X-SEL
IX-Series Controller
Operation Manual
Second Edition

X-SEL
CONTROLLER

IAI America, Inc.

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]

- If you are using a pace maker or other mechanical implant, do not come within one meter of the product. The strong magnetic field generated by the product may cause the pace maker, etc., to malfunction.
- 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 loading capacity 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.
- 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). 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.
- If the LEDs on the product do not illuminate after turning on the power, turn off the power immediately. The protective device (fuse, etc.) on the live side may remain active. Request repair to the IAI sales office from which you purchased the product.

[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]
- Turn on the power to individual equipment one by one, starting from the equipment at the highest level in the system hierarchy. 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 work to drop in case of power failure. Configure a control circuit that will prevent the table or work from dropping when the power to the machine is cut off or an emergency stop is actuated.

[Installation, Operation, Maintenance]
- When handling the product, wear protective gloves, protective goggles, safety shoes or other necessary gear 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.”
- If you have any question regarding the product, please contact your nearest IAI sales office. The addresses and phone numbers of our sales offices are provided at the end of this operation manual.

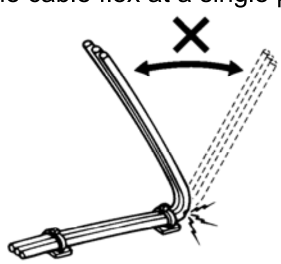
Prohibited Handling of Cables

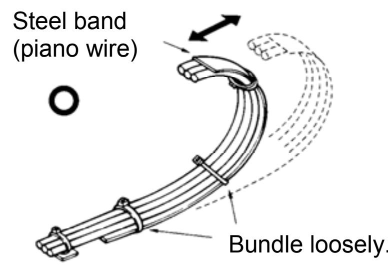
⚠ Caution

When designing an application system using actuators and controllers, incorrect wiring or connection of each cable may cause unexpected problems such as a disconnected cable or poor contact, or even a runaway system. This section explains prohibited handling of cables. Read the information carefully to connect the cables properly.

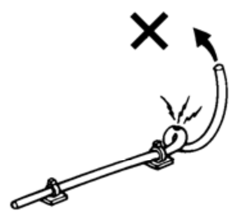
<u>Ten Rules for Handling Cables (Must be Observed!)</u>

1. Do not let the cable flex at a single point.

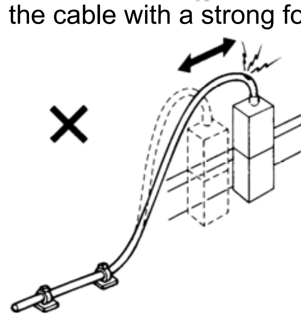




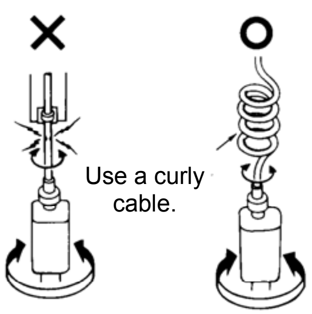
2. Do not let the cable bend, kink or twist.



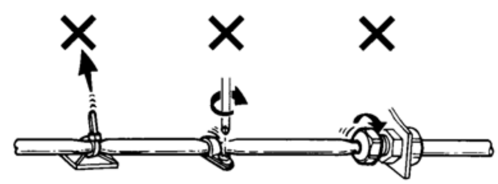
3. Do not pull the cable with a strong force.

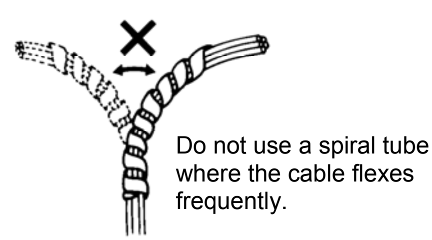


4. Do not let the cable receive a turning force at a single point.

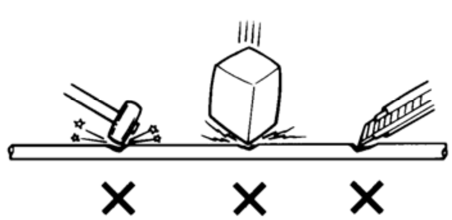


5. When fixing the cable, provide a moderate slack and do not tension it too tight.

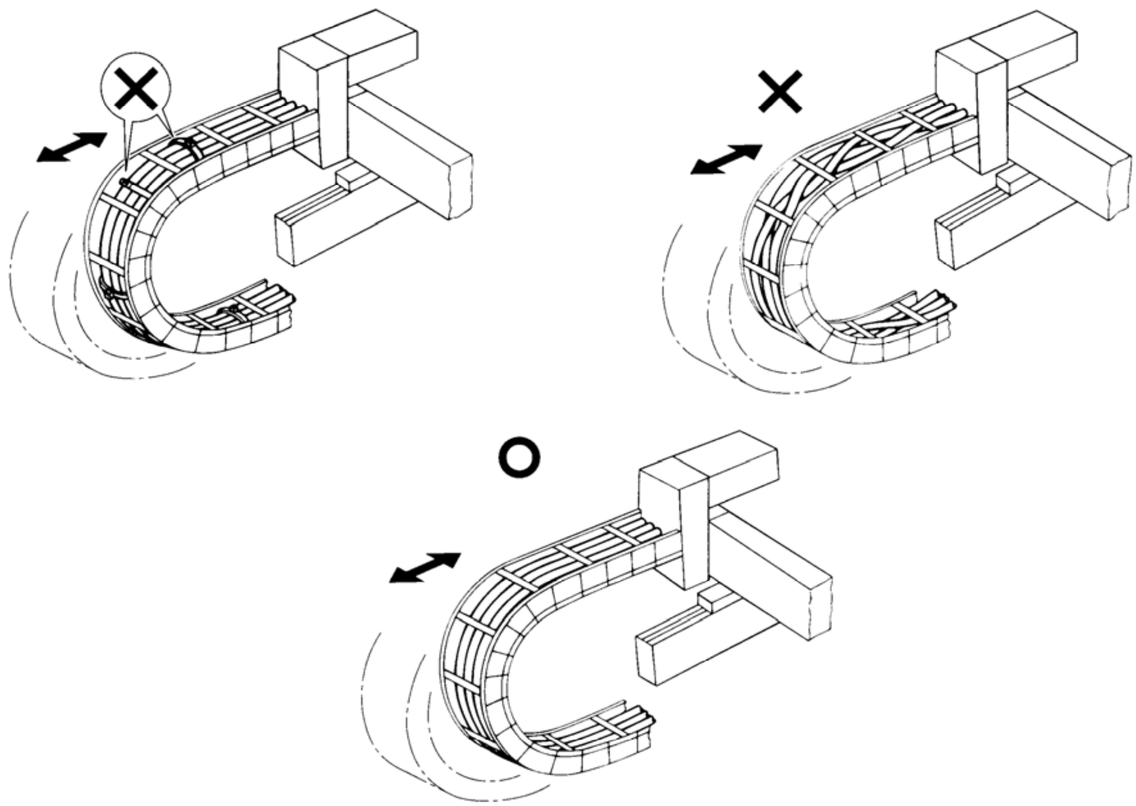




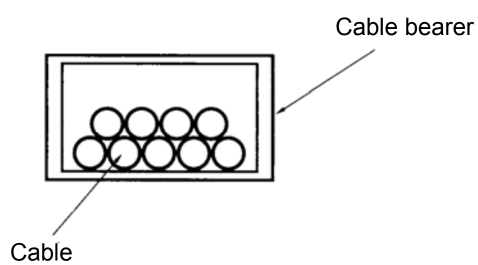
6. Do not pinch, drop a heavy object onto or cut the cable.



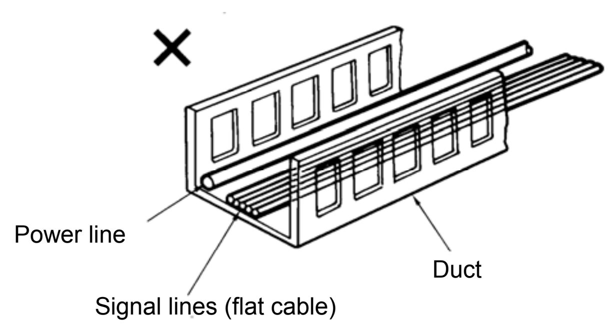
7. Do not let the cable got tangled or kinked in a cable bearer or flexible tube. When bundling the cable, keep a certain degree of flexibility (so that the cable will not become too taut when bent).



8. Do not cause the cables to occupy more than 60% of the space in the cable bearer.

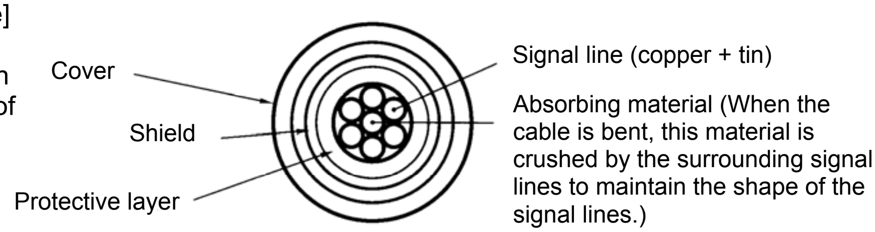


9. Do not lay signal lines together with circuit lines that create a strong electric field.



10. Always use a robot cable if the cable is likely to flex significantly.

[Standard structure of cable]
The standard structure of cable will vary depending on the manufacturer and type of cable.



★ Need for Robot Cables
A cable connected to a moving part of an actuator system will inevitably receive repeated bending loads at the base of the cable. As a result, the cores in the cable may break over time. To minimize the risk of cable breakage, we strongly recommend that a robot cable offering significantly higher flexibility be used in this type of application.

Before Use

⚠ Caution

■ Caution

1. Be sure to read this operation manual to ensure the proper use of this product.
2. Unauthorized use or reproduction of a part or all of this operation manual is prohibited.
3. Always handle or operate the product in manners specified in this operation manual, by assuming that whatever is not specified herein is not feasible. The warranty does not cover any defect arising from a handling or operation not specified in this operation manual.
4. The information contained in this operation manual is subject to change without notice for the purpose of modification and improvement.
   * If you have purchased PC software:
     Always back up the parameters after installing the product or changing the parameter settings.
5. The specifications in this manual may not apply to a custom product.

⚠ Caution

■ Action to Be Taken in Case of Emergency

If this product is found to be in a dangerous condition, immediately turn off all power switches of the main unit and connected equipment or immediately disconnect all power cables from the outlets. ("Dangerous condition" refers to a situation where the product is generating abnormal heat or smoke or has ignited and a fire or danger to human health is anticipated.)

Table of Contents

Introduction

Thank you for purchasing the X-SEL IX-Series Controller.

Inappropriate use or handling will prevent this product from demonstrating its full function and may even cause unexpected failure or result in a shortened service life. 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.

The IX-Series Controller you have purchased is a new-generation controller that uses a 32-bit RISC (Reduced Instruction Set Computer) CPU and an advanced version of the proven "SEL" language to perform a range of high functions such as multi-tasking, sequencer-less operation, palletizing function and pseudo-sequence function.

- Actuator duty
  IAI recommends that our actuators be used at a duty of 50% or less in view of the relationship of service life and accuracy. The duty is calculated by the formula below:

$$\text{Duty (\%)} = \frac{\text{Operating hours}}{\text{Operating hours} + \text{Stopped hours}} \text{ X } 100$$

- After turning off the main power, be sure to wait for at least 5 seconds before turning it on.
  Any shorter interval may generate "E88: Power system error (Other)."

- Do not plug in/out the connectors while the power is still supplied to the controller. Doing so may result in 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.

Part 1 Installation

Caution

Chapter 1 Safety Precautions

This controller has been designed for use exclusively with IAI's IX-Series Horizontal Articulated Robot, and is able to provide integrated control over the entire system including peripherals. In other words, the X-SEL Controller has the ability to control systems of all sizes ranging from a small system to a large factory automation system. In general, however, the occurrence rate of accidents due to wrong operation or carelessness will rise as the system becomes larger and more complex. Please give due consideration to safety measures.

Strict observance of the following items is requested to prevent unforeseen danger. Also read the appendix entitled, "Safety Rules and Others."

1. Do not handle this product in manners not specified in this manual. If you have any question regarding the content of this manual, please contact IAI.

2. Always use the specified, genuine IAI cables for wiring between the controller and the actuator.

3. Do not enter the operation area of the machine while the machine is operating or ready to operate (the controller power is on). If the machine is used in a place accessible to other people, provide an appropriate safety measure such as enclosing the machine with a cage.

4. When assembling/adjusting or maintaining/inspecting the machine, always turn off the controller power at the source beforehand. The operator should display in a conspicuous place a plate or other sign saying that operation is in progress and that the power should not be turned on. The operator should keep the entire power cable beside him or her to prevent another person from inadvertently plugging in the cable.

5. When two or more operators are to work together, set call-out signals to ensure safety of all personnel during the work. In particular, a person turning on/off the power or moving an axis—either via a motor or manually—must always say what he or she is going to do out loud and confirm the responses from the others first before actually performing the operation.

Chapter 2 Warranty Period and Scope of Warranty

The X-SEL 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 3 Installation Environment and Noise Measures

1. 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
Operating temperature range	0°C ~ 40°C
Forced air-cooling	24-VDC fan x 5 (compact type) 24-VDC fan x 6 (general-purpose type)
Operating humidity range	30% ~ 85%
Storage temperature range	-10°C ~ 65°C
Dust protection structure	IP10
Power-source voltage	100 to 115/200 to 230 VAC, single-phase
Operating power-source voltage	±10%
Rated operating power-source frequency	50 Hz/60 Hz

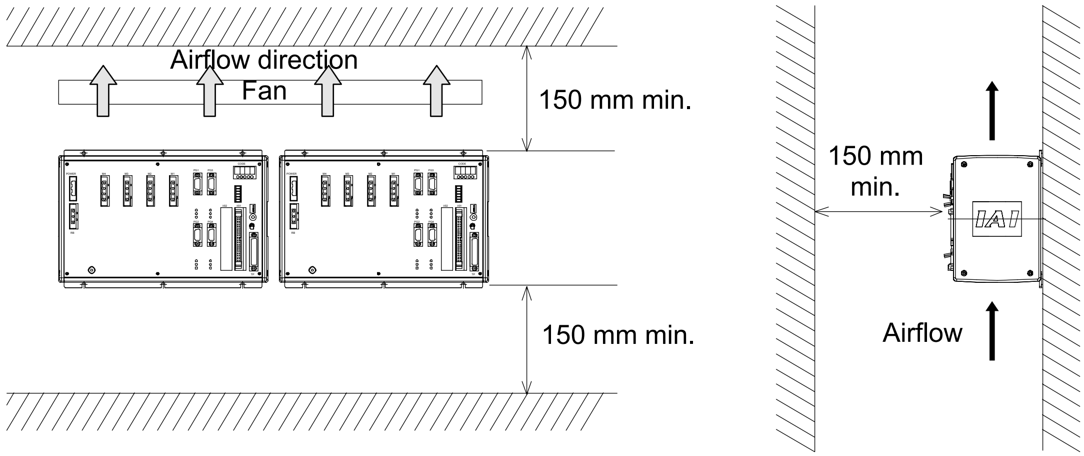
2. Heat Radiation and Installation

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 forced ventilation (exhaust air will be discharged from the top). Be sure to install the controller in the aforementioned direction and provide a minimum clearance of 150 mm above and 150 mm 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 150 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.

