position data to the flash memory using a teaching pendant or PC (software).

1. I/O Interface List

Pin No.	Category	Port No.	Signal name	Signal symbol	Function overview	Cable color
1A	P24		External power supply 24 V	P24		1-Brown
1B	Input	016	Axis 1 jog-	JOG1-	Axis 1 will move in the negative direction while this signal is ON.	1-Red
2A		017	Axis 2 jog+	JOG2+	Axis 2 will move in the positive direction while this signal is ON.	1-Orange
2B		018	Axis 2 jog-	KPG2-	Axis 2 will move in the negative direction while this signal is ON.	1-Yellow
3A		019	Inching (0.01 mm)	1C001	“0.01 mm” is specified as the inching distance.	1-Green
3B		020	Inching (0.1 mm)	1C01	“0.1 mm” is specified as the inching distance.	1-Blue
4A		021	Inching (0.5 mm)	1C05	“0.5 mm” is specified as the inching distance.	1-Purple
4B		022	Inching (1 mm)	1C1	“1 mm” is specified as the inching distance.	1-Gray
5A		023	Error reset	RES	Present alarms will be reset at the leading edge of this signal.	1-White
5B		000	Start	CSTR	The actuator will start moving at the leading edge of this signal.	1-Black
			Current position write	PWRT	The current position is written in the teaching mode.	
6A		001	Servo ON	SON	The servo will remain on while this signal is ON, and remain off while this signal is OFF.	2-Brown
6B		002	*Pause	*STP	The actuator can be moved when this signal is ON, and will decelerate to a stop when the signal turns OFF.	2-Red
7A		003	Position input 1	PC1	Input the position number corresponding to the position you want to move the actuator to. Be sure to specify a position input by no later than 6 msec before the start input signal turns ON. Position numbers are input as binary codes (factory setting). In the teaching mode, specify the position number for which the current position will be written. Position numbers are input as binary codes (factory setting).	2-Orange
7B		004	Position input 2	PC2		2-Yellow
8A		005	Position input 3	PC3		2-Green
8B		006	Position input 4	PC4		2-Blue
9A		007	Position input 5	PC5		2-Purple
9B		008	Position input 6	PC6		2-Gray
10A		009	Position input 7	PC7		2-White
10B		010	Position input 8	PC8		2-Black
11A		011	Position input 9	PC9		3-Brown
11B		012	Position input 10	PC10		3-Red
12A		013	Position input 11	PC11		3-Orange
12B		014	Teaching mode specification	MODE	ON: Teaching mode OFF: Positioner mode	3-Yellow
13A		015	Axis 1 jog-	JOG1+	Axis 1 will move in the positive direction while this signal is ON.	3-Green
13B	Output	300	*Alarm	*ALM	This signal remains ON if the controller is normal. It will turn OFF if an alarm occurs.	3-Blue
14A		301	Ready	RDY	This signal will turn ON when the controller becomes ready.	3-Purple
14B		302	Positioning complete	PEND	This signal will turn ON once the actuator has moved to the target position and entered the positioning band.	3-Gray
			Write complete	WEND	This signal will turn ON when writing of position data is completed.	
15A		303	Home return complete	HEND	This signal is OFF when the power is input, and will turn ON when home return is completed.	3-White
15B		304	Servo ON output	SVON	This signal will turn ON when the servo is turned on, and turn OFF when the servo is turned off.	3-Black
16A		305	Teaching mode output	TCMD	This signal will remain ON during the teaching mode.	4-Brown
16B		306	System battery error	SSER	This signal will turn ON when the voltage of the system-memory backup battery drops to the voltage-low warning level.	4-Red
17A		307				4-Orange
17B	N		External power supply 0 V	N		4-Yellow

*: Contact B (always ON)

2. Parameters

To use the controller in the teaching mode, set other parameter No. 25 to “4.”
Position numbers are specified as binary codes according to the factory setting. To change the input mode to BCD, set a value “other than 0” in other parameter No. 25.

	No.	Parameter	Function
Other	25	Operation mode type	4: Teaching mode
	71	Positioner mode parameter 1	Position-number input mode specification (0: Binary, ≠ 0: BCD) * Default value: 0 (Binary)

3. Details of Each Input Signal

■ Start (CSTR)

When the OFF → ON leading edge of this signal is detected, the controller will load the target point number specified by the 13-bit binary code consisting of PC1 through PC13, and perform positioning to the target position specified by the corresponding position data.
Before movement is started, the target position, speed and acceleration/deceleration operation data must be set in the position table using a PC or teaching pendant.
If this signal is input when no single home-return operation has been performed after the power was input (= when the HEND output signal is OFF), the actuator will perform home-return operation.

■ Position inputs 1 through 11 (PC1 through PC11)

When a movement command is executed upon the OFF → ON edge of the start signal, the controller will load the command position number specified by the 11-bit binary code consisting of signals PC1 through PC11.
The weight of each bit is as follows: 2^0 for PC1, 2^1 for PC2, 2^2 for PC3, 2^3 for PC4, ..., and 2^{10} for PC11.
By combining these bits, any position number between 0 and 1500 (maximum) can be specified.
In the teaching mode, specify the position number for which the current position will be written.
When the PWRT input signal is turned ON, the current position will be written to the position number specified by the binary code.
Also, the input mode can be changed to BCD by changing the setting of other parameter No. 71, as follows:
Other parameter No. 71 = 1 (other than 0) (BCD input)
(Default setting of other parameter No. 71 = 0 (Binary input))
In the BCD input mode, specify the one’s place in PC1 through 4, and ten’s place in PC5 through 8 (position Nos. 1 to 99 can be specified).

■ Pause (*STP)

If this signal turns OFF while the actuator is moving, the controller will cause the actuator to decelerate to a stop.
The remaining travel distance will be held, which means that when the signal turns ON again, the actuator will resume movement of the remaining travel distance.
To cancel the movement command altogether after turning OFF the pause signal, turn ON the alarm reset signal while this signal is OFF to cancel the remaining travel distance.
The pause signal can be used for the following purposes:
[1] As a sensor to detect entry into a specified area around the system or for other lower-level safety measures to stop the axis while the servo is on
[2] To prevent contact with other equipment
[3] For positioning based on sensor or LS signal detection
(Note) When this signal is input during home return, the movement command will be held if the actuator has not yet contacted the mechanical end. If the signal is input after the actuator has reversed upon contacting the mechanical end, home return will be performed again.

■ Servo ON (SON)

The servo remains on while this signal is ON.
Use this signal if servo ON/OFF control is required as part of the safety circuit for the entire system to be provided on the PLC side.
To operate the actuator using the start input/jog input, the servo ON input signal must be ON. If the servo ON input signal is OFF, these operation commands will not be accepted. (Only the commands will be ignored, and no error will generate.)
(Note) When this signal turns OFF while the actuator is moving, the actuator will not decelerate to a stop. It will complete the movement to the target position, after which the servo will turn off.

■ Error reset (RES)

This signal is used to reset the alarm output signal (*ALM) that has been generated due to an error.
If an error occurred, check the content of the error and then turn this signal ON.
The error will be reset upon detection of the leading edge of the signal.
(Note) Errors of cold start and higher level cannot be reset using this signal. The power must be reconnected to reset these errors. For details, refer to Appendix, “Error Level Management.”

■ Teaching mode specification (MODE)

When this signal turns ON, the normal positioning mode will change to the teaching mode. When the new mode becomes effective, the TCMD output signal will turn ON.
Program the PLC so that it will accept PWRT/JOG1+ and other operation commands after confirming that the TCMD output signal is ON.
To return the controller to the normal positioning mode, turn this signal OFF.
Program the PLC so that it will accept operation commands in the normal positioning mode after confirming that the TCMD output signal is OFF.
The controller will not return to the positioning mode right away when this signal is turned OFF while the actuator is jogging. It will not immediately stop the actuator, either. The controller will complete the movement first, and then return to the positioning mode.
Exercise caution because the actuator will start moving if this signal is turned ON when the servo is on in the positioning mode while any jog input signal (JOG1+, JOG1-, etc.) is also ON.

■ Current position write (PWRT)

This signal is effective when the aforementioned TCMD output signal is ON.
If this signal has remained on for at least 20 msec, the controller will load the position number corresponding to the binary code specified by PC1 through PC11 as currently detected, and write the current position data in the corresponding target position field of the position data table.
If any of the data fields other than the target position (such as speed, acceleration/deceleration and positioning band) is not yet defined, the default value of the applicable parameter (all-axis parameter Nos. 11, 12 or 13) will be written in that field.
When the data write is successfully completed, the WEND output signal will turn ON.
Program the PLC so that it will turn this signal OFF once the WEND signal turns ON. When this signal turns OFF, the controller will turn OFF the WEND signal.
(Note) An error will generate if position data is written before home return is completed. Position data cannot be written while the actuator is jogging.

■ Axis 1 jog (JOG1+, JOG1-)

These signals are effective when the aforementioned MODES output signal is ON.
The actuator of axis 1 will move to the + or - soft limit position upon detection of the OFF → ON leading edge of each signal.
Although the actuator will be forcibly decelerated to a stop after reaching the soft limit, no alarm will generate.
The speed and acceleration/deceleration to be used are the values set in user parameter No. 26 (PIO jog speed) and No. 9 (Default acceleration/deceleration).
If both the JOG+ and JOG- signals turn ON at the same time, the actuator will move to the direction corresponding to the signal that was input first.
The actuator will decelerate to a stop upon detection of the ON → OFF trailing edge of the signal while the actuator is moving.
(Note) Exercise due caution not to perform jogging before home return is complete, because the soft limits are still invalid and the actuator may collide with the mechanical end.

■ Inching (IN001 through 1)

These signals are used to specify the inching distance for inching operation performed in the teaching mode.
The four bits of IN001 through 1 indicate different inching distances, as follows:
IN001: 0.01 mm, IN01: 0.1 mm, IN05: 0.5 mm, IN1: 1 mm
The actuator will perform inching operation when a jog movement command is input while the bit or bits corresponding to a given inching distance is/are ON (if all four bits are OFF, the actuator will jog).
When multiple bits are turned ON, the sum of the distances represented by the applicable bits will become the inching distance.

4. Details of Each Output Signal

■ Positioning complete (PEND)

This signal indicates that the actuator reached the target position and the positioning has completed.
The signal will turn ON when the servo has turned on after the main power was input, and the controller becomes ready.
Thereafter, this signal will turn OFF when the start signal is turned ON to execute a movement command.
The signal will turn ON if the position deviation from the target position is within the in-position band after the start signal has turned OFF.
Once this signal turns ON, it will not turn OFF even after the position deviation subsequently exceeds the in-position band.

(Note) If the start signal is ON, this signal will not turn ON even when the position deviation from the target position falls within the in-position band. The signal will turn ON after the start signal turns OFF.
The signal will remain OFF while the servo is off.

■ Home return complete (HEND)

This signal is OFF when the power is input. It will turn ON upon completion of home return (if the actuator is of incremental specification).
To perform home return, specify a desired position number, and then turn ON the start input signal.
Use this signal as a condition for moving the actuator and also for writing the current position in the teaching mode.

(Note) Actuators of incremental specification must always perform home return after the power is input.
In the teaching mode, the actuator can be jogged before it completes home return, but the soft limits are still ineffective. Since coordinate values have no meaning in this condition, exercise due caution not to let the actuator contact the stroke end.

Once this signal turns ON, it will not turn OFF until the input power is cut off or the home-return signal is input again.

■ Teaching mode specification (MODES)

This signal will turn ON when the teaching mode was selected by the teaching mode input signal (turning ON the MODE signal) and the teaching mode has become effective.
Thereafter, this signal will remain ON until the MODE signal turns OFF.
Program the PLC so that it will start teaching operation after confirming that this signal has turned ON.

■ Write complete (WEND)

This signal is effective only in the teaching mode.
The signal is OFF immediately after the controller has entered the teaching mode, and will turn ON upon completion of the position data write initiated by the current position write signal.
When the current position write signal turns OFF thereafter, this signal will also turn OFF.
Program the PLC so that it will recognize completion of write operation upon turning OFF of this signal.

■ Alarm (*ALM)

This signal remains ON while the controller is normal, and will turn OFF if an alarm occurs.
This signal will turn OFF when an error of operation-cancellation level or higher generates.
Program the PLC so that it will monitor this signal and implement appropriate safety measures to protect the entire system when the signal turns OFF.
For details on alarms, refer to Appendix “⊙ Error Level Management” and “⊙ Error List.”

■ Ready (RDY)

This signal will turn ON when the initialization has completed successfully after the main power was input, and the controller enters the mode where it can control the actuator.
This signal will turn OFF when an error of cold level or higher generates.
Use this signal as a condition to start control on the PLC side.

■ Servo ON output (SVON)

This signal will turn ON when the servo turns on. Issue a movement command after the servo ON output signal has turned ON.

■ System battery error

This signal will turn ON when the voltage of the optional system-memory backup battery drops to a specified level.

5. Timing Chart

5.1 Recognition of I/O Signals

An input time constant is set for the input signals of this controller to prevent malfunction due to chattering, noise, etc.
Except for certain signals, the input signal will switch if the new signal level has remained for at least 6 [msec].
For example, when an input signal is turned ON, the controller will recognize that the signal is ON after elapse of 6 [msec]. The same applies when the signal is turned OFF. (Fig. 1)

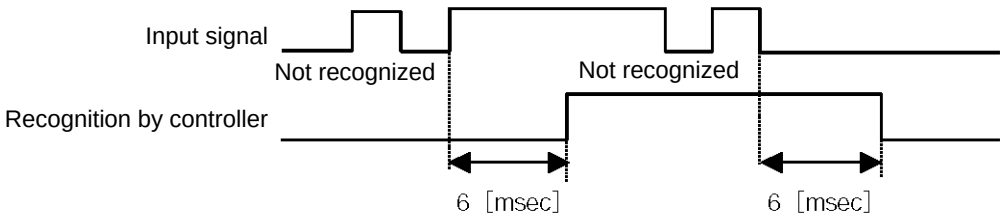
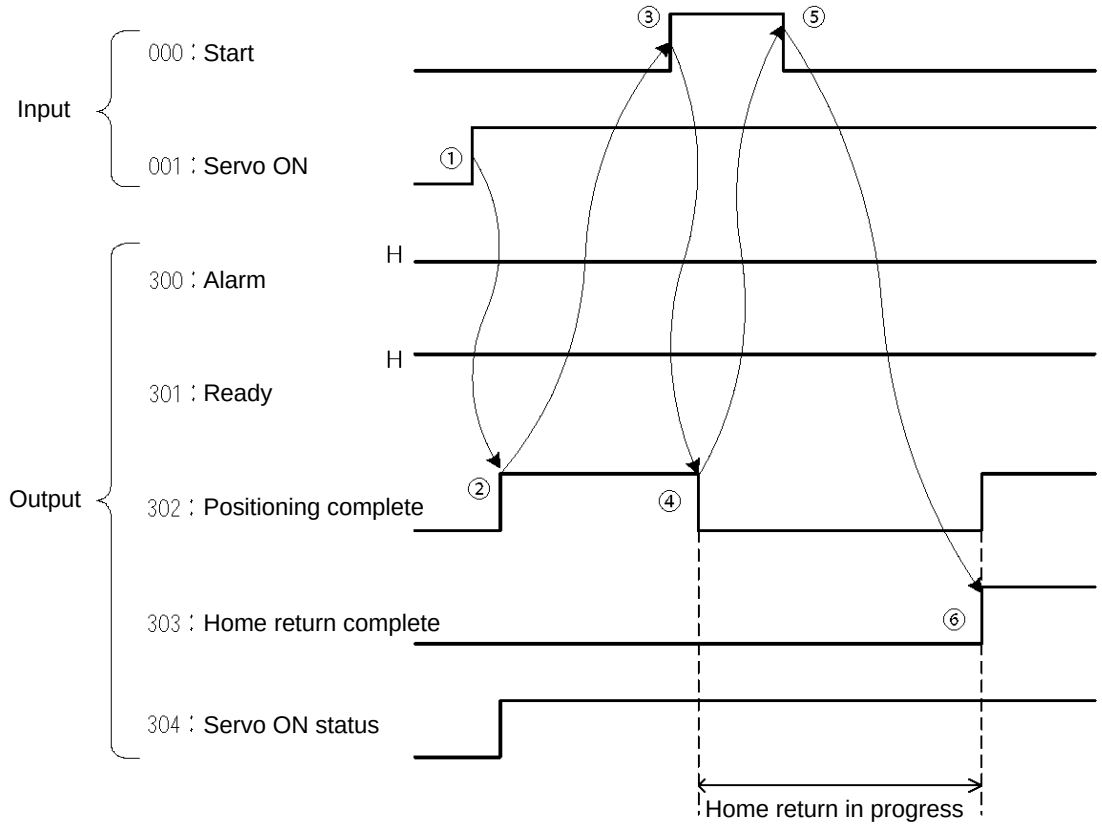


Fig. 1 Recognition of Input Signal

5.2 Home Return

In the teaching mode, no dedicated home-return input is available.
Home return will be performed when the start signal is input after specifying a desired position in a condition where home return is not yet completed.
Timings associated with home-return operation are illustrated below.



Timing Chart of Home-return Operation (Teaching Positioner Mode)

Perform home-return operation by following the procedure explained below.

* Before commencing the procedure, confirm that the ready output signal is ON, alarm output signal is OFF, and home-return complete output signal is OFF.

[1] Turn ON the servo ON input signal.
[2] Confirm that the servo-ON status output signal is ON.
[3] Turn ON the start input signal.
[4] Confirm that the positioning complete output signal is OFF.
[5] Turn OFF the start input signal.
[6] Confirm that the home-return complete output signal is ON. Home return is now completed.

* Pause input is a contact-B input signal (always ON), so keep this signal ON while home return is in progress.

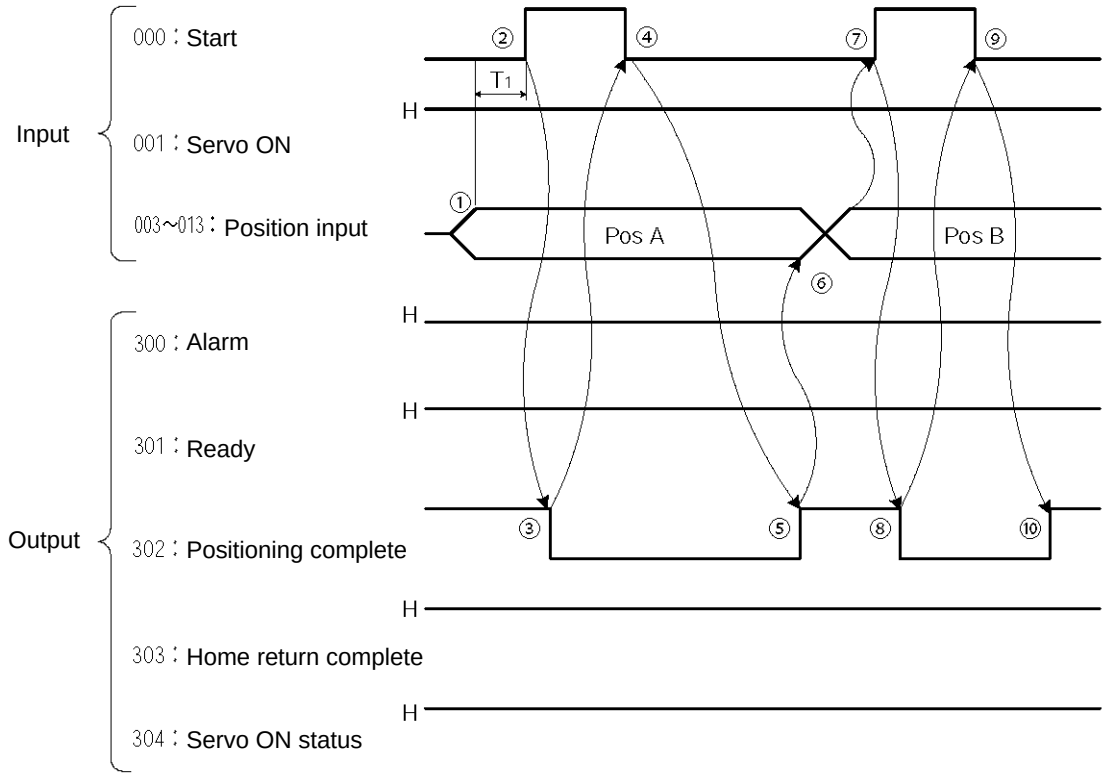
To operate the actuator using the start input, the servo ON input signal must be ON. If the servo ON input signal is OFF, this operation command will not be accepted. Note, however, that only the command will be ignored and no error will generate.

With the 2-axis specification, the controller has been configured at the factory so that the two axes will start home return simultaneously.
You can cause either axis to start home return earlier than the other axis by changing the applicable parameter setting.
Specifically, change the setting in axis-specific parameter No. 13, “SIO/PIO home-return order” so that the parameter value for the axis number corresponding to the axis for which you want to complete home return first, will become smaller than the parameter value for the other axis number.

Example) Cause axis 1 to perform home return after axis 2 has completed home return, set “1” for axis 1 and “0” for axis 2 in axis-specific parameter No. 13.

5.3 Movements through Positions

Timings of how the actuator moves through positions are illustrated below.



Timing Chart of Movement through Positions (Standard Positioner Mode)
Ti: At least 6 msec

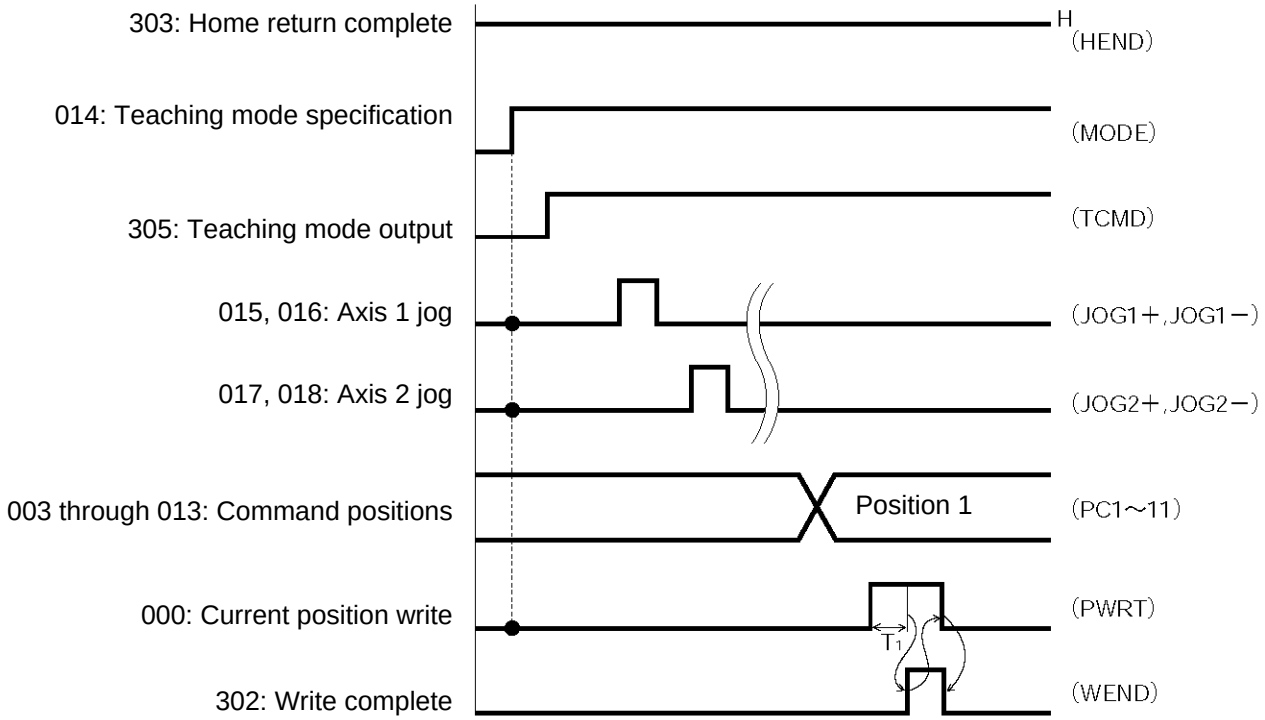
Operate the actuator to move through positions by following the procedure explained below.

* Confirm beforehand that the positioning complete output signal, home-return complete output signal and servo-ON status output signal are all ON.
  [1] Change the previous position number input to a different position number.
  [2] Turn ON the start input signal.
  [3] Confirm that the positioning complete output signal is OFF.
  [4] Turn OFF the start input signal.
  [5] Confirm that the positioning complete output signal is ON.
  Repeat steps [1] through [5] sequentially.
* Pause input is a contact-B input signal (always ON), so keep this signal ON while home return is in progress.
* While the actuator is moving to the target position, only the pause or cancellation input is accepted. The servo cannot be turned off even if the servo ON input signal is turned OFF. (The servo can be turned off only when the positioning complete output signal is ON.)
* While the start input signal is ON, the positioning complete output signal will not turn ON even after the actuator physically completes moving to the target position. Therefore, always turn OFF the start input signal ([4]) to detect the completion of positioning.

* As for the positioning complete output signal and push-motion complete output signal, they will not be output until the start signal turns OFF (based on the I/O control handshake rules).

* For the actuator to operate upon start signal input, the servo ON input signal must be ON. If the servo ON input signal is OFF, these operation commands will not be accepted. Note, however, that only the commands will be ignored and no error will generate.

5.4 Timings in the Teaching Mode



T1: At least 20 msec. T1 represents the time after the position-information write input signal turns ON, until writing of the current position starts.

When the teaching mode specification (MODE) input signal is turned ON, the teaching mode (TCMD) output signal will turn ON. The controller will enter the teaching mode and jogging/teaching via PIOs will become possible.
To confirm if the controller is in the teaching mode, check if the TCMD signal is ON.
If both the jog+ and jog- input signals turn ON at the same time, the actuator will move to the position corresponding to the signal that was input first.
*Pause signal is a contact-B input signal (always ON), so keep this signal ON while teaching is in progress.
To perform inching, specify a desired inching distance (IC001 through 1) before the jog command is input.
If the current position write (PWRT) input signal has remained ON for at least 20 msec, the current actuator position will be written to the selected command position number.
Once the data write is complete, the write complete (WEND) output signal will turn ON. To confirm if the controller has finished writing data, check if the WEND signal is ON.
When the PWRT input signal turns OFF, the WEND output signal will turn OFF.

If the position table screen is open on the PC or teaching pendant, inputting a write signal from the PLC will not update the position data display. To check the acquired position data, do one of the following operations: PC --- Click the button. Teaching pendant --- Turn the PORT switch from OFF to ON.

Chapter 6 DS-S-C1 Compatible Mode

In this mode, the same I/O assignments used by the conventional controller model DS-S-C1 are used. As added functions, the cancellation (CANC) input, interpolation setting input, system battery error output, and absolute battery error output are available, and the number of positions has been increased.

1. I/O Interface List

Pin No.	Category	Port No.	Signal name	Signal symbol	Function overview	Cable color
1A	P24		External power supply 24 V	P24		1-Brown
1B	Input	016	Position No. 1000 input	PC1000	(Same as PC1 through 800)	1-Red
2A		017	-			1-Orange
2B		018	-			1-Yellow
3A		019	-			1-Green
3B		020	-			1-Blue
4A		021	-			1-Purple
4B		022	-			1-Gray
5A		023	CPU reset	CPRES	The CPU will be restarted at the leading edge of this signal.	1-White
5B		000	Start	CSTR	The actuator will start moving at the leading edge of this signal.	1-Black
6A		001	Pause	STP	The actuator can be moved when this signal is ON, and will decelerate to a stop when the signal turns OFF.	2-Brown
6B		002	Cancellation	CANC	The remaining travel distance will be cancelled if this signal turns ON.	2-Red
7A		003	Interpolation setting	LINE	With the 2-axis specification, linear interpolation operation will start when the start input signal is turned ON while this signal is ON.	2-Orange
7B		004	Position No. 1 input	PC1	Input the position number corresponding to the position you want to move the actuator to. Be sure to specify a position input by no later than 6 msec before the start input signal turns ON. Position numbers are input as BCD codes. (PC1 through 8 indicate the one's place, PC10 through 80 indicate the ten's place, PC100 through 800 indicate the hundred's place, and PC1000 indicates the thousand's place.)	2-Yellow
8A		005	Position No. 2 input	PC2		2-Green
8B		006	Position No. 4 input	PC4		2-Blue
9A		007	Position No. 8 input	PC8		2-Purple
9B		008	Position No. 10 input	PC10		2-Gray
10A		009	Position No. 20 input	PC20		2-White
10B		010	Position No. 40 input	PC40		2-Black
11A		011	Position No. 80 input	PC80		3-Brown
11B		012	Position No. 100 input	PC100		3-Red
12A		013	Position No. 200 input	PC200		3-Orange
12B		014	Position No. 400 input	PC400		3-Yellow
13A		015	*Position No. 800 input	PC800		3-Green
13B	Output	300	*Alarm	ALM	This signal remains ON if the controller is normal. It will turn OFF if an alarm occurs.	3-Blue
14A		301	Ready	RDY	This signal will turn ON when the controller becomes ready.	3-Purple
14B		302	Positioning complete	PEND	This signal will turn ON once the actuator has moved to the target position and entered the positioning band.	3-Gray
15A		303	-			3-White
15B		304	-			3-Black
16A		305	-			4-Brown
16B		306	System battery error	SSER	This signal will turn ON when the voltage of the system-memory backup battery drops to the voltage-low warning level.	4-Red
17A		307				4-Orange
17B	N		External power supply 0 V	N		4-Yellow

Caution: The power wiring polarities are reversed from those of the PNP specification applicable to the old DS-S-C1 controller. As shown above, pin Nos. 1A and 17B are connected to 24 V and 0 V, respectively, even in the PNP specification.

2. Parameters

To use the controller in the DS-S-C1 compatible mode, set other parameter No. 25 to “16.”
Other parameter No. 25 = 16, “DS-S-C1 compatible mode”

3. Details of Each Input Signal

■ Start (CSTR)

The actuator will start moving to the position corresponding to the specified position data upon detection of the OFF → ON leading edge of this signal. Position numbers are specified using a 13-bit BCD code consisting of PC1 through 1000.
Before movement is started, the target position, speed and acceleration/deceleration must be set as position data. Use a PC (software) or teaching pendant to set position data.
Turn on the power, specify position No. 0 (PC1 through 1000 are all OFF) and then turn this signal ON, and the actuator will start home return once the servo turns ON.
If a movement command is executed when no single home-return operation has been performed after the power was input, “C6F, Home-return incomplete error” will generate.

■ Position Nos. 1 through 1000 (PC1 through 1000)

When a movement command is executed upon OFF → ON of the start signal, the controller will load the command position number specified by the 13-bit BCD code consisting of PC1 through 1000.
A desired position number between 1 and 1500 can be specified. Specify the one’s place in PC1 through 8, ten’s place in PC10 through 80, hundred’s place in PC100 through 800, and thousand’s place in PC1000.

An example of position number specification based on ON/OFF levels of PC1 through 1000 is shown below.

PC 1000	PC 800	PC 400	PC 200	PC 100	PC 80	PC 40	PC 20	PC 10	PC 8	PC 4	PC 2	PC 1	Position No.
0	0	0	0	0	0	0	0	0	0	0	0	1	1
0	0	0	0	0	0	0	0	0	0	0	1	0	2
0	0	0	0	0	0	0	0	0	0	0	1	1	3
0	0	0	0	0	0	0	0	1	0	0	1	0	12
0	0	1	1	0	0	1	1	1	1	0	0	0	678
1	0	0	1	0	0	0	1	1	0	1	0	0	1234

■ Pause (STP)

If this signal turns ON while the actuator is moving, the controller will cause the actuator to decelerate to a stop.
The remaining travel distance will be held, which means that when the signal turns OFF again, the actuator will resume movement of the remaining travel distance.
To cancel the movement command altogether after turning ON the pause signal, turn ON the cancellation signal while this signal is ON to cancel the remaining travel distance.
The pause signal can be used for the following purposes:

[1] As a sensor to detect entry into a specified area around the system or for other lower-level safety measures to stop the axis while the servo is on
[2] To prevent contact with other equipment
[3] For positioning based on sensor or LS signal detection

(Note) When this signal is input during home return, the movement command will be held if the actuator has not yet contacted the mechanical end. If the signal is input after the actuator has reversed upon contacting the mechanical end, home return will be performed again.

■ Cancellation (CANC)

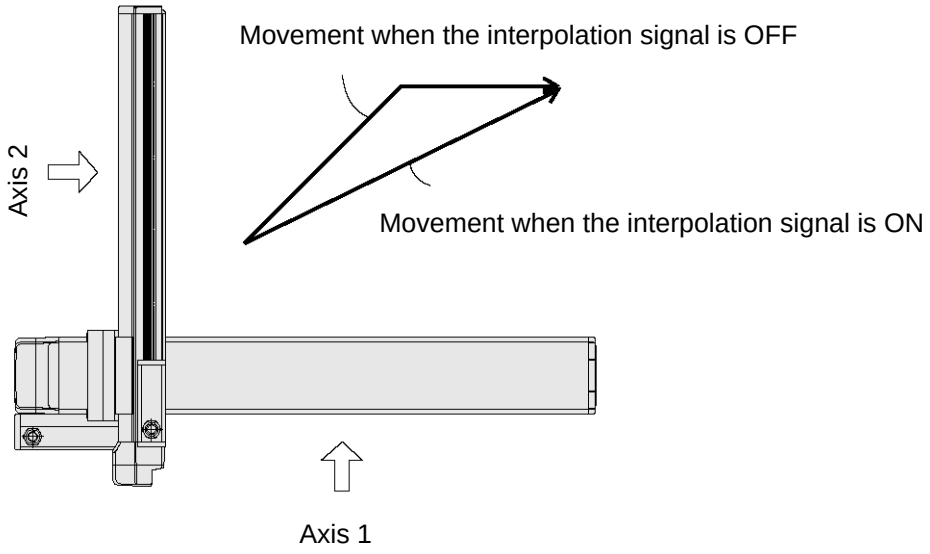
If this signal turns ON while the actuator is moving, the controller will cause the actuator to decelerate to a stop. The remaining travel distance will be cancelled and the movement will not resume even when the signal turns OFF thereafter.