3. Power Source

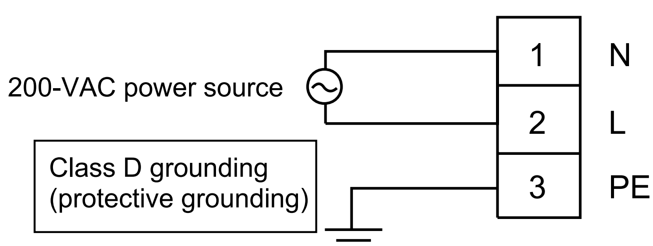
Provide a single-phase power source of 200 to 230 VAC.

4. Noise Measures and Grounding

(1) Wiring and power source

PE on the power terminal block is used for protective grounding. Provide Class D grounding from this terminal.

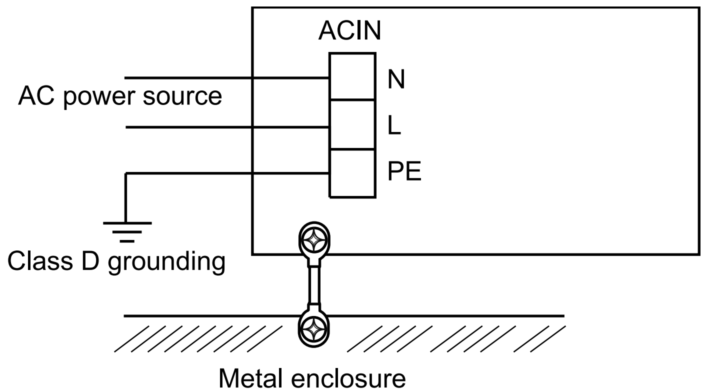
Use a grounding cable with a wire size of 1.0 mm^2 (#AWG17) or more, which should not be smaller than the AC power cable.



* Notes on wiring method
Use twisted cables for the AC power cable and 24-VDC external power cable. 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.

(2) Noise-elimination grounding

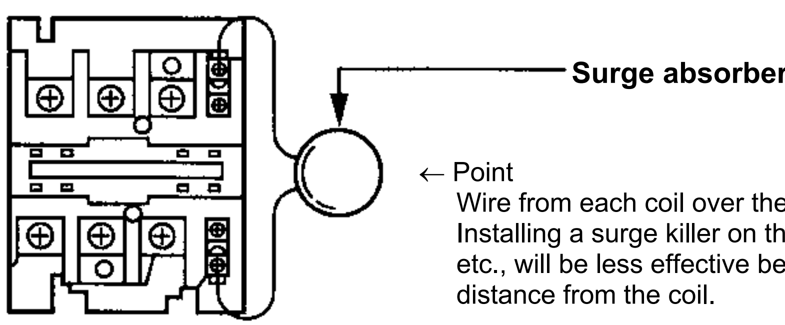


Connect the FG terminal with the metal enclosure using a cable of a maximum allowable size over the shortest distance.

(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:

a. AC solenoid valve, magnet switch, relay

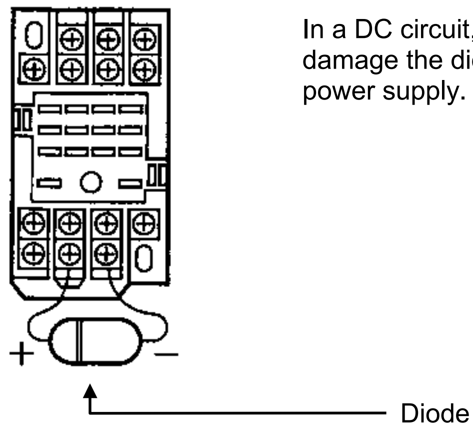
Measure --- Install a surge absorber in parallel with the coil.



← 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.

b. DC solenoid valve, magnet switch, relay

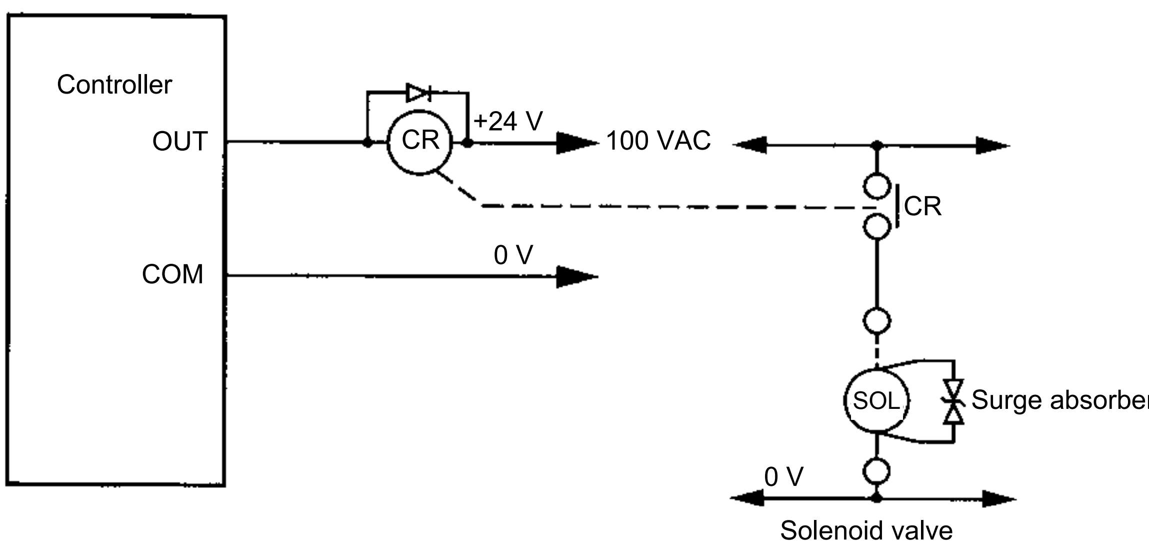
Measure --- Install a diode in parallel with the coil. Determine the diode capacity in accordance with the load capacity.



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.

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.

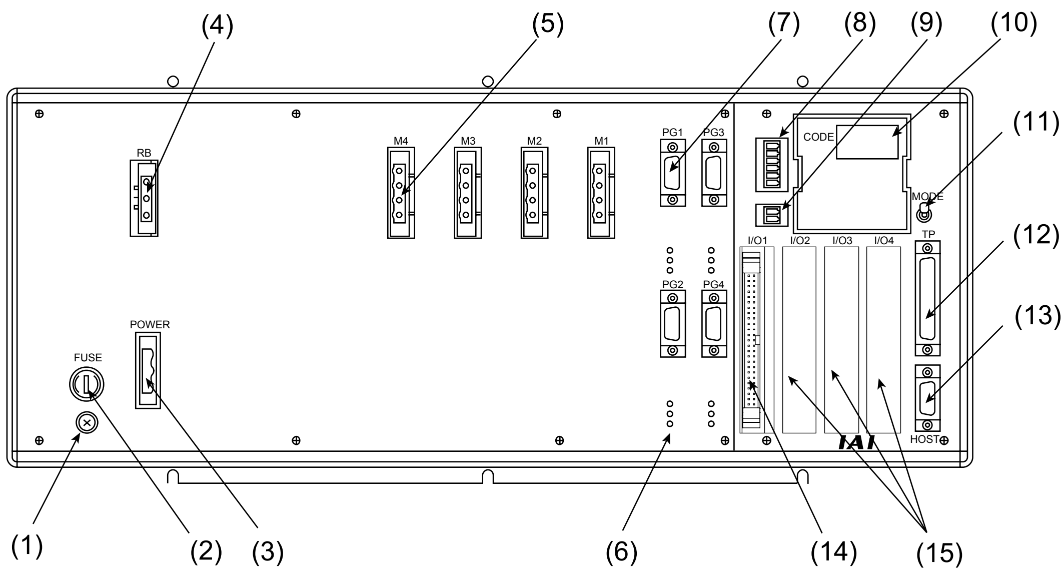
Reference Circuit Diagram



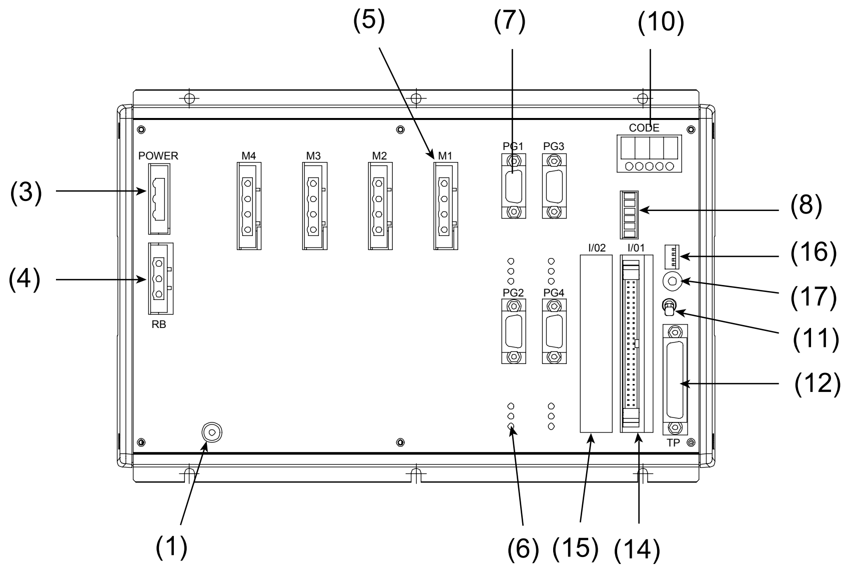
Chapter 4 Name and Function of Each Part

1. Front View of Controller

General-purpose type (for robots of all strokes)



Compact type (for robots of 250/350 strokes)



(1) FG terminal...... This terminal is used to ground FG on the enclosure. With a general-purpose type, the enclosure is connected to PE in the AC input part inside the controller. With a compact type, FG and PE are not connected inside the controller. Be sure to ground the FG terminal.

FG Terminal Specifications

Item	Overview
	M4 3-point SEMS screw, 5 mm
Cable size	2.0 ~ 5.5 mm^2
Grounding method	Class D grounding

(2) Fuse holder (general-purpose type only) .... This half-cut fuse holder is used to protect overcurrent in the AC input part. It prevents inflow of abnormal current generated by ground fault, etc., and uses a slow-blow fuse specified by IAI.

Overview of Fuse Holder Specifications

Item	Description
Holder	F-22001-A1 by Sato Parts
Fuse type	TWO 250V20A by Fuji Terminal *
Melting characteristics	Slow-blow

* With a compact type, the fuse holder is mounted on the board. Type: FGMT5 AC250V10A by Fuji Terminal

(3) AC input connector .... 200-VAC, single-phase input connector.

AC Power Connector Specifications

Item	Overview			
Connector name	POWER			
Connector	GMSTB2.5/3-STF-7.62 by Phoenix Contact 3-pin, 2-piece connector			
Supported cable size	1.25 ~ 2.5 mm^2 (AWG12 ~ 16)			
Connected to	AC power source			
Terminal assignments	1	In	N	AC power input, N side
	2	In	L	AC power input, L side
	3		⏚	PE (Protective grounding line)

(4) External regenerative unit connector .... This connector is used to connect a regenerative resistance unit that may be required when the controller is used in a high-speed/high-load environment, etc., and the built-in regenerative resistance capacity is not sufficient. Whether or not an external regenerative resistance is necessary will be determined by the specific application such as axis configuration.

Overview of Fuse Holder Specifications

Item	Overview			
Connector name	RB			
Connector	GIC2.5/3-STF-7.62 by Phoenix Contact 3-pin, 2-piece connector			
Cable size	1.0 mm^2 (AWG17 or equivalent), included in the external regenerative box			
Connected to	External regenerative box			
Terminal assignments	1	Out	RB+	Regenerative resistance + (Motor-driving DC voltage)
	2	In	RB–	Regenerative resistance –
			⏚	

(5) Motor connector.................. This connector is used to drive the motor inside the actuator.

Overview of Motor Connector Specifications

Item	Description			
Connector name	M			
Connector	GIC2.5/4-STF-7.62 by Phoenix Contact 4-pin, 2-piece connector			
Cable	Dedicated motor cable			
Connected to	Actuator			
Terminal assignments	1		⏚	PE (Protective grounding line)
	2	Out	U	Motor-driving phase U
	3	Out	V	Motor-driving phase V
	4	Out	W	Motor-driving phase W

(6) Driver status LEDs.............. These LEDs monitor the operating status of the driver CPU that controls the motor drive. The following three LEDs are provided.

Name	Color	Meaning when lit
ALM	Orange	The driver is detecting an error.
SVON	Green	The servo is on and the motor is being driven.
BAT ALM	Orange	The voltage of the absolute-data backup battery is low.

(7) Encoder connector............ This 15-pin, D-sub connector is used to connect the actuator's encoder.

Encoder Connector Specifications

Item	Description			
Connector name	PG			
Connector	High-density D-sub, 15-pin (female)			
Maximum connection distance	10 m			
Interface standard	Conforming to RS422			
Connected to	Actuator (Built-in encoder unit inside the actuator)			
Connection cable	Dedicated PG cable			
Terminal assignments	Pin No.	I/O	Signal name	Description
	1	In	A+	Phase-A differential + input (Phase U+)
	2	In	A–	Phase-A differential – input (Phase U–)
	3	In	B+	Phase-B differential + input (Phase V+)
	4	In	B–	Phase-B differential – input (Phase V–)
	5	In	Z+	Phase-Z differential + input (Phase W+)
	6	In	Z–	Phase-Z differential – input (Phase W–)
	7	IO	SRD+	Send/receive line+ (Pulse/magnetic-pole switching+)
	8	IO	SRD–	Send/receive line– (Pulse/ magnetic-pole switching–)
	9	Out	BATT	Backup-battery power supply
	10	Out	BATTGND	Battery ground
	11	Out	VCC	Encoder power source
	12	Out	GND	GND
	13	Out	BK–	Brake output
	14	Out	BK+	
	15		FG	Not used

(8) System I/O connector........ This connector is used to connect an emergency-stop switch, ENABLE contact, ready relay, etc.

Connector	MC1.5/6-ST-3.5 by Phoenix Contact; 6-pin, 2-piece connector		
Terminal assignments	1	RDY OUT	Ready-status output contact
	2	RDY OUT	
	3	ENB IN	Safety-gate input
	4	+24V OUT	+24-V power output for safety gate
	5	EMG IN	Emergency-stop input
	6	+24V OUT	+24-V power output for emergency stop

Pins 1 and 2 form a contact-A output that turns ON under the following condition:

- SYSRDY is output (software = PIO trigger program can be run) and hardware is normal (emergency stop is not being actuated and hardware error is not being detected).

Pins 3 and 4 form a contact-B safety-gate input. Operation is enabled when the pins are shorted, while the drive source is cut off when they are open.
Pins 5 and 6 form a contact-B emergency-stop input. Operation is enabled when the pins are shorted, while an emergency stop is actuated when they are open.
The controller is shipped with pins 3 and 4, and 5 and 6, shorted by a cable, respectively.

(9) I/O24V power connector (general-purpose type only).... This connector is used to externally supply I/O power to the insulated part when DI and DOs are mounted in the I/O connectors explained in (14) and (15) (2-pin, 2-piece connector by Phoenix Contact). 24 V must be supplied externally.