■ CPU reset (CPRES)

This input signal is used to restart the controller.
If an error occurs, identify and eliminate the cause, and then turn this signal ON.

■ Interpolation (LINE)

With the 2-axis specification, input of the position signal and start signal while this signal is ON will cause the two axes to perform interpolation operation (the two axes will start simultaneously and arrive at the target position simultaneously).
To perform interpolation operation, turn ON the interpolation input signal before turning ON the start input signal.



4. Details of Each Output Signal

■ Ready (RDY)

This signal will turn ON when the initialization has completed successfully after the main power was input, and the controller enters the mode where it can control the actuator.
This signal will turn OFF when an error of cold level or higher generates.
Use this signal as a condition to start control on the PLC side.

■ Alarm (*ALM)

This signal remains OFF while the controller is normal, and will turn ON if an alarm occurs.
Program the PLC so that it will monitor this signal and implement appropriate safety measures to protect the entire system when the signal turns ON.
For details on alarms, refer to 10, “Troubleshooting.”

■ Positioning complete (PEND)

This signal indicates that the actuator reached the target position and the positioning has completed.
When a movement command is executed by turning ON the start signal, this signal will turn OFF.
Thereafter, it will turn ON when the position deviation from the target position has entered the in-position band regardless of whether the start signal is ON or OFF.
Once this signal turns ON, it will not turn OFF even after the position deviation subsequently exceeds the in-position band.
(Note) Even if the motor is stopped, this signal will remain OFF if a pause signal is input or the servo is off.
This signal is OFF when the power is input. It will turn ON upon completion of home-return operation (if the actuator is of incremental specification).

■ System battery error

This signal will turn ON when the voltage of the optional system-memory backup battery drops to a specified level.

5. Timing Chart

5.1 Recognition of I/O Signals

An input time constant is set for the input signals of this controller to prevent malfunction due to chattering, noise, etc.
Except for certain signals, the input signal will switch if the new signal level has remained for at least 6 [msec].
For example, when an input signal is turned ON, the controller will recognize that the signal is ON after elapse of 6 [msec]. The same applies when the signal is turned OFF. (Fig. 1)

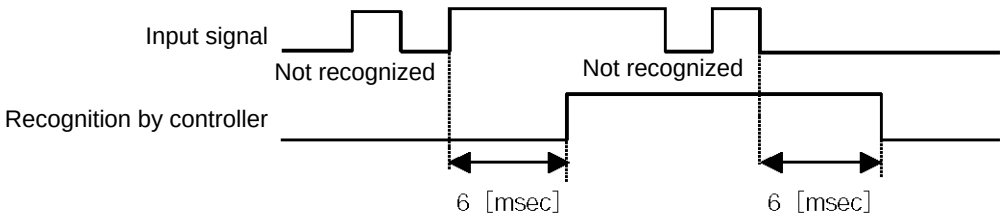
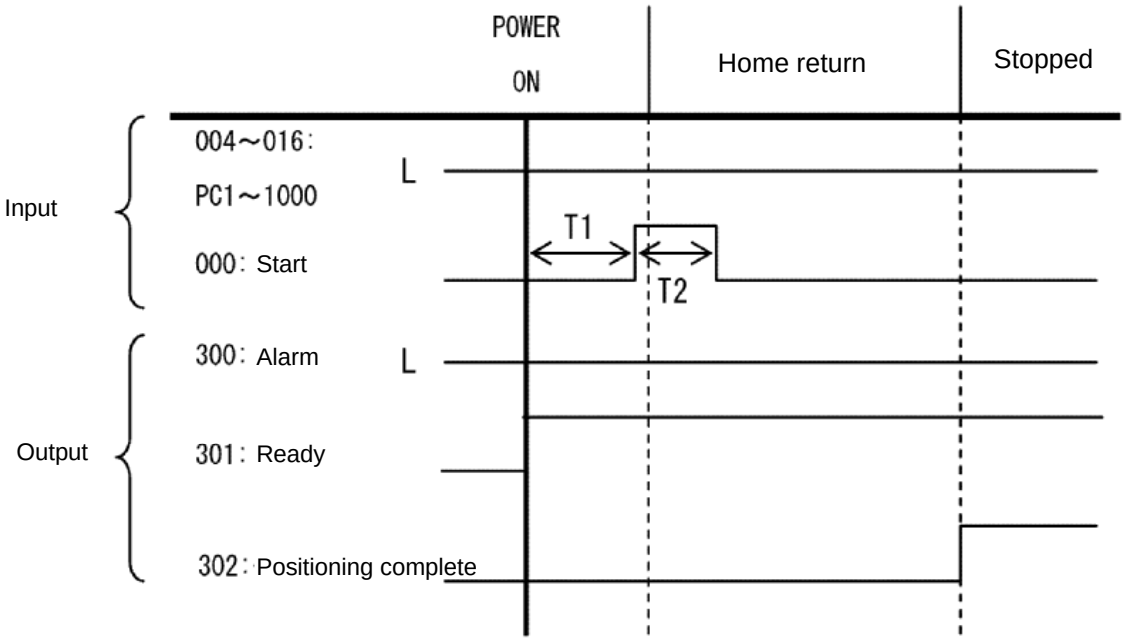


Fig. 1 Recognition of Input Signal

5.2 Home Return

In the DS-S-C1 compatible mode, no dedicated home-return input is available.
Home return will be performed when the start signal is input after specifying position No. 0. The positioning complete output signal is OFF after the power is input when home return is not yet completed.
Timings associated with home-return operation are illustrated below.



T1: Time after the ready output signal turns ON until input of the start signal becomes possible (50 msec or more)
T2: Start signal input (30 msec or more)

Timing Chart of Home-return Operation (Positioner Mode)

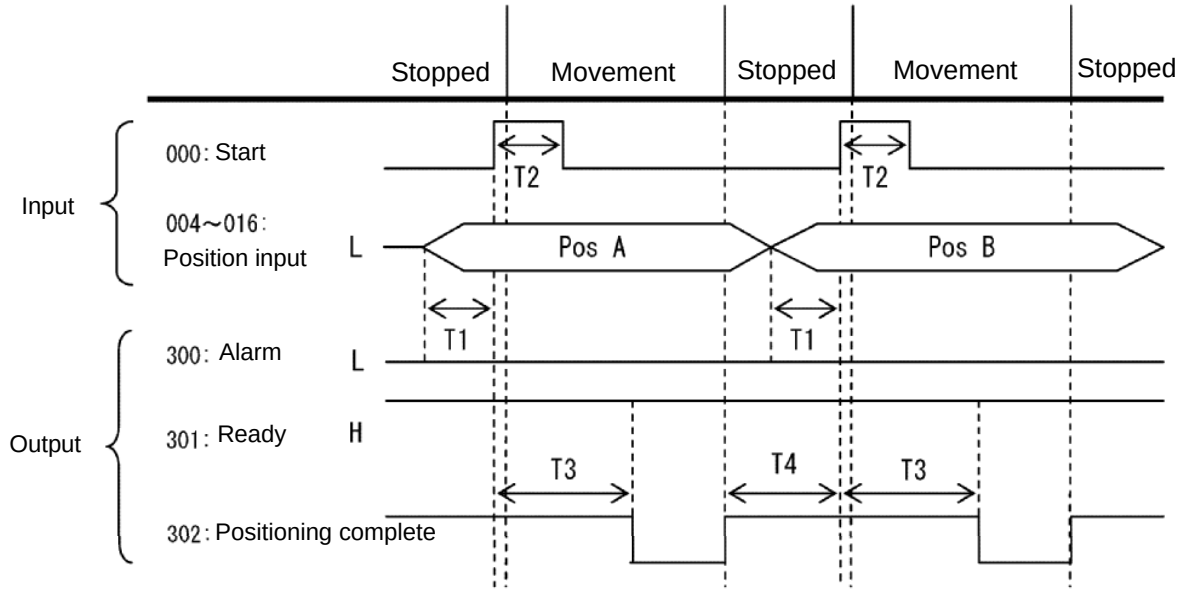
Perform home-return operation by following the procedure explained below.
* Before commencing the procedure, confirm that the ready output signal and alarm output signal are ON.

[1] Specify position No. 0 (PC1 through 1000 are all OFF).
[2] Turn ON the start input signal. (The signal should remain ON continuously for 30 msec or more (T2).)
[3] Turn OFF the start input signal.
[4] Confirm that the positioning complete output signal is ON. Home return is now completed.

With the 2-axis specification, the controller has been configured at the factory so that the two axes will start home return simultaneously.
You can cause either axis to start home return earlier than the other axis by changing the applicable parameter setting.
Specifically, change the setting in axis-specific parameter No. 13, “SIO/PIO home-return order” so that the parameter value for the axis number corresponding to the axis for which you want to complete home return first, will become smaller than the parameter value for the other axis number.
Example) Cause axis 1 to perform home return after axis 2 has completed home return, set “1” for axis 1 and “0” for axis 2 in axis-specific parameter No. 13.

5.3 Movements through Positions

Timings of how the actuator moves through positions are illustrated below.



Timing Chart of Movement through Positions (Positioner Mode)

T1: Time after the position number signal is input until input of the start signal becomes possible (30 msec or more)
T2: Start signal input (30 msec or more)
T3: Time after the start signal turns ON until the positioning complete output signal turns OFF (60 msec or less)
T4: Time after the previous positioning complete output signal turns ON until input of the next start signal becomes possible (50 msec or more)

Operate the actuator to move through positions by following the procedure explained below.

* Confirm beforehand that the positioning complete output signal, home-return complete output signal and servo-ON status output signal are all ON.
  [1] Change the previous position number input (BCD input) to a different position number.
  [2] Turn ON the start input signal. (The signal should remain ON continuously for 30 msec or more (T2).)
  [3] Turn OFF the start input signal.
  [4] Wait for T3 after [2].
  [5] Confirm that the positioning complete output signal is ON.
  Repeat steps [1] through [5] sequentially.
* To perform interpolation operation, turn ON the interpolation setting input signal at least 30 msec before turning ON the start input signal. Turn OFF the interpolation signal after the start input signal has turned OFF.
* The positioning complete output signal turns ON when the actuator completes moving to the specified position, regardless of whether the start input signal is ON or OFF.
* Take note that the time after the start signal turns ON until the positioning complete output signal turns OFF is 60 msec or less, which is different from 15 msec or less with the DS-S-C1 controller.

Caution:	Unlike in other modes, the pause input and cancellation input are contact-A input signals (always OFF). The alarm output is also a contact-A output signal (always OFF) unlike in other modes.

Appendix

⊙ List of Applicable Actuator Specifications

- Slider, ball-screw drive

	Model	Stroke (mm) and maximum speed (mm/sec) (Note 1)	Loading capacity (Note 2) Horizontal (kg)	Loading capacity (Note 2) Vertical (kg)	Rated acceleration Horizontal (G)	Rated acceleration Vertical (G)
Motor-straight	RCP2-SA5C-□-PM-12-□□□	600	4	1	0.3	0.2
	RCP2-SA5C-□-PM-6-□□□	300	8	2.5	0.3	0.2
	RCP2-SA5C-□-PM-3-□□□	150	8	4.5	0.2	0.2
	RCP2-SA6C-□-PM-12-□□□	600 / 540	6	1.5~1	0.3	0.2
	RCP2-SA6C-□-PM-6-□□□	300 / 270	12	3~2.5	0.3	0.2
	RCP2-SA6C-□-PM-3-□□□	150 / 135	12	6~4	0.2	0.2
	RCP2-SA7C-□-PM-16-□□□	533 / 480	35~7	5~0.5	0.3	0.2
	RCP2-SA7C-□-PM-8-□□□	266 / 240	40~10	10~1.5	0.3	0.2
	RCP2-SA7C-□-PM-4-□□□	133 / 120	40	15~5	0.2	0.2
	RCP2-SS7C-□-PM-12-□□□	600 / 470	30~6	4~1	0.3	0.2
	RCP2-SS7C-□-PM-6-□□□	300 / 230	30~20	8~2	0.3	0.2
	RCP2-SS7C-□-PM-3-□□□	150 / 115	30~20	12~4	0.2	0.2
	RCP2-SS8C-□-PM-20-□□□	666 (600) / 625 (600) / 515	40~10	5~0.5	0.3	0.2
	RCP2-SS8C-□-PM-10-□□□	333 (300) / 310 (300) / 255	50~4	12~2	0.3	0.2
	RCP2-SS8C-□-PM-5-□□□	165 (150) / 155 (150) / 125	55~10	20~0.5	0.2	0.2
Motor-reversing	RCP2-SA5R-□-PM-12-□□□	600	4	1	0.3	0.2
	RCP2-SA5R-□-PM-6-□□□	300	8	2.5	0.3	0.2
	RCP2-SA5R-□-PM-3-□□□	150	8	4.5	0.2	0.2
	RCP2-SA6R-□-PM-12-□□□	600 / 540	6	1.5~0.5	0.3	0.2
	RCP2-SA6R-□-PM-6-□□□	300 / 270	12	3~2	0.3	0.2
	RCP2-SA6R-□-PM-3-□□□	150 / 135	12	6~4	0.2	0.2
	RCP2-SA7R-□-PM-16-□□□	533 (400) / 480 (400)	25~4	5~1	0.3	0.2
	RCP2-SA7R-□-PM-8-□□□	266 / 240	35~7	10~1.5	0.3	0.2
	RCP2-SA7R-□-PM-4-□□□	133 / 120	35~20	15~3	0.2	0.2
	RCP2-SS7R-□-PM-12-□□□	600 (440) / 470 (440)	20~5.5	4~0.5	0.3	0.2
	RCP2-SS7R-□-PM-6-□□□	250 / 230	20~2.5	5~0.5	0.3	0.2
	RCP2-SS7R-□-PM-3-□□□	105 / 105	30~20	10~1.5	0.2	0.2
	RCP2-SS8R-□-PM-20-□□□	600 (333) / 515 (333)	23~1	3~0.5	0.3	0.2
	RCP2-SS8R-□-PM-10-□□□	300 (250) / 255 (250)	28~4	9~0.5	0.3	0.2
	RCP2-SS8R-□-PM-5-□□□	160 (140) / 155 (140) / 125	55~1.5	20~0.5	0.2	0.2

(Note 1) The value in each band indicates the maximum speed for each applicable stroke. Values in parentheses apply to vertical operation.
(Note 2) The loading capacity is measured at the rated acceleration.

- Slider, belt drive

Model	Stroke (mm) and maximum speed (mm/sec)	Loading capacity Horizontal (kg)	Loading capacity Vertical (kg)	Rated acceleration Horizontal (G)	Rated acceleration Vertical (G)
RCP2-BA6-□-PM-54-□□□	1000	4~2	–	0.5	–
RCP2-BA7-□-PM-54-□□□	1500	8~2	–	0.5	–
RCP2-BA6U-□-PM-54-□□□	1000	4~2	–	0.5	–
RCP2-BA7U-□-PM-54-□□□	1500	8~2	–	0.5	–

Rod type

	Model	Stroke (mm) and maximum speed (mm/sec) (Note 1)																Loading capacity (Note 2)		Rated acceleration	
		50	100	150	200	250	300	350	400	450	500	550	600	700	800	900	1000	Horizontal (kg)	Vertical (kg)	Horizontal (G)	Vertical (G)
Standard	RCP2-RA2C-I-PM-1-□□□	25																7	2.5	0.05	0.05
	RCP2-RA3C-□-PM-5-□□□	187																15~2	6~1	0.2	0.2
	RCP2-RA3C-□-PM-2.5-□□□	114																30~4	10~2	0.2	0.2
	RCP2-RA4C-□-PM-10-□□□	458				458	350											25~5	4.5~0.5	0.2	0.2
	RCP2-RA4C-□-PM-5-□□□	250				237	175											40~10	12~2	0.2	0.2
	RCP2-RA4C-□-PM-2.5-□□□	125 (110)				118	87											40	19~2.5	0.2	0.2
	RCP2-RA6C-□-PM-16-□□□	450 (400)																40~10	5~1	0.2	0.2
	RCP2-RA6C-□-PM-8-□□□	210																50~30	17.5~1.5	0.2	0.2
	RCP2-RA6C-□-PM-4-□□□	130																55~35	26~1.5	0.2	0.2
Splashproof	RCP2W-RA4C-I-PM-10-□□□	450 (250)																25~5	4.5~2	0.2	0.2
	RCP2W-RA4C-I-PM-5-□□□	190																40	12~2.5	0.2	0.2
	RCP2W-RA4C-I-PM-2.5-□□□	120 (115)																40	19~2.5	0.2	0.2
	RCP2W-RA6C-I-PM-16-□□□	320 (265)																40~20	5~1	0.2	0.2
	RCP2W-RA6C-I-PM-8-□□□	200																50	17.5~2	0.2	0.2
	RCP2W-RA6C-I-PM-4-□□□	100																55	26~5	0.2	0.2
Single-guide	RCP2-RGS4C-□-PM-10-□□□	458				458	350											2.5~0.5	3.5~0.5	0.2	0.2
	RCP2-RGS4C-□-PM-5-□□□	250				237	175											3.5~1	11~0.5	0.2	0.2
	RCP2-RGS4C-□-PM-2.5-□□□	125 (114)				118 (114)	87											4~1.5	18~1.5	0.2	0.2
	RCP2-RGS6C-□-PM-16-□□□	450 (400)																3~1	4~0.5	0.2	0.2
	RCP2-RGS6C-□-PM-8-□□□	210																4~1.5	16~1	0.2	0.2
	RCP2-RGS6C-□-PM-4-□□□	133																5~2	24~0.5	0.2	0.2
Double-guide	RCP2-RGD3C-□-PM-5-□□□	187																1.5~0.5	5~0.5	0.2	0.2
	RCP2-RGD3C-□-PM-2.5-□□□	114 (93)																2~0.5	9~1	0.2	0.2
	RCP2-RGD4C-□-PM-10-□□□	458				458	350											3.5~1	3.5~0.5	0.2	0.2
	RCP2-RGD4C-□-PM-5-□□□	250				237	175											4.5~2	11~0.5	0.2	0.2
	RCP2-RGD4C-□-PM-2.5-□□□	125 (114)				118 (114)	87											5~2.5	18~1.5	0.2	0.2
	RCP2-RGD6C-□-PM-16-□□□	450 (400)																4~1	4~0.5	0.2	0.2
	RCP2-RGD6C-□-PM-8-□□□	210																5~1.5	16~1	0.2	0.2
	RCP2-RGD6C-□-PM-4-□□□	133																5~2	24~0.5	0.2	0.2

(Note 1) The value in each band indicates the maximum speed for each applicable stroke. Values in parentheses apply to vertical operation.
(Note 2) The loading capacity is measured at the rated acceleration.

Gripper

Model		Stroke	Maximum gripping force	Maximum speed	Lead	Rated acceleration
2-finger	RCP2-GRS-I-PM-1-10-P1	10 mm (5 mm per side)	21 N	33.3 mm/s (per side)	1.0 mm	0.3 G
	RCP2-GRM-I-PM-1-14-P1	14 mm (7 mm per side)	80 N	36.7 mm/s (per side)	1.1 mm	0.3 G
3-finger	RCP2-GR3SS-I-PM-30-10-P1	10 mm (5 mm per side)	23 N	40 mm/s (per side)	2.5 mm	0.2 G
	RCP2-GR3SM-I-PM-30-14-P1	14 mm (7 mm per side)	120 N	50 mm/s (per side)	3.0 mm	0.2 G
	RCP2-GR3LS-I-PM-30-19-P1	19 deg	17 N	200 deg/sec (per side)	12 deg	0.2 G
	RCP2-GR3LM-I-PM-30-19-P1	19 deg	92 N	200 deg/sec (per side)	12 deg	0.2 G

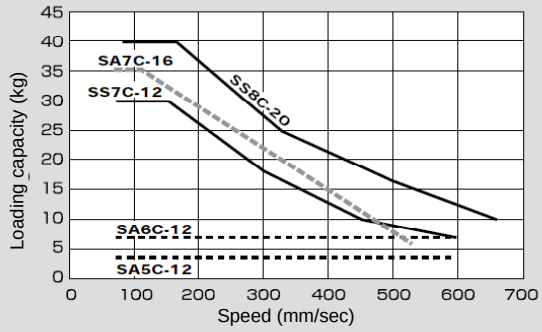
Rotary

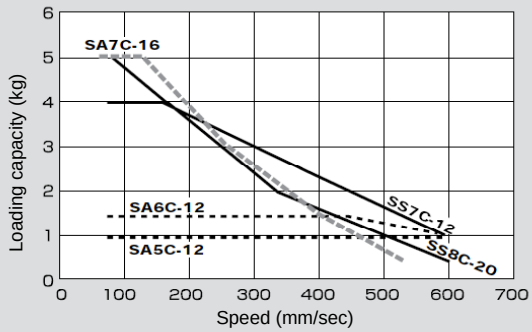
Model		Oscillation angle	Maximum torque	Maximum speed	Deceleration ratio	Rated acceleration
Vertical	RCP2-RTB-I-PM-20-330-P1	330 deg	1.1 N-m	600 deg/sec	1/20	0.3 G
	RCP2-RTB-I-PM-30-330-P1	330 deg	1.7 N-m	400 deg/sec	1/30	0.3 G
Flat	RCP2-RTB-I-PM-20-330-P1	330 deg	1.1 N-m	600 deg/sec	1/20	0.3 G
	RCP2-RTB-I-PM-30-330-P1	330 deg	1.7 N-m	400 deg/sec	1/30	0.3 G

- Correlation diagrams of speed and loading capacity for slider type (motor-straight type)

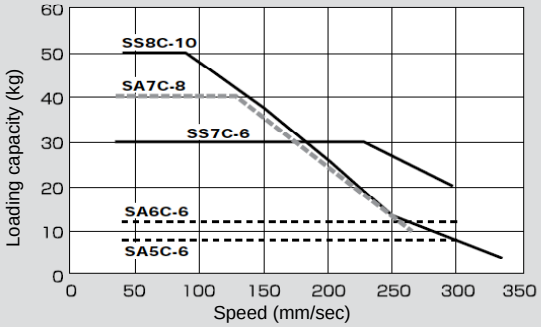
Horizontal installation | Vertical installation

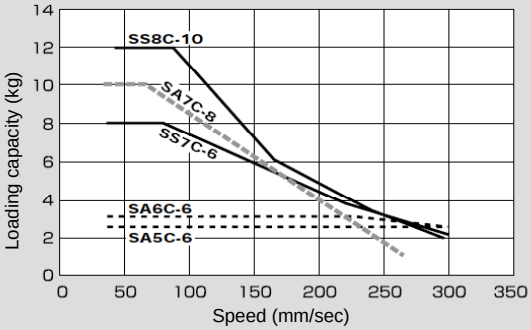
High-speed type



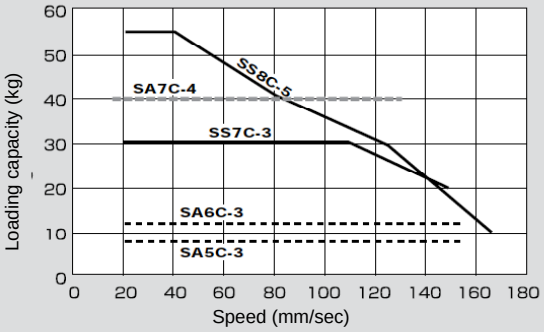


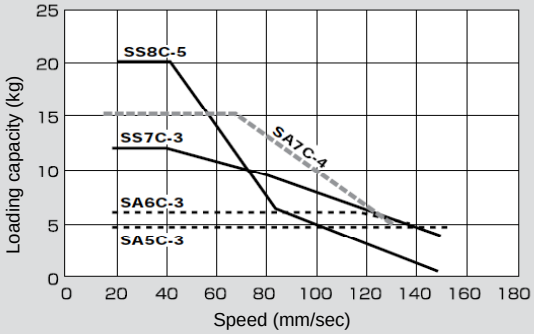
Medium-speed type





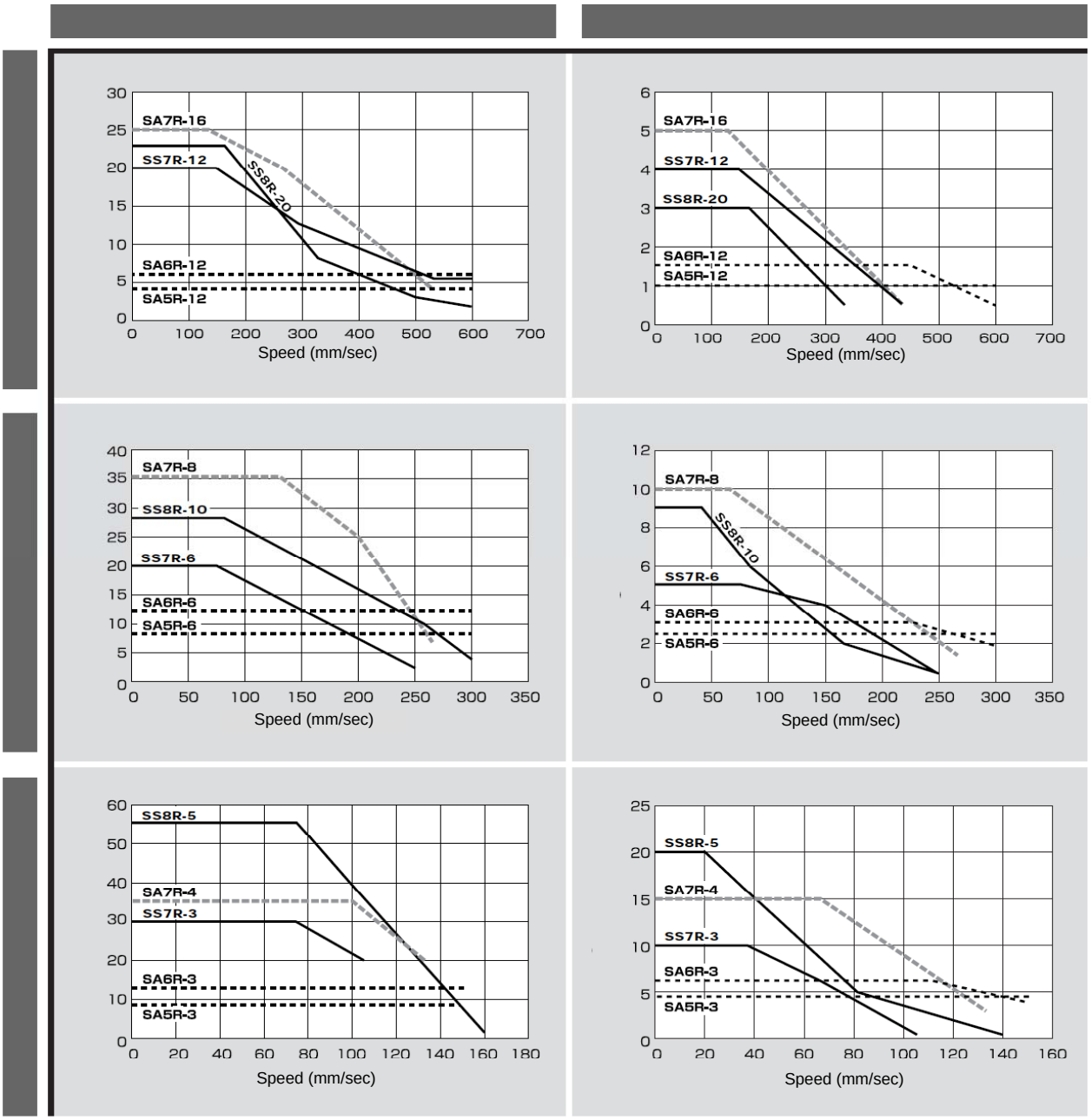
Low-speed type





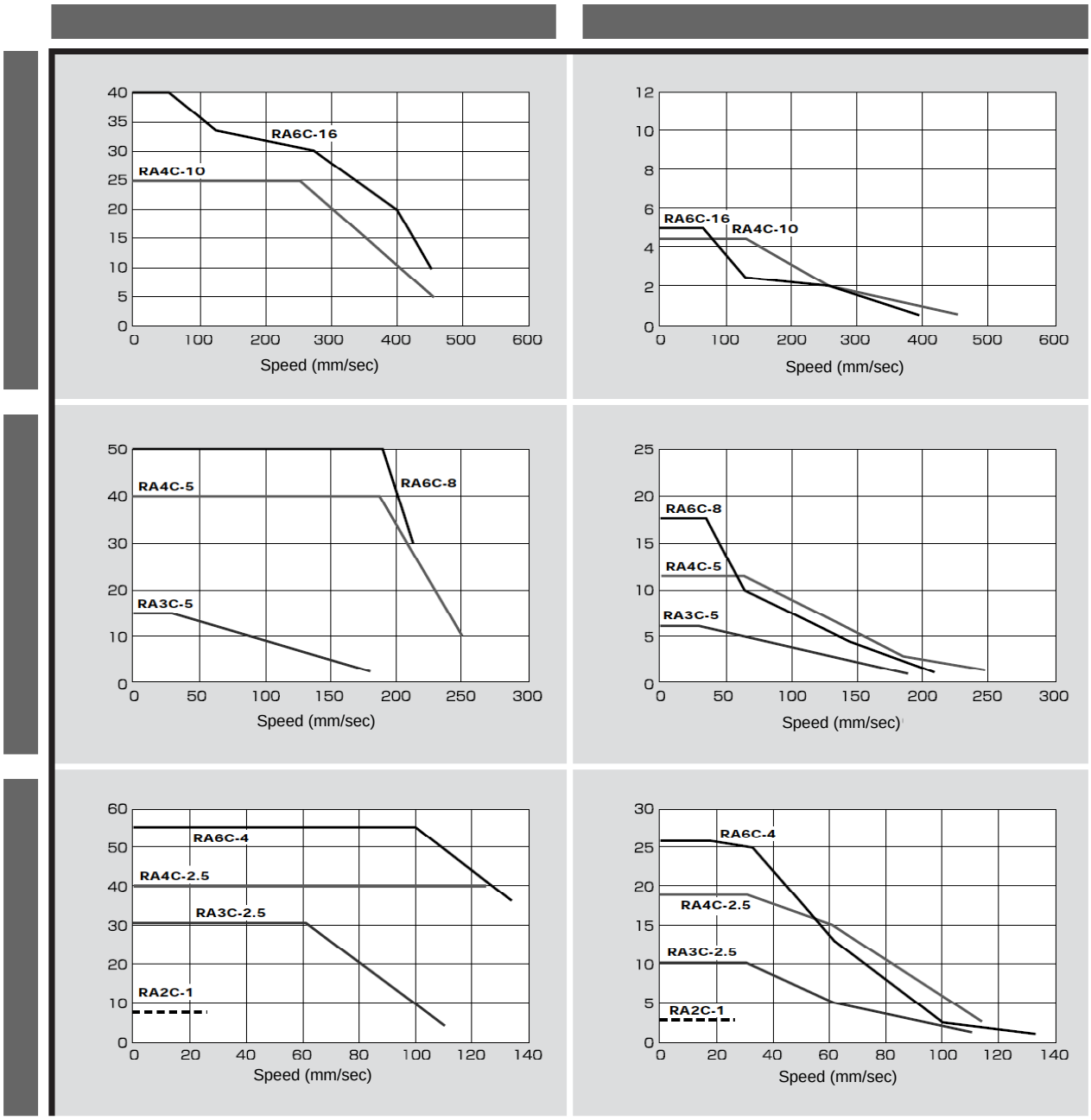
(Note) In the above graphs, the value shown after each type code indicates a lead.

- Correlation diagrams of speed and loading capacity for slider type (motor-reversing type)



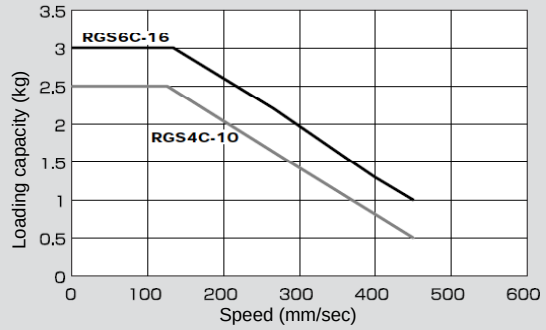
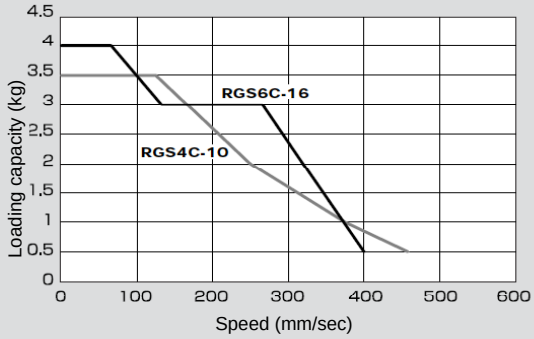
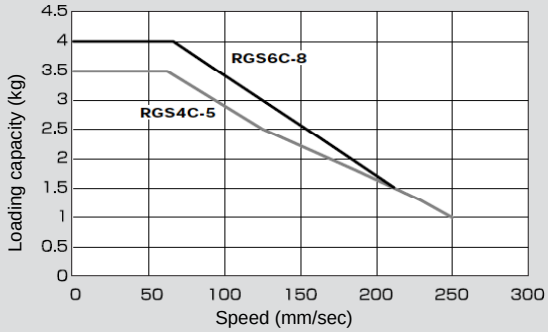
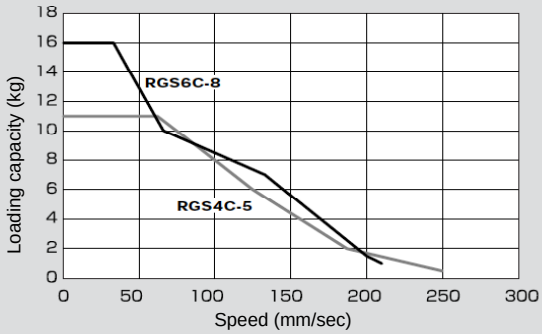
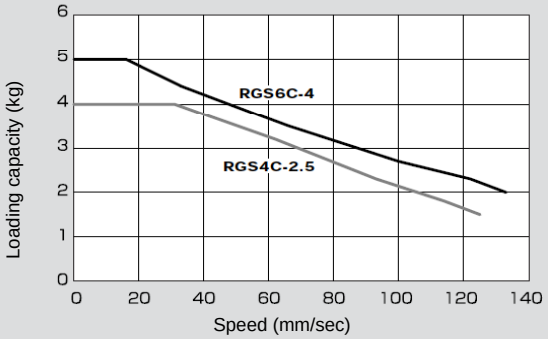
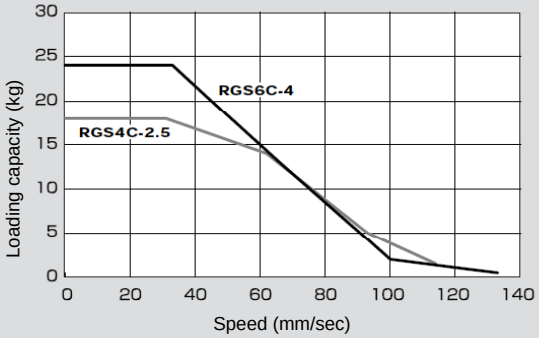
(Note) In the above graphs, the value shown after each type code indicates a lead.