Supported cable size	0.75 ~ 1.25 mm^2 (AWG16)		
Connector	MC1.5/6-ST-3.5 by Phoenix Contact; 2-pin, 2-piece connector		
Terminal assignments	1	0V	I/O GND
	2	24V IN	+24-V power input for I/Os

With a compact type, power is supplied externally to pin Nos. 1 and 50 of the I/O connector in (14).

(10) Panel window.................... This window consists of a 4-digit, 7-segment LED display and five LED lamps that indicate the status of the equipment.
For the information shown on the display, refer to 2, “Explanation of Codes Displayed on the Panel Window” or the “Error Code Table.”

Meanings of Five LEDs

Name	Status when the LED is lit
RDY	CPU ready (program can be run)
ALM	CPU alarm (system-down level error), CPU hardware error
EMG	Emergency stop has been actuated, CPU hardware error, power-system hardware error
PSE	Power-system hardware error
CLK	System clock error

(11) Mode switch...................... This alternate switch with lock is used to command a controller operation mode. To operate the switch, pull it toward you and tilt.
Tilting the switch upward will select MANU (manual mode), while tilting it downward will select AUTO (auto mode). Teaching can be performed only in the MANU mode, but auto program start is not enabled in the MANU mode.
(Refer to the types of manual operations explained on p.308.)

(12) Teaching-pendant connector .... When an optional teaching pendant or PC is connected, this D-sub, 25-pin connector will be used to input program and position data in the MANU mode.

Interface Specifications of Teaching Serial Interface

Item	Description
Connector name	TP
Connector	DSUB-25 XM3B-2542-502L (Omron)
Communication method	RS232C-compliant, start-stop synchronous method
Baud rate	38.4 kbps max.; half-duplex communication
Maximum connection distance	10 m (38.4 kbps)
Interface standard	RS232C
Connected to	X-SEL teaching pendant

Interface Specifications of Teaching Serial Interface

Item	No.	Direction	Signal name	Description
Terminal assignments	1		FG	Frame ground
	2	Out	TXD	Transmitted data
	3	In	RXD	Received data
	4	Out	RTS	Request to send
	5	In	CTS	Clear to send
	6	Out	DSR	Equipment ready
	7		SG	Signal ground
	8			
	9	In		Connection prohibited
	10	In		Connection prohibited
	11			
	12	Out	EMGOUT	Emergency stop
	13	In	EMGIN	
	14			
	15	Out		Connection prohibited
	16	Out		Connection prohibited
	17	Out		Connection prohibited
	18	Out	VCC	Power output (5-V power source for teaching pendant)
	19	In	ENBTBX	Enable input
	20	In	DTR	Terminal ready
	21			
	22			
	23	Out	EMGS	Emergency-stop status
	24			
	25		SG	Signal ground

(13) PC connector .... (general-purpose type only)

This D-sub, 9-pin connector is used to perform serial communication (RS232C) with the host equipment when AUTO is selected as the operation mode.
* (12) and (13) cannot be used simultaneously.

RS232 Host Connector Specifications
This connector is used to establish a serial connection with a PC or PLC to enable controller control.

Item	Description			
Connector name	HOST			
Connector	D-sub, 9-pin (DTE); XM2C-0942-502L by Omron			
Maximum connection distance	10 m (38.4 kbps)			
Interface standard	RS232C			
Connected to	AT-compatible PC, etc. (half-duplex communication)			
Connection cable	Dedicated cable			
Terminal assignments	Pin No.	I/O	Signal name	Description
	1		NC	
	2	In	RD	Received data (RXD)
	3	Out	SD	Transmitted data (TXD)
	4	In	DR	Data set ready (DSR)
	5	In	SG	Signal ground
	6	Out	ER	Equipment ready (DTR)
	7	Out	RS	Request to send (RTS)
	8	In	CS	Clear to send (CTS)
	9		NC	Not used

A dedicated cable must be used if an AT-compatible PC is to be connected.

The PC connector (D-sub, 9-pin) (13) will become available for use by the user only when the teaching-pendant connector (D-sub, 25-pin) (12) is not in use.

The PC connector (D-sub, 9-pin) (13) and teaching-pendant connector (D-sub, 25-pin) (12) cannot be used simultaneously. Setting the mode switch (11) to MANU will select the teaching-pendant connector (12), while setting it to AUTO will select the PC connector (13).

(14) Standard I/O connector ...... This connector consists of a 50-pin flat connector and comprises 32-input/16-output DIOs.

Overview of Standard I/O Interface Specifications

Item	Description
Connector name	I/O
Connector	Flat connector, 50-pin
Power supply	With a general-purpose type, power is supplied from the I/O24V power connector (9). With a compact type, power is supplied from connector pin Nos. 1 and 50.
Input	32 points (including general-purpose and dedicated inputs)
Output	16 points (including general-purpose and dedicated outputs)
Connected to	External PLC, sensor, etc.

The functions are at the time of shipment. The functions assigned to port Nos. 000 to 015, 300 to 308 and 313 can be changed via I/O parameters. (Refer to Nos. 30 to 54 and No. 59 in 1, "I/O Parameters," of Appendix, "List of Parameters.")

I/O Interface List

Pin No.	Category	Port No.	Function	Cable color
1	Input	-	General-purpose: NC, Compact: +24-V input	Brown-1
2		000	Program start	Red-1
3		001	General-purpose input	Orange-1
4		002	General-purpose input	Yellow-1
5		003	General-purpose input	Green-1
6		004	General-purpose input	Blue-1
7		005	General-purpose input	Purple-1
8		006	General-purpose input	Gray-1
9		007	Program specification (PRG No. 1)	White-1
10		008	Program specification (PRG No. 2)	Black-1
11		009	Program specification (PRG No. 4)	Brown-2
12		010	Program specification (PRG No. 8)	Red-2
13		011	Program specification (PRG No. 10)	Orange-2
14		012	Program specification (PRG No. 20)	Yellow-2
15		013	Program specification (PRG No. 40)	Green-2
16		014	General-purpose input	Blue-2
17		015	General-purpose input	Purple-2
18		016	General-purpose input	Gray-2
19		017	General-purpose input	White-2
20		018	General-purpose input	Black-2
21		019	General-purpose input	Brown-3
22		020	General-purpose input	Red-3
23		021	General-purpose input	Orange-3
24		022	General-purpose input	Yellow-3
25		023	General-purpose input	Green-3
26		024	General-purpose input	Blue-3
27		025	General-purpose input	Purple-3
28		026	General-purpose input	Gray-3
29		027	General-purpose input	White-3
30		028	General-purpose input	Black-3
31		029	General-purpose input	Brown-4
32		030	General-purpose input	Red-4
33		031	General-purpose input	Orange-4
34	Output	300	Alarm output	Yellow-4
35		301	Ready output	Green-4
36		302	Emergency-stop output	Blue-4
37		303	General-purpose output	Purple-4
38		304	General-purpose output	Gray-4
39		305	General-purpose output	White-4
40		306	General-purpose output	Black-4
41		307	General-purpose output	Brown-5
42		308	General-purpose output	Red-5
43		309	General-purpose output	Orange-5
44		310	General-purpose output	Yellow-5
45		311	General-purpose output	Green-5
46		312	General-purpose output	Blue-5
47		313	General-purpose output	Purple-5
48		314	General-purpose output	Gray-5
49		315	General-purpose output	White-5
50		-	General-purpose: NC, Compact: 0 V	Black-5

(15) Expansion I/O connectors .... These connectors are used to install I/O expansion boards.
I/O expansion boards are optional with a general-purpose type.
With compact types, only one expansion board can be installed.
Note) The connector pins are similar to those of the standard I/O connector.
24 VDC must be input to pin Nos. 1 and 50.

(16) System-operation setting switches (compact type only) .... These switches are used to set the system operation mode. Normally all switches should be set to OFF.

(17) Boot-target specification switch (compact type only) .... This switch is used to select the device that will be updated when the system implements program update. Normally this switch should be set to “1.”

2. Explanation of Codes Displayed on the Panel Window

2-1. Application

Display				Priority (*1)	Description
	A	C	F	1	AC power is cut off (including momentary power failure or drop in power-source voltage).
E	F	X	X	1	System-down level error
	P	r	d	2	Writing data to the flash ROM.
	E	r	G	3	Emergency stop is being actuated (except during the update mode).
	o	P	G	4	Safety gate is open (except during the update mode).
E	E	X	X	5	Cold-start level error
E	d	X	X	5	Cold-start level error
E	C	X	X	5	Operation-cancellation level error
E	b	X	X	5	Operation-cancellation level error
-		r	P	6	Waiting for a drive-source cutoff reset input (except during the update mode).
-		r	S	6	Operation is in pause (waiting for restart) (except during the update mode).
-	I	L	C	7	All servo axes are interlocked (except during the update mode).
E	A	X	X	8	Message level error
E	9	X	X	8	Message level error
r	U	d	C	9	Core update mode
	U	d	C	9	Core update is in progress.
F	U	d	c	9	Core update has completed.
r	U	d	S	9	Slave update mode
	U	d	S	9	Slave update is in progress.
F	U	d	S	9	Slave update has completed.
	P	N	0.	9	Running a program (last started program); “No.” indicates program number.
I	n	X	X	9	Initialization sequence number
	d	b	G	9	Debug mode
A	r	d	y	9	Ready status (auto mode)
	r	d	y	9	Ready status (manual mode)
	d	S	F	10	Deadman switch OFF (manual mode)

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

2-2. Core

Display				Priority (*1)	Description
	A	C	F	1	AC power is cut off (including momentary power failure or drop in power-source voltage).
E	E	x	x	1	Cold-start level error
E	d	x	x	1	Cold-start level error
E	C	x	x	1	Operation-cancellation level error
E	b	x	x	1	Operation-cancellation level error
E	A	x	x	2	Message level error
E	9	x	x	2	Message level error
r	U	d	A	2	Application update mode
	U	d	A	2	Application update is in progress.
F	U	d	A	2	Application update has completed.
P	_	_	_	2	Hardware test mode process
	E	r	A	2	Clearing the application flash ROM.
F	E	r	A	2	Application flash ROM has been cleared.
	J	P	A	2	Jump to the application
C	H	F	C	2	Core flash-ROM check process
C	H	F	A	2	Application flash-ROM check process
C	H	S	d	2	SDRAM check process

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

Chapter 5 Specifications

1. Controller Specifications

(1) Compact type (for actuators of strokes from 250 to 350)

Type	Compact
Total output when maximum number of axes are connected	450 W
Power-source voltage	Single-phase, 200 to 230 V
Operating power-source voltage range	±10%
Power-source frequency	50 Hz/60 Hz
Insulation resistance	10 MΩ min. (measured at 500 VDC between the power terminal and I/O terminals and between the external terminals (together) and case)
Withstand voltage	1500 VAC for 1 minute (Note)
Operating temperature range	0°C ~ 40°C
Operating humidity range	30% ~ 85%
Storage temperature range	-10°C ~ 65°C
Axis control method	AC full digital servo
Position detection methods	17-bit incremental encoder (wire-saving type) 17-bit rotation data backup absolute encoder (wire-saving type) (Both have a control resolution of 14 bits)
Batteries	For backup of absolute data: ER3V by Toshiba Battery For backup of system memory: CR2032 by Toshiba Battery
Speed setting	1 mm/sec ~ 2000 mm/sec
Acceleration/deceleration setting	0.01 G ~ 1 G
Programming language	Super SEL language
Program steps	6000 steps (total)
Number of positions	3000 positions (total)
Number of programs	64 programs
Multi-tasking	16 programs
Storage device	Flash ROM + SRAM battery backup
Data input methods	Teaching pendant or PC software
Standard inputs	32 points (total of dedicated inputs + general-purpose inputs)
Standard outputs	16 points (total of dedicated outputs + general-purpose outputs)
Expanded inputs/outputs	Only one board can be installed.
Serial communication	For connection of teaching pendant or PC
Other inputs/output	Emergency-stop input, safety-gate input, system ready output
Protective functions	Motor overcurrent, overload, motor-driver temperature check, overload check, encoder-open detection, soft limit over, system error, battery error
Drive-source cutoff method	Semiconductor
Regenerative resistance	Built-in (1 kΩ, 20 W); external regenerative resistance supported
Accessory	I/O flat cable
Optional parts and components	Teaching pendant, PC software, absolute-data backup battery unit, I/O shield cable, I/O expansion board (only one expansion board can be installed)

Note: The withstand voltage of the actuator motor is 1000 V for 1 minute.
When performing a withstand voltage test with the controller and actuator connected, make sure the test voltage and duration will not exceed 1000 V and 1 minute, respectively.

(2) General-purpose type (for actuators of strokes from 250 to 800)

Type	General-purpose
Total output when maximum number of axes are connected	1750 W
Power-source voltage	Single-phase, 200 to 230 V
Operating power-source voltage range	±10%
Power-source frequency	50 Hz/60 Hz
Insulation resistance	10 MΩ min. (measured at 500 VDC between the power terminal and I/O terminals and between the external terminals (together) and case)
Withstand voltage	1500 VAC for 1 minute (Note)
Operating temperature range	0°C ~ 40°C
Operating humidity range	30% ~ 85%
Storage temperature range	-10°C ~ 65°C
Axis control method	AC full digital servo
Position detection methods	17-bit incremental encoder (wire-saving type) 17-bit rotation data backup absolute encoder (wire-saving type) (Both have a control resolution of 14 bits)
Batteries	For backup of absolute data: ER3V by Toshiba Battery For backup of system memory: CR2032 by Toshiba Battery
Speed setting	1 mm/sec ~ 2000 mm/sec
Acceleration/deceleration setting	0.01 G ~ 1 G
Programming language	Super SEL language
Program steps	6000 steps (total)
Number of positions	3000 positions (total)
Number of programs	64 programs
Multi-tasking	16 programs
Storage device	Flash ROM + SRAM battery backup
Data input methods	Teaching pendant or PC software
Standard inputs	32 points (total of dedicated inputs + general-purpose inputs)
Standard outputs	16 points (total of dedicated outputs + general-purpose outputs)
Expanded inputs/outputs	48 points per unit (a maximum of 3 units can be added)
Serial communication	For connection of teaching pendant or PC
Other inputs/output	Emergency-stop input, safety-gate input, system ready output
Protective functions	Motor overcurrent, overload, motor-driver temperature check, overload check, encoder-open detection, soft limit over, system error, battery error
Drive-source cutoff method	Relay
Regenerative resistance	Built-in (220 kΩ, 80 W); external regenerative resistance supported
Accessory	I/O flat cable
Optional parts and components	Teaching pendant, PC software, absolute-data backup battery unit, I/O shield cable, I/O expansion board

Note: The withstand voltage of the actuator motor is 1000 V for 1 minute.
When performing a withstand voltage test with the controller and actuator connected, make sure the test voltage and duration will not exceed 1000 V and 1 minute, respectively.

2. External I/O Specifications

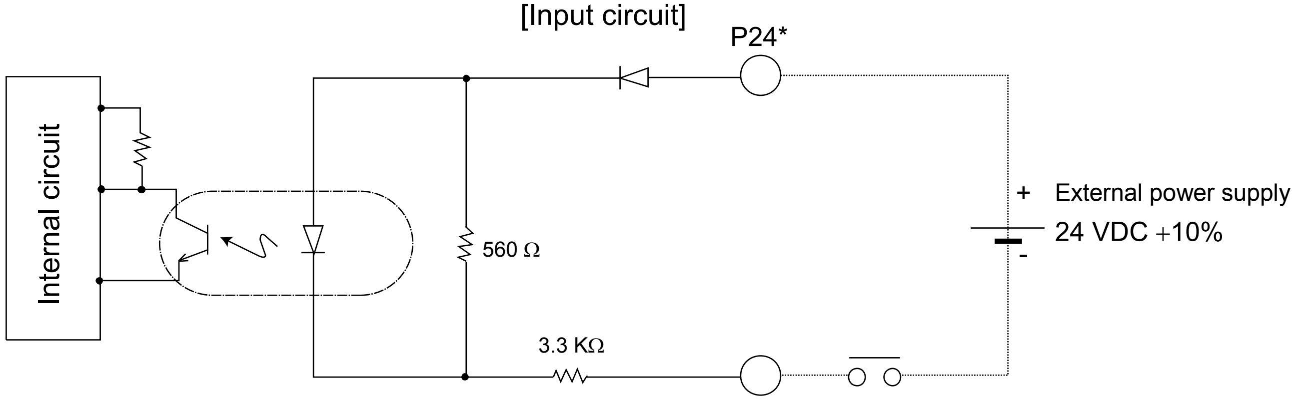
2-1. NPN Specification

(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)

[Input circuit]

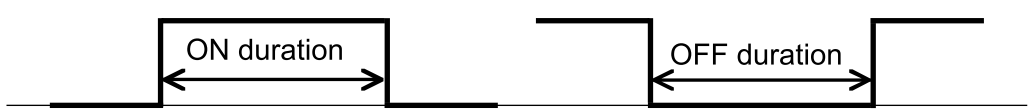


*	General purpose	Compact
P24	I/O24V connector 24V IN	I/O interface pin No. 1

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.

◎ X-SEL 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).

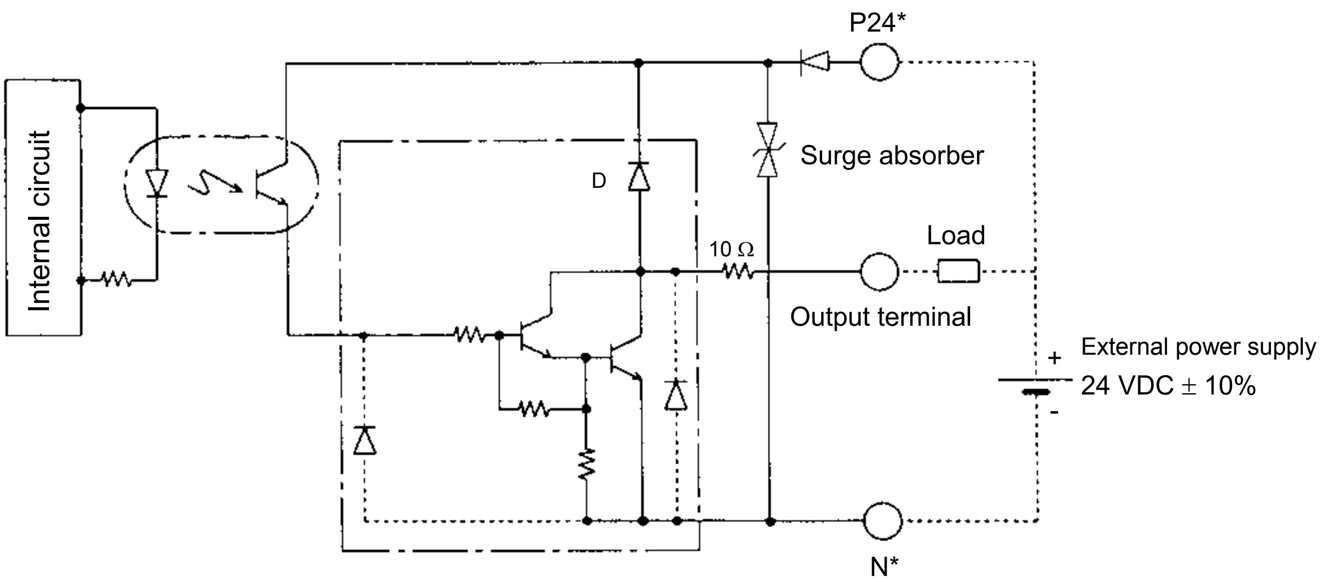
(2) Output part

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 every eight ports from output port No. 300. (The maximum total load current of output port No. 300 + n to No. 300 + n + 7 is 400 mA, where n is 0 or a multiple of 8.)

[Output circuit]



*	General purpose	Compact
P24	I/O24V connector 24V IN	I/O interface pin No. 1
N	I/O24V connector 0V	I/O interface pin No. 50

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.

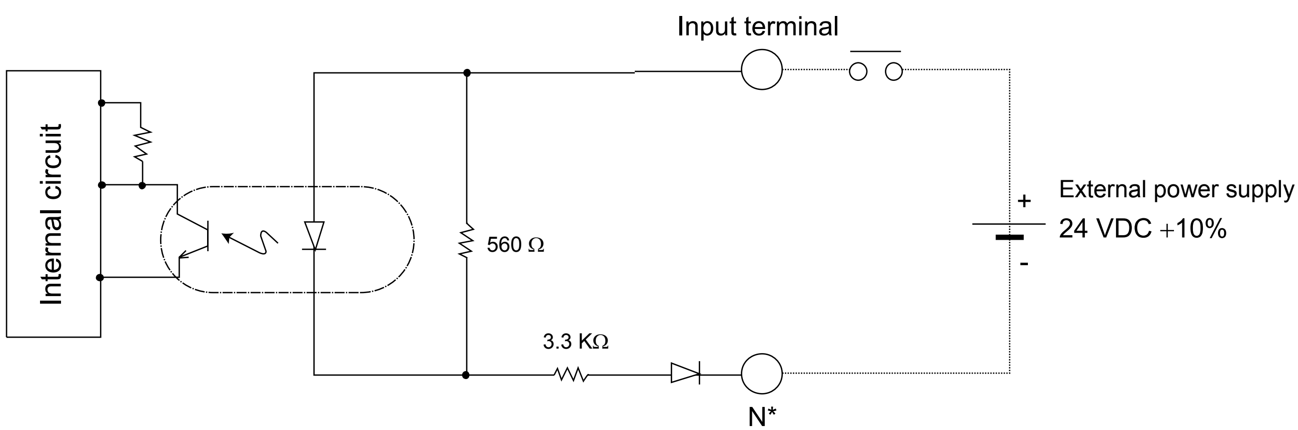
2-2. PNP Specification

(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]

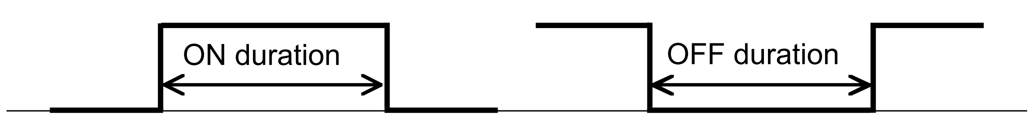


*	General purpose	Compact
N	I/O24V connector 0V	I/O interface pin No. 50

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.

◎ X-SEL 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).

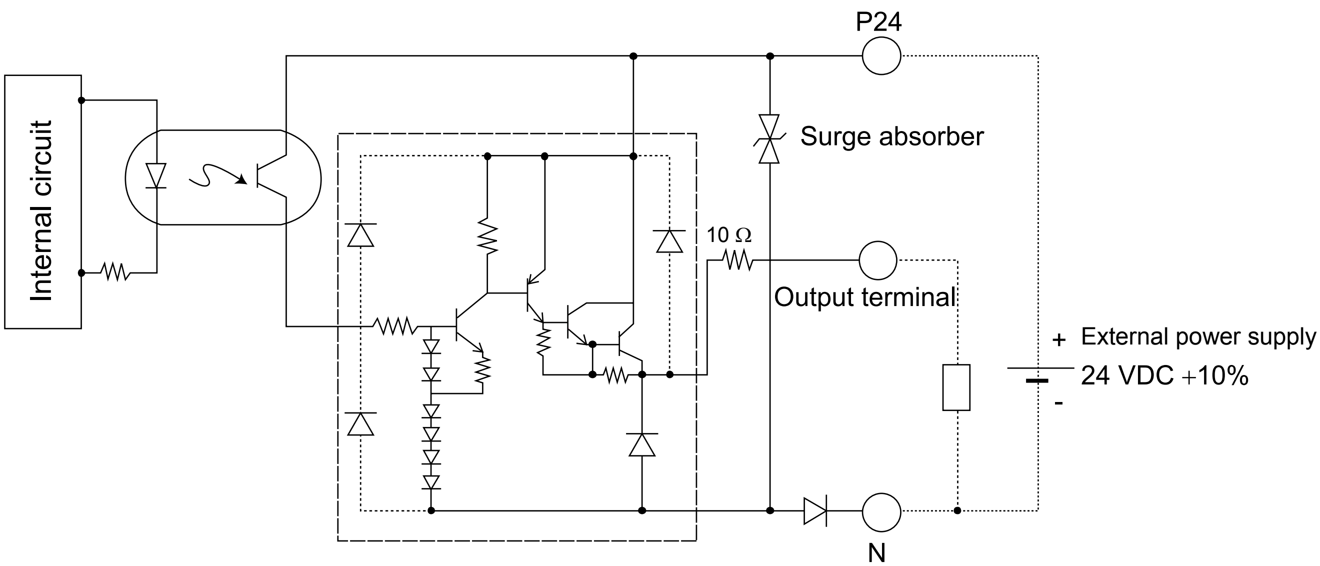
(2) Output part

External Output Specifications

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 every eight ports from output port No. 300. (The maximum total load current of output port No. 300 + n to No. 300 + n + 7 is 400 mA, where n is 0 or a multiple of 8.)

[Output circuit]



*	General purpose	Compact
P24	I/O24V connector 24V IN	I/O interface pin No. 1
N	I/O24V connector 0V	I/O interface pin No. 50

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.

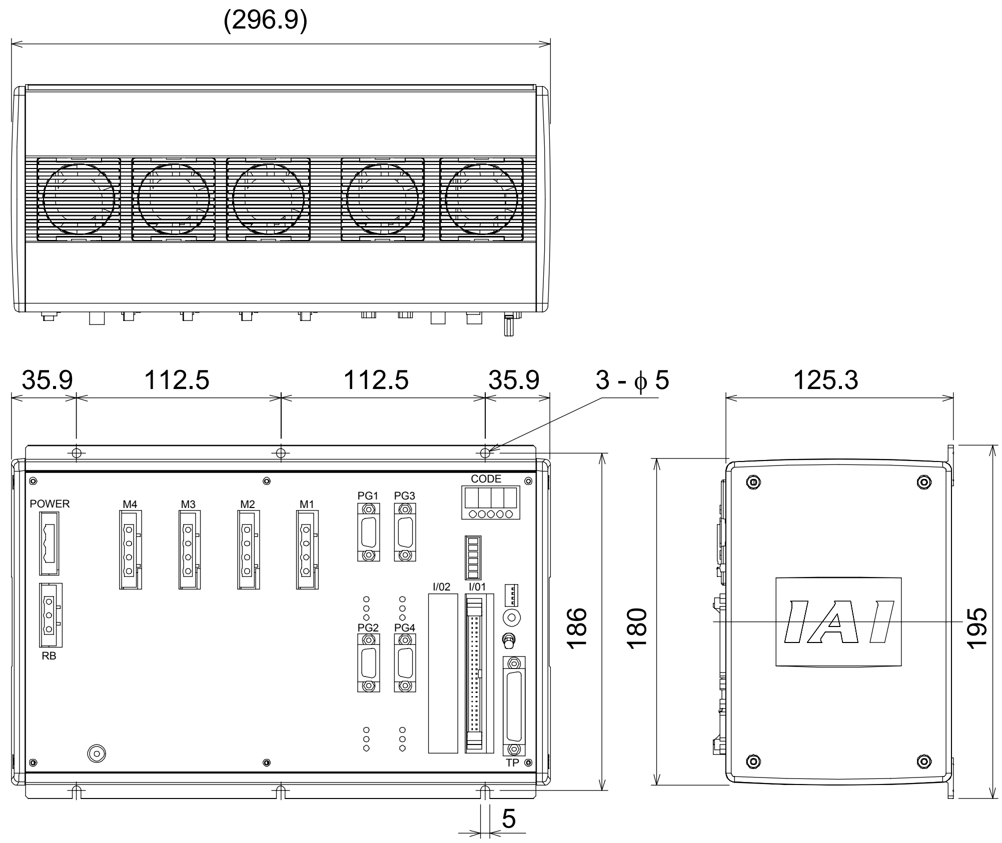
3. Power-Source Capacity and Heat Output of the Controller

The power-source capacity and heat output vary as follows, depending on the robot stroke.

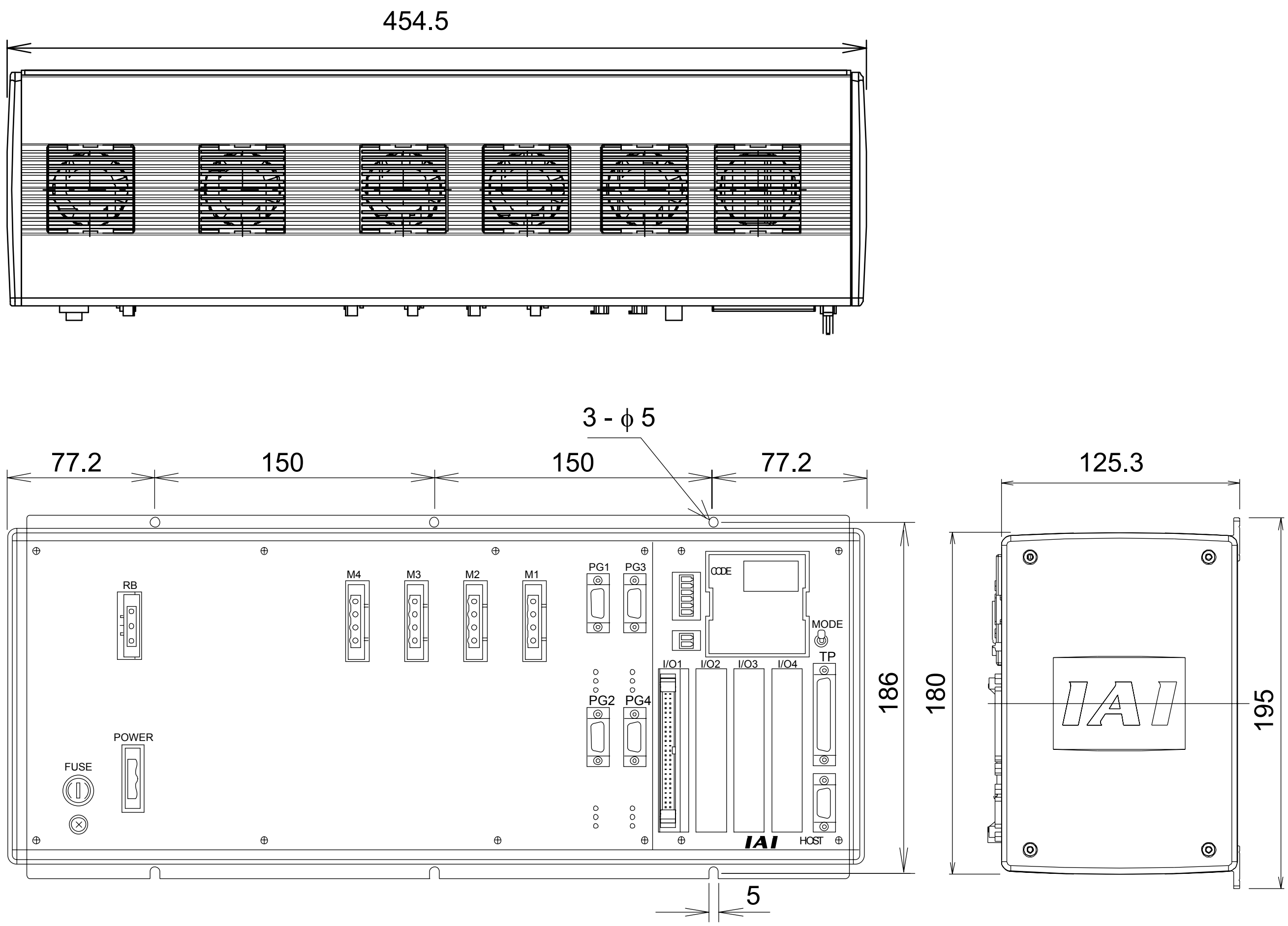
Robot stroke length	Power-source capacity	Heat output
700/800	3625 VA	133 W
500/600	1963 VA	99 W
250/350	1118 VA	81 W