- Correlation diagrams of speed and loading capacity for standard rod type



(Note) In the above graphs, the value shown after each type code indicates a lead.
(Note 1) The values for horizontal installation assume use of an external guide.

- Correlation diagrams of speed and loading capacity for single-guide type

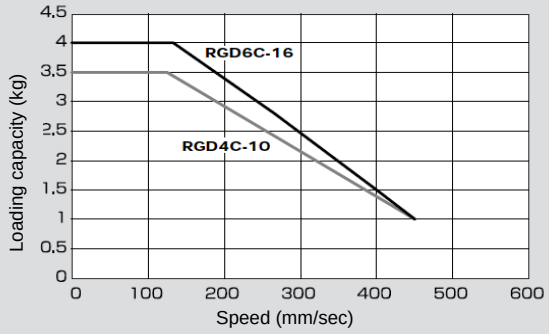
	Horizontal installation	Vertical installation
High-speed type		
Medium-speed type		
Low-speed type		

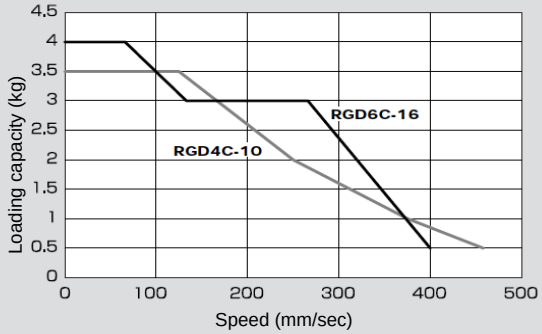
(Note) In the above graphs, the value shown after each type code indicates a lead.

- Correlation diagrams of speed and loading capacity for double-guide type

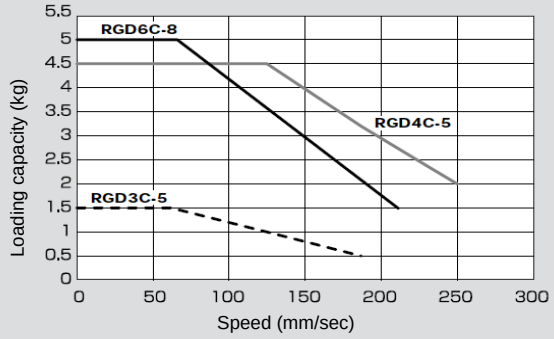
Horizontal installation | Vertical installation

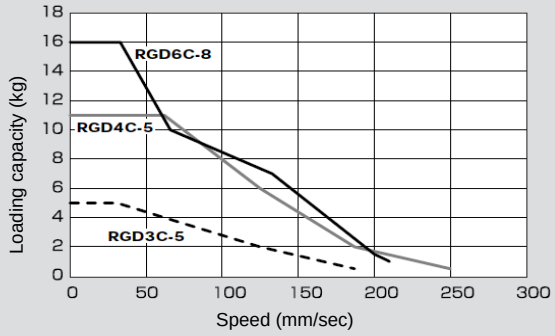
High-speed type



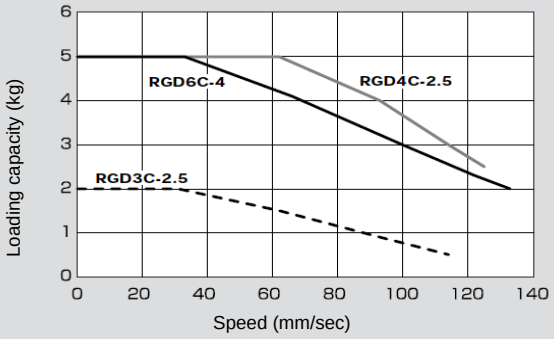


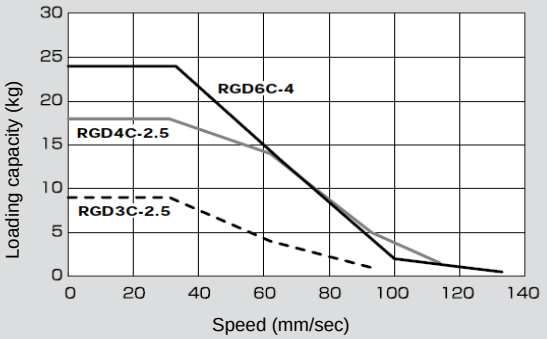
Medium-speed type





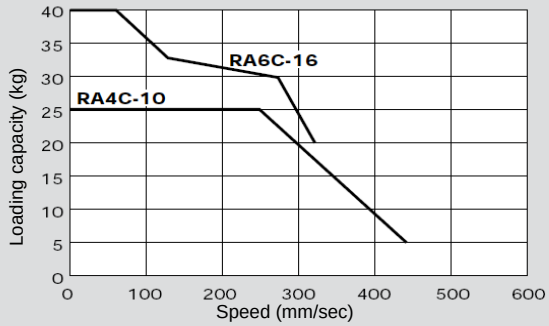
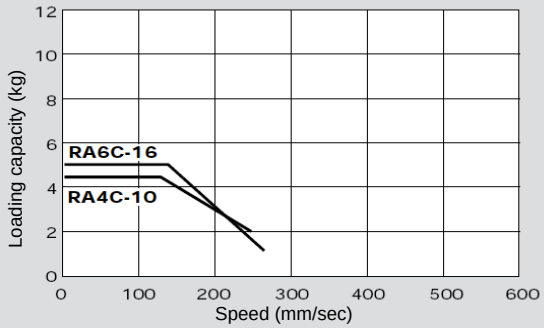
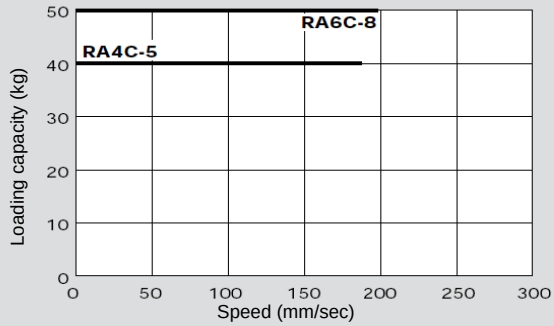
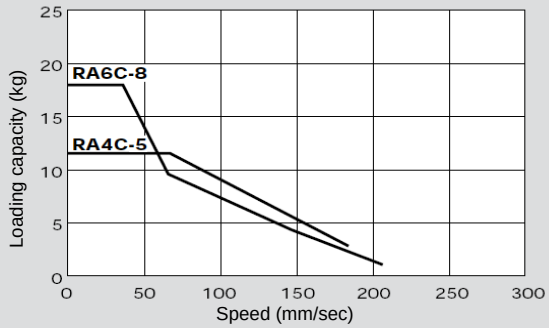
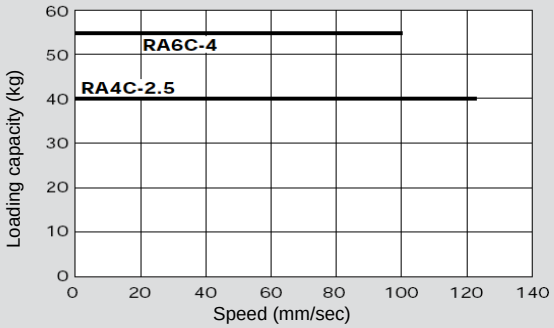
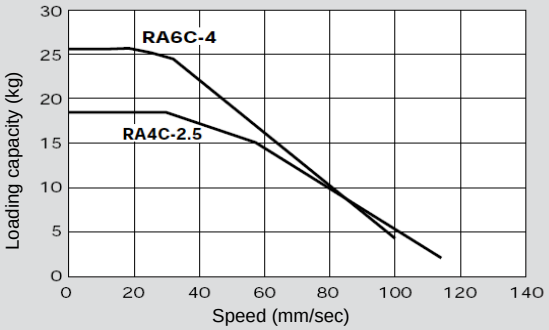
Low-speed type





(Note) In the above graphs, the value shown after each type code indicates a lead.

- Correlation diagrams of speed and loading capacity for dustproof/splashproof type

	Horizontal installation (Note 1)	Vertical installation (Note 2)
High-speed type		
Medium-speed type		
Low-speed type		

(Note) In the above graphs, the value shown after each type code indicates a lead.
(Note 1) The values for horizontal installation assume use of an external guide.
(Note 2) If the load is the same as the maximum loading capacity for the applicable speed, overshoot may occur due to vibration. Select an applicable specification so that the load becomes around 70% of the maximum loading capacity.

⦿ Battery Backup Function

The PSEL controller uses the following two batteries.

- System-memory backup battery (optional)
  The optional battery is available for backing up position data, SEL program variables and other data.

- Absolute-data backup battery
  A separate battery is used to retain the absolute encoder's rotation data, so that the motor rotation data can be retained/refreshed when the controller power is cut off. A controller specified with an absolute-type actuator is shipped with the absolute-data backup battery.

Each battery is explained in details.

1. System-Memory Backup Battery

The system-memory backup battery can be installed on the top face of the controller so that the data stored in the PSEL controller's SRAM will be retained even after the power is cut off.
Data to be backed up include controller parameters, SEL language variable data (global variables), position table data, and error list. The stored data will be retained even after the power is cut off.
(Use of the system-memory backup battery must be specified in the applicable controller parameter (other parameter No. 20 = 2).)

<Battery Replacement>
To replace the system-memory backup battery, disconnect the battery connector on the top face of the controller, and change the battery in the battery holder with a new battery.
It is recommended that you set a replacement schedule and replace the battery regularly.

The battery must be replaced as soon as the controller's battery voltage monitor function generates a battery voltage low alarm.

After an alarm is detected, a battery error will occur in approx. 10 days at an ambient temperature of 20°C if the power is supplied to the controller continuously. Once a battery error occurs, the data will be physically lost in approx. four days.

If the controller is not operated, the above periods should be reduced to 80% at 20°C or to 25% at 40°C.

The controller is designed so that the data will not be lost for at least 30 minutes without a battery if the controller is not detecting a battery error. Keep in mind to complete the battery replacement—taking out the current battery from the battery holder and placing a new battery in the holder—within 30 minutes.

To prevent the risk of data loss, you can use the PC software to evacuate the data in the SRAM to the flash ROM and then reload the flash ROM data to the SRAM after a new battery is installed.

The battery specifications are shown in the table below.

List of System-Memory Backup Battery Functions

Battery type	AB-5 (by IAI)	
Battery voltage	3.6 V	
Current capacity	2000 mAH	
Switching voltage at momentary power failure	(Typical) 2.81 V (2.7 V ~ 2.93 V)	System reset detection voltage
Power-source voltage drop at backup	(Typical) 0.3 V	
Detection voltage for battery voltage low alarm	(Typical) 2.65 V ± 5%	
Detection voltage for battery voltage low error	(Typical) 2.37 V ± 5%	
Time after alarm detection until error detection (reference)	10 days at 20°C based on continuous operation; 8 days if the power is not supplied. 10 days at 40°C based on continuous operation; 2.5 days if the power is not supplied.	
Minimum data retention voltage	Min. 2.0 V (Varies depending on the SRAM characteristics.)	
Time after error detection until data loss (reference)	4 days at 20°C based on continuous operation; 3 days if the power is not supplied. 4 days at 40°C based on continuous operation; 1 day if the power is not supplied.	
Data protection time during battery replacement	30 minutes (Maximum retention time when no battery is installed in the battery holder)	Data is retained by the super capacitor inside the controller.
Guide on when to replace battery	Approx. 5 years	

⊙ Parameter Utilization

Functions not initially available on the controller can be added, or dedicated functions can be assigned to input/output ports, by changing the values of corresponding parameters. Before changing a given parameter, always read the applicable section in the parameter list.

If you have any question regarding changing the parameters, please contact IAI's Sales Engineering Section. After changing a parameter, record the new and old parameter settings.

If you have purchased the PC software, we recommend that you back up the parameters immediately after the controller is delivered and when the system incorporating the controller is started. Since a number of customizing settings use parameters, you should back up the parameters regularly as you back up the programs.
To make the new parameters effective, write them to the flash ROM and then execute a software reset or reconnect the power.

Parameter classification

Parameters are classified into the following seven types based on what they specify:

1. I/O parameters
2. Parameters common to all axes
3. Axis-specific parameters
4. Driver parameters
5. Encoder parameters
6. I/O devices
7. Other parameters

1. Utilization Examples of I/O Parameters

I/Os include general-purpose inputs/outputs and dedicated inputs/outputs. General-purpose inputs/outputs are used by the user in SEL programs for sending/receiving ON/OFF signals to/from peripherals, among others.
Dedicated inputs are turned ON/OFF externally to activate specific functions.
Dedicated outputs turn ON or OFF in specific conditions. (Dedicated outputs cannot be turned ON/OFF in SEL programs.)

(1) I/O parameters
A desired input/output port can be specified as a dedicated input/output or general-purpose input/output. Set an appropriate input function specification value in the I/O parameter (Input/output function selection n) corresponding to the input/output port number you want to set.

The relationship of input port numbers and I/O parameter numbers is shown below.

Input port number	000	001	002	003	004	005	006	007	008	009	010	011
I/O parameter number	30	31	32	33	34	35	36	37	38	39	40	41

Input port number	012	013	014	015	016	017	018	019	020	021	022	023
I/O parameter number	42	43	44	45	251	252	253	254	255	256	257	258

Output port number	300	301	302	303	304	305	306	307
Output port number	46	47	48	49	50	51	52	53

Example 1) How to set input port No. 5 as an input to forcibly release the brake for axis 1
Change the input function specification value of I/O parameter No. 35, which corresponds to input port No. 5, to “22” (Axis 1 forced brake-release input).
I/O parameter No. 35 = 22

Example 2) How to set output port No. 307 as a servo-ON status output for axis 1
Change the output function specification value of I/O parameter No. 53, which corresponds to output port No. 307, to “24” (Axis 1 servo-ON status output).
I/O parameter No. 53 = 24

Example 3) How to set input port Nos. 21 and 22 as general-purpose inputs
Change the input function specification values of I/O parameter Nos. 256 and 257, which correspond to input port Nos. 21 and 22, respectively, to “0” (General-purpose input).
I/O parameter No. 256 = 0
I/O parameter No. 257 = 0

If the above parameter changes are made from their factory settings, the start-program number specification bits will change to the five bits represented by input port Nos. 16 through 20. The range of program numbers that can be specified will become 1 to 19.

(2) Explanation of input function specification values

Input function specification value 0: General-purpose input
The applicable input can be used freely in programs as a general-purpose input.

Input function specification value 1: Program start signal (BCD) (ON edge)
The applicable signal is set as a program start signal.
Once set, the signal can start the BCD program number specified by input function setting values 9 through 15.

Input function specification value 2: Program start signal (BIN) (ON edge)
The applicable signal is set as a program start signal.
Once set, the signal can start the binary program number specified by input function setting values 9 through 15.

Input function specification value 3: Soft reset signal (ON edge)
Allow the applicable signal to restart the controller in the event of an error, etc.
Note 1: The input signal must remain ON for at least 1 second.
Note 2: The coordinate values will be cleared, so home return must be performed again.

Input function specification value 4: Servo ON
Allow the applicable signal to turn on the servo of a valid axis at its ON edge.
The signal will turn off the servo of a valid axis at its OFF edge.
Note: There must be an interval of at least 1.5 seconds between ON and OFF edges.

Input function specification value 5: Auto-start program start signal
If an auto-start program is set, this signal can be used to start the program.
The program will start at the ON edge of this signal, while all operations and programs will be aborted at the OFF edge.

Input function specification value 6: Soft interlock for all servo axes (OFF level)
The active programs will turn off when this signal turns OFF.
(Any moving axis will decelerate to a stop.)

Input function specification value 7: Operation-pause reset signal (ON edge)
Allow the applicable signal to reset the operation pause signal set by input function selection value 8.

Input function specification value 8: Operation pause signal (OFF level)
Allow the applicable signal to pause all valid axes.
Note: The pause will be reset at the ON edge of the operation-pause reset signal (specified by input function selection 7) after turning this signal ON.

Input function specification value 9: Start-program number specification bit 1 (least significant bit)
This bit specifies the least significant bit of a program number.
Note: Start-program number specification bits x (input function setting values 9 through 15) cannot be assigned discontinuously from the least significant bit or in descending order from the least significant bit.

Input function specification value 10: Start-program number specification bit 2
This bit specifies the second bit of a program number.

Input function specification value 11: Start-program number specification bit 3
This bit specifies the third bit of a program number.

Input function specification value 12: Start-program number specification bit 4
This bit specifies the forth bit of a program number.

Input function specification value 13: Start-program number specification bit 5
This bit specifies the fifth bit of a program number.

Input function specification value 14: Start-program number specification bit 6
This bit specifies the sixth bit of a program number.

Input function specification value 15: Start-program number specification bit 7
This bit specifies the seventh bit of a program number.

Input function specification value 16: Error reset (ON edge)
This signal is used to reset errors.
Note: Only errors of operation-cancellation level or lower can be reset using this signal.

Input function specification value 17: Drive-source cutoff reset input (ON edge) (Effective when the problem factor has been removed)
This signal is used as a drive-source cutoff reset input when the emergency stop/enable switch recovery type is set to “Operation continued.”

Input function specification value 18: Home-return command signal for all valid axes (ON edge)
This signal commands home return of all valid axes.
Note: The servo ON input signal (input function specification value 4) must be turned ON first.

Input function specification value 19: Home-return command signal for all incremental axes (ON edge)
This signal commands home return of all incremental axes.
Note: The servo ON input signal (input function specification value 4) must be turned ON first.

Input function specification value 20: PC/TP-servo movement command acceptance permission input
Movements can be permitted from the PC software or teaching pendant.

Input function specification value 21: Remote-mode control input
This signal can be used to switch between the AUTO mode and MANUAL mode.
Note: Switching is enabled only when the mode switch is set to “AUTO.”

Input function specification value 22: Axis 1 forced brake release
Forcibly release the brake (axis 1).
Note: This function is effective only when the brake switch is tilted down (NOM).

Input function specification value 23: Axis 2 forced brake release
Forcibly release the brake (axis 2).
Note: This function is effective only when the brake switch is tilted down (NOM).

Input function specification value 24 ~ 27: For future expansion
Not used.

(3) Explanation of output function specification values

Output function specification value 0:	General-purpose output The applicable output can be used freely in programs as a general-purpose output.
Output function specification value 1:	Operation-cancellation level or higher error output (ON) The signal will turn ON when an error of operation-cancellation level or higher generates.
Output function specification value 2:	Operation-cancellation level or higher error output (OFF) The signal will turn OFF when an error of operation-cancellation level or higher generates.
Output function specification value 3:	Operation-cancellation level or higher error + emergency stop output (ON) This error output signal and emergency-stop output signal will turn ON when an error of operation-cancellation level or higher generates.
Output function specification value 4:	Operation-cancellation level or higher error + emergency stop output (OFF) This error output signal and emergency-stop output signal will turn OFF when an error of operation-cancellation level or higher generates.
Output function specification value 5:	READY output (PIO trigger program operation enabled) A signal will be output after the check is completed following the controller power input. The signal will turn ON only when the controller is able to perform program operation.
Output function specification value 6:	READY output (Absence of operation-cancellation level or higher error) The function is the same as that of output function specification value 5, but absence of operation-cancellation level or higher error is added as a condition.
Output function specification value 7:	READY output (Absence of cold-start level or higher error) The function is the same as that of output function specification value 5, but absence of cold-start level or higher error is added as a condition.
Output function specification value 8:	Emergency stop output (ON) The output signal will turn ON when the emergency-stop input signal turns ON. The signal will turn OFF when the emergency stop is reset.
Output function specification value 9:	Emergency stop output (OFF) The output signal will turn OFF when the emergency-stop input signal turns ON. The signal will turn ON when the emergency stop is reset.
Output function specification value 10:	AUTO mode output A signal will be output during the AUTO mode.
Output function specification value 11:	Auto operation status output A signal will be output during auto program operation.
Output function specification value 12:	All-valid-axes home (= 0) output A signal will be output when all valid axes are at the 0-mm position.

Output function specification value 13: All-valid-axes home-return complete (coordinate confirmed) output
A signal will be output when all valid axes have completed home return.

Output function specification value 14: All-valid-axes preset home coordinate output
A signal will be output when all valid axes have completed home return.
The value set by axis-specific parameter No. 12, “Home preset value” is used as the home position.

Output function specification value 15: Voltage-low warning output for system-memory backup battery
A signal will be output when the voltage of the system-memory backup battery drops to approx. 2.6 V.

Output function specification value 16: Voltage-low warning output for absolute-data backup battery
A signal will be output when the voltage of the absolute-data backup battery drops to approx. 3.2 V.
Once an abnormal voltage level is detected, the signal will remain ON until a power ON reset or software reset is performed.

Output function specification value 17: Drive-source cutoff (SDN) notification output
The output port will turn OFF when the drive source is cut off.

Output function specification value 18 ~ 23: For future expansion
Not used.

Output function specification value 24: Axis 1 servo ON output
A signal is output while the servo for axis 1 is ON.

Output function specification value 25: Axis 2 servo ON output
A signal is output while the servo for axis 2 is ON.

Output function specification value 26 ~ 29: Reserved by the system.
Not used.

2. Utilization Examples of Axis-specific Parameters

The following functions can be added to, or changed from the factory-set functions, by changing the values of the corresponding axis-specific parameters. Before changing a given parameter, always read the applicable section in the parameter list.

- Change the home return direction
- Set a home preset
- Set a home offset
- Apply length measurement correction
- About the axis operation type and rotational axis mode
- Operate a rotational axis in the multi-rotation mode or with short-cut control
- About the zone output

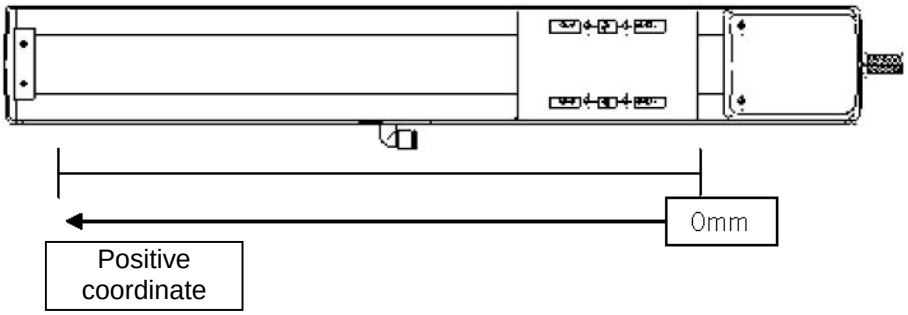
Change the home return direction
Axis-specific parameter No. 6, “Coordinate/physical-operation direction selection”

No.	Parameter name	Default value	Input range	Unit
6	Coordinate/physical-operation direction selection	1	0 ~ 1	None

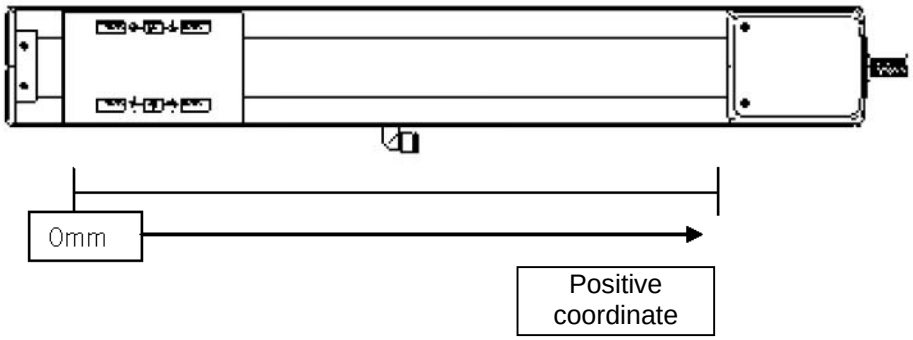
- Setting method
  A desired direction of home-return operation can be selected.

- Set value
  0: Motor CCW → Positive coordinate direction
  1: Motor CCW → Negative coordinate direction

Example 1: A linear axis whose home is at the standard position: When the parameter is set to “1”



Example 2: A linear axis whose home is at the standard position: When the parameter is set to “0”



Note: On rod-type actuators (except for the RCP2-RA3C, 4C and 6C), the home return direction cannot be reversed after the delivery only by changing the parameter.

Set a home preset
Axis-specific parameter No. 12, “Home preset value”

No.	Parameter name	Default value	Input range	Unit
12	Home preset value	0	-99999999 ~ 99999999	0.001 mm

- Explanation of setting
  Set a value indicating where the actuator should be upon completing home return.
  (Normally, the actuator should be at 0-mm coordinate upon completing home return.)

- Set value
  Unit: 0.001 mm

Example 1: “Do not set” a home preset value

Home return complete → [0.000] mm is displayed.

Example 2: Set “3000” as a home preset value

Home return complete → [3.000] mm is displayed.

- Note
  Take note that when a home-return preset value is set, the effective stroke will also change. In particular, the stroke will decrease if the preset position is on the positive side of the default home.

Set a home offset
Axis-specific parameter No. 21, “Offset travel distance at home return”

No.	Parameter name	Default value	Input range	Unit
21	Offset travel distance at home return	1000	-99999999 ~ 99999999	0.001 mm

- Explanation of setting
  An offset can be set that will be applied after detecting phase Z (point 0) during home return.
  * If the home position has shifted after replacing the motor, jig, etc., use this parameter to adjust the home.

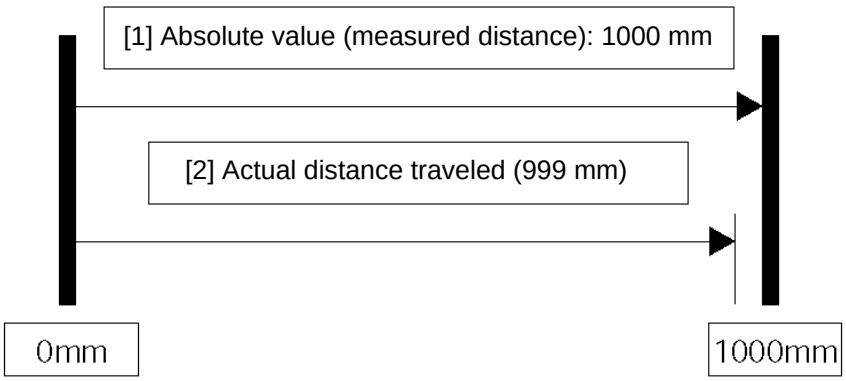
- Set value
  Setting unit: 0.001 mm
  Example:
  Set the offset to 0.5 mm = 500

- Note
  If the offset travel distance is near an integer multiple of the ball screw lead (such as 0, 6, 12 or 18 mm when the lead is 6 mm), the home will come directly above phase Z and thus rotation data may shift by one revolution upon absolute reset due to an “unstable” servo lock condition (a phenomenon where the coordinate values shift by one motor revolution). In this case, the position after home return will become the integer multiple of the lead length.
  * If the position after home return has become an integer multiple of the lead value, make adjustment using axis-specific parameter No. 12, “Home preset value.”

Apply length measurement correction
Axis-specific parameter No. 44, “Length measurement correction”

No.	Parameter name	Default value	Input range	Unit
44	Length measurement correction	0	-99999999 ~ 99999999	0.001 mm/1 m

- Explanation of setting
  Adjust the difference between the actual distance traveled and the measured distance, for the commanded travel distance.
  Example: Move the actuator from 0 mm to 1000 mm by specifying a position.



Correct the travel distance of [2] with respect to [1].
In the above example, enter “1000” because the actual distance traveled is 999 mm.
(Setting unit: 0.001/1 m)

* C10-class ball screws are subject to a margin of error of ± 0.21 mm per 300 mm.

Axis operation type and rotational axis mode
Axis-specific parameter No. 1, “Axis operation type”

No.	Parameter name	Default value	Input range	Unit
1	Axis operation type	Varies depending on the actuator.	0 ~ 1	None

- Explanation of setting
  This parameter defines the type of the actuator to be used.

- Set value
  0: Linear movement axis — Actuators other than rotational axes of multi-rotation type
  1: Rotational movement axis — Rotational axes of multi-rotation type (RCP2-RTBL,. RCP2-RTCL)

Axis-specific parameter No. 66, “Mode selection for rotational axis”

No.	Parameter name	Default value	Input range	Unit
66	Rotational axis mode selection	0	0 ~ 5	None

- Explanation of setting
  Select the mode for a rotational axis.
  Related parameter: Axis-specific parameter No. 7, “Soft limit +”

- Set value
  0: Normal
  1: Index mode
  If a rotational axis is used, this parameter will be fixed to “1.”
  * When the index mode is set, the soft limit will be fixed to 359.999 mm internally.
  2 ~ 5: Reserved by the system

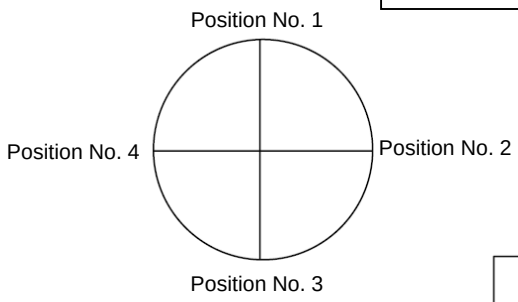
Note: If a simplified absolute unit is used, the following settings are disabled:
- Linear movement axis: If “0” is set, the infinite stroke mode cannot be set using parameter No. 68.
- Rotational movement axis: If “1” is set to enable the index mode, short-cut control cannot be selected using parameter No. 67.

Operate a rotational axis in the multi-rotation mode or with short-cut control
Axis-specific parameter No. 67, “Short-cut control selection for rotational movement axis”

- Set this parameter if you want to rotate a rotational axis in a specified direction.
  Set the parameter to “1” (short-cut selected) and repeatedly issue a movement command in the same rotating direction. The actuator will perform multi-rotation operation.
  What is short-cut control?
  The actuator moves to the position closest to the subsequent position.
- Set value
  0: Not selected
  1: Selected
  2 ~ 5: Reserved by the system

Note: On actuators using a simplified absolute unit, multi-rotation operation cannot be performed because this parameter cannot be set to “1” (short-cut control selected).

Example: Move the actuator continuously in the sequence of position Nos. 2 → 3 → 4 → 1 with reference to position No. 1



Program

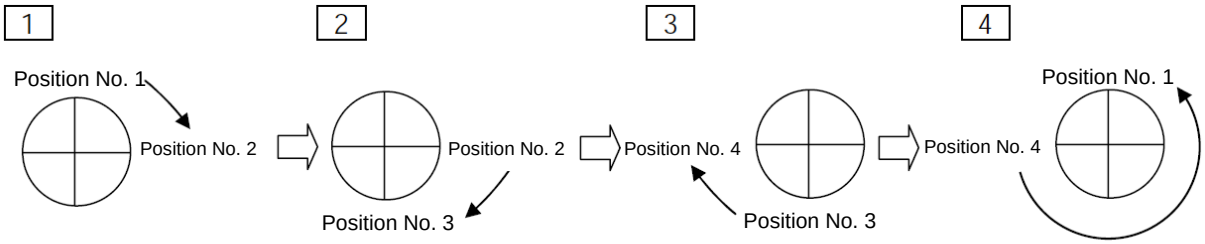
Step No.	Command	Operand
1	MOVP	2
2	MOVP	3
3	MOVP	4
4	MOVP	1

Position

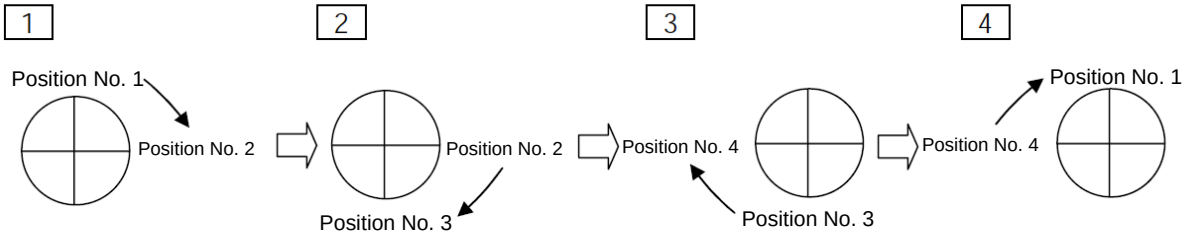
Position No.	Position data
1	0
2	90
3	180
4	270

In position data, 1° is equivalent to 1 mm.