4. External Dimensions

Compact type



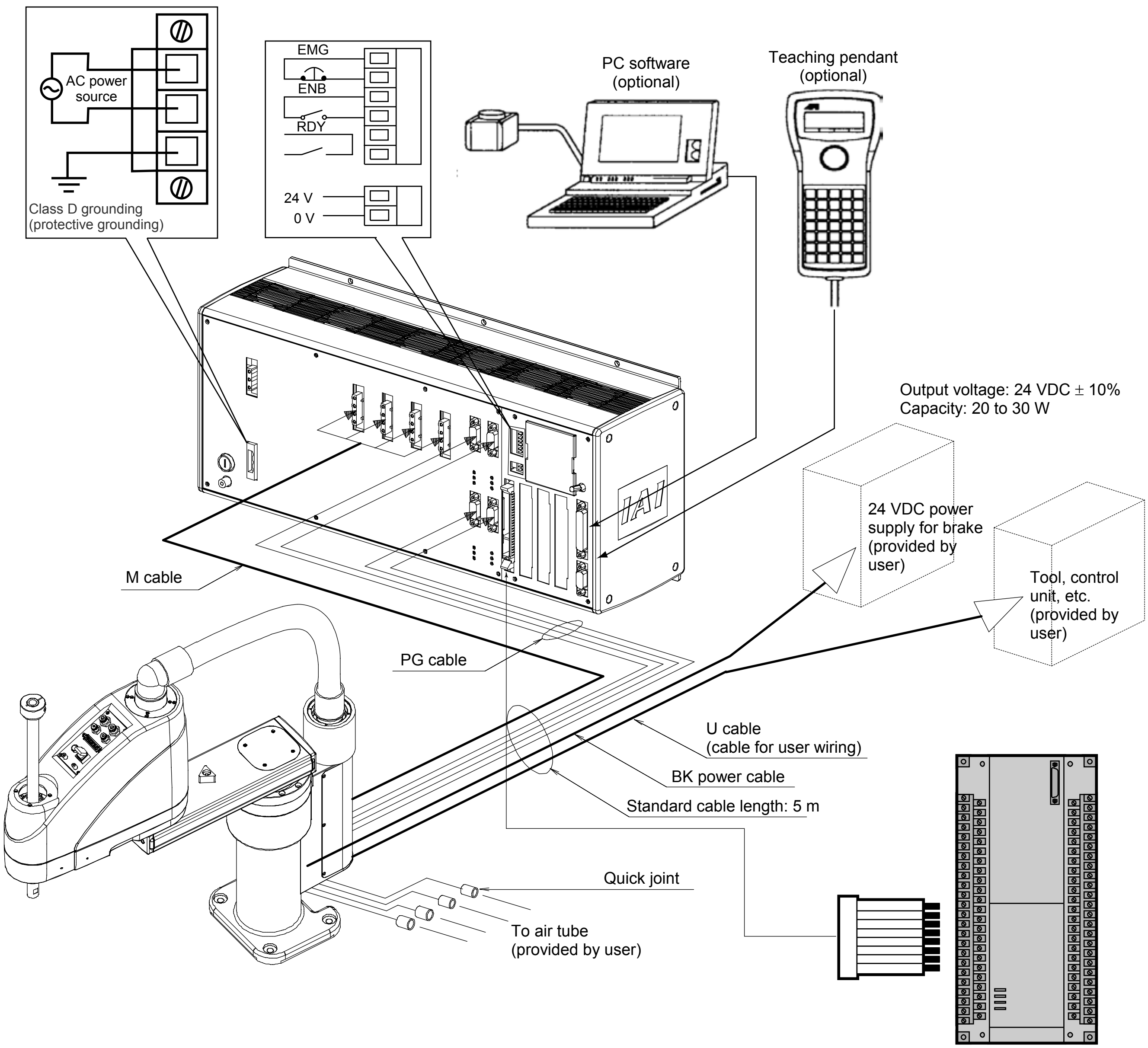
General-purpose type



Chapter 6 System Setup

A connection example is given below:

1. Connection Method of Controller and Robot (General-Purpose Type)



(1) Connect to the controller the motor cable and encoder cable from the actuator.
(2) Connect the teaching-pendant cable to the teaching-pendant connector. After the connection, set the mode switch to MANU (by tilting the switch upward).
(If the mode switch is set to AUTO, the teaching pendant will not operate and RS-232 communication will not establish after the power is turned on.)
(3) Connect the power cable to the controller.
(4) The panel window will display the code "rdy" to indicate that the preparation is complete.
If "ErG" is displayed, EMERGENCY STOP signal is being input. Reset the emergency stop.

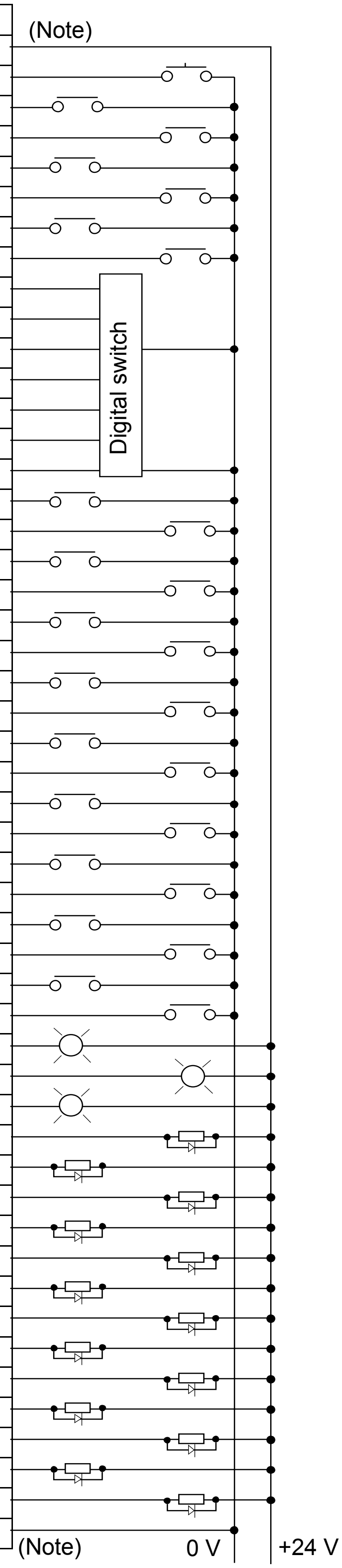
This completes the preparation.

- The RDY terminals (Nos. 1 and 2) on the system I/O connector are relay-contact terminals shorted in the ready mode.
- The ENB terminals (Nos. 3 and 4) on the system I/O connector are enable terminals. The controller can operate when these terminals are shorted. The drive source will be cut off when the terminals are open.
- The EMG terminals (Nos. 5 and 6) on the system I/O connector are emergency-stop switch connection terminals. An emergency stop will be actuated when these terminals are open.
  The controller is shipped with terminal Nos. 3 and 4, and 5 and 6, shorted, respectively.

2. I/O Connection Diagram

(1) NPN specification

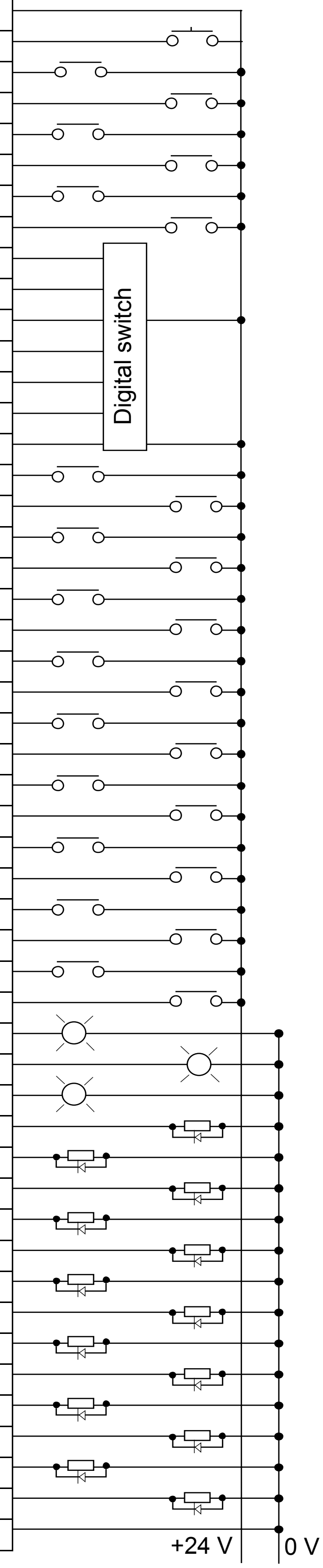
Pin No.	Category	Port No.	Function
1	Input	-	General-purpose: NC, Compact: +24-V input
2		000	Program start
3		001	General-purpose input
4		002	General-purpose input
5		003	General-purpose input
6		004	General-purpose input
7		005	General-purpose input
8		006	General-purpose input
9		007	Program specification (PRG No. 1)
10		008	Program specification (PRG No. 2)
11		009	Program specification (PRG No. 4)
12		010	Program specification (PRG No. 8)
13		011	Program specification (PRG No. 10)
14		012	Program specification (PRG No. 20)
15		013	Program specification (PRG No. 40)
16		014	General-purpose input
17		015	General-purpose input
18		016	General-purpose input
19		017	General-purpose input
20		018	General-purpose input
21		019	General-purpose input
22		020	General-purpose input
23		021	General-purpose input
24		022	General-purpose input
25		023	General-purpose input
26		024	General-purpose input
27		025	General-purpose input
28		026	General-purpose input
29		027	General-purpose input
30		028	General-purpose input
31		029	General-purpose input
32		030	General-purpose input
33		031	General-purpose input
34	Output	300	Alarm output
35		301	Ready output
36		302	Emergency-stop output
37		303	General-purpose output
38		304	General-purpose output
39		305	General-purpose output
40		306	General-purpose output
41		307	General-purpose output
42		308	General-purpose output
43		309	General-purpose output
44		310	General-purpose output
45		311	General-purpose output
46		312	General-purpose output
47		313	General-purpose output
48		314	General-purpose output
49		315	General-purpose output
50		-	General-purpose: NC, Compact: 0 V



With a general-purpose (K) type, pin Nos. 1 and 50 are not connected.
With a compact (J) type, +24 V is connected to pin No. 1, while 0 V is connected to pin No. 50.

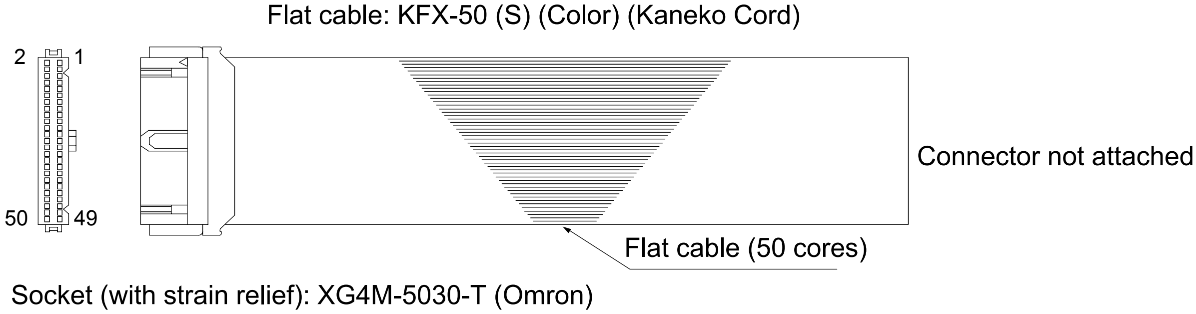
(2) PNP specification

Pin No.	Category	Port No.	Function
1	Input	-	General-purpose: NC, Compact: +24-V input
2		000	Program start
3		001	General-purpose input
4		002	General-purpose input
5		003	General-purpose input
6		004	General-purpose input
7		005	General-purpose input
8		006	General-purpose input
9		007	Program specification (PRG No. 1)
10		008	Program specification (PRG No. 2)
11		009	Program specification (PRG No. 4)
12		010	Program specification (PRG No. 8)
13		011	Program specification (PRG No. 10)
14		012	Program specification (PRG No. 20)
15		013	Program specification (PRG No. 40)
16		014	General-purpose input
17		015	General-purpose input
18		016	General-purpose input
19		017	General-purpose input
20		018	General-purpose input
21		019	General-purpose input
22		020	General-purpose input
23		021	General-purpose input
24		022	General-purpose input
25		023	General-purpose input
26		024	General-purpose input
27		025	General-purpose input
28		026	General-purpose input
29		027	General-purpose input
30		028	General-purpose input
31		029	General-purpose input
32		030	General-purpose input
33		031	General-purpose input
34	Output	300	Alarm output
35		301	Ready output
36		302	Emergency-stop output
37		303	General-purpose output
38		304	General-purpose output
39		305	General-purpose output
40		306	General-purpose output
41		307	General-purpose output
42		308	General-purpose output
43		309	General-purpose output
44		310	General-purpose output
45		311	General-purpose output
46		312	General-purpose output
47		313	General-purpose output
48		314	General-purpose output
49		315	General-purpose output
50		-	General-purpose: NC, Compact: 0 V



With a general-purpose (K) type, pin Nos. 1 and 50 are not connected.
With a compact (J) type, +24 V is connected to pin No. 1, while 0 V is connected to pin No. 50.

3. I/O Flat Cable



No.	Color	No.	Color	No.	Color	No.	Color	No.	Color
1	Brown-1	11	Brown-2	21	Brown-3	31	Brown-4	41	Brown-5
2	Red-1	12	Red-2	22	Red-3	32	Red-4	42	Red-5
3	Orange-1	13	Orange-2	23	Orange-3	33	Orange-4	43	Orange-5
4	Yellow-1	14	Yellow-2	24	Yellow-3	34	Yellow-4	44	Yellow-5
5	Green-1	15	Green-2	25	Green-3	35	Green-4	45	Green-5
6	Blue-1	16	Blue-2	26	Blue-3	36	Blue-4	46	Blue-5
7	Purple-1	17	Purple-2	27	Purple-3	37	Purple-4	47	Purple-5
8	Gray-1	18	Gray-2	28	Gray-3	38	Gray-4	48	Gray-5
9	White-1	19	White-2	29	White-3	39	White-4	49	White-5
10	Black-1	20	Black-2	30	Black-3	40	Black-4	50	Black-5

Chapter 7 Maintenance

- Routine maintenance and inspection are necessary so that the system will operate properly at all time. 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 --- Battery life: Approx. 1 year*
- Absolute-data backup battery --- Battery life: Approx. 3 years*
  (Absolute-data backup battery is installed on the robot.)
- Fuses

* The actual battery life will vary depending on the use condition.

Memory backup

The X-SEL Controller saves program, position, coordinate system and parameter data to its flash memory (when written to the flash memory). The data saved by the system-memory backup battery are positions, coordinate system data, SEL global data and error lists. (Refer to Chapter 1, “How to Save Data,” of Part 3.)

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
Absolute-data backup battery	A03 or A23

In the case of a low battery voltage of the absolute-data backup battery, the axis-driver status LED will also illuminate.

(3) Replacement procedure for system-memory backup battery (general-purpose type)

Backing up the system memory
If “Other parameter No. 20, System-memory backup battery installation function type” is set to “2” (installed), the following SRAM data in the X-SEL Controller will be backed up by the system-memory backup battery on the panel board:

- Position data
- Coordinate system data
- SEL global data (flags, integer/real variables, string variables)
- Error lists

Therefore, the above SRAM data will be destroyed if the system-memory backup battery is removed when “Other parameter No. 20, System-memory backup battery installation function type” is set to “2” (installed). For this reason, always follow the procedure below when replacing the system-memory backup battery:

1. Turn on the controller power.

2. Record (write down) the current setting of “Other parameter No. 20, System-memory 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, System-memory 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 (restart) to restart the controller. (The SEL global data and error lists will be saved to the special area in the flash ROM.)

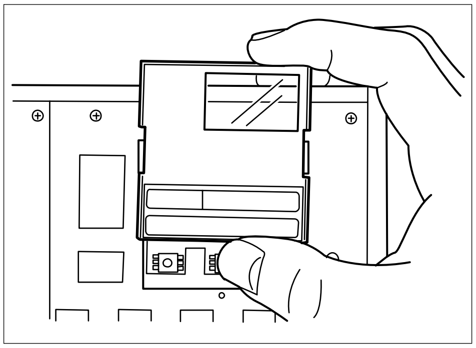
6. When the controller has been restarted, turn off the power.

   * Once the controller has been restarted, be sure to keep the power on until the initialization sequence number is no longer displayed on the panel window (while “InXX” is displayed following “8888”; XX indicates a number).

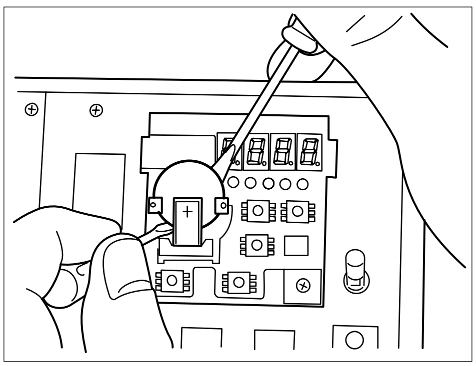
7. Replace the system-memory backup battery. The battery is mounted on the panel board and can be removed using two small screwdrivers or the like. When removing the current battery and installing a new one, pay due attention to the following points (SRAM data will be destroyed if steps 1 through 6 are not performed properly):

   - Do not damage the board patterns.
   - Do not short the circuits.
   - Connect the battery in the correct polarities.

Battery Replacement Procedure



1) Remove the 7-segment LED panel from the controller. Slide the panel upward and pull it toward you to remove.



2) As shown at left, remove the battery from the holder using two screwdrivers, and then install a new battery in the holder.

3) Install the panel in the original position.

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, System-memory 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).

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

(4) Replacement procedure for system-memory backup battery (compact type)

[Absolute specification]
In the case of a compact controller of absolute specification, an absolute reset must be performed after replacing the system-memory backup battery. (Since all cables and absolute-data backup battery are disconnected from the controller during the replacement procedure, the absolute data will be destroyed.)

[Memory backup]
If "Other parameter No. 20, System-memory backup battery installation function type" is set to "2" (installed), the following SRAM data in the X-SEL Controller will be backed up by the system-memory backup battery on the main CPU board:

- Position data
- Coordinate system data
- SEL global data (flags, integer/real variables, string variables)
- Error lists

Therefore, the above SRAM data will be destroyed if the system-memory backup battery is removed when "Other parameter No. 20, System-memory backup battery installation function type" is set to "2" (installed). For this reason, always follow the procedure below when replacing the system-memory backup battery:

[Replacement procedure for system-memory backup battery]

1. Turn on the controller power.

2. Record (write down) the current setting of "Other parameter No. 20, System-memory 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, System-memory 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.

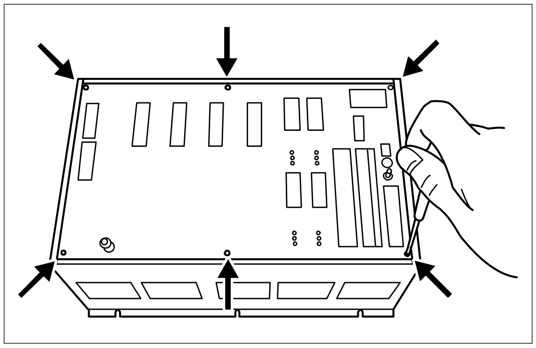
5. Perform a software reset to restart the controller. (The SEL global data and error lists will be saved to the special area in the flash ROM.)

6. When the controller has been restarted, turn off the power.
   * Once the controller has been restarted, be sure to keep the power on until the initialization sequence number is no longer displayed on the panel window (while "InXX" is displayed following "8888"; XX indicates a number).

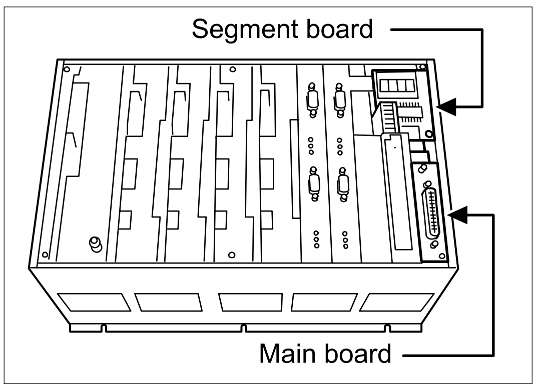
7. Replace the system-memory backup battery. When removing the current battery and installing a new one, pay due attention to the following points (SRAM data will be destroyed if steps 1 through 6 are not performed properly):

   - Do not damage the board patterns.
   - Do not short the circuits.
   - Connect the battery in the correct polarities.

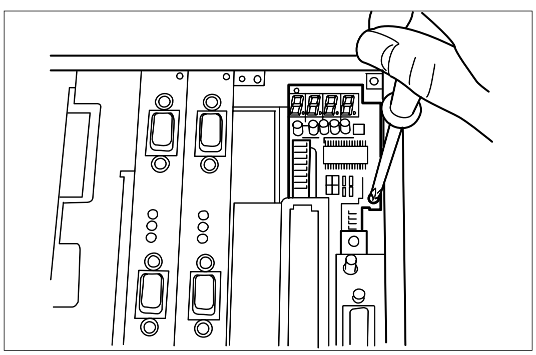
[Battery replacement method]
1) Disconnect all cables from the controller.



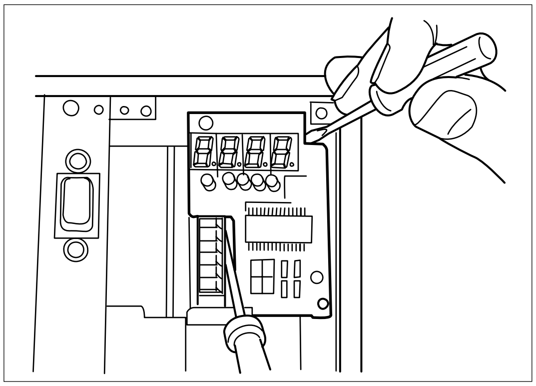
2) Remove the screws shown by arrows that are mounting the front panel.



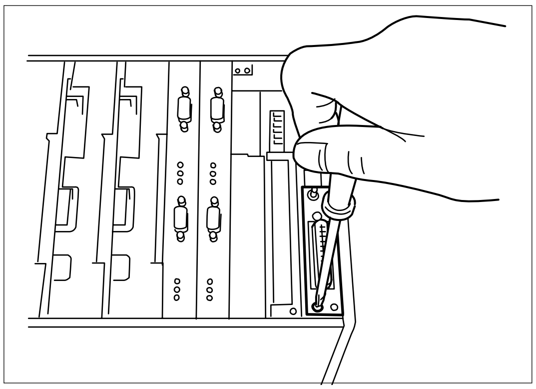
3) Remove the front panel.



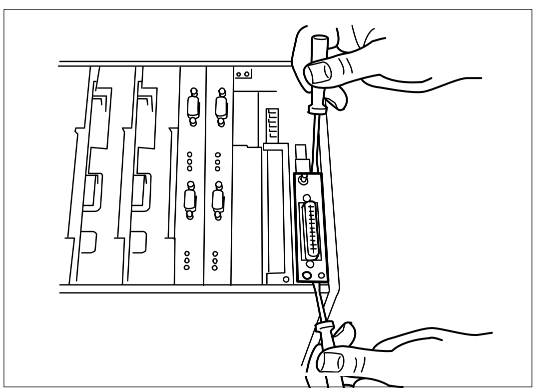
4) Remove the screws mounting the segment board.



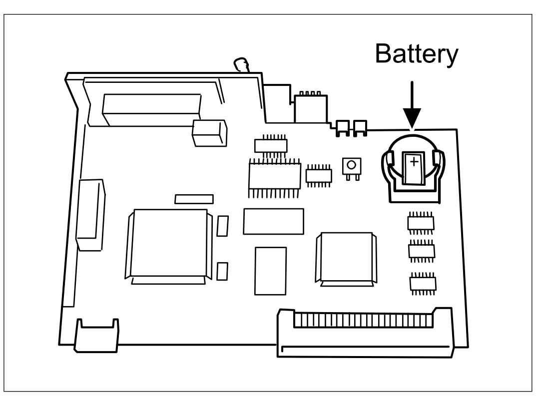
5) As shown at left, use small screwdrivers to remove the segment board.



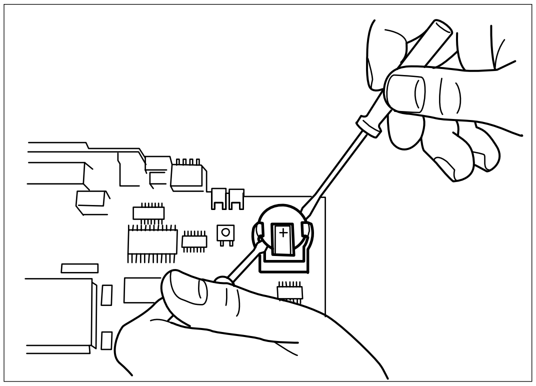
6) Remove the screws mounting the main CPU board.



7) As shown at left, use small screwdrivers to remove the main CPU board. (The main CPU board is sandwiched between the enclosure guides.)



8) Replace the battery located in the position shown by an arrow in the figure at left.



As shown at left, use small screwdrivers to remove the battery from the holder and then install a new battery in the holder.

8. When the replacement of system-memory backup battery is complete, confirm that the battery is installed securely. (Pay attention to the polarities of the battery.)

9. Install the main CPU board, segment board and front panel in the reverse steps.

10. Connect the controller and the actuator using the cables. Connect the power cable and turn on the controller power.

11. Revert "Other parameter No. 20, System-memory 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.

12. Perform a software reset and restart the controller.

13. When the controller has been restarted, confirm that the SRAM data have been restored.

Part 2 Operation

Chapter 1 Operation

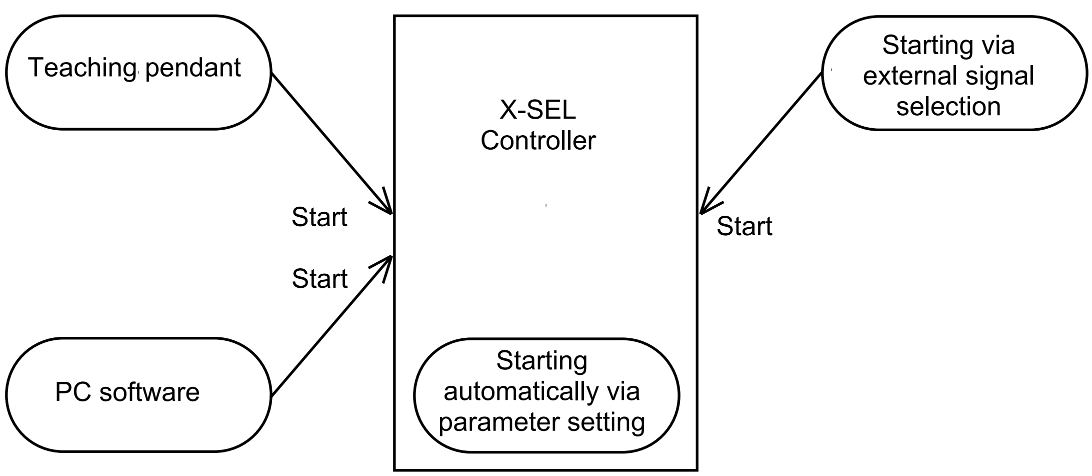
How to Start a Program

With the X-SEL 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.”



1. Starting a Program by Auto-Start via Parameter Setting

I/O parameter No. 33 (input function selection 003) = 1 (default 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, 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.