When short-cut control is not selected



When short-cut control is selected



* By selecting this option, the actuator can be rotated in a specified direction.

A table summarizing the information on “Axis operation type,” “Mode selection for linear movement axis,” “Mode selection for rotational movement axis,” “Short-cut control selection for rotational movement axis,” “Expression of current position (approx.),” “Software limit +” and “Software limit -,” etc., is provided as “⊙ Combination Table of PSEL Linear/Rotary Control Parameters” in the Appendix. Use this table as a reference.

Zone output

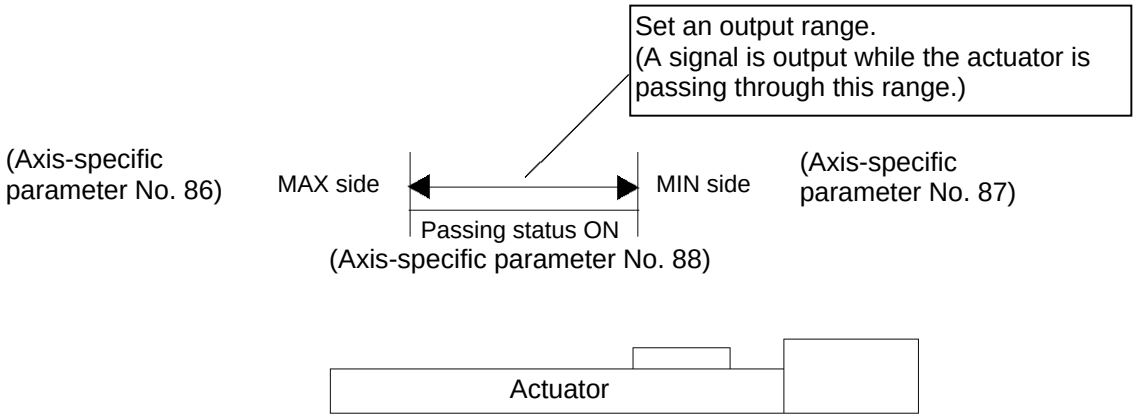
A signal can be output when the actuator has entered a desired zone specified by the user.
Three parameters must be set to specify a zone.
A zone is set for each axis.

No.	Parameter name	Default value	Input range	Unit
86	Zone 1 MAX	0	-99999999 ~ 99999999	0.001 mm
87	Zone 1 MIN	0	-99999999 ~ 99999999	0.001 mm
88	Zone 1 output number	0	0 ~ 899	None

Axis-specific parameter No. 86, “Zone 1 MAX”
Set the maximum limit of the zone, in units of 0.001 mm.
Example: To set 50 mm, set the value “50000.”

Axis-specific parameter No. 87, “Zone 1 MIN”
Set the minimum limit of the zone, in units of 0.001 mm.
Example: To set 10 mm, set the value “10000.

Axis-specific parameter No. 88, “Zone 1 output number”
Set an output port or flag number for the zone.
The output number set in this parameter cannot be used in programs.



- Note
  Set the zone so that the passing time through the zone will become at least 3 msec.

The zone output function allows four zones (zones 1 through 4) to be set for each axis.

No.	Parameter name	Default value	Input range	Unit
86	Zone 1 MAX	0	-99999999 ~ 99999999	0.001 mm
87	Zone 1 MIN	0	-99999999 ~ 99999999	0.001 mm
88	Zone 1 output number	0	0 ~ 899	None
89	Zone 2 MAX	0	-99999999 ~ 99999999	0.001 mm
90	Zone 2 MIN	0	-99999999 ~ 99999999	0.001 mm
91	Zone 2 output number	0	0 ~ 899	None
92	Zone 3 MAX	0	-99999999 ~ 99999999	0.001 mm
93	Zone 3 MIN	0	-99999999 ~ 99999999	0.001 mm
94	Zone 3 output number	0	0 ~ 899	None
95	Zone 4 MAX	0	-99999999 ~ 99999999	0.001 mm
96	Zone 4 MIN	0	-99999999 ~ 99999999	0.001 mm
97	Zone 4 output number	0	0 ~ 899	None

3. Parameter Utilization Examples (Reference)

	Description	Action	Parameter setting	Operation/outcome
1	Suppress generation of errors pertaining to the standard I/O board (so that trial operation can be performed before the board is wired, for example).	The I/O-board error monitor can be disabled to suppress error generation.	Set “0” in the I/O parameter corresponding to the I/O whose error monitor is to be disabled. Standard I/O: I/O parameter No. 10 = 0	To disable the error monitor of the standard I/O board, set “0” in I/O parameter No. 10. Note: Before operating the I/O board again, be sure to reset the parameter value to “1.”
2	Implement a restart (software reset) using an external input signal.	A desired input port can be set as a restart input.	Set the following value in the I/O parameter “Input function selection n” corresponding to the selected input port: I/O parameter “Input function specification value” = 3	The controller will be restarted when the specified port has remained ON for at least 1 second.
3	Turn on the servo using an external input signal.	A desired input port can be set as a servo ON input.	Set the following value in the I/O parameter “Input function selection n” corresponding to the selected input port: I/O parameter “Input function specification value” = 4	The servo will turn ON at the ON edge of the specified port. The servo will turn OFF at the OFF edge.
4	Start an auto-start program using an external input signal. (Under the default setting, the auto-start program will start when the power is input or the controller is restarted (by software reset) in the AUTO mode.) (The steps to start the auto-start program will increase.)	A desired input port can be set as an input for auto-program start signal.	Set the following value in the I/O parameter “Input function selection n” corresponding to the selected input port: I/O parameter “Input function specification value” = 5 Other parameter No. 7 = 0	The program will start at the ON edge of the specified port. The program will end at the OFF edge.
5	Pause operations using an external input signal.	A desired input port can be set as a pause input. A desired input port can be set as a pause reset input.	Set the following value in each I/O parameter “Input function selection n” corresponding to the selected input port: I/O parameter “Input function specification value” = 7 I/O parameter “Input function specification value” = 8 Setting example) To set input port No. 5 as the pause reset input and input port No. 6 as the pause input, set “8” in I/O parameter No. 35 and “7” in I/O parameter No. 36.	Operations will pause at the OFF edge of the specified port set as the operation-pause signal input. Pause will be reset at the ON edge of the port set as the operation-pause reset signal input. (The port set as the operation-pause signal input is always ON.)
6	Reset errors (errors of operation-cancellation level or lower) using an external input signal.	A desired input port can be set as an error reset input.	Set the following value in the I/O parameter “Input function selection n” corresponding to the selected input port: I/O parameter “Input function specification value” = 16	Errors will be reset at the ON edge of the specified port.
7	Perform home return using an external input signal.	A desired input port can be set as a home return input.	Set the following value in the I/O parameter “Input function selection n” corresponding to the selected input port: I/O parameter “Input function specification value” = 18	Home return will be performed at the ON edge of the specified port. (The servo must be turned ON first.)

	Description	Action	Parameter setting	Operation/outcome
8	Enter program numbers as binary codes using input ports (default setting: BCD input).	Program numbers to be specified can be input as binary codes using the ports set as start-program number specification bits 1 through 7.	Set the following value in the I/O parameter “Input function selection n” corresponding to the selected input port: I/O parameter “Input function specification value” = 2	
9	Check the level of each error currently present, using an output port.	Error levels can be checked based on the combination of the output function specification values (1 through 4, 5 through 7) and the ON/OFF levels of the applicable output ports.	Set the following value in each I/O parameter “Output function selection n” corresponding to the selected output port: I/O parameter “Output function specification value” = 2 I/O parameter “Output function specification value” = 7 (I/O parameter No. 46 and No. 47 have been set to “2” and “7,” respectively, at the factory.)	Note) Factory-set parameters
10	Have emergency stop status notified via an output port.	Whether or not an emergency stop is currently actuated can be checked from the ON/OFF levels of the output ports for which function specification values of 8 and 9 are specified.	Set the following value in the I/O parameter “Output function selection n” corresponding to the selected output port: I/O parameter “Output function specification value” = 9	Note) Not set at the factory.
11	Output a signal during the AUTO mode.	A desired output port can be set as an AUTO mode output.	Set the following value in the I/O parameter “Output function selection n” corresponding to the selected output port: I/O parameter “Output function specification value” = 10	The specified port will turn ON during the AUTO mode.
12	Output a signal during auto operation.	A desired output port can be set as an auto operation status output.	Set the following value in the I/O parameter “Output function selection n” corresponding to the selected output port: I/O parameter “Output function specification value” = 11	The specified port will turn ON during auto operation.
13	How auto operation status is recognized during auto operation can be changed using the setting of other parameter No. 12.	• Auto operation status will be recognized if a program is running (regardless of the MANU or AUTO mode). • Auto operation status will be recognized if a program is running or when the controller is in the AUTO mode (regardless of whether or not a program is running). In either case, no all-operation-cancellation factor must be present. Auto operation status will be recognized based on one of the two conditions specified above.	• Other parameter No. 12 = 0 Auto operation will be recognized when a program is running. • Other parameter No. 12 = 1 Auto operation will be recognized when a program is running or the controller is in the AUTO mode. • “No all-operation-cancellation factor is present” refers to a condition in which no error of operation-cancellation level or higher is present AND no emergency stop signal is input AND no safety gate signal is input AND the deadman switch is ON (teaching pendant option).	
14	Output a signal when all valid axes are at their home.	A desired output port can be set as an all-valid-axis home position signal output. Note: Do not use a HOME command if the controller is of absolute specification.	Set the following value in the I/O parameter “Output function selection n” corresponding to the selected output port: I/O parameter “Output function specification value” = 12	The specified port will turn ON when all valid axes are at their home.

	Description	Action	Parameter setting	Operation/outcome
15	Output a signal when all valid axes have completed home return.	A desired output port can be set as an all-valid-axes home-return complete output.	Set the following value in the I/O parameter “Output function selection n” corresponding to the selected input port: I/O parameter “Output function specification value” = 13	The specified port will turn ON when all valid axes have completed home return.
16	Output a warning signal when the voltage of the system-memory backup battery became low.	A desired output port can be set as a voltage-low warning output for the system-memory backup battery.	Set the following value in the I/O parameter “Output function selection n” corresponding to the selected input port: I/O parameter “Output function specification value” = 15	The specified port will turn ON when the voltage of the system-memory backup battery became low.
17	Output a warning signal when the voltage of the absolute encoder battery became low.	A desired output port can be set as a voltage-low warning output for the absolute encoder battery.	Set the following value in the I/O parameter “Output function selection n” corresponding to the selected input port: I/O parameter “Output function specification value” = 16	The specified port will turn ON when the voltage of the absolute encoder battery became low.
18	Release the brake using an external input signal.	A desired input port can be set as a forced brake-release input.	Set the following value in each I/O parameter “Input function selection n” corresponding to the selected input port: I/O parameter “Input function specification value” = 22 (Axis 1) I/O parameter “Input function specification value” = 23 (Axis 2) Setting example) To set input port No. 12 as the brake release input for axis 2, set “23” in I/O parameter No. 42.	The brake will be forcibly released when the specified port turns ON.
19	Retain output conditions upon actuation of an emergency stop or opening of the safety gate.	The minimum and maximum output port numbers can be set to specify a range of outputs whose condition is to be retained.	I/O parameter No. 70 = Min. output port number I/O parameter No. 71 = Max. output port number Setting example) To retain the conditions of output port Nos. 303 through 307, set as follows: I/O parameter No. 70 = 303 I/O parameter No. 71 = 307	← The conditions of output port Nos. 303 through 307 will be retained when the emergency stop input turns ON or the safety gate opens.
20	Start a program when the emergency stop input turns ON or the safety gate opens. Programs that can be started in these conditions are limited to those not containing I/O processing, calculation processing or any other processing involving actuator operation (PIO processing programs).	A PIO processing program to be started in these conditions can be set. The program number of the applicable PIO processing program, and the minimum and maximum output port numbers indicating the range of processed outputs, are set by parameters.	Other parameter No. 2 = PIO processing program number Other parameter No. 70 = Min. output port number Other parameter No. 71 = Max. output port number Setting example) To start program No. 5 that processes output port Nos. 303 through 307, set as follows: Other parameter No. 2 = 5 Other parameter No. 70 = 303 Other parameter No. 71 = 307	← Program No. 5 will start when the emergency stop input turns ON or the safety gate opens. Output port Nos. 303 through 307 can be processed.

	Description	Action	Parameter setting	Operation/outcome
21	Switch between the AUTO mode and MANUAL mode using an input port.	A desired input port can be set as a mode switching input.	Set the following value in the I/O parameter “Input function selection n” corresponding to the selected input port: I/O parameter “Input function specification value” = 21	Set the mode switch to the “AUTO” side. The controller will switch to the AUTO mode when the specified input port turns OFF, and to the MANU mode when the port turns ON. If the mode switch is set to the “MANU” side, the controller will remain in the MANU mode regardless of the ON/OFF level of the input port.
22	Automatically restart the controller (effect a software reset) and start the auto-start program after an emergency stop has been reset.	The emergency-stop recovery type can be set to “Abort operations/programs (Software reset when the emergency stop is reset).”	Other parameter No. 10 = 3 Other parameter No. 7 = 1	After the emergency stop button has been reset, the controller will be reset (software reset will be effected) automatically and the auto-start program will start.
23	Automatically reset errors and start the auto-start program after an emergency stop has been reset.	The emergency-stop recovery type can be set to “Abort operations/programs (Error reset and auto-start program start when the emergency stop is reset).”	Other parameter No. 10 = 4 Other parameter No. 7 = 1 “17” must not be set as the “input function specification value” in the I/O parameter “Input function selection n.”	After the emergency stop button has been reset, errors will be reset automatically and the auto-start program will start.
24	Continue to operate the actuator after an emergency stop has been reset (= resume actuator operation from immediately before the emergency-stop input signal turned ON). When an emergency-stop input signal is ON, all programs remain active and only programs involving actuator operation will be stopped. (When an emergency stop is actuated, all programs in which actuator operations are not specified will remain active. Programs in which actuator operations are specified will run until reaching a step in which an actuator operation command is specified.)	The emergency-stop recovery type can be set to “Operation continued.” A desired port can be selected as a pause reset input. A desired port can be selected as a restart input.	Other parameter No. 10 = 2. Set the following value in each I/O parameter “Input function selection n” corresponding to the selected input port: I/O parameter “Input function specification value” = 7 I/O parameter “Input function specification value” = 3 (To ensure the specified operation cancellation method will work) Setting example) To set input port No. 5 as the pause reset input and input port No. 1 as the restart input, set “7” in I/O parameter No. 35 and “3” in I/O parameter No. 31.	After the emergency stop button has been reset, the actuator operation will resume at the ON edge of the port for which the input function specification value 7 (operation-pause reset signal) is set. To abort the remaining operation, do not allow the port for which the input function specification value 7 is set to receive an ON signal edge. Instead, turn ON for at least 1 second the port for which the input function specification value 3 (software reset signal) is set, in order to restart the controller.
25	Use the system-memory backup battery.	Install the optional system-memory backup battery.	Other parameter No. 20 = 2	When this setting is enabled, SEL global data and error list will be retained even after the main power is turned off.

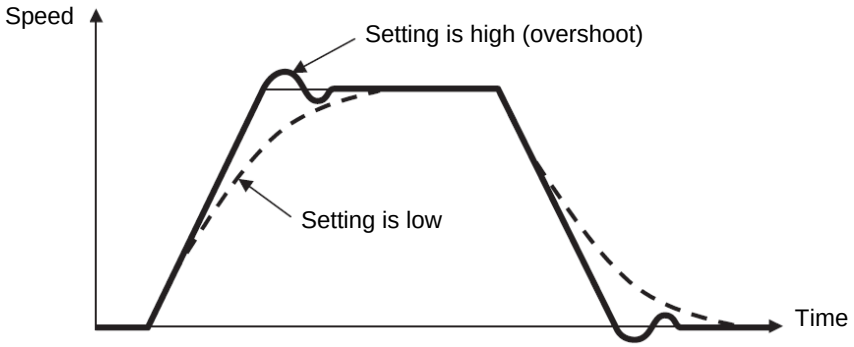
4. Servo Gain Adjustment

Since the servo has been adjusted at the factory in accordance with the standard specification of the actuator, the servo gain need not be changed in normal conditions of use.
However, vibration or noise may occur depending on how the actuator is affixed, specific load condition, and so on, and therefore the parameters relating to servo adjustment are disclosed to allow the customer to take quick actions should adjustment become necessary.
Particularly with custom models (whose ball screw lead or stroke is longer than that of the standard model), vibration/noise may occur due to external conditions.
In this case, the parameters shown below must be changed. Contact IAI for details.

- Position gain

Axis-specific parameter number	Unit	Input range	Default value (reference)
60	/sec	1 ~ 9999	45

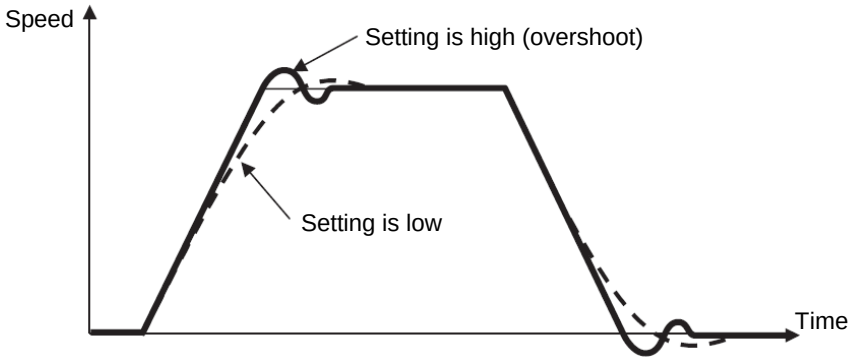
This parameter determines the level of response with respect to a position control loop.
Increasing the setting improves compliance with the position command.
However, increasing the setting too much increases the tendency of the actuator to overshoot.
If the setting is low, compliance with the position command drops and the positioning time increases as a result.



- Speed loop proportional gain

Driver parameter number	Unit	Input range	Default value (reference)
43	---	1 ~ 32767	300

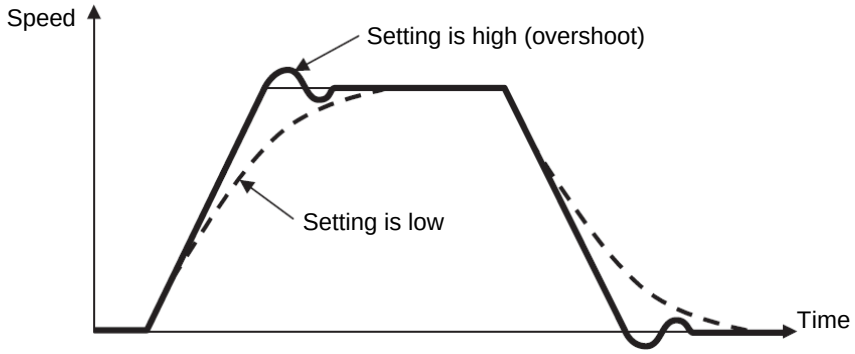
This parameter determines the level of response with respect to a speed control loop.
Increasing the setting improves compliance with the speed command (i.e., servo rigidity increases).
The greater the load inertia, the higher the setting should be.
However, increasing the setting too much increases the tendency of the actuator to overshoot or oscillate, resulting in increased mechanical vibration.



- Speed loop integral gain

Driver parameter number	Unit	Input range	Default value (reference)
44	---	1 ~ 3276700	4601

This parameter determines the level of response with respect to a speed control loop.
Decreasing the setting results in lower response with the speed command and decreases the reactive force upon load change. If the setting is low, compliance with the position command drops and the positioning time increases as a result.
Decreasing the setting too much increases the tendency of the actuator to overshoot or oscillate, resulting in increased mechanical vibration.



- Torque filter time constant

Driver parameter number	Unit	Input range	Default value (reference)
45	---	1 ~ 2500	0

This parameter determines the filter time constant applicable to the torque command.
If the mechanical resonance frequency is equal to or lower than the servo loop response frequency, the motor will vibrate.
This mechanical resonance can be suppressed by increasing the setting of this parameter.
It should be noted, however, that increasing the setting too much may affect the stability of the control system.

- Current control band number

Driver parameter number	Unit	Input range	Default value (reference)
46	---	0 ~ 4	4

This parameter is used to change the current control band of a PI current control system.
The setting should not be changed in normal conditions of use.
Changing the parameter carelessly may affect the stability of the control system and a very dangerous situation may occur.
Changing this parameter may be useful in some situations such as when resonance noise generates, in which case a parameter change can help suppress the noise.
If you wish to change this parameter, please contact IAI.

⊙ List of Parameters

If you have any question regarding changing the parameters, please contact IAI's Sales Engineering Section. After changing a parameter, record the new and old parameter settings.

If you have purchased the PC software, we recommend that you back up the parameters immediately after the controller is delivered and when the system incorporating the controller is started. Since a number of customizing settings use parameters, you should back up the parameters regularly as you back up the programs.
To make the new parameters effective, write them to the flash ROM and then execute a software reset or reconnect the power.
The lists below are examples of default values displayed on the PC software. The default parameter settings vary depending on the operating condition and actuators used.

The values in the "Input range" column represent input limitations on the teaching pendant or in PC software. For the actual settings, enter the values defined in the "Remarks" column.
Values other than those defined in the "Remarks" column are for future expansion, even when they are inside the input range.
Therefore, do not enter values other than those defined in the "Remarks" column.

1. I/O Parameters

1.1 I/O Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
1	I/O port assignment type	1	0 ~ 20		0: Fixed assignment 1: Automatic assignment (Priority: Network I/F module → Standard I/O; * Ports are assigned only for the installed adjoining slots, starting from the standard I/O slot = For safety reasons)
2	Input port start number with fixed standard I/O assignments (I/O1)	000	-1 ~ 599		0 + (Multiple of 8) (Invalid if a negative value is set)
3	Output port start number with fixed standard I/O assignments (I/O1)	300	-1 ~ 599		300 + (Multiple of 8) (Invalid if a negative value is set)
4 ~ 9	For future expansion	-1	-1 ~ 599		
10	Standard I/O error monitor	1	0 ~ 5		0: Do not monitor 1: Monitor 2: Monitor (Do not monitor errors relating to 24-V I/O power source) 3: Monitor (Monitor only errors relating to 24-V I/O power source) * Some exceptions apply. * If this parameter is set to “0” (= Do not monitor) or “2” (= Do not monitor errors relating to 24-V I/O power source), a system error will not generate even when the 24-V I/O power source presents abnormality. However, the actual outputs from digital I/Os will be cut off by circuitry thereafter to protect the controller.
11 ~ 13	For future expansion	1	0 ~ 5		
14	Network system reservation	0	0 ~ 256		
15	Network system reservation	0	0 ~ 256		
16	Network system reservation	-1	-1 ~ 599		
17	Network system reservation	-1	-1 ~ 599		
18	Network system reservation	1	0 ~ 5		
19	(For expansion)	0			
20	Input filtering periods	2	1 ~ 9	msec	Input signal is recognized when the status is held for twice the period set by this parameter.
21	For future expansion (change prohibited)	0	1 ~ 9		
22	For future expansion	0	0 ~ 99999	msec	
23	For future expansion	0H	0H ~ FFFFFFFFH		
24	I/O setting bit pattern 1	10000H	0H ~ FFFFFFFFH		Bits 0 to 3: RDY OUT function selection (System IO) (0: SYSRDY (Software = PIO trigger program can be run) and hardware is normal (emergency stop has not been actuated and hardware error is not present) 1: Error of operation-cancellation level or higher is not present 2: Error of cold-start level or higher is not present) Bits 4 to 7: RDY LED function selection (0: Program can be run 1: Error of operation-cancellation level or higher is not present 2: Error of cold-start level or higher is not present) Bits 8 to 19: (For future expansion) Bits 20 to 23: ALM LED function selection (0: Error of message level or higher error is present 1: Error of operation-cancellation level or higher is present 2: Error of cold-start level or higher is present 3: Error of system-down level or higher is present)
25	I/O setting bit pattern 2	0H	0H ~ FFFFFFFFH		
26	(For expansion)	0			

I/O Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
27	(For expansion)	0			
28	(For expansion)	0			
29	For future expansion	0	0 ~ 599		
30	Input function selection 000	1	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters”
31	Input function selection 001	0	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
32	Input function selection 002	0	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
33	Input function selection 003	0	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
34	Input function selection 004	0	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
35	Input function selection 005	0	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
36	Input function selection 006	0	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
37	Input function selection 007	0	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
38	Input function selection 008	0	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
39	Input function selection 009	0	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
40	Input function selection 010	0	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
41	Input function selection 011	0	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
42	Input function selection 012	0	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
43	Input function selection 013	0	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
44	Input function selection 014	0	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
45	Input function selection 015	0	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
46	Output function selection 300	2	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
47	Output function selection 301	7	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
48	Output function selection 302	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
49	Output function selection 303	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
50	Output function selection 304	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
51	Output function selection 305	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
52	Output function selection 306	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
53	Output function selection 307	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”

I/O Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
54	Output function selection 308	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
55	Output function selection 309	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
56	Output function selection 310	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
57	Output function selection 311	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
58	Output function selection 312	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
59	Output function selection 313	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
60	Output function selection 314	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
61	Output function selection 315	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
62	For future expansion	0	0 ~ 299		
63	For future expansion	0	0 ~ 299		
64 ~ 67	For future expansion	0	0 ~ 299		
68	(For expansion)	0			
69	(For expansion)	0			
70	Unaffected general-purpose output area number (MIN) when all operations/programs are aborted	0	0 ~ 599		* Important: Outputs in this area must be operated under the responsibility of user programs including the “I/O processing program at operation/program abort.” Outputs outside this area will be forcibly turned OFF. (Invalid if “0” is set)
71	Unaffected general-purpose output area number (MAX) when all operations/programs are aborted	0	0 ~ 599		
72	Unaffected general-purpose output area number (MIN) when all operations are paused (servo-axis soft interlock + output-port soft interlock)	300	0 ~ 599		* Important: Outputs in this area must be operated (including recovery) under the responsibility of user programs including the “I/O processing program at all operations pause.” Outputs outside this area will be forcibly turned OFF, reflecting/holding the results of operations performed while all operation pause is effective (only during automatic operation). (Invalid if “0” is set)
73	Unaffected general-purpose output area number (MAX) when all operations are paused (servo-axis soft interlock + output-port soft interlock)	599	0 ~ 599		
74	Number of TP user output ports used (hand, etc.)	0	0 ~ 8		Referenced by TP. (Invalid if “0” is set)
75	TP user output port start number (hand, etc.)	0	0 ~ 599		Referenced by TP.
76	For future expansion	0	0 ~ 599		
77	For future expansion	0	0 ~ 299		
78	Axis pattern permitted to receive PC/TP servo movement command for	0	0B ~ 11111111B		
79	For future expansion	0	0 ~ 299		
80	(PC/TP SIO usage)	0	1 ~ 1		Switching of DIP switches
81	(PC/TP SIO station code)	1	153 ~ 153		Fixed to 153 (99H).
82	(PC/TP SIO reservation)	153			

I/O Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
83	(PC/TP SIO reservation)	0			
84	(PC/TP SIO reservation)	0			
85	(PC/TP SIO reservation)	0			
86	(PC/TP SIO reservation)	0			
87	(PC/TP SIO reservation)	0			
88	(PC/TP SIO reservation)	0			
89	(PC/TP SIO reservation)	0			
90	Usage of SIO channel 0 opened to user (AUTO mode)	0	0 ~ 9		0: Open SEL program 1: Open SEL program (Connect PC/TP when both devices are closed = Used exclusively by the manufacturer) 2: IAI protocol B (Slave)
91	Station code of SIO channel 0 opened to user	153	0 ~ 255		Valid only with IAI protocol.
92	Baud rate type of SIO channel 0 opened to user	0	0 ~ 5		0: 9.6, 1: 19.2, 2: 38.4, 3: 57.6, 4: 76.8, 5: 115.2 kbps
93	Data length of SIO channel 0 opened to user	8	7 ~ 8		
94	Stop bit length of SIO channel 0 opened to user	1	1 ~ 2		
95	Parity type of SIO channel 0 opened to user	0	0 ~ 2		0: None 1: Odd 2: Even
96	Receive operation type of SIO channel 0 opened to user	0	0 ~ 1		0: Forcibly enable receive after send 1: Do not forcibly enable receive at send
97	IAI-protocol minimum response delay for SIO channel 0 opened to user	0	0 ~ 999	msec	Valid only with IAI protocol.
98	(Reservation of SIO channel 0 opened to user)	0			
99	(Reservation of SIO channel 0 opened to user)	0			
100 ~ 115	SIO system reservation	0	0H ~ FFFFFFFFH		
116	(For expansion)	0			
117	(For expansion)	0			
118	(For expansion)	0			
119	(For expansion)	0			
120	Network system reservation	1H	0H ~ FFFFFFFFH		
121	Network system reservation	0	0H ~ FFFFFFFFH		
122	Network system reservation	0	0H ~ FFFFFFFFH		
123	Network system reservation	0H	0H ~ FFFFFFFFH		
124	Network system reservation	0H	0H ~ FFFFFFFFH		
125	Network system reservation	1E32H	0H ~ FFFFFFFFH		
126	Network system reservation	7D007D0H	0H ~ FFFFFFFFH		
127	Network system reservation	5050214H	0H ~ FFFFFFFFH		
128	Network system reservation	0H	0H ~ FFFFFFFFH		
129	Network system reservation	0H	0H ~ FFFFFFFFH		
130	Network system reservation	0H	Reference only (HEX)		
131	Network system reservation	0H	Reference only (HEX)		
132	Network system reservation	192	1 ~ 255		
133	Network system reservation	168	0 ~ 255		
134	Network system reservation	0	0 ~ 255		
135	Network system reservation	1	1 ~ 254		

I/O Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
136	Network system reservation	255	0 ~ 255		
137	Network system reservation	255	0 ~ 255		
138	Network system reservation	255	0 ~ 255		
139	Network system reservation	0	0 ~ 255		
140	Network system reservation	0	0 ~ 255		
141	Network system reservation	0	0 ~ 255		
142	Network system reservation	0	0 ~ 255		
143	Network system reservation	0	0 ~ 255		
144	Network system reservation	64511	1025 ~ 65535		
145	Network system reservation	64512	1025 ~ 65535		
146	Network system reservation	64513	1025 ~ 65535		
147	Network system reservation	64514	1025 ~ 65535		
148	Network system reservation	64515	1025 ~ 65535		
149	Network system reservation	192	0 ~ 255		
150	Network system reservation	168	0 ~ 255		
151	Network system reservation	0	0 ~ 255		
152	Network system reservation	100	0 ~ 254		
153	Network system reservation	64611	0 ~ 65535		
154	Network system reservation	192	0 ~ 255		
155	Network system reservation	168	0 ~ 255		
156	Network system reservation	0	0 ~ 255		
157	Network system reservation	100	0 ~ 254		
158	Network system reservation	64611	0 ~ 65535		
159	Network system reservation	64516	1025 ~ 65535		
160 ~ 169	(For network expansion)	0			
170 ~ 200	(For expansion)	0			
201 ~ 224	SIO system reservation	00000000H	0H ~ FFFFFFFFH		
225 ~ 250	(For expansion)	0			
251	Input function selection 016	9	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
252	Input function selection 017	10	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
253	Input function selection 018	11	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
254	Input function selection 019	12	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
255	Input function selection 020	13	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
256	Input function selection 021	14	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
257	Input function selection 022	15	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
258	Input function selection 023	3	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
259	Input function selection 024	0	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”

I/O Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
260	Input function selection 025	0	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
261	Input function selection 026	0	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
262	Input function selection 027	0	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
263	Input function selection 028	0	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
264	Input function selection 029	0	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
265	Input function selection 030	0	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
266	Input function selection 031	0	0 ~ 99		Input function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
267	Output function selection 316	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
268	Output function selection 317	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
269	Output function selection 318	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
270	Output function selection 319	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
271	Output function selection 320	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
272	Output function selection 321	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
273	Output function selection 322	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
274	Output function selection 323	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
275	Output function selection 324	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
276	Output function selection 325	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
277	Output function selection 326	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
278	Output function selection 327	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
279	Output function selection 328	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
280	Output function selection 329	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
281	Output function selection 330	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
282	Output function selection 331	0	0 ~ 99		Output function specification value * For details, refer to 1.2, “I/O Function Lists” under “I/O Parameters.”
283 ~ 300	(For expansion)	0			

1.2 I/O Function Lists

(1) Input Function List

Input function specification value	Function name	Remarks
0	General-purpose input	
1	Program start signal (BCD) (ON edge)	Specify a BCD program number using the ports to which start-program number specification bits x (input function specification values 9 through 15) are assigned. * To ensure starting of the program, keep these bits ON for at least 100 msec. * The following input functions cannot be assigned at the same time: • Program start signal (BCD) (input function specification value = 1) • Program start signal (BIN) (input function specification value = 2)
2	Program start signal (BIN) (ON edge)	Specify a binary program number using the ports to which start-program number specification bits x (input function specification values 9 through 15) are assigned. * To ensure starting of the program, keep these bits ON for at least 100 msec. * The following input functions cannot be assigned at the same time: • Program start signal (BCD) (input function specification value = 1) • Program start signal (BIN) (input function specification value = 2)
3	Soft reset signal (ON for 1 second)	If the emergency-stop recovery type is set to “Operation continued,” enable the soft reset signal (to ensure the specified operation cancellation method will work.)
4	Servo ON	ON edge: Same as the all-valid-axes servo ON command OFF edge: Same as the all-valid-axes servo OFF command (an interval of at least 1.5 seconds is required). * The signal must be input when the actuator is not operating.
5	Auto-start program start signal	ON edge: Start the program OFF edge: Abort all operations/programs (excluding the I/O processing program at operation/program abort) * Turn ON the signal for at least 100 msec to ensure starting of the program.
6	Soft interlock for all servo axes (OFF level)	Effective when the servo OFF command is not active. Operations will be put on hold if the interlock signal is input during auto operation. Operations will be aborted if the interlock signal is input during non-auto operation.
7	Operation-pause reset signal (ON edge)	
8	Operation pause signal (OFF level)	Effective only during auto operation. * Pause is reset using the operation-pause reset signal.
9	Start-program number specification bit 1 (least significant bit)	* Start-program number specification bits x (input function setting values 9 through 15) cannot be assigned discontinuously from the LSB or in descending order from the LSB (port numbers are not considered). Program No. 1 (BIN or BCD)
10	Start-program number specification bit 2	(Same as “Input function specification value = 9”) Program No. 2 (BIN or BCD)
11	Start-program number specification bit 3	(Same as “Input function specification value = 9”) Program No. 4 (BIN or BCD)
12	Start-program number specification bit 4	(Same as “Input function specification value = 9”) Program No. 8 (BIN or BCD)
13	Start-program number specification bit 5	(Same as “Input function specification value = 9”) Program No. 16 (BIN) or 10 (BCD)
14	Start-program number specification bit 6	(Same as “Input function specification value = 9”) Program No. 32 (BIN) or 20 (BCD)
15	Start-program number specification bit 7	(Same as “Input function specification value = 9”) Program No. 64 (BIN) or 40 (BCD)
16	Error reset (ON edge)	
17	Drive-source cutoff reset input (ON edge) (Effective when the problem factor has been removed)	Drive-source cutoff control is not available for axes whose motor-drive power source is not installed in this controller, or axes whose drive-source cutoff circuit is not controlled by this controller.
18	Home return command signal for all valid axes (ON edge)	The servo must be turned on first (Input function specification value = 4, axis-specific parameter No. 13)
19	Home return command signal for all incremental axes (ON edge)	The servo must be turned on first (Input function specification value = 4, axis-specific parameter No. 13)
20	PC/TP-servo movement command acceptance permission input	* Caution: Ineffective once operation is started.
21	Remote-mode control input	Is the specified DI is ON or the AUTO/MANU switch is set to “MANU,” the system mode will become MANU. * Debug filter is disabled on the remote-mode control input port.
22	Axis 1 forced brake-release input	When the applicable port turns ON, the brake will be unlocked forcibly (pay attention to falling load). * Brake release of the synchronized slave axis conforms to brake release of the synchronized master axis.
23	Axis 2 forced brake-release input	When the applicable port turns ON, the brake will be unlocked forcibly (pay attention to falling load). * Brake release of the synchronized slave axis conforms to brake release of the synchronized master axis.
24 ~ 27	For future expansion	

(2) Output Function List

Output function specification value	Function name	Remarks
0	General-purpose output	
1	Operation-cancellation level or higher error output (ON)	* The following output functions cannot be assigned at the same time: • Operation-cancellation level or higher alarm output (ON) (Output function specification value = 1) • Operation-cancellation level or higher alarm output (OFF) (Output function specification value = 2) • Operation-cancellation level or higher alarm + emergency stop output (ON) (Output function specification value = 3) • Operation-cancellation level or higher alarm + emergency stop output (OFF) (Output function specification value = 4)
2	Operation-cancellation level or higher error output (OFF)	(Same as “Output function specification value = 1”)
3	Operation-cancellation level or higher error + emergency stop output (ON)	(Same as “Output function specification value = 1”)
4	Operation-cancellation level or higher error + emergency stop output (OFF)	(Same as “Output function specification value = 1”)
5	READY output (PIO trigger program operation enabled)	* The following output functions cannot be assigned at the same time: • READY output (PIO trigger program operation enabled) (Output function specification value = 5) • READY output (PIO trigger program operation enabled AND absence of operation-cancellation level or higher error) (Output function specification value = 6) • READY output (PIO trigger program operation enabled AND absence of cold-start level or higher error) (Output function specification value = 7)
6	READY output (PIO trigger program operation enabled AND absence of operation-cancellation level or higher error)	(Same as “Output function specification value = 5”)
7	READY output (PIO trigger program operation enabled AND absence of cold-start level or higher error)	(Same as “Output function specification value = 5”)
8	Emergency stop output (ON)	* The following output functions cannot be assigned at the same time: • Emergency stop output (ON) (Output function specification value = 8) • Emergency stop output (OFF) (Output function specification value = 9)
9	Emergency stop output (OFF)	(Same as “Output function specification value = 8”)
10	AUTO mode output	
11	Auto operation status output	(Other parameter No. 12)
12	All-valid-axes home (= 0) output	* To move the absolute-encoder axis to coordinate 0 or the preset home coordinate, use a MOVP command instead of a HOME command.
13	All-valid-axes home return complete (coordinate confirmed) output	* To move the absolute-encoder axis to coordinate 0 or the preset home coordinate, use a MOVP command instead of a HOME command.
14	All-valid-axes preset home coordinate output	* To move the absolute-encoder axis to coordinate 0 or the preset home coordinate, use a MOVP command instead of a HOME command.
15	Voltage-low warning output for system-memory backup battery	
16	Voltage-low warning output for absolute-data backup battery	All axes are checked by the OR gate. Once an abnormal level has been detected, the signal will remain ON until a power ON reset or software reset is performed.
17	Drive-source cutoff (SDN) notification output	The output port will turn OFF when the drive source is cut off. (* Caution: This notification output is implemented only by software means.)
18	For future expansion	
19	For future expansion	
20 ~ 23	For future expansion	
24	Axis 1 servo-ON status output	
25	Axis 2 servo-ON status output	
26 ~ 29	For future expansion	

The following assignments are prohibited:
- Assign a specification value not included in the I/O function lists.
- Assign the same input function specification value, which is not for general-purpose input, to multiple input ports.
- Assign the same output function specification value, which is not for general-purpose output, to multiple output ports.

(For the conditions associated with each specification value, refer to the Remarks field of the applicable item.)

If a prohibited assignment is set, an error “I/O function assignment error” will generate and all input ports and output ports will become general-purpose inputs and general -purpose outputs, respectively.

* In the positioner mode, input and output function assignments are ignored. Each function will follow the corresponding specification in the positioner mode.

2. Parameters Common to All Axes

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
			~		
1	Valid axis pattern	0000B	00B ~ 11111111B		An OFF bit indicates that no driver is installed.
2	Default override	100	1 ~ 100		Used if not specified in program. (Invalid for SIO operation)
3 ~ 8	(For expansion)	0	~		
9	Physical axis pattern for which enable switch (deadman switch/safety gate) is effective	11111111B	00B ~ 11111111B		Not affected by a BASE command. (To make the enable switch effective for all axes (= it must be effective for all axes, as a rule), always specify “11111111.” Only when “11111111” is set will the enable switch be included in the drive-source cutoff factor. If a value other than “11111111” is set, the drive source will not be cut off and only the servo of the specified axis will be turned off.) * All axes are specified if “Other parameter No. 11: Deadman switch/safety-gate open recovery type” is set to 1 (Reset required for recovery). * The drive-source cannot be cut off for axes whose motor-drive power unit is not housed inside this controller or whose drive-source cutoff circuit is not controlled by this controller. * If the optional (custom) specification is available, the optional (custom) specification will be given priority over the deadman-switch-enabled physical axis/drive-source cutoff specification, servo OFF specification or 7-segment display specification.
10	(For expansion)	0	0H ~ FFFFFFFFH		
11	Default acceleration	30	1 ~ 200	0.01 G	Used if not specified in position data, program or SIO message, etc.
12	Default deceleration	30	1 ~ 200	0.01 G	Used if not specified in position data, program or SIO message, etc.
13	Default speed	30	1 ~ 250	mm/s	Used if not specified in SIO message or position data, when movement is to be continued, etc.
14	Valid selection when operation point data deceleration is 0	0	0 ~ 5		0: “Deceleration = Acceleration” when the deceleration in the operation point data is “0” 1: “Deceleration = 0” when the deceleration in the operation point data is “0”
15	Maximum jog speed when home return is incomplete	30	1 ~ 250	mm/s	
16 ~ 18	(For expansion)	0	~		
19	Processing type upon stationary (non-push) torque limit over				0: Operation-cancellation level error (recommended) 1: Operation cancellation (SEL command outputs will turn OFF) * Driver errors resulting from overload, etc., will be given priority.
20	Maximum operating speed check timing	1	0 ~ 1		0: Check at input 1: Check at operation * If “Check at operation” is selected, the distribution speed (CP) of specified speed or the specified speed (PTP) will be compared against the maximum operating speed of each axis and clamped at the allowable speed. Accordingly, the system can achieve its maximum performance in accordance with the operation command. However, complete check cannot be performed at input (since the command/operation start position is indeterminable). In the case of CP, the distribution speed will vary depending on the operation start position. Therefore, specifying CP at an unspecified position (first point movement, etc.) will cause the speed to fluctuate depending on where the operation is started.
21	Maximum operating speed for input value check	1000	1 ~ 9999	mm/s	If “Input” is selected as the maximum speed check timing, this parameter will be used to check for input error.
22	Maximum acceleration	200	1 ~ 999	0.01 G	
23	Maximum deceleration	200	1 ~ 999	0.01 G	
24	Minimum emergency deceleration	30	1 ~ 300	0.01 G	

Parameters Common to All Axes

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
25	(Acceleration/deceleration at home return (old))	30	1 ~ 300	0.01 G	(Invalid)
26	Acceleration/deceleration specification type	0	Reference only		0: T system, 1: P, M system
27	Master axis type	0	Reference only		0: T system, 1: P system
28	Selection of inching → jog auto-switching prohibition	0	Reference only		0: Execute auto-switching (Continuous button ON timer), 1: Prohibited * Referenced by the PC/TP (no handy terminal auto-switching function)
29	All-axis setting bit pattern 1	0	0H ~ FFFFFFFFH		Bits 0 to 3: Selection of use of last PC/TP inching distance (0: Do not use, 1: Use) * Referenced by the PC/TP (Excluding ANSI-compatible TP) Bits 4 to 7: Overrun (servo) error level (0: Operation-cancellation level, 1: Cold-start level, 2: Operation-cancellation level at reset, thereafter cold-start level) Bits 8 to 11: “Actual-position soft limit over (servo)” error level (0: Operation-cancellation level, 1: Cold-start level, 2: Operation-cancellation level at reset, thereafter cold-start level) Bits 12 to 15: For future expansion Bits 16 to 19: Absolute-data backup battery voltage error level (0: Operation-cancellation level, 1: Message level)
30	Default division angle	150	0 ~ 1200	0.1 degree	
31	Default division distance	0	0 ~ 10000	mm	
32	Arch-trigger start-point check type	0	0 ~ 5		0: Check operation amount and actual position, 1: Check operation amount only
33	Safety speed in manual mode	250	1 ~ 250	mm/s	* This parameter is treated as a value equivalent to or below the minimum value set in “Axis-specific parameter No. 29, VLMX speed” for all valid axes.
34 ~ 100	(For expansion)	0	~		
			~		
101	For future expansion	0H	0H ~ FFFFFFFFH		
102	For future expansion	0H	0H ~ FFFFFFFFH		
103	For future expansion	0H	0H ~ FFFFFFFFH		
104	For future expansion	0H	0H ~ FFFFFFFFH		
105 ~ 120	(For expansion)	0	~		
			~		
			~		

3. Axis-Specific Parameters

No	Parameter name	Default value (Reference)	Input range	Unit	Remarks
			~		
1	Axis operation type	0	0 ~ 1		0: Linear movement axis, 1: Rotational movement axis (Angle control)
2 ~ 5	(For expansion)	0	~		
6	Coordinate/physical-operation direction selection	1	0 ~ 1		0: Motor CCW → Positive direction on the coordinate system 1: Motor CCW → Negative direction on the coordinate system
7	Soft limit +	50000	-99999999 ~ 99999999	0.001 mm	Fixed to 359.999 degrees internally in the index mode. Invalid in the infinite-stroke mode.
8	Soft limit –	0	-99999999 ~ 99999999	0.001 mm	Fixed to 0 degree internally in the index mode. Invalid in the infinite-stroke mode.
9	Soft-limit actual position margin	2000	0 ~ 9999	0.001 mm	Actual position margin in the positioning boundary critical zone in the infinite-stroke mode
10	Home-return method	0	0 ~ 5		0: Search phase Z after end search, 1: Current position 0 home (This parameter can be specified only with an incremental encoder. Pay attention to contact.), 2: Current position = Preset home (This parameter can be specified only with an incremental encoder. Pay attention to contact.)
11	Home-return end-search direction selection	0	0 ~ 1		0: Negative end of the coordinate system 1: Positive end of the coordinate system
12	Home preset value	0	-99999999 ~ 99999999	0.001 mm	(Refer to axis-specific parameter No. 76)
13	SIO/PIO home-return order	0	0 ~ 16		Executed from the smallest one.
14	Home-sensor input polarity	0	0 ~ 2		0: Do not use, 1: Contact a, 2: Contact b
15	For future expansion	0	Reference only		
16	For future expansion	0	Reference only		
17	Initial home-sensor pull-out speed at home return	10	1 ~ 100	mm/sec	
18	For future expansion	100	Reference only		
19	End search speed at home return	20	1 ~ 100	mm/sec	
20	Phase-Z search speed at home return	3	1 ~ 10	mm/sec	Exercise caution, since limitations apply depending on the read/encoder pulse count.
21	Offset travel distance at home return	1000	-99999999 ~ 99999999	0.001 mm	Offset travel distance from the ideal phase-Z position (Positive value = Applied in the direction of moving away from the end) (Refer to axis-specific parameter No. 76) * Note on absolute encoders When a value near an integer multiple of the phase-Z distance (including an offset travel distance of 0) is set in this parameter, the servo will lock above phase Z upon absolute reset. As a result, the coordinates may shift by the pulses corresponding to the phase-Z distance. Therefore, never set a value near an integer multiple of the phase-Z distance. (Provide a sufficient margin with respect to the servo amplitude.)
22	Allowable phase-Z position error check value at home return	200	0 ~ 99999999	0.001 mm	Minimum allowable distance between the end (mechanical or LS) and phase Z in a rotary encoder specification. Phase-Z search limit in a linear encoder specification.
23	Phase-Z count per encoder revolution	1	1 ~ 8		Only “1” can be set, in the case of an absolute encoder. Invalid in the case of a linear encoder.
24	Push stop check time at home return	700	1 ~ 5000	msec	Used to confirm push action during home return.
25	Push stop check time at positioning	500	1 ~ 5000	msec	Used to confirm push action during PUSH command operation.

Axis-Specific Parameters

No	Parameter name	Default value (Reference)	Input range	Unit	Remarks
26	(Phase-Z evacuation distance at absolute home return (old))	1000	0 ~ 99999	0.001 mm	Evacuation distance from the actual phase-Z position (Positive value = Applied in the direction of moving away from the end) (Phase-shift prevention margin) (Refer to axis-specific parameter No. 76)
27	Maximum motor speed	5000	Reference only		In rpm when a rotary encoder is used, or in mm/sec when a linear encoder is used (cannot be changed).
28	Maximum operating speed of each axis	1000	1 ~ 9999	mm/s	
29	VLMX speed	1000	1 ~ 9999	mm/s	During VLMX operation, the maximum operating speed of each axis or VLMX speed, whichever is lower, is used as the maximum speed of the applicable axis.
30	Servo ON check time	150	0 ~ 5000	msec	Brake equipped: Time after receiving a servo-ON start response until start of brake unlocking Brake not equipped: Time after receiving a servo ON start response until transition to an operation-enabled status
31	Offset travel speed at home return	3	1 ~ 500	mm/sec	
32	Actual distance between phase Z and end	-1	-1 ~ 99999	0.001 mm	Absolute distance from the end (mechanical or LS). Obtained automatically if the distance is a negative value. When multiple actuators are combined, it is recommended to write the flash ROM after automatic acquisition. (Refer to axis-specific parameter No. 76)
33	Ideal distance between phase Z and end	0	0 ~ 99999	0.001 mm	Absolute distance from the end (mechanical or LS). (Refer to axis-specific parameter No. 76)
34	Brake equipment specification	0	0 ~ 1		0: Not equipped, 1: Equipped
35	Brake unlock check time	150	0 ~ 3000	msec	Time after receiving a brake-unlock start response until transition to an operation-enabled status
36	Brake lock check time	300	0 ~ 1000	msec	Time after receiving a brake-lock start response until start of servo OFF
37	Encoder linear/rotary type	0	0 ~ 1		0: Rotary encoder 1: Linear encoder
38	Encoder ABS/INC type	0	0 ~ 1		0: INC, 1: ABS
39	Magnetic-pole sensor equipment specification	0	0 ~ 1		0: Not equipped, 1: Equipped
40	For future expansion (change prohibited)	0	0 ~ 1		
41	For future expansion (change prohibited)	25	1 ~ 100	DRVVR	
42	Encoder resolution	800	0 ~ 99999999	Pulse/rev, 0.001 μm/pulse	Pulses (before division)/rev, in the case of a rotary encoder. 0.001 μm/pulse (before division), in the case of a linear encoder.
43	Encoder division ratio	0	-7 ~ 7		Pulses are multiplied by (“n”th power of 1/2).
44	Length measurement correction	0	-99999999 ~ 99999999	0.001 mm/ 1M	Valid only for linear movement axes. (Coordinates other than the encoder reference Z point will change proportionally.)
45 ~ 46	(For expansion)	0			
47	Screw lead	6000	1 ~ 99999999	0.001 mm	Valid only for linear movement axes. Invalid in the case of a linear encoder.
48 ~ 49	(For expansion)	0			
50	Gear ratio numerator	1	1 ~ 99999999		Invalid in the case of a linear encoder.
51	Gear ratio denominator	1	1 ~ 99999999		Invalid in the case of a linear encoder.
52	(For expansion)	0			
53	Setting bit pattern 1 of each axis	0	0H ~ FFFFFFFFH		
54	Travel distance for push stop detection at home return	20	1 ~ 99999	0.001 mm	Used to confirm push action during home return.
55	Travel distance for push stop detection at positioning	30	1 ~ 99999	0.001 mm	Used to confirm push action during PUSH command operation.
56	Push-abort deviation ratio at home return	2000	1 ~ 99999		Deviation is compared against “Steady-state deviation of push speed + Push-speed pulse speed x Abort deviation ratio.”

Axis-Specific Parameters

No	Parameter name	Default value (Reference)	Input range	Unit	Remarks
57	Push-abort deviation ratio at positioning	5000	1 ~ 99999		Deviation is compared against “Steady-state deviation of push speed + Push-speed pulse speed x Abort deviation ratio.”
58	Positioning band	100	1 ~ 99999	0.001 mm	
59	Allowable deviation error ratio (Maximum speed pulse ratio)	138	1 ~ 9999		Deviation is compared against “Steady-state deviation of maximum operating speed of each axis + Pulse speed of maximum operating speed of each axis x Allowable deviation error ratio.”
60	Position gain	45	1 ~ 9999	/s	
61	FAG	0	0 ~ 999		
62	Synchro FB gain	77	0 ~ 1000		
63	Stop special output range	1	0 ~ 9999	Pulse	Invalid if “0” is set.
64	Stop special output value	1	0 ~ 999	DRVVR	
65	Mating synchro-axis number	0	0 ~ 8		Must be input for both axes. (Of the axis pair, the axis with the smaller axis number becomes the master axis. Both axes must have the same resolution characteristics. Commands cannot be issued to the slave axis.) (Invalid if “0” is set)
66	Mode selection for rotational movement axis	0	0 ~ 5		0: Normal, 1: Index mode
67	Short-cut control selection for rotational movement axis	0	0 ~ 5		0: Do not select, 1: Select (Valid only in the index mode AND when an incremental encoder is used)
68	Mode selection for linear movement axis	0	0 ~ 5		0: Normal, 1: Infinite-stroke mode (Note: Positioning boundary applies. This setting can be specified only when an incremental encoder is used.)
69	(For expansion)	0	~		
70	For future expansion	0	Reference only		
71	For future expansion	0	Reference only		
72	DRVVR + offset	0	Reference only	DRVVR	(Change prohibited) To maintain symmetry of the positive and negative sides.
73	DRVVR – offset	0	Reference only	DRVVR	(Change prohibited) To maintain symmetry of the positive and negative sides.
74	For future expansion	0	Reference only		
75	For future expansion	0	Reference only		
76	Home-adjustment parameter set selection	1	Reference only		(Change prohibited) 0: P21 = Phase-Z evacuation distance at INC home return P12 = Ideal phase-Z position coordinate 1: P32 is read automatically even when P33 = 0. P33 = 0 indicates “actual distance.” P21 = Offset travel at home return P12 = Coordinate achieved by offset travel at home return P26 = Invalid (To facilitate adjustment)
77	Synchro S pulse	3	0 ~ 99999	Pulse	
78	Maximum takeoff command amount	0	-3000 ~ 3000	0.001 mm	Maximum lift command amount before brake unlock (Input with sign) (Suppression of momentary drop upon servo ON when a heavy object is placed) * Important: Input using the same sign as the rising coordinate direction. (0.100 mm to 0.500 mm in absolute value as a guideline) * The servo-ON check time (axis-specific parameter No. 30) must also be extended (approx. 1000 to 1500 msec) to provide a sufficient time for rise-direction torque to follow. (Valid only when installation of brake is specified.)
79	Actual takeoff check distance	5	0 ~ 3000	0.001 mm	Absolute value input
80	Maximum forced-feed range	0	0 ~ 9999	0.001 mm	For reduction of settling time. (Invalid range if “0” is set) (Approx. 1.000 mm as a guideline)
81	Minimum forced-feed range	200	0 ~ 9999	0.001 mm	
82	Medium forced-feed range	600	0 ~ 9999	0.001 mm	
83	Absolute synchro slave-axis initialization cancellation	0	0 ~ 5		Valid only with a synchro slave axis.

Axis-Specific Parameters

No	Parameter name	Default value (Reference)	Input range	Unit	Remarks
84	Maximum synchronization correction speed of synchro slave axis	5	0 ~ 100	mm/sec	Maximum travel speed for synchronization position correction of slave axis. Valid only with a synchro slave axis. * Note: Not limited by the safety speed.
85	Home-return acceleration/ deceleration	15	1 ~ 300	0.01 G	
86	Zone 1 MAX	0	-99999999 ~ 99999999	0.001 mm	Valid only when MAX > MIN. * Must be inside the range for at least 3 msec.
87	Zone 1 MIN	0	-99999999 ~ 99999999	0.001 mm	Valid only when MAX > MIN. * Must be inside the range for at least 3 msec.
88	Zone 1 output number	0	0 ~ 899		Physical output port or global flag (Output is invalid if “0” is input; multiple specification is invalid)
89	Zone 2 MAX	0	-99999999 ~ 99999999	0.001 mm	Valid only when MAX > MIN. * Must be inside the range for at least 3 msec.
90	Zone 2 MIN	0	-99999999 ~ 99999999	0.001 mm	Valid only when MAX > MIN. * Must be inside the range for at least 3 msec.
91	Zone 2 output number	0	0 ~ 899		Physical output port or global flag (Output is invalid if “0” is input; multiple specification is invalid)
92	Zone 3 MAX	0	-99999999 ~ 99999999	0.001 mm	Valid only when MAX > MIN. * Must be inside the range for at least 3 msec.
93	Zone 3 MIN	0	-99999999 ~ 99999999	0.001 mm	Valid only when MAX > MIN. * Must be inside the range for at least 3 msec.
94	Zone 3 output number	0	0 ~ 899		Physical output port or global flag (Output is invalid if “0” is input; multiple specification is invalid)
95	Zone 4 MAX	0	-99999999 ~ 99999999	0.001 mm	Valid only when MAX > MIN. * Must be inside the range for at least 3 msec.
96	Zone 4 MIN	0	-99999999 ~ 99999999	0.001 mm	Valid only when MAX > MIN. * Must be inside the range for at least 3 msec.
97	Zone 4 output number	0	0 ~ 899		Physical output port or global flag (Output is invalid if “0” is input; multiple specification is invalid)
98	For future expansion	4	Reference only		
99	For future expansion	2	Reference only		
100 ~ 118	(For expansion)	0	~		
119	FSG	0	0 ~ 100		
120	FFF	10	0 ~ 100		* Change is prohibited unless instructed by the manufacturer.
121~ 170	(For expansion)	0	~		
171		0	~		
172		0	~		
173		0	~		
174		0	~		
175		0	~		
176		0	~		
~ 200	(For expansion)	0	~		

4. Driver Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
1	Type (upper) (Manufacturing information)	Space	Reference only		For adjustment by the manufacturer
2	Type (middle) (Manufacturing information)	Space	Reference only		For adjustment by the manufacturer
3	Type (lower) (Manufacturing information)	Space	Reference only		For adjustment by the manufacturer
4	Manufacturing data (Manufacturing information)	Space	Reference only		For adjustment by the manufacturer
5	Manufacturing data (Manufacturing information)	Space	Reference only		For adjustment by the manufacturer
6	Manufacturing data (Manufacturing information)	Space	Reference only		For adjustment by the manufacturer
7	Manufacturing data (Manufacturing information)	Space	Reference only		For adjustment by the manufacturer
8	Board type (Function information)	0	Reference only		For adjustment by the manufacturer
9	Installation type word 1 (Function information)	0101H	Reference only		For adjustment by the manufacturer
10	Installation type word 2 (Function information)	0000H	Reference only		For adjustment by the manufacturer
11	(Function information)	0000H	Reference only		
12	Software version (Function information)	0000H	Reference only		For adjustment by the manufacturer
13	Maximum supported motor ID number (Function information)	0000H	Reference only		For adjustment by the manufacturer
14	Motor control data use selection (Function information)	0000H	Reference only		For adjustment by the manufacturer
15	(Function information)	0000H	Reference only		For adjustment by the manufacturer
16	(Function information)	0000H	Reference only		For adjustment by the manufacturer
17	(Function information)	0000H	Reference only		For adjustment by the manufacturer
18	(Function information)	0000H	Reference only		For adjustment by the manufacturer
19	(Function information)	0000H	Reference only		For adjustment by the manufacturer
20	(Function information)	0000H	Reference only		For adjustment by the manufacturer
21	(Function information)	0000H	Reference only		For adjustment by the manufacturer
22	(Function information)	0000H	Reference only		For adjustment by the manufacturer
23	(Configuration information)	0000H	Reference only		For adjustment by the manufacturer
24	Configuration capacity (rated motor output) (compatible with E, priority on E) (configuration information)	0014H	Reference only	W	For adjustment by the manufacturer
25	Configuration voltage (motor voltage) (compatible with E, priority on E) (configuration information)	0018H	Reference only	V	For adjustment by the manufacturer
26	Motor/encoder configuration information (compatible with E, priority on E) (configuration information)	0005H	Reference only		For adjustment by the manufacturer
27	(Configuration information)	0000H	Reference only		For adjustment by the manufacturer
28	(Configuration information)	0000H	Reference only		For adjustment by the manufacturer

Driver parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
29	Motor/encoder characteristic word (compatible with E, priority on E) (configuration information)	0000H	Reference only		For adjustment by the manufacturer
30	Motor/encoder control word 1 (compatible with E, priority on E) (configuration information)	5000	Reference only	0.1 K (Kelvin = temperature unit)	For adjustment by the manufacturer
31	Motor/encoder control word 2 (compatible with E, priority on E) (configuration information)	0000H	Reference only		For adjustment by the manufacturer
32	Motor/encoder control word 3 (configuration information) (encoder cable length) [m]	2	Reference only		Encoder cable length (m) If the encoder has been replaced, don't forget to change the setting of this parameter.
33	Motor/encoder control word 4 (configuration information)	14H	Reference only		For adjustment by the manufacturer
34	Motor/encoder control word 5 (configuration information)	0000H	Reference only		For adjustment by the manufacturer
35	(Configuration information)	0000H	Reference only		For adjustment by the manufacturer
36	(Configuration information)	0000H	Reference only		For adjustment by the manufacturer
37	(Configuration information)	0000H	Reference only		For adjustment by the manufacturer
38	Push torque limit at positioning	70	0 ~ 70	%	
39	Push torque limit at home return	40	0 ~ 100	%	
40	Maximum torque limit	100	10 ~ 100	%	*The maximum value that can be set varies depending on the motor, etc.
41	Dynamic brake operation specification	0	0 ~ 1		(Data for other model)
42	Software DB operation specification	0	0 ~ 1		(Data for other model)
43	Speed loop proportional gain	300	1 ~ 32767		Proportional gain
44	Speed loop integral gain	4601	1 ~ 3276700		Integral gain
45	Torque filter time constant	0	0 ~ 2500		
46	Current control band number	4	0 ~ 4		
47	Current ON time for excited-phase signal detection step	10	0 ~ 32767	ms	
48	Excited-phase signal detection method	0	0 ~ 2		(Data for other model)
49	Excited-phase signal detection direction	0	0 ~ 1		0: CW, 1: CCW
50	Excited-phase fixed mode: Torque-limit switching type	1	0 ~ 1		0: Immediate, 1: Added for each cycle
51	Excited-phase fixed mode: Torque limit	35	0 ~ 100	%	Torque limit in the excited-phase fixed mode
52	(For expansion)	0H	00000000H ~ FFFFFFFFH		
53	Current control word 1	0H	Reference only		
54	Current control word 2	0H	Reference only		
55	Current control word 3	0H	Reference only		
56	Current control word 4	0H	Reference only		
57	Current control word 5	0H	Reference only		
58	Current control word 6	0H	Reference only		
59	Current control word 7	0H	00000000H ~ FFFFFFFFH		
60	Current control word 8	0H	00000000H ~ FFFFFFFFH		

Driver parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
61 ~ 67	(For expansion)	0H	00000000H ~ FFFFFFFFH		
68 ~ 97	For future expansion	0H	Reference only		

5. Encoder Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
1	Type (upper) (Manufacturing information)	Space	Reference only		
2	Type (middle) (Manufacturing information)	Space	Reference only		
3	Type (lower) (Manufacturing information)	Space	Reference only		
4	Manufacturing data (Manufacturing information)	Space	Reference only		
5	Manufacturing data (Manufacturing information)	Space	Reference only		
6	Manufacturing data (Manufacturing information)	Space	Reference only		
7	Manufacturing data (Manufacturing information)	Space	Reference only		
8	Board type (Function information)	0	Reference only		
9	Configuration capacity (rated motor output) (compatible with X/E) (function information)	0000H	Reference only	W	For adjustment by the manufacturer
10	Configuration voltage (motor voltage) (compatible with X/E) (function information)	0000H	Reference only	V	For adjustment by the manufacturer
11	Motor/encoder configuration information (compatible with X/E) (function information)	0000H	Reference only		For adjustment by the manufacturer
12	Encoder resolution (upper word) (compatible with X/E) (function information)	0000H	Reference only		For adjustment by the manufacturer
13	Encoder resolution (lower word) (compatible with X/E) (function information)	0000H	Reference only		For adjustment by the manufacturer
14	Motor/encoder characteristic word (compatible with X/E) (function information)	0000H	Reference only		For adjustment by the manufacturer
15	Motor/encoder control word 1 (function information)	0000H	Reference only	0.1 K (Kelvin = temperature unit)	For adjustment by the manufacturer
16	Motor/encoder control word 2 (function information)	0000H	Reference only		For adjustment by the manufacturer
17	Motor/encoder control word 3 (function information)	0000H	Reference only		For adjustment by the manufacturer
18	Motor/encoder control word 4 (function information)	0001H	Reference only		For adjustment by the manufacturer
19	(Function information)	0000H	Reference only		For adjustment by the manufacturer
20	(Function information)	0000H	Reference only		For adjustment by the manufacturer
21	(Function information)	0000H	Reference only		For adjustment by the manufacturer
22	(Function information)	0000H	Reference only		For adjustment by the manufacturer
23 ~ 30	Card parameter (by board type)	0000H	Reference only		For adjustment by the manufacturer

6. I/O Devices

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
1	Type (upper) (Manufacturing information)	Space	Reference only		For adjustment by the manufacturer
2	Type (middle) (Manufacturing information)	Space	Reference only		For adjustment by the manufacturer
3	Type (lower) (Manufacturing information)	Space	Reference only		For adjustment by the manufacturer
4	Manufacturing data (Manufacturing information)	Space	Reference only		For adjustment by the manufacturer
5	Manufacturing data (Manufacturing information)	Space	Reference only		For adjustment by the manufacturer
6	Manufacturing data (Manufacturing information)	Space	Reference only		For adjustment by the manufacturer
7	Manufacturing data (Manufacturing information)	Space	Reference only		For adjustment by the manufacturer
8	Board type (Function information)	0	Reference only		For adjustment by the manufacturer
9	Function information 01 (by board type)	0000H	Reference only		For adjustment by the manufacturer
10	Function information 02 (by board type)	0000H	Reference only		For adjustment by the manufacturer
11	Function information 03 (by board type)	0000H	Reference only		For adjustment by the manufacturer
12	Function information 04 (by board type)	0000H	Reference only		For adjustment by the manufacturer
13	Function information 05 (by board type)	0000H	Reference only		For adjustment by the manufacturer
14	Function information 06 (by board type)	0000H	Reference only		For adjustment by the manufacturer
15	Function information 07 (by board type)	0000H	Reference only		For adjustment by the manufacturer
16	Function information 08 (by board type)	0000H	Reference only		For adjustment by the manufacturer
17	Function information 09 (by board type)	0000H	Reference only		For adjustment by the manufacturer
18	Function information 10 (by board type)	0000H	Reference only		For adjustment by the manufacturer
19	Function information 11 (by board type)	0000H	Reference only		For adjustment by the manufacturer
20	Function information 12 (by board type)	0000H	Reference only		For adjustment by the manufacturer
21	Function information 13 (by board type)	0000H	Reference only		For adjustment by the manufacturer
22	Function information 14 (by board type)	0000H	Reference only		For adjustment by the manufacturer
23 ~ 52	Device parameter (by board type)	0000H	Reference only		For adjustment by the manufacturer
53 ~ 82	Query information 01 to 30 (by board type)	0000H	Reference only		For adjustment by the manufacturer

7. Other Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
1	Auto-start program number	0	0 ~ 64		(Invalid if “0” is set)
2	I/O processing program number at operation/program abort	0	0 ~ 64		The start trigger is determined from the “I/O processing program start type at operation/program abort.” (Note: This program will be started before confirming an abort of other programs.) (Invalid if “0” is set) * If the setting is valid, the number of user program tasks that can be used will decrease by 1.
3	I/O processing program number at all operation pause	0	0 ~ 64		This program will be started when an all-operation-pause command is issued due to an all-operation-pause factor. (Only when a program is running) (Invalid if “0” is set) * If the setting is valid, the number of user program tasks that can be used will decrease by 1.
4	Program abort type at error	0	0 ~ 5		0: Cancel only the program in which an error of operation-cancellation level or higher has generated. (If the error requires the drive source to be cut off or a servo-OFF or all-axis servo-OFF request to be issued, all programs other than the “I/O processing program at operation/program abort” will be cancelled.) 1: Cancel all programs other than the “I/O processing program at operation/program abort” when an error of operation-cancellation level or higher has generated.
5	I/O processing program start type at operation/program abort	0	0 ~ 5		0: When all-operation-cancellation factor has generated (Only when a program is running) 1: When all-operation-cancellation factor has generated (Always) 2: All-operation-cancellation factor + Error of operation-cancellation level or higher (“Other parameter No. 4 = 0” is considered) (Only when a program is running) 3: All-operation-cancellation factor + Error of operation-cancellation level or higher (“Other parameter No. 4 = 0” is considered) (Always)
6	PC/TP reconnection delay at software reset	14000	1 ~ 99999	msec	* The setting will become effective after the controller, PC or TP is restarted.
7	Auto program start setting	1	0 ~ 5		0: Do not start the auto-start program upon power ON reset/software reset 1: Start the auto-start program
8	(For expansion)	0			
9	For future expansion (change prohibited)	0	0 ~ 2		
10	Emergency-stop recovery type	0	0 ~ 4		0: Abort operations/programs 1: Recovery after reset 2: Operation continued (Only during automatic operation. * Operation commands from the PC/TP will be aborted on the PC/TP side.) 3: Abort operations/programs (Software reset when the emergency stop is reset. The home-return completion status of incremental-encoder axes will be reset (EG approximation swap).) 4: Abort operations/programs (Error reset (only with an error of operation-cancellation level or lower) and auto-start program start (only if AUTO mode AND other parameter No. 7 = 1 AND I/O parameter “Input function selection” ≠ 17 AND all-operation-cancellation factor is not present) when the emergency stop is reset. There must be a minimum interval of 1 second after an emergency stop is actuated before it is reset. The home-return completion status of incremental-encoder axes will be retained.
11	Enable switch (deadman/enable switch) recovery type	0	0 ~ 2		0: Abort operations/programs 1: Recovery after reset 2: Operation continued (Only during automatic operation. * Operation commands from the PC/TP will be aborted on the PC/TP side.)