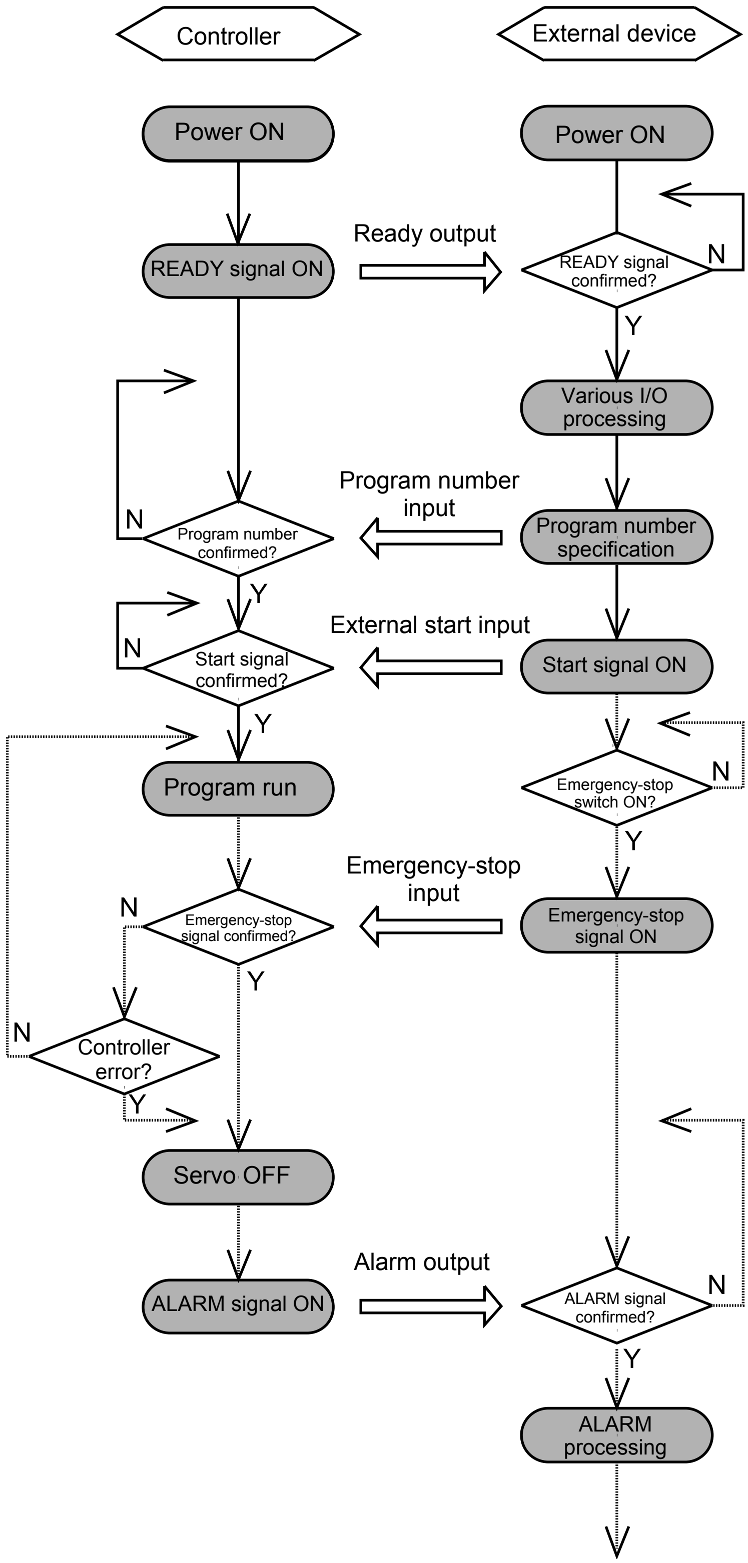
If you wish to start multiple programs at the same time, write multiple "EXPG" commands at the beginning of the main program to start the remaining programs. Provide safety measures for each program to be started.

* When I/O parameter No. 33 is set to "2"
The program of the selected number will start automatically at the ON edge of input signal received by input port No. 3.
The program will be terminated at the OFF edge.

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 turns ON, the RDY lamp (green) on the controller front panel will illuminate.

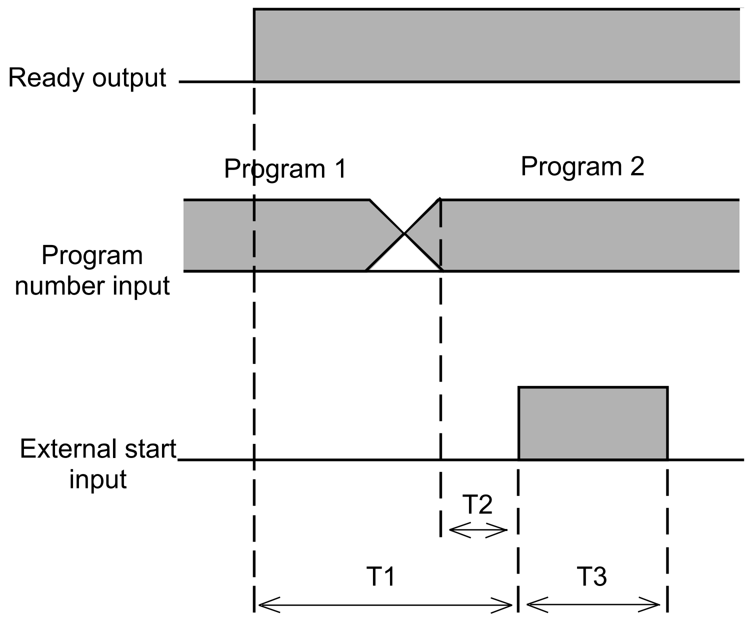
Input a desired program number as a BCD code from the external device.

Input a start signal from the external device.

When the program is run, the number of the started program will be shown in the CODE display area of the controller front panel.

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.)

(2) Timing chart



T1: Duration after the ready output turns ON until input of external start signal is permitted

T1 = 3 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.

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

(1) Drive-source recovery request

1. How to request a drive-source recovery

   A drive-source recovery request can be issued using one of the following methods:
   - Set I/O parameter No. 44 to "1" (Input selection function 014 = Drive-source cutoff reset input), then input the ON edge to input port No. 14.
   - 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. Case where a drive-source request is required

   A drive-source recovery request is required in the following case:
   - A drive-source cutoff factor occurred when I/O parameter No. 44 was set to "1" → Recovery after the cutoff factor is removed.

(2) Operation-pause reset request

1. How to request an operation-pause reset

   An operation-pause reset request can be issued using one of the following methods:
   - Set I/O parameter No. 35 to "1" (Input selection function 005 = Operation-pause reset signal), then input the ON edge to input port No. 5.
   - 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.

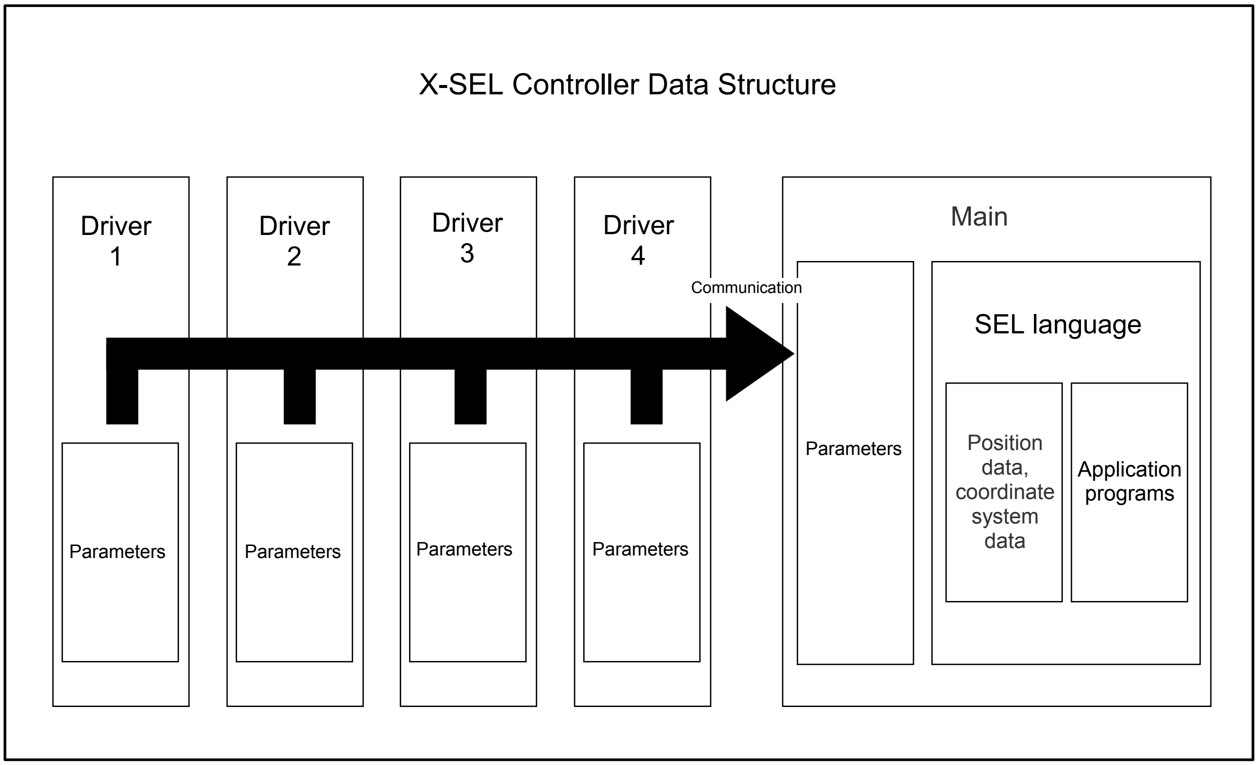
2. Cases where an operation-pause reset request is required

   An operation-pause reset request is required in any of the following cases:
   - The automatic operation was stopped using the deadman switch when other parameter No. 9 was set to "2" (Deadman-switch recovery type = Continued operation) (only during automatic operation) → Recovery (reset of operation pause) after the stop is reset.
   - 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 safety gate was opened during automatic operation when other parameter No. 11 was set to "2" (Safety-gate open recovery type = Continued operation) (only during automatic operation) → Recovery (reset of operation pause) after the safety gate is closed.
   - An OFF-level input signal was received by input port No. 6 when I/O parameter No. 36 was set to"1" (Input selection function 006 = Operation-pause signal) → Recovery (reset of operation pause) after an ON-level input signal is received by input port No. 6.

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

Part 3 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.

Chapter 1 How to Save Data

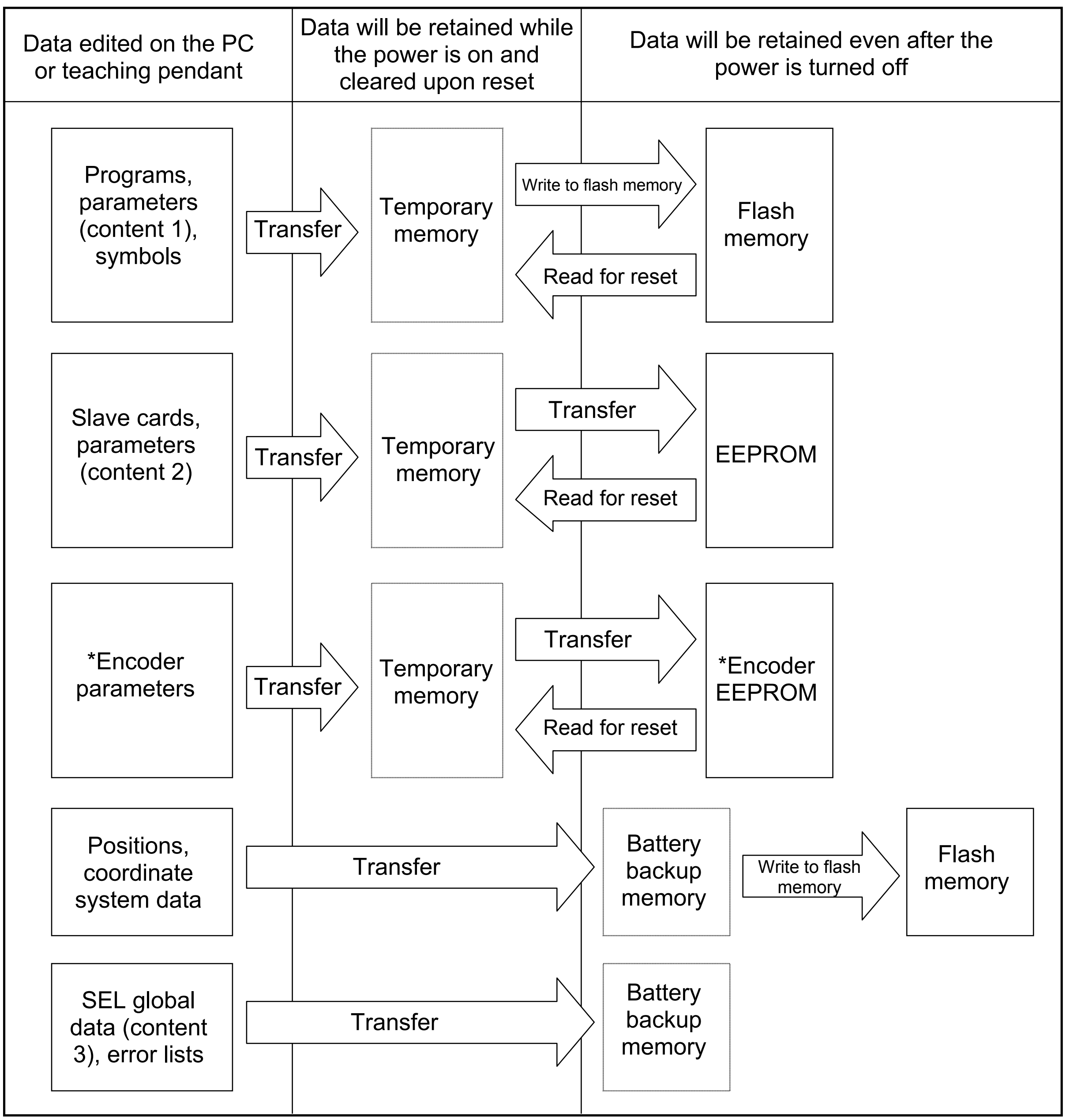
Since the IX-Series Controller uses flash memory, some data are saved by battery backup while others are saved in the flash memory.

When data is transferred from the PC software or teaching pendant to the controller, the data is only written to the temporary memory 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.

1. Factory Settings: When the System-Memory Backup Battery is Used

(Other parameter No. 20 = 2 (System-memory backup battery installed))



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

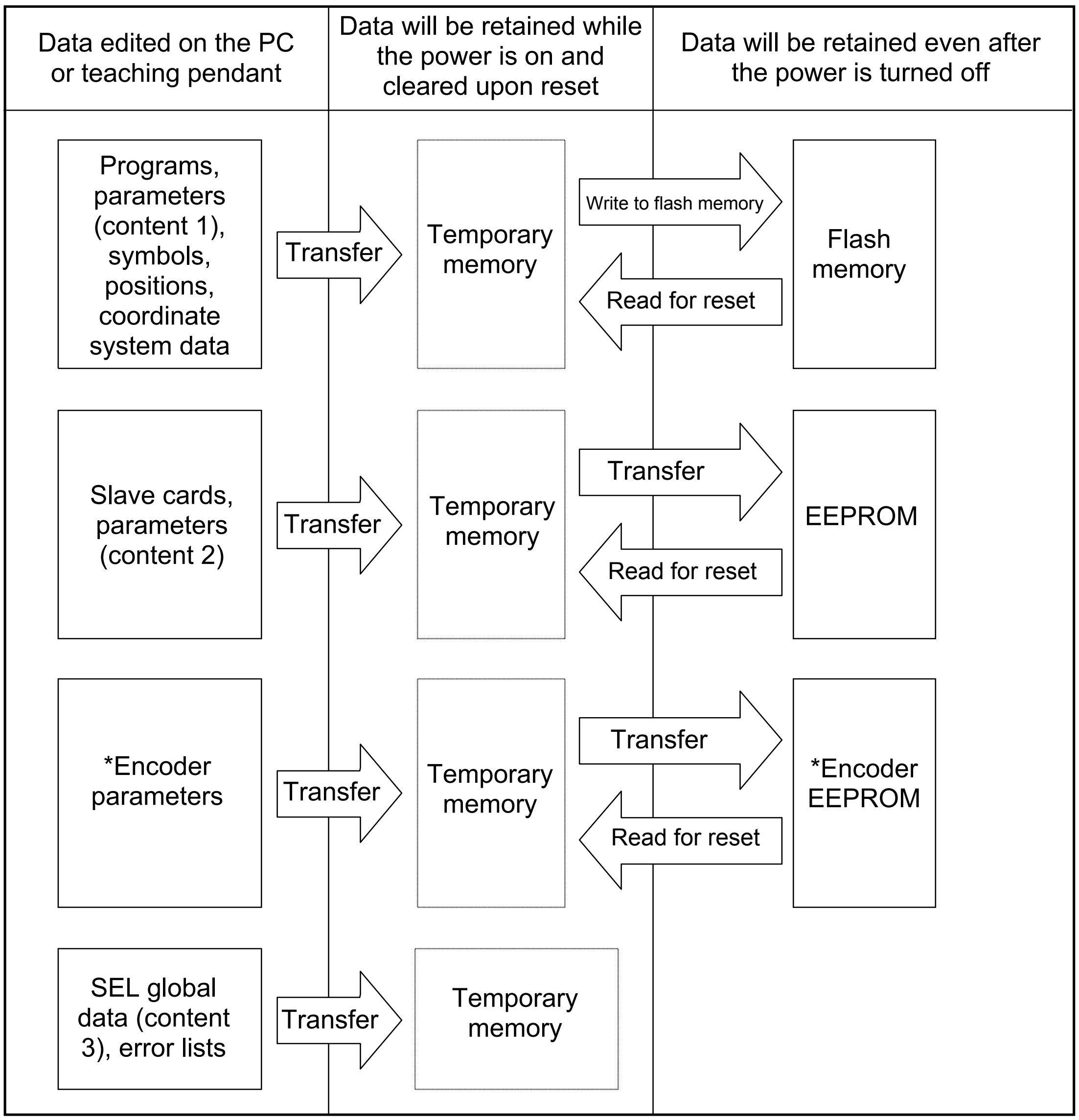
Since the programs, parameters and symbols are read from the flash memory at restart, the data in the temporary memory will remain the same as the original data before edit unless the edited data are written to the flash memory.