Other Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
12	Automatic operation recognition type	0	0 ~ 3		0: Program is running AND all-operation-cancellation factor is not present 1: [Program is running OR in AUTO mode] AND all-operation-cancellation factor is not present
13 ~ 19	(For expansion)	0			
20	System-memory backup battery installation function type	0	0 ~ 2		0: Not installed (SEL global data/error lists cannot be recovered from the flash ROM) 1: Not installed (SEL global data/error lists can be recovered from the flash ROM) 2: Installed * When the power is turned on without battery installed, point data can be copied from the flash ROM. * Use of setting “1” will be prohibited for the time being due to limitations. * When point data is lost due to a battery error, the point data valid before the flash ROM was written can be restored → Input “0” (not installed) and transfer the setting to the controller, and then perform a software reset without writing the flash ROM. The point data last written to the flash ROM will be restored. Thereafter, reset this parameter to the original value. (No remedy is available for recovery of SEL global data/error lists.)
21	Manual mode type	0	0 ~ 5		0: Always enable edit and SIO/PIO start (Initial condition after connection = With safety speed) 1: Select edit and start (with password) (EU, etc.) 2: Always enable edit and SIO/PIO start (Initial condition after connection = Without safety speed (cancellation)) * Referenced by the PC/TP.
22	Control use region	0	0 ~ 99		0: J, 1: E, 2: EU
23	PSIZ command function type	0	0 ~ 5		0: Maximum number of point data areas 1: Number of point data used
24	Local variable number for storing SEL communication command return code	99	1 ~ 99 1001 ~ 1099		
25	Operation mode type	0	0 ~ 16		0: Program mode 1 to 16: Positioner mode
26 ~ 29	(For expansion)	0			
30	Option Password 00	0H	0H ~ FFFFFFFFH		HOME command option (Change prohibited) * Change is prohibited unless instructed by the manufacturer.
31	Option Password 01	0H	0H ~ FFFFFFFFH		Reserved (Change prohibited) * Change is prohibited unless instructed by the manufacturer.
32	Option Password 02	0H	0H ~ FFFFFFFFH		Reserved (Change prohibited) * Change is prohibited unless instructed by the manufacturer.
33 ~ 35	(For expansion)	0	0H ~ FFFFFFFFH		

Other Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
36	PC/TP data protect setting (Program)	0H	0H ~ FFFFFFFFH		Bits 0 to 3: Protect type (0: Read/write, 1: Read only, 2: No read/write) Bits 4 to 7: Protect release method (0: Special operation) Bits 8 to 11: Protect range maximum number (1's place, BCD) Bits 12 to 15: Protect range maximum number (10's place, BCD) Bits 16 to 19: Protect range minimum number (1's place, BCD) Bits 20 to 23: Protect range minimum number (10's place, BCD) * Referenced by the PC/TP
37	PC/TP data protect setting (Position)	0H	0H ~ FFFFFFFFH		Bits 0 to 3: Protect type (0: Read/write, 1: Read only, 2: No read/write) Bits 4 to 7: Protect release method (0: Special operation) Bits 8 to 11: Protect range maximum number (10's place, BCD) Bits 12 to 15: Protect range maximum number (100's place, BCD) Bits 16 to 19: Protect range maximum number (1000's place, BCD) Bits 20 to 23: Protect range minimum number (10's place, BCD) Bits 24 to 27: Protect range minimum number (100's place, BCD) Bits 28 to 31: Protect range minimum number (1000's place, BCD) * The value in the 1's place is considered "0" for both the protect range maximum/minimum numbers. * Referenced by the PC/TP
38	PC/TP data protect setting (Symbol, parameter)	0H	0H ~ FFFFFFFFH		Bits 0 to 3: Protect type (Parameter) (0: Read/write, 1: Read only, 2: No read/write) Bits 4 to 7: Protect release method (Parameter) (0: Special operation) Bits 8 to 11: Protect type (Symbol) (0: Read/write, 1: Read only, 2: No read/write) Bits 12 to 15: Protect release method (Symbol) (0: Special operation) * Referenced by the PC/TP
39	(For future expansion)	0H	0H ~ FFFFFFFFH		

Other Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
40	EEPROM information check type	02H	0H ~ FFFFFFFFH		0: Disable checksum, 1: Enable checksum Bit 0 = (For future expansion) Bit 1 = Encoder Bits 2 to 7 = (For future expansion) 0: Do not use EEPROM, 1: Use EEPROM Bits 16 to 23 = (For future expansion)
41	Hardware information check type	0H	0H ~ FFFFFFFFH		Bits 0 = (For future expansion)
42	Hardware test type	0H	0H ~ FFFFFFFFH		Bits 0 to 2 = (For future expansion)
43	For future expansion	0H	0H ~ FFFFFFFFH		
44	(For expansion)	0			
45	Special start condition setting	0	0H ~ FFFFFFFFH		Bits 0 to 3: Enable start from PC/TP in AUTO mode = Used exclusively by the manufacturer (0: Do not enable, 1: Enable) Bits 4 to 7: PIO program start (Input port 000) Single start selection (0: Normal, 1: Single start) * In accordance with the input port for which the I/O parameter “Input function selection” has been set to “1” or “2.” * When single start is selected, the next PIO program start will not be accepted as long as a program with the same program number as the one started by the last PIO program start is running. Bits 8 to 11: Permission of auto program start when all-operation-cancellation factor is present (0: Do not permit, 1: Permit) Bits 12 to 15: Permission of ON edge acceptance for PIO-program start when all-operation-cancellation factor is present (0: Do not permit, 1: Permit) * In accordance with the input port for which the I/O parameter “Input function selection” has been set to “1” or “2.” * This parameter specifies an ON-edge acceptance condition. If the starting condition is not satisfied, an “Error No. A1E: Start condition non-satisfaction error” will generate.
46	Other setting bit pattern 1	2011H	0H ~ FFFFFFFFH		Bits 0 to 3: Variable-value format type in response message to real-number/variable query (0: Big endian with four upper/lower binary-converted bytes reversed, 1: Big endian) Bits 4 to 7: Decimal-place rounding selection for real-number → integer-variable assignment in LET/TRAN commands (0: Do not round, 1: Round) Bits 8 to 11: For future expansion * Change strictly prohibited unless specified by the manufacturer. Bits 12 to 15: Selection of processing to be performed when subroutine first step input condition is not specified when TPCD command = 1 (0: Do not execute, 1: Execute, 2: Error)
47 ~ 48	(For expansion)	0			

Other Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
49	Panel 7-segment display data type	0	0 ~ 9		0: Display controller status 1: Display motor current indicator The current pattern of each axis is displayed instead of "ready status" or "program run number." "Minimum indicator-displayed axis number" (far-right column) is specified by "Other parameter No. 50."  0 < Motor current to rating ratio (%) ≤ 25  25 < Motor current to rating ratio (%) ≤ 50  50 < Motor current to rating ratio (%) ≤ 75  75 < Motor current to rating ratio (%) ≤ 100  100 < Motor current to rating ratio (%) ≤ 150  150 < Motor current to rating ratio (%) ≤ 200  200 < Motor current to rating ratio (%) 2: Display user information number (U001 to U999) The user information number is displayed instead of "ready status" or "program run number" only when the user information number is not "0." "Global integer variable number for specifying user information number" is specified by "Other parameter No. 50."
50	Auxiliary specification for panel 7-segment display data type	0	-99999999 ~ 99999999		* Refer to the Remarks field for "Other parameter No. 49."
51	Monitoring-data buffering period	10	1 ~ 100	msec	
52 ~ 70	(For expansion)	0			
71	Positioner mode parameter 1	0	-99999999 ~ 99999999		
72	Positioner mode parameter 2	0	-99999999 ~ 99999999		
73	Positioner mode parameter 3	0	-99999999 ~ 99999999		
74	Positioner mode parameter 4	0	-99999999 ~ 99999999		
75	Positioner mode parameter 5	0	-99999999 ~ 99999999		
70 ~ 100	(For expansion)	0			

8. Manual Operation Types

The selectable operation types will vary depending on the setting of the “Manual operation type” parameter (Other parameter No. 21).

(1) PC software

[1] Setting = 0 (Always enable edit and SIO/PIO start)

Operation type	Password	Functions				
		Edit	Safety speed	Jog, move, continuous move	SIO program start	PIO program start
With safety speed	Not required.	○	○	○	○	○
Without safety speed	Not required.	○		○	○	○

[2] Setting = 1 (Select edit and start (with password))

Operation type	Password	Functions				
		Edit	Safety speed	Jog, move, continuous move	SIO program start	PIO program start
Edit and jog	Not required.	○	○	○		
SIO start and jog (safety speed)	1817 (*1)		○	○	○	
SIO start and jog	1818 (*1)			○	○	
SIO/PIO start and jog	1819 (*1)			○	○	○

(*1) PC software version 0.0.6.0 or later (“0000” in versions 0.0.0.0 through 0.0.5.x)

(2) Teaching pendant

[1] Setting = 0 (Always enable edit and SIO/PIO start)

Safety-speed enable selection	Password	Functions				
		Edit	Safety speed	Jog, move, continuous move	SIO program start	PIO program start
Enable	Not required.	○	○	○	○	○
Disable	Not required.	○		○	○	○

[2] Setting = 1 (Select edit and start (with password))

Safety-speed enable selection	Password	Functions					
		Edit	Safety speed	Jog, move, continuous move	SIO program start	PIO program start	*2
Enable	Not required.	○	○	○	○	(*3)	
Disable	1818 (*1)	○		○	○	(*3)	

PIO start prohibition selection	Password	Functions					
		Edit	Safety speed	Jog, move, continuous move	SIO program start	PIO program start	*2
Prohibit	Not required.	○	(*4)	○	○		
Enable	1819 (*1)	○	(*4)	○	○	○	

(*1) Teaching pendant application version 0.02 or later (not supported by version 0.01 or earlier)
(*2) PIO program start is enabled only in modes other than the edit mode.
(*3) In accordance with the “PIO start prohibition selection” setting.
(*4) In accordance with the “Safety-speed enable” setting.

⊙ Combination Table of PSEL Linear/Rotary Control Parameters

Axis-specific parameter No. 1, Axis operation type	Axis-specific parameter No. 68, Mode selection for linear movement axis	Axis-specific parameter No. 66, Mode selection for rotational movement axis	Axis-specific parameter No. 67, Short-cut control selection for rotational movement axis	Permitted encoder processing method		Expression of current position (approx.)	Axis-specific parameter No. 7, Soft limit +	Axis-specific parameter No. 8, Soft limit -	Axis-specific parameter No. 44, Length measurement correction	Axis-specific parameter No. 47, Screw lead	Axis-specific parameter No. 50, Gear ratio numerator	Axis-specific parameter No. 51, Gear ratio denominator	Input unit
				Simplified absolute unit	INC								
0 (Linear movement axis)	0 (Normal mode)	Invalid	Invalid	○	○	Counter range	Valid	Valid	Valid	Valid	Valid	Valid	• Distance mm • Speed mm/sec • Acceleration/ deceleration G
	1 (Infinite stroke mode) *Duty-cycle timeout check must be examined.			X	○	-10000 ~ 9999.99 (Rotary)	Invalid (Note)	Invalid (Note)					
1 (Rotational movement axis)	Invalid	0 (Normal mode)	0 (Short-cut control not selected) * Specification of values other than "0" is prohibited in the normal mode.	○	○	Counter range	Valid	Valid	Invalid	Invalid	Valid	Valid	• Angle mm → deg • Angular velocity mm/sec → deg/sec • Angular acceleration/deceleration G = 9807 mm/sec2 → 9807 deg/sec2 = 9807 x 2π / 360 rad/sec2 * Degree values indicate the angles of the rotating body at the end.
		1 (Index mode)	0 (Short-cut control not selected)	○	○	Counter range	Invalid (Fixed to "359.999" internally.)	Invalid (Fixed to "0" internally.)					
			1 (Short-cut control selected)	X	○	0 ~ 359.999 (Rotary)							

(Note) If a positioning command other than "J□W□" is issued by specifying a value outside the coordinate range of approx. -9990 to 9990, an error (CBE, "Target-locus boundary over error") will occur.
If a positioning command other than "J□W□" is issued outside the coordinate range of approx. -9990 to 9990, an error (CC5, "Positioning boundary pull-out error") will occur.

Note: On actuators using a simplified absolute unit, the following settings are disabled:
- Set the infinite stroke mode ("1") for a linear movement axis
- Select the short-cut control ("1") for a rotational movement axis in the index mode

⊙ Error Level Control

Error level	System error assignment source	Error No. (HEX)	Display (7-segment display, etc.)	Error list (Application only)	Error LED output (MAIN only)	Program run (Application only)		Error reset (Application only)	Remarks
						Other parameter No. 4 = 0	Other parameter No. 4 = 1		
Secret level	MAIN application	800 ~ 88F		○					Special error level provided for maintenance purposes
	MAIN core	890 ~ 8AF							
	PC	8B0 ~ 8DF							
	TP	8E0 ~ 8FF							
Message level	MAIN application		○	△ (Battery and fieldbus errors will be registered in an error list.)				Enabled.	Status display, input error, etc.
	MAIN core	-							
	PC								
	PC (Update tool)								
	TP								
	MAIN application	200 ~ 24F							
	MAIN core	-							
	PC	250 ~ 29F							
	PC (Update tool)	2A0 ~ 2CF							
	TP	2D0 ~ 2FF							
	MAIN application	900 ~ 93F							
	MAIN core	940 ~ 97F							
	PC	980 ~ 9AF							
	PC (Update tool)	9B0 ~ 9BF							
	TP	9C0 ~ 9FF							
	MAIN application	A00 ~ A6F							
	MAIN core	A70 ~ A9F							
	PC	AA0 ~ ACF							
	TP	AD0 ~ AFF							
Operation-cancellation level	MAIN application		○	○		The program in which the error generated will be cancelled. (Except for axis errors, a cancellation factor is present only for the moment the error occurs.) * However, in the case of an error requiring servo OFF or all-axis servo OFF, all programs other than the “I/O processing program at operation/program abort” will be cancelled.	All programs other than the “I/O processing program at operation/program abort” will be cancelled. (Except for axis errors, a cancellation factor is present only for the moment the error occurs.)	Enabled.	Errors affecting operation. The system will attempt to reset minor errors below this level using an auto-reset function via external active command (SIO/PIO) (application only).
	MAIN core	-							
	PC								
	PC (Update tool)								
	TP								
	MAIN application	400 ~4CF							
	MAIN core	-							
	PC	4D0 ~ 4DF							
	PC (Update tool)	4E0 ~ 4EF							
	TP	4F0 ~ 4FF							

Error level	System error assignment source	Error No. (HEX)	Display (7-segment display, etc.)	Error list (Application only)	Error LED output (MAIN only)	Program run (Application only)		Error reset (Application only)	Remarks
						Other parameter No. 4 = 0	Other parameter No. 4 = 1		
Operation-cancellation level	MAIN application	B00 ~ B9F	○	○		The program in which the error generated will be cancelled. (Except for axis errors, a cancellation factor is present only for the moment the error occurs.) * However, in the case of an error requiring servo OFF or all-axis servo OFF, all programs other than the "I/O processing program at operation/program abort" will be cancelled.	All programs other than the "I/O processing program at operation/program abort" will be cancelled. (Except for axis errors, a cancellation factor is present only for the moment the error occurs.)	Enabled.	Errors affecting operation. The system will attempt to reset minor errors below this level using an auto-reset function via external active command (SIO/PIO) (application only).
	MAIN core	BA0 ~ BBF							
	PC	BC0 ~ BDF							
	TP	BE0 ~ BFF							
	MAIN application	C00 ~ CCF							
	MAIN core	CD0 ~ CDF							
	PC	CE0 ~ CEF							
	TP	CF0 ~ CFF							
Cold-start level	MAIN application		○	○	○ (Core only)	The program in which the error generated will be cancelled. * However, in the case of an error requiring drive-source cutoff, servo OFF or all-axis servo OFF (initialization error, power error, etc.), all programs other than the "I/O processing program at operation/program abort" will be cancelled.	All programs other than the "I/O processing program at operation/program abort" will be cancelled.	Not enabled.	The controller power must be reconnected (MAIN only). (The CPU and OS will run properly.)
	MAIN core	-							
	PC								
	PC (Update tool)								
	TP								
	MAIN application	600 ~ 6CF							
	MAIN core	-							
	PC	6D0 ~ 6DF							
	PC (Update tool)	6E0 ~ 6EF							
	TP	6F0 ~ 6FF							
	MAIN application	D00 ~ D8F							
	MAIN core	D90 ~ DAF							
	PC	DB0 ~ DCF							
	PC (Update tool)	DD0 ~ DDF							
	TP	DE0 ~ DFF							
	MAIN application	E00 ~ E8F							
	MAIN core	E90 ~ EBF							
	PC	EC0 ~ EDF							
	TP	EE0 ~ EFF							
System-down level	MAIN application		○	○	○	All programs will be cancelled.		Not enabled.	The controller power must be reconnected (MAIN only). (The CPU and OS will not run.)
	MAIN core	-							
	PC								
	PC (Update tool)								
	TP								
	MAIN application	FF0 ~ FBF							
	MAIN core	FC0 ~ FCF							
	PC	FD0 ~ FDF							
	TP	FE0 ~ FEF							

Note) Secret-level errors are not actual errors. Internal statuses are registered in an error list as secret-level errors, when deemed necessary, in order to facilitate error analysis.
PC: PC software TP: Teaching pendant

⊙ Error List (MAIN application) (In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
200	Encoder parameter data version mismatch warning	The version of encoder parameter data is not supported by this controller. Update the encoder parameters.
203	Drive-source cutoff relay DET (MELT) error	The drive-source cutoff relay may have fused.
206	Updating system mode error (IAI protocol)	An update command was received other than in the update mode.
207	Update file name error (IAI protocol)	The name of the update program file selected in the update mode is invalid. Select the correct file and repeat the updating procedure from the beginning.
208	Time data error	The time data is invalid. Check the data.
209	Unsupported control constant table ID error	The control constant table ID is not supported. Check the data.
20A	Control constant table change/query error	The message of the control constant table change/query command contains error. Check the message that has been sent.
20B	Control constant table write data type specification error	The specified control constant table write data type is invalid. Check the message that has been sent.
20C	Control constant table management information mismatch error	The management information regarding the control constant table is invalid. Confirm that the control constant table is supported by the controller.
20D	Flash busy reset timeout error	Error erasing/writing the flash ROM
20E	Motorola S-byte count error	The update program file is invalid. Check the file.
20F	Updating target specification error (Received by the application)	The system application received an updating target specification command. To update the program, restart the controller and repeat the updating procedure from the beginning.
210	Program-related data change/run command rejection error in positioner mode	Change of program-related data or running of programs is prohibited in the positioner mode.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
406	Flash busy reset timeout	Error erasing/writing the flash ROM
407	Control constant table management information mismatch error	The management information regarding the control constant table is invalid. If this error occurs when the controller is started, the control constant table may need to be updated.
408	Control constant table ID error	The control constant table ID is invalid.
409	Encoder control constant error (power-source voltage control)	An encoder control constant relating to power-source voltage control is invalid. The encoder power-source voltage cannot be adjusted (the encoder power will be supplied without voltage adjustment).
40A	Encoder power-source voltage calculation error	The encoder power-source voltage cannot be adjusted (the encoder power will be supplied without voltage adjustment). Check the “motor/encoder configuration information” in driver parameter No. 26 and encoder parameter No. 11.
40B	Speed control parameter calculation error	Check driver parameter Nos. 38, 39, 40, 43, 44, 45, etc.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
605	Forced discharge error	Abnormal forced discharge. The drive-source cutoff relay may be abnormal. The power must be reconnected.
606	Regenerative discharge error	Abnormal regenerative discharge. The power must be reconnected.
607	Motor power-source voltage low error	Low voltage was detected in the motor power circuit.
608	Power-supply board FRCDCSTR-ON timeout error	Power-supply board FRCDCSTR-ON could not be confirmed within the specified time.
609	Power-supply board RBONSTR-ON timeout error	Power-supply board RBONSTR-ON could not be confirmed within the specified time.
60A	Power-supply board RBONSTR-OFF timeout error	Power-supply board RBONSTR-OFF could not be confirmed within the specified time.
60B	Power-supply board FRCDCSTR-OFF timeout error	Power-supply board FRCDCSTR-OFF could not be confirmed within the specified time.
60C	Power-system overheat error	An overheated power-supply board, regenerative resistor, etc., was detected. The power must be reconnected.
60D	Slave board CPU ready OFF error (other than power supply)	A ready status of the driver board, etc. (other than power-supply board) cannot be confirmed.
60E	Dynamic brake ON/OFF timeout error	Dynamic brake ON/OFF cannot be confirmed within the specified time.
613	Driver synchronous communication driver read error	A communication failure occurred between the driver board and FPGA (main).
614	Driver synchronous communication LRC error	A communication failure occurred between the driver board and FPGA (main).
615	Driver synchronous communication toggle error	A communication failure occurred between the driver board and FPGA (main).
623	Driver error detail code acquisition error	A driver error occurred, but an error detail code could not be acquired.
624	Undefined driver error	A driver error occurred.
625	Driver-side detection synchronous communication error	A communication failure occurred between the driver board and FPGA (main).
626	Driver IPM15V voltage low error	A low voltage was detected in the driver IPM15V circuit.
627	Driver current detection A/D offset over error	A driver current detection A/D offset error was detected.
628	Driver error	(Driver error for future expansion)
629	Driver error	(Driver error for future expansion)
62A	Driver error	(Driver error for future expansion)
62B	Driver error	(Driver error for future expansion)
62C	Driver error	(Driver error for future expansion)
62D	Driver error	(Driver error for future expansion)
62E	Driver error	(Driver error for future expansion)
62F	Driver error	(Driver error for future expansion)

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
630	Updating system code error (Application detection)	The updating system code is invalid.
631	Updating unit code error (Application detection)	The updating unit code is invalid.
632	Updating device number error (Application detection)	The updating device number is invalid.
633	Feedback pulse synchronization error (Detected in the speed loop)	Abnormal feedback pulse synchronization (detected in the speed loop).
634	Feedback pulse synchronization error (Detected in the position loop)	Abnormal feedback pulse synchronization (detected in the position loop).
635	Deadman/enable switch requiring reset recovery open	Reset the deadman/enable switch, and then reconnect the power.
636	Serial encoder command busy error	The system was busy when the serial encoder command was issued.
637	Serial encoder command timeout error	Completion of the serial encoder command cannot be confirmed after the specified time.
638	Speed control parameter setting command busy error	The system was busy when the speed control parameter setting command was issued.
639	Speed control parameter setting command timeout error	Completion of the speed control parameter setting command cannot be confirmed after the specified time.
63A	ABZ encoder logic error	An encoder phase-A/B electrical level pattern error was detected. The power must be reconnected.
63B	Encoder/motor control constant table flash ROM status error	Data is not written correctly to the flash ROM, or the data is of an old, incompatible version.
63C	Encoder/motor control constant table checksum error	The flash ROM data is corrupted.
63D	ABZ encoder specification error	An ABZ encoder cannot be installed for this axis. Check the “motor/encoder configuration information” in driver parameter No. 26 and encoder parameter No. 11.
63E	ABZ encoder magnetic-pole sensor signal logic error	Check if the encoder cable is connected.
63F	Encoder control constant error	The encoder control constant is invalid.
640	Motor control constant error	The motor control constant is invalid.
641	Encoder power-source voltage control parameter error	Check driver parameter Nos. 32, 33, etc.
642	Speed loop parameter error	Check driver parameter Nos. 43, 44, 45, etc.
643	Encoder resolution division error	Check “Axis-specific parameter No. 43: Encoder division ratio.”
644	Encoder/motor combination mismatch error (encoder resolution)	Check driver parameter No. 26, encoder parameter No. 11.
645	DAC transfer completion check timeout error when encoder power was supplied	A timeout occurred during DAC transfer when the encoder power was supplied.
646	Encoder EEPROM read busy error	The encoder is faulty or an encoder communication failure occurred.
647	Encoder EEPROM write address mismatch error	The encoder is faulty or an encoder communication failure occurred.
648	Encoder EEPROM read address mismatch error	The encoder is faulty or an encoder communication failure occurred.
649	Undefined serial encoder installation error	Installation of serial encoder is not defined. Check the “motor/encoder configuration information” in driver parameter No. 26 and encoder parameter No. 11.
64A	Undefined serial encoder command error	The serial encoder command is not defined.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
64B	Serial encoder command packet error	The serial encoder command packet is invalid.
64C	1-revolution data reset error at servo ON (serial encoder command)	A 1-revolution data reset was commanded when the servo was ON. Turn OFF the servo.
64D	Encoder reset command timeout error (serial encoder command)	An encoder communication failure.
64E	ABS data query command timeout error (serial encoder command)	An encoder communication failure.
64F	Encoder error reset error at servo ON (serial encoder command)	Turn OFF the servo before resetting an encoder error.
650	Encoder receive timeout error (during initialization communication)	An encoder communication failure.
651	Speed control interruption control job error	The speed control interruption error job is invalid.
652	Serial encoder command control job error	The serial encoder command control job is invalid.
653	Encoder control job logic error	The encoder control job logic is invalid.
654		
655	Encoder receive timeout error at serial encoder command issuance	An encoder communication failure.
656	Torque limit logic error	The torque limit logic is invalid.
657	Torque limit parameter error	Check driver parameter Nos. 38, 39, 40, etc.
658	Movement error during ABZ encoder counter initialization	Axis movement was detected while initializing the ABZ encoder counter following power on. The power may have been turned on or a software reset executed while the actuator was moving due to external force such as reactive force of a self-supported cable or while the installation location was vibrating.
65A	Unsupported encoder ID error	The encoder is not supported. No encoder control constant record is available that corresponds to the encoder ID. Check the installed encoder.
65B	Unsupported encoder error (main information)	The encoder is not supported. No encoder control constant record is available that corresponds to the encoder ID, or the record is invalid. Check the “motor/encoder configuration information” in driver parameter No. 26 and encoder parameter No. 11.
65C	Unsupported motor error (main information)	The motor is not supported. No motor control constant record is available that corresponds to the motor ID, or the record is invalid. Check the “motor/encoder configuration information” in driver parameter No. 26 and encoder parameter No. 11.
65D	Unsupported motor error (driver information)	The motor is not supported. The motor ID bit number is outside the range of “maximum supported motor ID number” when the driver parameter, “Use motor control data in driver flash ROM” is specified. Check the “motor/encoder configuration information” in driver parameter No. 26 and encoder parameter No. 11.
65E	Current detection circuit type mismatch error	The motor control constant, “Current detection circuit specification” does not match the driver parameter, “Installation type word 1, current detection circuit type.” Check the “motor/encoder configuration information” in driver parameter No. 26 and encoder parameter No. 11.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
65F	Main/driver motor control data mismatch error	A motor control constant does not match the corresponding driver parameter (rated speed, maximum speed, rated current, maximum current number of pole pairs, linear motor lead, linear motor specification). Check the “motor/encoder configuration information” in driver parameter No. 26 and encoder parameter No. 11.
660	Maximum motor speed mismatch error	The axis-specific parameter, “Maximum motor speed” does not match the motor control constant, “Maximum speed.” Check the “motor/encoder configuration information” in driver parameter No. 26 and encoder parameter No. 11.
661	Encoder/motor combination mismatch error (linear/rotary type)	The linear/rotary type does not match between the encoder and motor. Check the “motor/encoder configuration information” in driver parameter No. 26 and encoder parameter No. 11.
662	Mechanical angle 360-degree pulse count calculation error	The calculated pulse count based on 360 mechanical angle degrees is invalid. (The calculated value is “0,” or in the case of a linear encoder, the calculated value has fraction.)
663	Software DB specification error	The value in the driver parameter, “Software DB specification” is invalid.
664	Current control band number specification error	The value in the driver parameter, “Current control band number” is invalid.
665	Driver/encoder communication line channel number specification error	All-axis parameter No. 101 or 102, “Driver/encoder communication line channel setting” is invalid (invalid value, duplicate specifications).
666	Driver initialization communication type specification error	All-axis parameter No. 103 or 104, “Driver initialization communication type setting” is invalid (invalid value, duplicate specifications, mismatch).
667	Invalid driver initialization communication line specification error at specification of valid axis	Initialization communication line channel number is not specified for a valid axis. Check all-axis parameter No. 1, “Valid axis pattern,” Nos. 101 and 102, “Driver/encoder communication line channel setting” and Nos. 103 and 104, “Driver initialization communication type setting.”
668	Driver target information initialization error	The initialization sequence of driver target information did not complete successfully. Check the installed driver board. Check all-axis parameter Nos. 101, 102, 103 and 104, or driver parameter No. 26, encoder parameter No. 11.
669	Encoder target information initialization error	The initialization sequence of encoder target information did not complete successfully. Check the installed encoder. Check all-axis parameter Nos. 101, 102, 103 and 104, or driver parameter No. 26, encoder parameter No. 11.
66A	Power-system target information initialization error	The initialization sequence of power-system target information did not complete successfully. Check the installed power-supply board. Check the power-supply board parameters.
66B	Slave communication error response error	An error response was received during slave communication.
66C	SCI LRC error (slave communication)	The message LRC of slave communication is invalid.
66D	Slave communication target ID error	The target ID of slave communication is invalid.
66E	Slave communication block number error	The block number of slave communication is invalid.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
66F	Target specification error due to no axis number	The specified target of slave communication (driver or encoder) is invalid (no axis number is assigned for the target ID, or an internal driver board axis is specified).
670	Target board type error	The target board type is invalid.
671	Encoder control data error	The encoder control data is invalid or cannot be acquired. Take the same actions specified for error Nos. 65A, 65B and 669.
672	Motor control data error	The motor control data is invalid or cannot be acquired. Take the same actions as specified for error Nos. 65C, 65D, 668 and 669.
680	Magnetic-pole detection parameter error	Invalid parameter used for magnetic-pole detection. Check driver parameter Nos. 49, 50, etc.
682	I/O function specification error	Wrong I/O function specification. Check I/O parameter Nos. 30 through 61 and 251 through 282.
683	Axis operation error in system semi-locked (encoder stopped) status	An attempt was made to operate an axis by turning on the servo, executing an absolute reset, etc., when the system was in semi-locked status (encoder was stopped).
690	Motor overcurrent error	Excessive current flew through the motor.
691	Driver error	(Driver error for future expansion)
692	Driver error	(Driver error for future expansion)
693	Driver error	(Driver error for future expansion)
694	Driver error	(Driver error for future expansion)
695	Driver error	(Driver error for future expansion)
696	Driver error	(Driver error for future expansion)
697	Driver error	(Driver error for future expansion)
698	Driver error	(Driver error for future expansion)
699	Driver error	(Driver error for future expansion)

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
801	SCIF overrun status (IAI protocol reception)	Communication failure. Check for noise, connected equipment and communication setting.
802	SCIF receive ER status (IAI protocol reception)	Communication failure. Check for noise, shorted/disconnected communication cable, connected equipment and communication setting. This error will also occur when establishing communication with the PC/TP wrongly connected to SIO-CH1 being opened to the user.
803	Receive timeout status (IAI protocol reception)	The transfer interval after the first received byte is too long. Possible causes include disconnected communication cable and error in the connected equipment.
804	SCIF overrun status (SEL reception)	Communication failure. Check for noise, connected equipment and communication setting.
805	SCIF receive ER status (SEL reception)	Communication failure. Check for noise, shorted/disconnected communication cable, connected equipment and communication setting.
806	SCIF receive ER status due to other factor (SEL reception)	Communication failure. Take the same action specified for error No. 804 or 805.
807	Drive-source cutoff relay ER status	The motor-drive power ON status remains ON even when the drive source is cut off. The drive-source cut-off relay contacts may have been melted.
808	Power OFF status during slave parameter write	The power was turned off while writing slave parameters. (This error can be detected only when a backup battery is used.)
809	Power OFF status during data write to flash ROM	The power was turned off while writing data to the flash ROM. (This error can be detected only when a backup battery is used.)
80F	Ethernet control status 1	Ethernet control information (for analysis)
810	Ethernet control status 2	Ethernet control information (for analysis)
811	Maintenance information 1	Maintenance information (for analysis)
812	Maintenance information 2	Maintenance information (for analysis)
813	Maintenance information 3	Maintenance information (for analysis)
814	Maintenance information 4	Maintenance information (for analysis)
815	Maintenance information 5	Maintenance information (for analysis)
820	DRV status 820 (TO_SELECTEDDATA)	(This is not an error, but maintenance information.)

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
900	Blank step shortage error	There are not enough blank steps to save step data. Provide enough blank steps needed to save step data.
901	Step number error	The step number is invalid.
902	Symbol-definition table number error	The symbol-definition table number is invalid.
903	Point number error	The point number is invalid.
904	Variable number error	The variable number is invalid.
905	Flag number error	The flag number is invalid.
906	I/O port/flag number error	The I/O port/flag number is invalid.
910	Command error (IAI protocol HT reception)	The command ID is not supported or invalid. (For future expansion)
911	Message conversion error (IAI protocol HT reception)	The transmitted message does not match the message format or contains invalid data. (For future expansion)
912	PC/TP servo-movement command acceptance-enable input OFF error	Any axis movement command issued to the axis specified in I/O parameter No. 78 from the PC/TP will not be accepted while the input port specified in I/O parameter No. 77 is OFF. (Important: The acceptance-enable input port will become invalid once the operation is started.)
913	Multiple-program simultaneous start inhibition error	Simultaneously starting of multiple programs is inhibited.
914	Absolute-data backup battery voltage error	Check the connection of the absolute-data backup battery and replace the battery if necessary, and also check the connection of the encoder cable, and then perform an absolute reset.
A01	System-memory backup battery voltage-low warning	The voltage of the system-memory backup battery is low. Replace the battery. (Above the minimum data-backup voltage)
A02	Abnormal system-memory backup battery voltage	The voltage of the system-memory backup battery is low. Replace the battery. (Below the minimum data-backup voltage)
A03	Absolute-data backup battery voltage-low warning (Driver analysis)	The voltage of the absolute-data backup battery is low. Check the battery connection or replace the battery.
A04	System mode error at core update	An update command was received when the system was not in the core update mode. Before updating the core, confirm that a chip resistance for setting core update mode is provided on the board. (For maintenance)
A05	Motorola S record format error	The update program file is invalid. Check the file.
A06	Motorola S checksum error	The update program file is invalid. Check the file.
A07	Motorola S load address error	The update program file is invalid. Check the file.
A08	Motorola S write address over error	The update program file is invalid. Check the file.
A09	Flash-ROM timing limit over error (Write)	Error writing the flash ROM
A0A	Flash-ROM timing limit over error (Erase)	Error erasing the flash ROM

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
A0B	Flash-ROM verify error	Error erasing/writing the flash ROM
A0C	Flash-ROM ACK timeout	Error erasing/writing the flash ROM
A0D	Head sector number specification error	Error erasing the flash ROM
A0E	Sector count specification error	Error erasing the flash ROM
A0F	Write-destination offset address error (Odd-numbered address)	Error writing the flash ROM
A10	Write-source data buffer address error (Odd-numbered address)	Error writing the flash ROM
A11	Invalid core-code sector block ID error	The core program already written to the flash ROM is invalid.
A12	Core-code sector block ID erase count over	The number of times the flash ROM can be erased was exceeded.
A13	Flash-ROM write request error when erase is incomplete	When updating, a flash-ROM write command was received before a flash-ROM erase command. Check the update program file and perform update again.
A14	Busy-status reset timeout error at EEPROM write	A busy-status reset timeout occurred after executing EEPROM write.
A15	EEPROM write request error due to no-EEPROM in target	An EEPROM write request was received for a driver or other unit with CPU not equipped with EEPROM.
A16	EEPROM read request error due to no-EEPROM in target	An EEPROM read request was received for a driver or other unit with CPU not equipped with EEPROM.
A17	Message checksum error (IAI protocol reception)	The checksum in the received message is invalid.
A18	Message header error (IAI protocol reception)	The header in the received message is invalid. Invalid header position (message is 9 bytes or less) is suspected, among other reasons.
A19	Message station number error (IAI protocol reception)	The station number in the received message is invalid.
A1A	Message ID error (IAI protocol reception)	The ID in the received message is invalid.
A1C	Message conversion error	The transmitted message does not match the message format or contains invalid data. Check the transmitted message.
A1D	Start mode error	A start not permitted in the current mode (MANU/AUTO) was attempted.
A1E	Start condition non-satisfaction error	Start was attempted when the start condition was not satisfied, such as when an all-operation-cancellation factor (see the 7-segment display: Drive-source cutoff, mode switching, error, auto-start switch OFF edge, deadman switch, safety gate, emergency stop, etc.) was present or the flash ROM was being written.
A1F	Axis duplication error (SIO・PIO)	The applicable axis is currently in use.
A20	Servo-control-right acquisition error (SIO・PIO)	The servo control right is not available.
A21	Servo-control-right duplicate-acquisition error (SIO・PIO)	The servo control right has already been acquired.
A22	Servo-control-right non-acquisition error (SIO・PIO)	An attempt to retain the servo control right has failed.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
A23	Absolute-data backup battery voltage-low warning (Main analysis)	The voltage of the absolute-data backup battery is low. Check the battery connection or replace the battery.
A25	Step count specification error	The specified number of steps is invalid.
A26	Program count specification error	The specified number of programs is invalid.
A27	Program non-registration error	The applicable program is not registered.
A28	Reorganization disable error during program run	A program-area reorganization operation was attempted while a program was running. End all active programs first.
A29	Active-program edit disable error	An edit operation was attempted to a program currently not running. End the applicable program first.
A2A	Program inactive error	The specified program is not running.
A2B	Program-run command refusal error in AUTO mode	Programs cannot be run from the TP/PC software connector in the AUTO mode.
A2C	Program number error	The program number is invalid.
A2D	Inactive program resumption error	A resumption request was received for a program currently not running.
A2E	Inactive program pause error	A pause request was received for a program currently not running.
A2F	Breakpoint error	The step number specified as a breakpoint is invalid.
A30	Breakpoint setting-count specification error	The number of breakpoints to be set exceeds the limit value.
A31	Parameter change value error	The value of parameter changed is invalid.
A32	Parameter type error	The parameter type is invalid.
A33	Parameter number error	The parameter number is invalid.
A34	Card-parameter buffer read error	Error reading the card-parameter buffer
A35	Card-parameter buffer write error	Error writing the card-parameter buffer
A36	Parameter change refusal error during operation	Parameters cannot be changed during operation (program is running, servo is in use, etc.).
A37	Card manufacturing/function information change refusal error	The card manufacturing/function information cannot be changed.
A38	Parameter change refusal error during servo ON	An attempt was made to change a parameter whose change is not permitted while the servo is ON.
A39	Non-acquired card parameter change error	An attempt was made to change a parameter for a card not recognized at reset.
A3A	Device number error	The device number is invalid.
A3C	Memory initialization type specification error	The specified memory initialization type is invalid.
A3D	Unit type error	The unit type is invalid.
A3E	SEL write data type specification error	The specified SEL write data type is invalid.
A3F	Flash-ROM write refusal error during program run	The flash ROM cannot be written while a program is running.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
A40	Data change refusal error during flash ROM write	Data cannot be changed while the flash ROM is being written.
A41	Duplicate flash-ROM write commands refusal error	Another flash-ROM write command was received while the flash ROM was being written.
A42	Direct monitor prohibition error during flash ROM write	Direct monitor is prohibited while the flash ROM is being written.
A43	P0/P3-area direct monitor prohibition error	Direct monitor in the P0/P3 areas is prohibited.
A44	Point-data count specification error	The specified number of point data is invalid.
A45	Symbol-record count specification error	The specified number of symbol records is invalid.
A46	Variable-data count specification error	The specified number of variable data is invalid.
A48	Error-detail query type 1 error	Error-detail query type 1 is invalid.
A49	Error-detail query type 2 error	Error-detail query type 2 is invalid.
A4A	Monitoring data type error	The data type for monitoring data query is invalid.
A4B	Monitoring-record count specification error	The specified number of records for monitoring data query is invalid.
A4C	Monitoring-operation special command register busy error	The driver special command ACK generated a timeout during monitoring operation.
A4E	Parameter register busy error at issuance of slave command	The driver special command ACK generated a timeout at issuance of a slave command.
A4F	Software reset refusal error during operation	Software reset (SIO) is prohibited during operation (program is running, servo is in use, etc.).
A50	Drive-source recovery request refusal error	The drive-source cutoff factor (error, deadman switch, safety gate, emergency stop, etc.) has not been removed.
A51	Operation-pause reset request refusal error	The all-operation-pause factor (drive-source cutoff, operation-pause signal, deadman switch, safety gate, emergency stop, etc.) has not been removed.
A53	Refusal error due to servo ON	A processing not permitted during servo ON was attempted.
A54	Refusal error due to unsupported function	The function is not supported.
A55	Refusal error due to exclusive manufacturer function	A processing not opened to users other than the manufacturer was attempted.
A56	Refusal error due to invalid data	The data is invalid.
A57	Program start duplication error	An attempt was made to start a program currently running.
A58	BCD error warning	The BCD value being read may be invalid, or the value being written (variable 99) may be a negative value, among other reasons.
A59	IN/OUT command port flag error warning	The number of I/O ports (flags) may have exceeded 32, among other reasons. Check the I/O port (flag) specifications.
A5B	Character-string → value conversion error warning	The specified number of converting characters is invalid or characters that cannot be converted to value are included.
A5C	Copying-character count error warning with SCPY command	The specified number of copying characters is invalid.
A5D	SCIF open error in non-AUTO mode	The channel was opened in a non-AUTO mode. In the MANU mode, the PC/TP connection must be forcibly disconnected before opening the serial channel opened to the user. Exercise caution.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
A5E	I/O-port/flag count specification error	The specified number of I/O ports/flags is invalid.
A5F	Fieldbus error (LERROR-ON)	A LERROR-ON was detected.
A60	Fieldbus error (LERROR-BLINK)	A LERROR-BLINK was detected.
A61	Fieldbus error (HERROR-ON)	A HERROR-ON was detected.
A62	Fieldbus error (HERROR-BLINK)	A HERROR-BLINK was detected.
A63	Fieldbus not ready	Fieldbus ready cannot be confirmed.
A69	Data change refusal error during operation	An attempt was made to change data whose change is prohibited during operation (program is running, servo is in use, etc.).
A6A	Software reset refusal error during write	Software reset is prohibited while data is being written to the flash ROM or slave parameters are being written.
A6B	Fieldbus error (FBRS link error)	A FBRS link error was detected.
A6C	PC/TP start command refusal error in AUTO mode	Starting from the PC software/TP connector is prohibited in the AUTO mode.
A6D	P0/P3/FROM-area direct write prohibition error	Direct write to the P0/P3/FROM areas is prohibited.
A6E	Refusal error during write	A processing not permitted while data is being written to the flash ROM or slave parameters are being written was attempted.
A6F	Driver monitor type mismatch error	The monitor type supported by the standard DIO board or based on the capacity of FROM on the main CPU board does not match the monitor type on the PC software side (selected on the monitor screen).

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
B00	SCHA setting error	The setting of SCHA command is invalid.
B01	TPCD setting error	The setting of TPCD command is invalid.
B02	SLEN setting error	The setting of SLEN command is invalid.
B03	Home-return method error	The setting of “Axis-specific parameter No. 10, Home-return method” is invalid. (Not incremental encoder AND current position 0 home is specified, etc.)
B04	1-shot-pulse output excessive simultaneous use error	The number of BTPN and BTPF timers operating in one program simultaneously exceeds the upper limit (16).
B05	Estimate-stroke over error at home return	The operation at home return exceeded the estimate stroke. The home sensor or creep sensor may be faulty, among other reasons.
B10	Phase-Z search timeout error	Phase Z cannot be detected. Check for operation restriction, wiring, encoder, motor, etc.
B11	Home-sensor pull-out timeout error	Pull-out from the home sensor cannot be confirmed. Check for operation restriction, wiring, motor, home sensor, etc.
B12	Storage variable number error for SEL command return code	The variable number specified for storing SEL command’s return code is invalid.
B13	Backup SRAM data checksum error	The backup SRAM data has been destroyed. Check the battery.
B15	Input-port debug filter type error	The setting of input-port debug filter type is invalid.
B16	SEL operand specification error	The operand specification of SEL command is invalid.
B17	Parameter register busy error at issuance of slave command	The driver special command ACK generated a timeout at issuance of a slave command.
B18	Device number error	The device number is invalid.
B19	Unit type error	The unit type is invalid
B1A	Absolute reset specification error	The specification for absolute reset using an optional function, etc., is invalid. (Two or more axes are specified simultaneously, non-absolute-encoder axis is specified, etc.)
B1B	Ethernet non-closed socket open error	An attempt was made to open a socket without closing it first.
B1C	Ethernet in-use-by-other-task error	An attempt was made to open a channel already opened by other task.
B1D	Ethernet non-open error	An attempt was made to use a channel not opened by own task.
B1E	Ethernet multiple WRIT execution error	WRIT commands were executed simultaneously by multiple tasks for the same channel.
B1F	Ethernet job busy error	An attempt was made to start a new process when the Ethernet mailbox control job was busy.
B20	Ethernet non-initialization device use error	An attempt was made to use the Ethernet system when Ethernet device initialization was not yet complete. Check I/O parameter Nos. 123 to 159, 14, 15, etc., depending on the purpose of use.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
B21	Ethernet IP address error	An error will generate under the following conditions during normal use. When IP address (H) (first octet) through IP address (L) (fourth octet) are given as IP_H, IP_MH, IP_ML and IP_L, the error conditions are described as follows: IP_H ≤ 0 or IP_H = 127 or IP_H > 255 or IP_MH < 0 or IP_MH > 255 or IP_ML < 0 or IP_ML > 255 or IP_L ≤ 0 or IP_L ≥ 255 Check I/O parameter Nos. 132 to 135, 149 to 152, and 154 to 157, the IP address of connection destination specified by an IPCN command in an integer variable, or the like.
B22	Ethernet port number error	An error will generate if own port number < 1025, or own port number > 65535, or own port number duplication, or connection-destination port number for client ≤ 0, or connection-destination port number for client > 65535, or connection-destination port number for server < 0, or connection-destination port number for server > 65535 is satisfied. Check I/O parameter Nos. 144 to 148, 159, 153, and 158, the port number of connection destination specified by an IPCN command in an integer variable, or the like.
B86	SEL PTRQ command preprocessing error	The PTRQ command setting is abnormal. Check the setting for abnormality, such as deviation from the allowable range.
B92	Excessive arc interpolation radius error	The radius of arc interpolation is excessive. Use a CIR/ARC command, etc.
C02	Executable program count over error	Execution requests were received for programs exceeding the number that can be executed simultaneously.
C03	Non-registered program specification error	The specified program is not registered.
C04	Program entry point non-detection error	A request was made to execute a program number for which no program steps are registered.
C05	Program first-step BGSR error	The program specified for execution starts with BGSR.
C06	Executable step non-detection error	The program specified for execution does not contain executable program steps.
C07	Subroutine non-definition error	The subroutine specified for call is not defined.
C08	Subroutine duplicate-definition error	The same subroutine number is defined at multiple locations.
C0A	Tag duplicate-definition error	The same tag number is defined at multiple locations.
C0B	Tag non-definition error	The tag specified as the jump destination of a GOTO statement is not defined.
C0C	DW/IF/IS/SL pair-end mismatch error	The branching command syntax is invalid. Correspondence with the last appearing branching command is invalid when EDIF, EDDO or EDSL is used. Check the correspondence between IF/IS command and EDIF, DO command and EDDO or SLCT command and EDSL.
C0D	DW/IF/IS/SL no pair-end error	EDIF, EDDO or EDSL is not found. Check the correspondence between IF/IS command and EDIF, DO command and EDDO or SLCT command and EDSL.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
C0E	BGSR no pair-end error	There is no EDSR for BGSR, or no BGSR for EDSR. Check the correspondence between BGSR and EDSR.
C0F	DO/IF/IS over-nesting error	The number of nests in a DO or IF/IS command exceeds the limit value. Check for excessive nesting or branching out of or into the syntax using a GOTO command.
C10	SLCT over-nesting error	The number of nests in a SLCT command exceeds the limit value. Check for excessive nesting or branching out of or into the syntax using a GOTO command.
C11	Subroutine over-nesting error	The number of nests in a subroutine exceeds the limit value. Check for excessive nesting or branching out of or into the syntax using a GOTO command.
C12	DO/IF/IS under-nesting error	The EDIF or EDDO position is invalid. Check the correspondence between IF/IS command and EDIF or DO command and EDDO, or branching out of or into the syntax using a GOTO command.
C13	SLCT under-nesting error	The EDSL position is invalid. Check the correspondence between SLCT and EDSR, or branching out of or into the syntax using a GOTO command.
C14	Subroutine under-nesting error	The EDSR position is invalid. Check the correspondence between BGSR and EDSR, or branching out of or into the syntax using a GOTO command.
C15	SLCT next-step command code error	The program step next to SLCT must be WHEQ, WHNE, WHGT, WHGE, WHLT, WHLE, WSEQ, WSNE, OTHE or EDSL.
C16	Create stack failed	Initialization of the input-condition-status storage stack has failed.
C17	Expansion-condition code error	Input program step error. The expansion condition code is invalid.
C18	Expansion-condition LD simultaneous processing over error	The number of LDs processed simultaneously exceeds the limit value.
C19	Expansion-condition LD shortage error 1	There is not enough LD when expansion condition A or O is used.
C1A	Expansion-condition LD shortage error 2	There is not enough LD when expansion condition AB or OB is used.
C1C	Unused-LD detection error	An attempt was made to execute a command based on multiple LD condition that has been saved, without using it in expansion condition AB or OB.
C1F	Input-condition CND shortage error	The necessary input condition is not found when an expansion condition is used.
C21	Input-condition use error with input-condition prohibited command	Input-condition prohibited commands prohibit the use of input conditions.
C22	Invalid command position error with input-condition prohibited command	A command for which input condition is prohibited cannot be included in an input condition nest.
C23	Invalid operand error	Program step error. The necessary operand data is invalid.
C24	Operand type error	Program step error. The operand data type is invalid.
C25	Actuator control declaration error	The setting of actuator control declaration command is invalid.
C26	Timer setting-range over error	The timer setting is invalid.
C27	Timeout setting-range over error during wait	The timeout setting is invalid.
C28	Tick count setting-range error	The Tick count setting is invalid.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
C29	DIV command divisor 0 error	“0” was specified as the divisor in the DIV command.
C2A	SQR command range error	The operand value in the SQR command is invalid. Input a value larger than “0” as data in a SQR command.
C2B	BCD display digit range error	The specified number of BCD display digits is invalid. Specify a value between 1 and 8.
C2C	Program number error	The program number is invalid.
C2D	Step number error	The step number is invalid.
C2E	Blank step shortage error	There are not enough blank steps to save step data. Provide enough blank steps needed to save step data.
C2F	Axis number error	The axis number is invalid.
C30	Axis pattern error	The axis pattern is invalid.
C32	Operating-axis addition error during command execution	An operating axis for point data was added during continuous point movement or push-motion movement calculation.
C33	Base axis number error	The base axis number is invalid.
C34	Zone number error	The zone number is invalid.
C35	Point number error	The point number is invalid.
C36	I/O port/flag number error	The I/O port/flag number is invalid.
C37	Flag number error	The flag number is invalid.
C38	Tag number error	The tag number is invalid.
C39	Subroutine number error	The subroutine number is invalid.
C3A	User-open communication channel number error	The channel number of the communication channel opened to the user is invalid.
C3B	Parameter number error	The parameter number is invalid.
C3C	Variable number error	The variable number is invalid.
C3D	String number error	The string number is invalid.
C3E	String-variable data count specification error	The specified number of string variables exceeds the area, etc.
C40	String-variable delimiter non-detection error	Delimiter cannot be detected in the string variable.
C41	String-variable copy size over error	The copy size of string variable is too large.
C42	Character count non-detection error during string processing	The character-string length is not defined in string processing. Execute a string processing command after defining the length with a SLEN command.
C43	Character-string length error during string processing	The character-string length used in string processing is invalid. Check the value of character-string length defined by a SLEN command.
C45	Symbol definition table number error	The symbol definition table number is invalid.
C46	Blank area shortage error with source-symbol storage table	There is not enough area to store the source symbols. Check the number of times source symbol can be used.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
C47	Symbol search error	Definitions are not found for the symbols used in the program steps.
C48	SIO-message continuous conversion error	The transmitted SIO message does not match the message format or contains invalid data. Check the transmitted message.
C49	SEL-SIO in-use error	The SIO is being used by other interpreter task.
C4A	SCIF unopen error	Serial channel 1 opened to the user is not opened in the target task. Open the channel using an OPEN command first.
C4B	Delimiter non-definition error	An end character is not defined. Set an end character using a SCHA command first.
C4E	SIO1 invalid usage OPEN error	The usage of serial channel opened to the user does not match the parameter. Check “I/O parameter No. 90, Usage of SIO channel opened to user.”
C4F	SEL program/source symbol checksum error	The flash ROM data has been destroyed.
C50	Symbol definition table checksum error	The flash ROM data has been destroyed.
C51	Point data checksum error	The flash ROM data has been destroyed.
C52	Backup SRAM data destruction error	The backup SRAM data has been destroyed. Check the battery.
C53	Invalid flash-ROM SEL global data/error list error	The SEL global data/error lists in the flash ROM are invalid.
C54	Flash-ROM SEL global data/error list duplication error	The SEL global data/error lists in the flash ROM are duplicated.
C55	Flash-ROM erase count over error for SEL global data/error lists	The number of time the flash ROM containing SEL global data/error lists can be erased was exceeded.
C56	Timing limit over error (Flash ROM erase)	Error erasing the flash ROM
C57	Flash-ROM verify error (Flash ROM erase)	Error erasing the flash ROM
C58	Flash-ROM ACK timeout error (Flash ROM erase)	Error erasing the flash ROM
C59	Head sector number specification error (Flash ROM erase)	Error erasing the flash ROM
C5A	Sector count specification error (Flash ROM erase)	Error erasing the flash ROM
C5B	Timing limit over error (Flash ROM write)	Error writing the flash ROM
C5C	Flash-ROM verify error (Flash ROM write)	Error writing the flash ROM
C5D	Flash-ROM ACK timeout error (Flash ROM write)	Error writing the flash ROM
C5E	Write-destination offset address error (Flash ROM write)	Error writing the flash ROM
C5F	Write-source data buffer address error (Flash ROM write)	Error writing the flash ROM
C60	No SEL global data/error list write area error	There is no area to write the erased SEL global data/error lists.
C61	SEL-data flash-ROM erase count over error	The number of times the flash ROM containing SEL data can be erased was exceeded.
C62	Operation command error at servo OFF	An attempt was made to execute an operation command when the servo was OFF.
C63	Servo operation condition error	The servo is not in an operation-enabled condition.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
C64	Invalid servo acceleration/deceleration error	The internal servo acceleration/deceleration is invalid.
C65	Servo ON/OFF logic error	The servo ON/OFF logic between the main and driver is invalid.
C66	Axis duplication error	An attempt was made to acquire the control right to an axis already in use.
C67	Servo-control-right acquisition error	There is no space in the servo user management area.
C68	Servo-control-right duplicate-acquisition error	The servo control right has already been acquired.
C69	Servo-control-right non-acquisition error	A user who doesn’t have the servo control right attempted to retain the control right.
C6A	Push-motion flag logic error	The internal logic for push-motion processing is invalid.
C6B	Deviation overflow error	The command cannot be followed. Check for operation restriction, wiring, encoder, motor, etc.
C6C	Movement error during absolute data acquisition	Axis movement was detected while acquiring absolute encoder data after the power was turned on. The power may have been turned or a software reset executed while the actuator was moving due to external force such as reactive force of a self-supported cable or while the installation location was vibrating. Or, a software reset may have been executed. Absolute coordinates cannot be confirmed in this condition.
C6D	Maximum installable axes over error	The specified number of axes exceeded the number of installable axes as a result of axis shift with a base command.
C6E	Servo-OFF axis use error	An attempt was made to use an axis whose servo is OFF.
C6F	Home-return incomplete error	Home return has not completed yet. This error may also occur if operation is performed immediately after changing an encoder parameter, performing an absolute encoder reset or resetting an encoder error, without first executing a software reset or reconnecting the power.
C70	Absolute coordinate non-confirmation error	Absolute coordinates have not been confirmed. The power must be reconnected. This error may also occur if operation is performed immediately after changing an encoder parameter, performing an absolute encoder reset or resetting an encoder error, without first executing a software reset or reconnecting the power.
C71	Synchro slave-axis command error	A command was issued to the synchro slave axis.
C72	Overrun error	The overrun sensor was actuated.
C73	Target-locus soft limit over error	The target position or movement locus exceeds a soft limit. * In the case of a SCARA specification, position data may not exist for the applicable axis.
C74	Actual-position soft limit over error	The actual position exceeds a soft limit by the “soft limit/actual position margin” or more.
C75	Motion-data-packet generation logic error	The motion-data-packet generation logic is invalid.
C76	Movement-point count over error	Too many packets are generated simultaneously.
C77	Handling-packet overflow error	The servo handling packets overflowed.
C78	Motion-data-packet overflow error	The servo motion data packets overflowed.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
C79	Pole sense operation error	Operation is disabled in the pole sense mode.
C7A	Servo unsupported function error	An attempt was made to use an unsupported function.
C7B	Odd-pulse slide error	Internal servo calculation error
C7C	Odd-pulse processing logic error	Internal servo calculation error
C7D	Packet pulse shortage error	Internal servo calculation error
C7E	Quadratic equation solution error	An error was detected while calculating a quadratic equation solution.
C7F	No valid specified axis error	No valid axes are specified.
C80	Servo-packet calculation logic error	Internal servo calculation error If the controller is of absolute encoder specification and the system has just been moved or “Error No. C74, Actual-position soft limit over error” has also generated, the controller may be experiencing a servo-packet calculation overflow caused by abnormal current position resulting from an unsuccessful absolute reset. Perform an absolute reset again by following the operation manual. (Simply selecting “Encoder error reset” on the absolute reset screen will not allow the controller to recognize the correct position. Always perform an absolute reset by strictly following the specified procedure.)
C81	Operation-amount logic during servo ON	Servo processing logic error
C82	Servo direct command type error	Servo processing logic error
C83	Servo calculation method type error	The servo calculation method type is invalid.
C84	In-use axis servo OFF error	The servo of an axis currently in use (being processed) was turned off.
C85	Non-installed driver error	Driver is not installed for the applicable axis.
C86	Driver servo ready OFF error	The ready signal for the driver of the applicable axis is OFF.
C87	SEL unsupported function error	An attempt was made to use a function not supported by SEL.
C88	Speed specification error	The specified speed is invalid.
C89	Acceleration/deceleration specification error	The specified acceleration/deceleration is invalid.
C8B	Circle/arc calculation logic error	The arc calculation logic is invalid.
C8D	Circle/arc calculation error	Position data that cannot be used in arc movement was specified. Check the position data.
C8E	Point deletion error during command execution	The final point data was deleted while continuous point movement was being calculated.
C8F	Axis operation type error	The axis operation type is invalid. Check “Axis-specific parameter No. 1, Axis operation type” and perform operation appropriate for the operation type specified.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
C90	Spline calculation logic error	The spline processing logic is invalid.
C91	Push-motion axis multiple specification error	Two or more push-motion axes were specified.
C92	Push-motion approach distance/speed specification error	The specified push-motion approach distance/speed is invalid.
C93	System output operation error	The user attempted a system output operation (through the port specified by I/O parameter for output function selection or the zone output port specified by axis-specific parameter).
C94	PIO program number error	The PIO-specified program number is invalid.
C95	AUTO program number error	The setting of “Other parameter No. 1, Auto-start program number” is invalid.
C96	Start error from operation-abort program	(This error should not occur now that the specification has been changed.)
C97	Program number error for I/O processing program at operation/program abort	The setting of “Other parameter No. 2, I/O processing program number at operation/program abort” is invalid.
C98	Program number error for I/O processing program at operation pause	The setting of “Other parameter No. 3, I/O processing program number at all operation pause” is invalid.
C99	Home sensor non-detection error	The home sensor cannot be detected. Check the wiring and sensor.
C9A	Creep sensor non-detection error	The creep sensor cannot be detected. Check the wiring and sensor.
C9B	Phase Z non-detection error	Phase Z cannot be detected. Check the wiring and encoder.
C9C	Defective phase-Z position error	The phase-Z position is defective. Normal wear and tear of the mechanical ends and home sensor may also be a reason. Readjustment is necessary.
C9D	Card parameter write error	Error writing card parameters
C9E	Servo calculation overflow error	Internal servo calculation error
CA1	Abnormal absolute-data backup battery voltage (Driver analysis)	Check the connection of the absolute-data backup battery/replace the battery and/or check the encoder cable connection, and then perform an absolute reset.
CA2	Abnormal absolute-data backup battery voltage (Main analysis)	Check the connection of the absolute-data backup battery/replace the battery and/or check the encoder cable connection, and then perform an absolute reset.
CA3	Slave setting data out-of-range error	The data set to the slave is outside the allowable range.
CA4	Slave error response	An error response was returned from the slave.
CA5	Stop deviation overflow error	Movement may have occurred during stopping due to external force or operation may have been restricted during deceleration. This error may also generate when jog operation is restricted (due to contact with an obstacle, contact with a mechanical end before home return, etc.) or when wiring error, faulty encoder or faulty motor is detected during deceleration.
CA6	Palletizing number error	The specified palletizing number is invalid.
CA7	Setting error of even-numbered row count for palletizing zigzag	The set even-numbered row count for palletizing zigzag is invalid.
CA8	Setting error of palletizing pitches	The set palletizing pitches are abnormal.
CA9	Setting error of placement points in palletizing-axis directions	The set X/Y-axis direction counts for palletizing are invalid.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
CAA	Palletizing PASE/PAPS non-declaration error	Neither PASE nor PAPS palletizing-setting command is set. Set either command.
CAB	Palletizing position number error	The specified palletizing position number is invalid.
CAC	Palletizing position number setting over	The specified palletizing position number exceeds the position number range calculated for the current palletizing setting.
CAD	Palletizing PX/PY/PZ-axis duplication error	Any two of the specified PX, PY and PZ-axes for palletizing are the same axis.
CAE	Insufficient valid axes for palletizing 3-point teaching data	There are not enough valid axes in the point data for palletizing 3-point teaching. Axes to comprise the palletizing PX/PY planes cannot be specified.
CAF	Excessive valid axes for palletizing 3-point teaching data	There are too many valid axes in the point data for palletizing 3-point teaching. Axes to comprise the palletizing PX/PY planes cannot be specified.
CB0	Mismatched valid axes for palletizing 3-point teaching data	The valid axis pattern in the point data for palletizing 3-point teaching does not match.
CB1	Offset setting error at palletizing 3-point teaching	Zigzag offset (not zero) cannot be set in palletizing 3-point teaching, if the reference point is the same as the end point of the PX-axis.
CB2	BGPA/EDPA pair-end mismatch error	The BGPA/EDPA syntax is invalid. EDPA was declared before BGPA, or another BGPA was declared after BGPA without first declaring EDPA.
CB4	Arch-motion Z-axis non-declaration error	Z-axis has not been declared by PCHZ or ACHZ.
CB5	BGPA non-declaration error during palletizing setting	Palletizing setting cannot be performed without first declaring BGPA. Declare BGPA.
CB6	Palletizing point error	The palletizing points are invalid (non-Z-axis components for arch-motion movement are absent, etc.).
CB7	Arch-trigger non-declaration error	Declare arch triggers using PTRG or ATRG.
CB8	No 3-point teaching setting error at palletizing angle acquisition	The palletizing angle cannot be acquired until setting by palletizing 3-point teaching is complete.
CB9	PX/PY-axis indeterminable error at palletizing angle acquisition	Angle cannot be calculated because there are too many valid axes in the 3-point teaching data and thus PX/PY-axes cannot be specified.
CBA	Reference-axis/PY/PY-axis mismatch error at palletizing angle acquisition	Angle cannot be calculated because the reference axis for angle calculation is neither of the axes comprising the PX/PY-axes as set by 3-point teaching.
CBB	Reference-point/PX-axis end-point duplication error at palletizing angle acquisition	Angle cannot be calculated because the reference point of 3-point teaching is the same as the PX-axis end-point data other than the PZ-axis component and thus arc tangent cannot be calculated.
CBC	Palletizing motion calculation error	Trapezoid control calculation error for palletizing motion
CBD	MOD command divisor 0 error	“0” was specified as the divisor in the MOD command.
CBE	Target-locus boundary over error	The target position or movement locus exceeded the positioning boundary in the infinite-stroke mode.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
CBF	Positioning distance overflow error	The positioning distance is too large. If the controller is of absolute encoder specification and the system has just been moved or “Error No. C74, Actual-position soft limit over error” has also generated, the controller may be experiencing a servo-packet calculation overflow caused by abnormal current position resulting from an unsuccessful absolute reset. Perform an absolute reset again by following the operation manual. (Simply selecting “Encoder error reset” on the absolute reset screen will not allow the controller to recognize the correct position. Always perform an absolute reset by strictly following the specified procedure.)
CC0	Axis mode error	The axis mode is invalid.
CC1	Speed change condition error	An attempt was made to change the speed of an axis whose speed cannot be changed (axis operating in S-motion, etc.).
CC2	Driver parameter list number error	The driver parameter list number is invalid.
CC3	Angle error	The angle is invalid.
CC4	SEL data error	The SEL data is invalid.
CC5	Positioning boundary pull-out error	An attempt was made to execute a command not permitted outside the positioning boundary.
CC6	Driver error primary detection	A driver error was found by primary detection.
CC7	Palletizing movement PZ-axis pattern non-detection error	PZ-axis component is not found in the axis pattern during palletizing movement.
CC8	Arch top Z-axis pattern non-detection error	Z-axis component relating to the highest point of arch motion is not found in the axis pattern during arch motion operation.
CC9	Arch trigger Z-axis pattern non-detection error	Z-axis component relating to arch motion is not found in the axis pattern of the arch-trigger declaration point data.
CCA	Arch top/end-point reversing error	The coordinates of highest point and end point are reversed during arch motion operation.
CCB	Arch start-point/trigger reversing error	The coordinates of start point and start-point arch trigger are reversed during arch motion operation.
CCC	Arch end-point/trigger reversing error	The coordinates of end point and end-point arch trigger are reversed during arch motion operation.
CCD	Drive-source cutoff axis use error	An attempt was made to use an axis whose drive source is cut off.
CCE	Error axis use error	An attempt was made to use an axis currently generating an error.
CCF	Palletizing reference-point/valid-axis mismatch error	The PX/PY(/PZ)-axes set by PASE/PCHZ are not valid in the axis pattern of the reference-point data set by PAST.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
D01	Encoder EEPROM-write timeout error	The encoder is faulty or failure occurred in the encoder communication.
D02	Encoder EEPROM-read timeout error	The encoder is faulty or failure occurred in the encoder communication.
D03	Encoder count error	Faulty encoder or defective encoder assembly condition is suspected.
D04	Encoder one-revolution reset error	The encoder is faulty or has turned.
D05	Encoder-EEPROM write acceptance error	The encoder is faulty or failure occurred in the encoder communication.
D06	Encoder received-data error	The encoder is faulty or failure occurred in the encoder communication.
D07	Driver logic error	The driver CPU board is in a condition where it cannot operate normally.
D08	Encoder CRC error	The encoder is faulty or failure occurred in the encoder communication.
D09	Driver overspeed error	The motor speed exceeded the upper limit.
D0A	Driver overload error	The power input to the motor exceeded the upper limit.
D0B	Driver EEPROM data error	Failure during write or EEPROM failure
D0C	Encoder EEPROM data error	Failure during write or EEPROM failure
D0E	Axis sensor error	An error occurred in the axis sensor.
D0F	Power stage temperature error	The power stage board exceeded the upper temperature limit.
D10	IPM error	A failure occurred in the motor drive circuit.
D11	Driver abnormal interruption error	The driver CPU board is in a condition where it cannot operate normally.
D12	Encoder disconnection error	The encoder cable is disconnected. The power must be reconnected.
D13	FPGA watchdog timer error	Failure in the interface with the main CPU
D14	Current loop underrun error	Failure in the interface with the main CPU
D15	Driver-CPU down status error	An error occurred in the driver CPU board.
D17	Main-CPU alarm status error	Failure in the interface with the main CPU
D18	Speed loop underrun error	Failure in the interface with the main CPU
D19	Encoder receive timeout error	The encoder is faulty or failure occurred in the encoder communication.
D1A	Driver command error	An error occurred in the CPU bus command.
D1B	Serial bus receive error	Failure in the interface with the main CPU
D1C	Encoder overspeed error	The motor speed exceeded the upper limit.
D1D	Encoder full-absolute status error	The motor speed exceeded the upper limit.
D1E	Encoder counter overflow error	The encoder rotation counter exceeded the upper limit.
D1F	Encoder rotation error	Faulty encoder or defective encoder assembly condition is suspected.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
D20	Driver error	(Refer to error No. CA1.)
D22	Encoder rotation reset error	The encoder is faulty or has turned.
D23	Encoder alarm reset error	Faulty encoder
D24	Encoder ID error	The encoder is faulty or failure occurred in the encoder communication.
D25	Encoder configuration mismatch error	The encoder configuration information is outside the function information range.
D26	Motor configuration mismatch error	The motor configuration information is outside the function information range.
D50	Fieldbus error (FBMIRQ timeout)	A FBMIRQ timeout was detected. Check the status of the monitor LED on the front face of the board by referring to the operation manual for the field network board.
D51	Fieldbus error (FBMIRQ reset)	A FBMIRQ reset error was detected. Check the status of the monitor LED on the front face of the board by referring to the operation manual for the field network board.
D52	Fieldbus error (FBMBSY)	A FBMBSY was detected. Check the status of the monitor LED on the front face of the board by referring to the operation manual for the field network board.
D53	Fieldbus error (BSYERR)	A BSYERR was detected. The power must be reconnected. Check the status of the monitor LED on the front face of the board by referring to the operation manual for the field network board.
D54	Window lock error (LERR)	A LERR was detected. The power must be reconnected. Check the status of the monitor LED on the front face of the board by referring to the operation manual for the field network board.
D55	Fieldbus error (Min busy)	A Min busy error was detected. Check the status of the monitor LED on the front face of the board by referring to the operation manual for the field network board.
D56	Fieldbus error (MinACK timeout)	A Min ACK timeout was detected. Check the status of the monitor LED on the front face of the board by referring to the operation manual for the field network board.
D57	Fieldbus error (MoutSTB timeout)	A Mout STB timeout was detected. Check the status of the monitor LED on the front face of the board by referring to the operation manual for the field network board.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
D58	Fieldbus error (INIT timeout)	An INIT timeout was detected. Check the status of the monitor LED on the front face of the board by referring to the operation manual for the field network board.
D59	Fieldbus error (DPRAM write/read)	A DPRAM write/read error was detected. Check the status of the monitor LED on the front face of the board by referring to the operation manual for the field network board.
D5A	Fieldbus error (TOGGLE timeout)	A TOGGLE timeout was detected. Check the status of the monitor LED on the front face of the board by referring to the operation manual for the field network board.
D5B	Fieldbus error (Access-privilege retry over)	An access-privilege retry over error was detected. Check the status of the monitor LED on the front face of the board by referring to the operation manual for the field network board.
D5C	Fieldbus error (Access-privilege open error)	An access-privilege open error was detected. Check the status of the monitor LED on the front face of the board by referring to the operation manual for the field network board.
D5D	Fieldbus error (FBRS link error)	A FBRS link error was detected. Check the status of the monitor LED on the front face of the board by referring to the operation manual for the field network board.
D5E	Fieldbus error (Mailbox response)	A mailbox response error was detected. Check the status of the monitor LED on the front face of the board by referring to the operation manual for the field network board.
D67	Motor/encoder configuration information mismatch error	The “motor/encoder configuration information” (motor identification number and encoder identification number) in driver parameter No. 26 does not match the “motor/ encoder configuration information” (motor identification number and encoder identification number) in encoder parameter No. 11. Check the parameter values, encoder cable connection, etc.
D68	No remote-mode control support board error	Hardware supporting remote-mode control is not installed, although remote-mode control (AUTO/MANU) is specified in I/O parameter No. 79.
D69	External terminal block overcurrent or power-supply error	Overcurrent or power-supply error in the external terminal block
D70	Option use permission error	Check if any option whose use is not permitted is specified in the system program.
D6A	Hardware unsupported function error	An attempt was made to use a function not supported by the hardware.
D6B	Overrun error	The overrun sensor was actuated.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
D6C	Actual-position soft limit over error	The actual position exceeded a soft limit by the “soft limit/actual position margin” or more.
D6D	Logic error	A logic error occurred.
D6F	Optional password error	The optional function the controller is attempting to use requires an optional password. Check other parameter Nos. 30 through 32, etc., in accordance with the applicable function.
E01	DMA address error	DMA transfer error
E02	SCIF send-buffer overflow error	The SCIF send buffer overflowed.
E03	SCI send-buffer overflow error	The SCI send buffer overflowed.
E04	SCIF receive-buffer overflow error	The SCIF receive buffer overflowed. Excessive data was received from outside.
E05	SCI receive-buffer overflow error	The SCI receive buffer overflowed. Excessive data was received from the slave.
E06	Receive timeout error (Slave communication)	Response from the slave cannot be recognized.
E07	SCI overrun error (Slave communication)	Communication failure. Check for noise, circuit failure and slave card.
E08	SCI framing error (Slave communication)	Communication failure. Check for noise, shorting, circuit failure and slave card.
E09	SCI parity error (Slave communication)	Communication failure. Check for noise, shorting, circuit failure and slave card.
E0A	SCI CRC error (Slave communication)	The CRC in the message is invalid.
E10	SCIF communication mode error	The communication mode is invalid.
E11	SCI communication mode error	The communication mode is invalid.
E14	SCI receive-data-register full wait timeout error	Communication failure. Check for noise, shorting, circuit failure and slave card.
E15	SCI overrun error	Communication failure. Check for noise, shorting, circuit failure and slave card.
E16	Program end confirmation timeout error	The program cannot be ended.
E17	I/O-processing-program start logic error	The I/O-processing-program start logic is invalid.
E18	Task ID error	The task ID is invalid.
E19	WAIT factor error	The WAIT factor is invalid.
E1A	WAIT logic error	The WAIT logic is invalid.
E1B	Point-data valid address error	Point-data valid address is not set.
E1C	Source data error	The source data is invalid.
E1D	Unaffected output number error	The unaffected output number is invalid. A value other than an output port number (“0” is acceptable) may be input in I/O parameter Nos. 70 to 73.
E1E	Zone parameter error	A value other than an output port/global flag number (“0” is acceptable) or duplicate numbers may be input in axis-specific parameter Nos. 88, 91, 94 and 97, or the output number specified as system output in the I/O parameter for output function selection may be duplicated, among other reasons.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
E1F	I/O assignment parameter error	A value other than an I/O port number (“-1” is acceptable) or other than an I/O head port number + [multiple of 8] may be input in I/O parameter Nos. 2 to 9, or a value other than a [multiple of 8] may be input in I/O parameter Nos. 14 to 17.
E20	I/O assignment duplication error	I/O assignments are duplicated. Check I/O parameter Nos. 2 to 9 and 14 to 17 and the I/O slot card type (number of I/Os), etc.
E21	I/O assignment count over error	The I/O assignments exceed the specified range. Check I/O parameter Nos. 2 to 9 and 14 to 17 and the I/O slot card type (number of I/Os).
E22	Header error (Slave communication)	The header in the message received from the slave card is invalid.
E23	Card ID error (Slave communication)	The card ID in the message received from the slave card is invalid.
E24	Response type error (Slave communication)	The response type in the message received from the slave card is invalid.
E25	Command type error (Slave communication)	The command type of the transmitting command is invalid.
E26	Target type error	The target type is invalid.
E27	No target error	Target (driver card, I/O card, encoder or other slave card) is not installed.
E29	EEPROM error (EWEN/EWDS not permitted)	EEPROM access error (when writing)
E2A	Read compare mismatch error during EEPROM write	EEPROM access error (when writing)
E2B	Abnormal response error when sending EEPROM information acquisition command	An abnormal response was received when a slave-EEPROM information acquisition command was sent.
E2C	Maximum receive size over error when sending EEPROM information acquisition command	The maximum receive size exceeds the limit value when a slave-EEPROM information acquisition command is sent.
E2D	Receive-data checksum error when sending EEPROM information acquisition command	The checksum of receive data is invalid when a slave-EEPROM information acquisition command is sent.
E33	Slave response logic error	The slave response logic is invalid.
E34	Slave block number out of range	The slave block number is out of range.
E37	Slave data setting prohibited	Setting of slave data is prohibited.
E38	Faulty slave EEPROM	The slave EEPROM is faulty.
E39	No encoder EEPROM error	The encoder is not equipped with EEPROM.
E3A	Absolute encoder error	Absolute encoder is specified illegally.
E3C	Undefined slave-command error code detected	An undefined slave-command error code was detected.
E3D	SEL program/point/parameter flash ROM status error	Data is not written to the flash ROM correctly or written in an old, incompatible application version.
E3E	Parameter checksum error	The flash ROM data has been destroyed.
E3F	Gain parameter error	The setting of “Axis-specific parameter No. 60, Position gain,” etc., is invalid.
E40	Rotational-movement axis parameter error	Check axis-specific parameter Nos. 67, 66, 38, 37, 1, etc.
E41	Servo-motion data packet shortage error	There are not enough servo-motion data packets.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
E42	Servo job error	The servo job is invalid.
E45	Servo undefined command detection error	An undefined command was detected during servo processing.
E46	Maximum receive size over error at absolute-data acquisition	The receive size is too large when acquiring absolute data.
E47	No normal response error at absolute-data acquisition	Normal response is not received when acquiring absolute data.
E49	Encoder rotation error	An encoder rotation error was detected.
E4A	Encoder rotation counter overflow error	An encoder rotation counter overflow error was detected.
E4B	Encoder count error	An encoder count error was detected.
E4C	Encoder overspeed error	An encoder overspeed error was detected.
E4D	Driver phase-Z detection logic error	A phase-Z detection completion status was notified from the driver in a mode other than the phase-Z detection operation mode.
E4E	Phase-Z count parameter error	Check axis-specific parameter Nos. 23, 38, 37, etc.
E4F	Synchro parameter error	Check axis-specific parameter Nos. 65, 39, all-axis parameter No. 1, etc.
E50	Driver special command ACK-timeout error	ACK cannot be detected for the driver special command.
E51	Drive unit error (DRVESR)	Error notification from the driver
E52	Encoder error (DRVESR)	Error notification from the driver
E53	Driver CPU error (DRVESR)	Error notification from the driver
E54	Servo control error (DRVESR)	Error notification from the driver
E55	Command error (DRVESR)	Error notification from the driver
E56	Motor temperature error (DRVESR)	Error notification from the driver
E58	Servo ON/OFF timeout error	Servo ON/OFF cannot be confirmed.
E59	Brake ON/OFF timeout error	Brake ON/OFF cannot be confirmed.
E5A	Pole sense non-detection error	Motor magnetic pole cannot be detected.
E5B	Detection OFF error upon pole sense completion	The motor-magnetic-pole detection status bit (Psenex) is turned OFF after completion of pole sense.
E5C	Hold-at-stop servo job error	The servo job is invalid.
E5D	Servo packet error	The servo packets are invalid.
E5E	Servo-control-right management array number error	The servo-control-right management array number is invalid.
E5F	Length conversion parameter error	Check axis-specific parameter Nos. 47, 50, 51, 42, 1, etc.
E60	Slave maximum receive size over error	The slave receive size is too large.
E61	Slave no normal response reception error	Normal response cannot be received from the slave.
E62	Sending-slave CPU type error	The CPU type of the sending slave is invalid.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
E63	Message-buffer information type error	The message-buffer information type is invalid.
E64	Abnormal standby power detection error	Abnormal standby power was detected.
E65	Regenerative resistance temperature error	A regenerative resistance temperature error was detected.
E66	AC-power overvoltage error	An AC-power overvoltage error was detected.
E67	Motor-power overvoltage error	A motor-power overvoltage error was detected.
E68	Emergency-stop status requiring reset recovery (not error)	Reset the emergency stop and then reconnect the power.
E69	Abnormal 24-V I/O power source	The 24-V I/O power source is abnormal.
E6A	Safety-gate open status requiring reset recovery (not error)	Close the safety gate and then reconnect the power.
E6B	Shutdown factor indeterminable error	Shutdown factor cannot be determined.
E6C	DO output current error	The DO output current is abnormal.
E6D	Drive-source cutoff relay error	The drive-source cutoff relay may have been melted.
E71	Encoder configuration information outside supported function information range	An encoder whose configuration information is outside the range supported by the driver unit is installed.
E72	Motor configuration information outside supported function information range	A motor whose configuration information is outside the range supported by the driver unit is installed.
E73	Encoder resolution mismatch error	The encoder resolution in the system’s axis-specific parameter and that of the installed encoder do not match.
E74	Encoder division ratio mismatch error	The encoder division ratio in the system’s axis-specific parameter and that of the installed encoder do not match.
E75	Encoder linear/rotary type mismatch error	The encoder linear/rotary type in the system’s axis-specific parameter and that of the installed encoder do not match.
E76	Encoder ABS/INC type mismatch error	The encoder ABS/INC type in the system’s axis-specific parameter and that of the installed encoder do not match.
E77	Magnetic-pole sensor installation specification mismatch error	The magnetic-sensor installation specification in the system’s axis-specific parameter and that of the installed encoder do not match.
E78	Brake installation specification mismatch error	The brake installation specification in the system’s axis-specific parameter and that of the installed encoder do not match.
E79	Abnormal response error when sending EEPROM-data setting slave command	An abnormal response was received when an EEPROM-data setting slave command was sent.
E7A	Maximum receive size over error when sending EEPROM-data setting slave command	The receive size exceeded the limit value when an EEPROM-data setting slave command was sent.
E7B	Motor-drive power ON timeout error	Abnormal current flow from the motor-drive power source
E7C	Register read/write test error	Error reading/writing the register
E7D	Linear-movement axis parameter error	Check axis-specific parameter Nos. 38, 68, 1, etc.
E7E	Parameter error	The parameter is invalid.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
E7F	Stroke parameter error	Check axis-specific parameter Nos. 7, 8, 1, etc.
E80	Unsupported card error	An unsupported card is installed in an I/O slot.
E81	Priority auto-assignment card non-detection error	Priority auto-assignment card cannot be detected.
E82	Card mismatch error	The combination or positioning of I/O slot cards has a problem.
E83	I/O slot card error	The I/O slot card is invalid.
E84	Resolution parameter error	Check axis-specific parameter Nos. 47, 50, 51, 44, 42, 43, 1, 37, etc.
E85	Driver ready OFF factor indeterminable error	Driver ready OFF factor cannot be determined.
E86	Fieldbus error (FBVCCER)	A fieldbus error (FBVCCER) was detected.
E87	Fieldbus error (FBPOWER)	A fieldbus error (FBPOWER) was detected.
E88	Power error (Other)	A power error (Other) was detected. This error also generates when the power OFF → ON interval is short. After the power has been turned off, be sure to wait for at least 5 seconds before turning it back on. Abnormal regenerative resistance temperature is also suspected.
E89	SCIF open error in non-AUTO mode (Servo in use)	In a mode other than AUTO, opening of the serial 1 channel (also used by the PC software/TP port) from a SEL program is prohibited while the servo is in use (to ensure safety).
E8A	SEL program flash-ROM status error	Data is not written to the flash ROM correctly or written in an old, incompatible application version.
E8B	Symbol definition table flash-ROM status error	Data is not written to the flash ROM correctly or written in an old, incompatible application version.
E8C	Point data flash-ROM status error	Data is not written to the flash ROM correctly or written in an old, incompatible application version.
E8D	Parameter flash-ROM status error	Data is not written to the flash ROM correctly or written in an old, incompatible application version.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
FF0 ~ F00	Shutdown error (hi_sysdwn () definition)	A shutdown error (hi_sysdwn () definition) was detected.
F03 ~ F58	Shutdown error (OS call error)	A shutdown error (OS call error) was detected.
F60	System-down level error-call procedure error	A system-down level error-call procedure error was detected.
F61	Interpreter-task end task ID error	An interpreter-task end task ID error was detected.
F62	Abnormal standby power detection error	Abnormal standby power was detected.
F63	Regenerative resistance temperature error	A regenerative resistance temperature error was detected.
F64	AC-power overvoltage error	An AC-power overvoltage error was detected.
F65	Motor-power overvoltage error	A motor-power overvoltage error was detected.
F66	Servo control underrun error	A servo control underrun error was detected.
F67	FROM-write bus width error	A write operation other than 32-bit long word access was detected while writing the flash ROM.
F68	FROM write protect error	Write operation to a write-protected flash ROM area (FRMWE bit in DEVCTR = 1) was detected.
F69	Boot watchdog error	A FPGA boot watchdog was detected. The core program may not be running properly.
F6A ~ FA0	Undefined exception/interruption error	An undefined exception/interruption occurred.
FB0	TMU0 interruption error	A TMU0 interruption error was detected.
FB1	Application code SDRAM copy error (Checksum)	The sum of 4 bytes does not match between the corresponding sections after FROM → SDRAM program copy.
FB2	Installed flash ROM type mismatch (Application)	The flash ROM type anticipated in the software does not match the flash ROM type actually installed. Check the combination of software and hardware.
FB8	Undefined NMI error	An undefined NMI interruption occurred.

⊙ Error List (MAIN core) (In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
A70	SCIF overrun error	Communication error. Check for noise, connected equipment and communication setting. (When updating the application, connect to a PC and use IAI’s update tool.)
A71	SCIF framing error	Communication error. Check for noise, shorted/disconnected communication cable, connected equipment and communication setting. (When updating the application, connect to a PC and use IAI’s update tool.)
A72	SCIF parity error	Communication error. Check for noise, shorted/disconnected communication cable, connected equipment and communication setting. (When updating the application, connect to a PC and use IAI’s update tool.)
A73	IAI protocol header error	Communication protocol error. Check for noise and connected equipment. (When updating the application, connect to a PC and use IAI’s update tool.)
A74	IAI protocol terminal ID error	Communication protocol error. Check for noise and connected equipment. (When updating the application, connect to a PC and use IAI’s update tool.)
A75	IAI protocol command ID error	Communication protocol error. Check for noise and connected equipment. (When updating the application, connect to a PC and use IAI’s update tool.)
A76	IAI protocol checksum error	Communication protocol error. Check for noise and connected equipment. (When updating the application, connect to a PC and use IAI’s update tool.)
A77	Motorola S record type error	The update program file is invalid. Check the file.
A78	Motorola S checksum error	The update program file is invalid. Check the file.
A79	Motorola S load address error	The update program file is invalid. Check the file.
A7A	Motorola S write address over error	The update program file is invalid. Check the file.
A7B	Flash timing limit over error (Write)	Error writing the flash ROM (When updating)
A7C	Flash timing limit over error (Erase)	Error erasing the flash ROM (When updating)
A7D	Flash verify error	Error erasing/writing the flash ROM (When updating)
A7E	Flash ACK timeout	Error erasing/writing the flash ROM (When updating)
A7F	Head sector number specification error	Error erasing the flash ROM (When updating)
A80	Sector count specification error	Error erasing the flash ROM (When updating)
A81	Write-destination offset address error (Odd-numbered address)	The address written during flash ROM write (when updating) is invalid. Check the update program file.
A82	Write-source data buffer address error (Odd-numbered address)	Error writing the flash ROM (When updating)
A83	Invalid code sector block ID error	The flash ROM is new, or the program currently written to the flash ROM is invalid because the last update was aborted. The ROM can be updated without problem.
A84	Code sector block ID erase count over	The number of times the flash ROM was erased exceeded the allowable count.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
A85	FROM write request error before erase is complete	When updating, a flash-ROM write command was received before a flash-ROM erase command. Confirm that the update program file is valid and then perform update again.
A86	Absolute-encoder backup battery voltage-low warning (Driver detection)	The voltage of the absolute-data backup battery is low. Check the battery connection or replace the battery.
A87	Motorola S-byte count error (Core detection)	The update program file is invalid. Check the file.
A88	Message conversion error (Core detection)	The received message does not conform to the message format or contains invalid data. Check the message sent from the host communication device.
A89	Updating target non-specification error (Core detection)	During update, an update command was received before the updating target was specified properly. Check if an appropriate updating PC tool is used and the target specification and other settings in the updating PC tool are correct.
A8A	Updating system code error (Core detection)	The system code in the message received with the updating target specification command does not match the controller system. Check the target specification and other settings in the updating PC tool.
A8B	Updating unit code error (Core detection)	The unit code in the message received with the updating target specification command does not match any updatable unit in the controller. Check the target specification and other settings in the updating PC tool.
A8C	Updating device number error (Core detection)	The specified device number in the message received with the updating target specification command is not appropriate. Check the target specification and other settings in the updating PC tool.
A8D	Flash busy reset timeout (Core detection)	Error erasing/writing the flash ROM
A8E	Unit type error (Core detection)	The unit type specified in the message received with the command is invalid or not supported.
CD0	Drive error (Driver detection)	Error notification from the driver
CD1	Encoder error (Driver detection)	Error notification from the driver
CD2	Driver CPU error (Driver detection)	Error notification from the driver
CD3	Servo control error (Driver detection)	Error notification from the driver
CD4	Command error (Driver detection)	Error notification from the driver
CD5	Motor temperature error (Driver detection)	Error notification from the driver

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
E90	Core code flash-ROM status error	The core program is invalid. Contact the manufacturer.
E91	Application code flash-ROM status error	The application program is invalid. Contact the manufacturer.
E92	Core code sum error	The core program is invalid. Contact the manufacturer.
E93	Application code sum error	The application program is invalid. Contact the manufacturer.
E94	Timing limit over error (Flash erase)	Error erasing the flash ROM
E95	Flash verify error (Flash erase)	Error erasing the flash ROM
E96	Flash ACK timeout (Flash erase)	Error erasing the flash ROM
E97	Head sector number specification error (Flash erase)	Error erasing the flash ROM
E98	Sector count specification error (Flash erase)	Error erasing the flash ROM
E99	Timing limit over error (Flash write)	Error writing the flash ROM
E9A	Flash verify error (Flash write)	Error writing the flash ROM
E9B	Flash ACK timeout (Flash write)	Error writing the flash ROM
E9C	Write-destination offset address error (Flash write)	Error writing the flash ROM
E9D	Write-source data buffer address error (Flash write)	Error writing the flash ROM
E9E	Watchdog reset occurrence error	A WDT (watchdog timer) was manually reset (error detection).
E9F	Exception occurrence error while BL = 1 (NMI)	An exception occurred while the block bit in the CPU status register was “1.” (NMI)
EA0	Exception occurrence error while BL = 1 (Other than NMI)	An exception occurred while the block bit in the CPU status register was “1.” (Other than NMI)
EA1	Bit exception reset due to command/data TLB duplication	This reset occurs when there are multiple TLB entries corresponding to the virtual address.
EA2	Undefined exception/interruption error	An undefined exception/interruption occurred.
EA3	AC-power cutoff detection error	An AC-power cutoff was detected.
EA4	Abnormal standby power detection error	Abnormal standby power was detected.
EA5	Regenerative resistance temperature error	A regenerative resistance temperature error was detected.
EA6	AC-power overvoltage error	An AC-power overvoltage error was detected.
EA7	Motor-power overvoltage error	A motor-power overvoltage error was detected.
EA8	FROM-write bus width error	A write operation other than 32-bit long word access was detected while writing the flash ROM.
EA9	FROM write protect error	Write operation to a write-protected flash ROM area (FRMWE bit in DEVCTR = 1) was detected.
EAA	SDRAM write/read test error	The SDRAM is faulty. Contact the manufacturer.
EAB	Application-update SCIF send-queue overflow error	An overflow occurred in the send queue.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
EAC	Servo control underrun error	A servo control underrun error was detected.
EAD	Boot error	A FPGA boot watchdog was detected. The core program may not be running properly.
EAE	Application-update SCIF receive-queue overflow error	Excessive data is received from outside. (Confirm that a PC and IAI’s update tool are used to update the application.)
EAF	Installed flash ROM type mismatch (Core)	The flash ROM type anticipated in the software does not match the flash ROM type actually installed. Check the combination of software and hardware.
EB0	Undefined NMI error (Core)	An undefined NMI interruption occurred.
EB1	FPGA read/write test error (Core)	A read/write error of the FPGA.
EB2	Flash busy reset timeout (Core detection)	Flash ROM malfunction. The busy status of the flash ROM is not reset.

⊙ Troubleshooting of PSEL Controller

After the optional panel unit was connected, the panel window began displaying an error number every time an error generates.
When the power is turned on, normally “rdy” or “Ardy” will be displayed. “P01” or other code will be displayed while a program is running.
When an error generates, the panel window will show “EA1D” or other code starting with “E.” (Some errors do not begin with “E.”)

Status	Panel window display
After turning on the power	rdy, Ardy
Program is running	P01, P64, etc.
Error has generated	EA1D, ED03, etc.

* Among the alphabets, B and D are shown in lower case.

Depending on the error number, it may be possible to reset the error after removing the cause of the error, or the power must be reconnected to reset the error.
Also, some error numbers are output to the LED display in the panel window, while others are not.
For details, see “⊙ Error Level Control.”

Troubleshooting (Causes and Countermeasures for Key Errors)

Error No.	Error name	Cause	Countermeasure
dCF	DC power cutoff	Momentary power failure has occurred or the voltage has dropped.	Check the power-source voltage. (24-VDC specification)
ErG	Emergency stop (This is not an error.)	Emergency-stop signal is input.	Emergency-stop signal is input in the following condition: 1. The emergency-stop button on the teaching pendant is pressed. 2. The applicable input terminal in the system connector is turned ON. 3. The port switch on the front panel is set to the manual side. (The teaching-pendant/PC-software connector is not connected.) 4. The actuator is of sensor specification and the slider is stopped on either end of the slider.
enb	Safety gate open	The safety gate is open.	Check the system connector wiring.
C9C	Defective phase-Z position error	The phase-Z position is defective or the reversing amount at home return is small.	Check to see if foreign object has entered the actuator. Check to see if the mounting bolts are contacting the slider. * Change axis-specific parameter No. 22 to “100.”
914 CA2	Abnormal absolute-data backup battery voltage	The PG cable was disconnected from the controller. Absolute reset has not been executed after the initial setup. The voltage of the absolute-data backup battery has dropped.	Connect the PG cable to the controller and execute an absolute reset. Replace the absolute-data backup battery and execute an absolute reset.
CA5	Stop deviation overflow error	Operation is mechanically disabled. If there is no problem in the mechanical function, the power stage board is faulty.	Check to see if the actuator mounting bolts are contacting inside the axes, or if the slider attachment is contacting any surrounding mechanical parts. Replace the board.
C6b	Deviation overflow error	Operation is mechanically disabled.	Check to see if the actuator mounting bolts are contacting inside the axes, or if the slider attachment is contacting any surrounding mechanical parts.

Error No.	Error name	Cause	Countermeasure
d03	Faulty encoder or attachment of dust	The encoder is faulty or dust is attached.	Remove the motor cover and apply cleaning air spray for OA equipment, etc., over the cord wheel. If the problem persists, replace/readjust the encoder.
d06	Encoder received-data error	The encoder cable is disconnected.	Replace the encoder cable.
690	Motor overcurrent error	The motor coil is damaged.	Measure the inter-phase resistances among U, V and W. If the measured resistances are not the same, burn damage is suspected. Replace the motor. If the measured resistances are roughly the same, there is no burn damage.
		If the motor coil is not damaged, the driver's CPU board (the board to which the motor drive cable is connected) is faulty.	Replace the board.
d19	Encoder receive timeout error	The encoder cable is disconnected.	Replace the encoder cable.
d18	Speed loop underrun error	The driver CPU board was damaged due to noise in the encoder cable.	Replace the board and implement noise control measures.
807	Shutdown relay ER status	The transistor on the power-supply board (to which the power cable is connected) is damaged.	Replace the board.

MEMO

Trouble Report Sheet

Trouble Report Sheet				Date:	
Company name		Department		Reported by	
TEL	(Ext)	FAX			
IAI agent		Purchase date			
Serial number		Manufacture date			

[1] Number of axes ☐ axis(es)

Type

[2] Type of problem

1. Disabled operation 2. Position deviation 3. Runaway machine
4. Error Error code = ☐☐☐
5. Other ( )

[3] Problem frequency and condition

Frequency =

Condition _____

_____

_____

_____

_____

[4] When did the problem occur?

1. Right after the system was set up
2. After operating for a while (Operating hours: _____ year(s) and _____ month(s))

[5] Operating direction

1. Horizontal 2. Horizontal + Vertical

[6] Load condition

1. Load transfer 2. Push-motion operation 3. Load: Approx. _____ kg
4. Speed: Approx. _____ mm/sec

[7] Special specification (option, etc.)

Catalog No.: ME0172-4A (September 2008)

IAI
Quality and Innovation

IAI America, Inc.

Head Office: 2690 W. 237th Street, Torrance, CA 90505
TEL (310) 891-6015 FAX (310) 891-0815
Chicago Office: 1261 Hamilton Parkway, Itasca, IL 60143
TEL (630) 467-9900 FAX (630) 467-9912
Atlanta Office: 1220-E Kennestone Circle, Marrietta, GA 30066
TEL (678) 354-9470 FAX (678) 354-9471

website: www.intelligentactuator.com

IAI Industrieroboter GmbH

Ober der Röth 4, D-65824 Schwalbach am Taunus, Germany
TEL 06196-88950 FAX 06196-889524