The controller always operates in accordance with the data in the temporary memory (inside the dotted line) (excluding the parameters).

Content 1: Parameters other than content 2 and encoder parameters
Content 2: Parameters of driver card, I/O slot card (power system card)
Content 3: Flags, variables, strings

2. When the System-Memory Backup Battery is Not Used

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



Since the programs, parameters, symbols and positions 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 temporary memory (inside the dotted line) (excluding the parameters).

Note: SEL global data cannot be retained if the backup battery is not installed.

3. 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 2 X-SEL Language Data (IX-Series Controller)

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 (99)	199 is used for PPUT, PGET, PARG, etc.
String	300 ~ 999 (700)	1 ~ 299 (299)	
Tag number		1 ~ 99 (99)	
Subroutine number		1 ~ 99 (99)	
Work coordinate system	0 ~ 31 (32)		
Tool coordinate system	0 ~ 127 (128)		
Simple interference check zone number	1 ~ 10 (10)		
Pallet number		1 ~ 10 (10)	
Axis number	1 ~ 4 (4)		Varies depending on the function.
Axis pattern	0 ~ 1111		
Position number	1 ~ 3000 (3000)		
Program number	1 ~ 64 (64)		
Step number	1 ~ 6000 (6000)		
Task level	NORMAL/HIGH (2)		
SIO channel number	1 ~ 1 (1) (Also used for TP/PC)		
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	1000		
Number of times symbol can be used in commands	5000 (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.

- The variables and flags in the global range will be retained even after the controller power is turned off. (When other parameter No. 20 is set to “2.” Refer to Chapter 1, “How to Save Data,” of Part 3.)

- 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 X-SEL 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 031 (standard)

(2) Output ports
Used as various output ports.

Output number assignment
300 to 315 (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 extension = Use strictly prohibited)
7005	(For future extension = 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 extension = 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 extension = 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 (restart))
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 (restart))
7019	Voltage low warning for axis-3 absolute-data backup battery
7020	Abnormal voltage of axis-3 absolute-data backup battery (latched until power-on reset or software reset (restart))
7021	Voltage low warning for axis-4 absolute-data backup battery
7022	Abnormal voltage of axis-4 absolute-data backup battery (latched until power-on reset or software reset (restart))
7023 ~ 7030	(For future extension = Use strictly prohibited)
7031	Reading SIO CH1 (standard SIO) (reception ready) (*OFF if used for PC/TP connection)
7032	Reading SIO CH2 (expanded SIO)(reception ready)
7033	Reading SIO CH3 (expanded SIO)(reception ready)
7034	Reading SIO CH4 (expanded SIO)(reception ready)
7035	Reading SIO CH5 (expanded SIO)(reception ready)
7036	Reading SIO CH6 (expanded SIO)(reception ready)
7037	Reading SIO CH7 (expanded SIO)(reception ready)
7038 ~ 7100	(For future extension = Use strictly prohibited)
7101	Running program No. 01 (including during pause)
~	~
7164	Running program No. 64 (including during pause)
7165 ~ 7299	(For future extension = 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 extension = Use strictly prohibited)
7381 ~ 7399	(For future extension = Use strictly prohibited)
7400 ~ 7599	(For future extension = Use strictly prohibited)

1-4 Flags

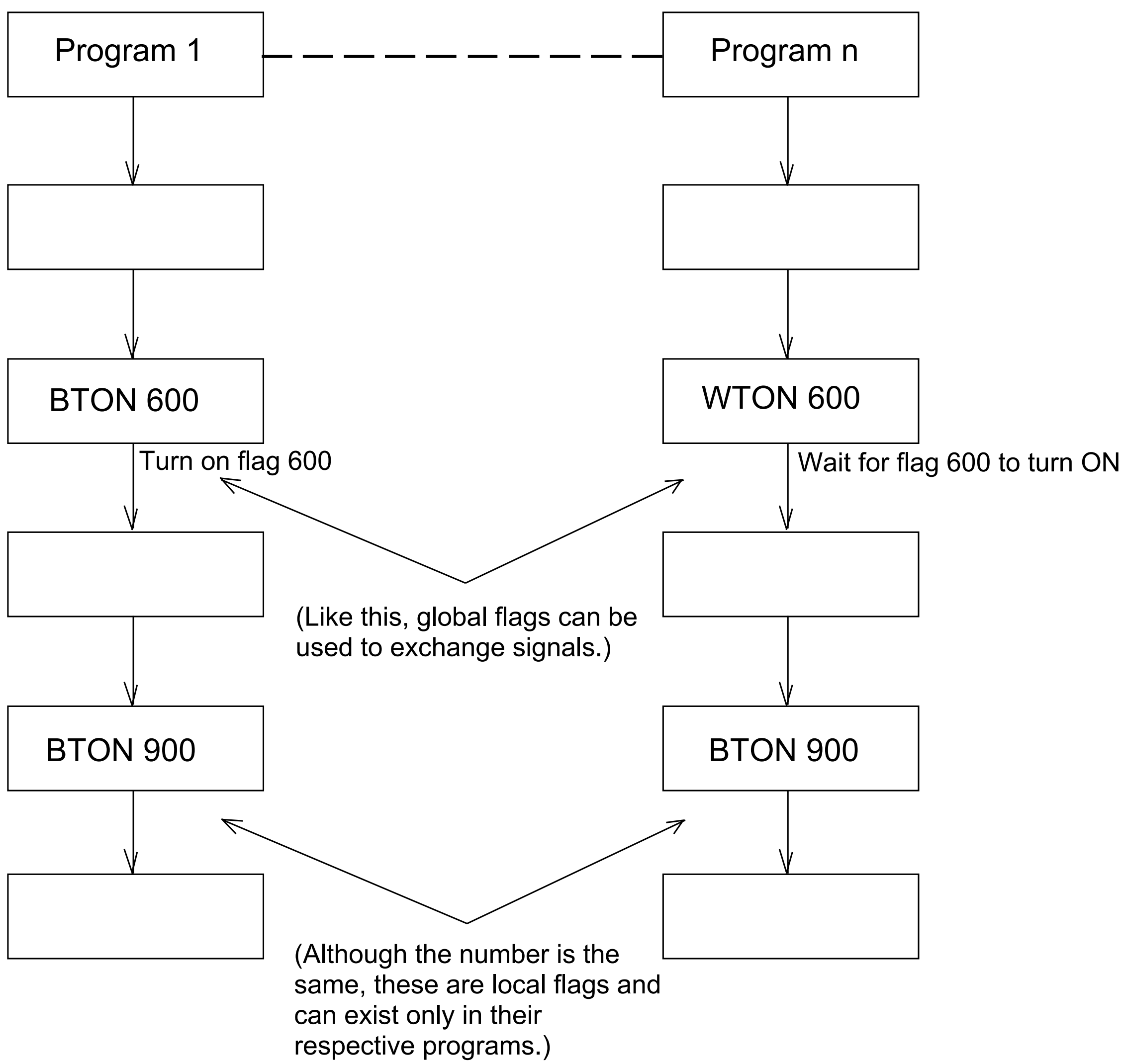
Contrary to its common meaning, the term “flag” as used in programming means “memory.” Flags are used to set or reset data. They correspond to “auxiliary relays” in a sequencer.

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.

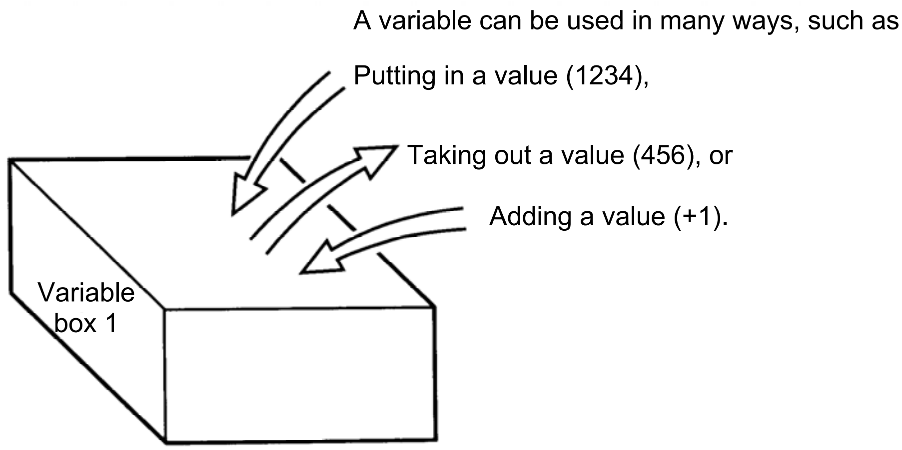
Flag number	600 ~ 899	Can be used in all programs “Global flags”
Flag number	900 ~ 999	Used only in each program “Local flags”



1-5 Variables

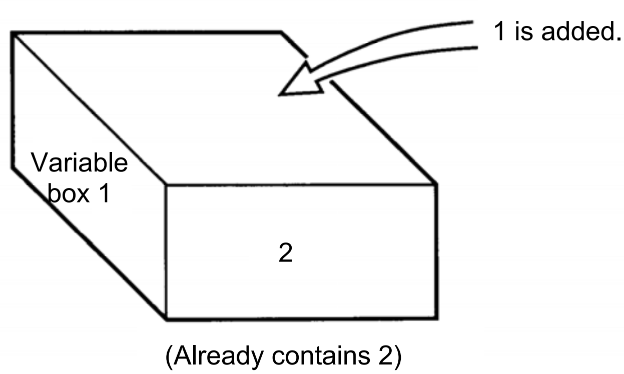
(1) Meaning of variable

"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.



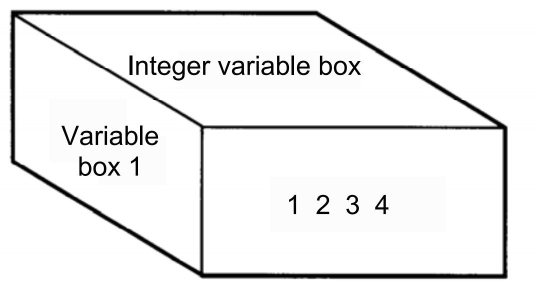
Command	Operand 1	Operand 2
ADD	1	1

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.



(2) Types of variables
Variables are classified into two types, as follows:

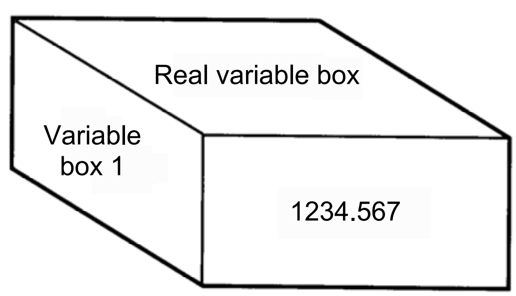
1. Integer variables
These variables cannot handle decimal places.
[Example] 1234



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)

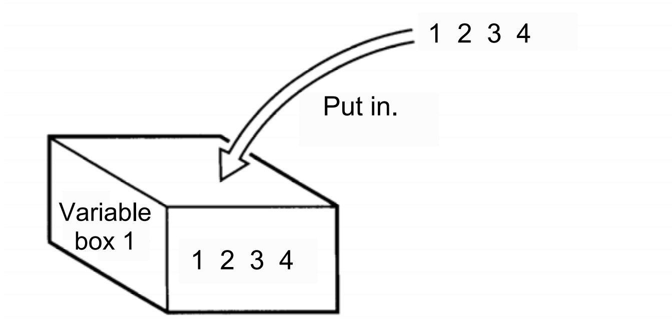


Real variable number	300 ~ 399 1300 ~ 1399	Can be used in all programs	"Global real variables"
Real variable number	100 ~ 199 1100 ~ 1199	Used only in each program	"Local real variables"

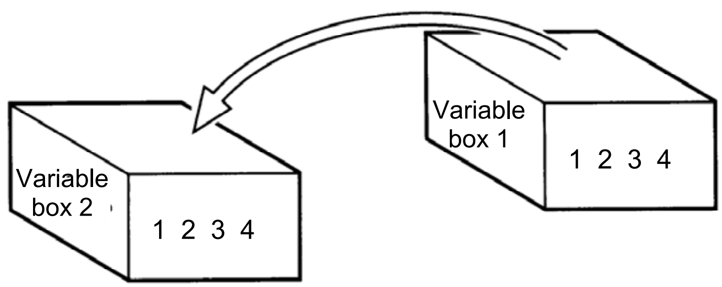
Caution Real number 199 is a special register this system uses in real-number operations. Any value in the range from –99,999.9 to 999,999.9 (eight digits including a sign) can be input in programs.

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



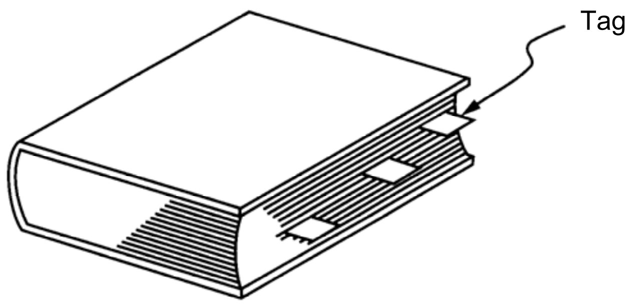
The above use of variables is called "indirect specification."

1-6 Tags

The term “tag” means “heading.”

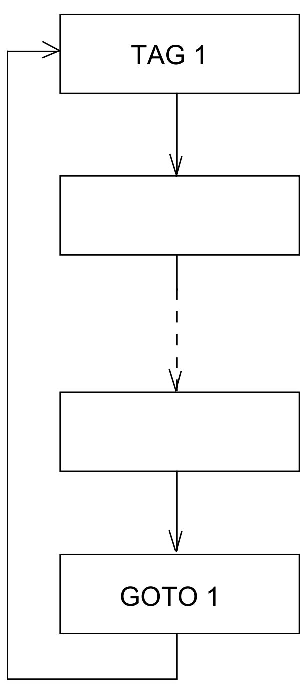
Tags are used in the same way you attach labels to the pages in a book you want to reference frequently.

A tag is a destination specified in a jump command “GOTO.”



Command	Operand 1
TAG	Tag number (Integer between 1 and 99)

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)

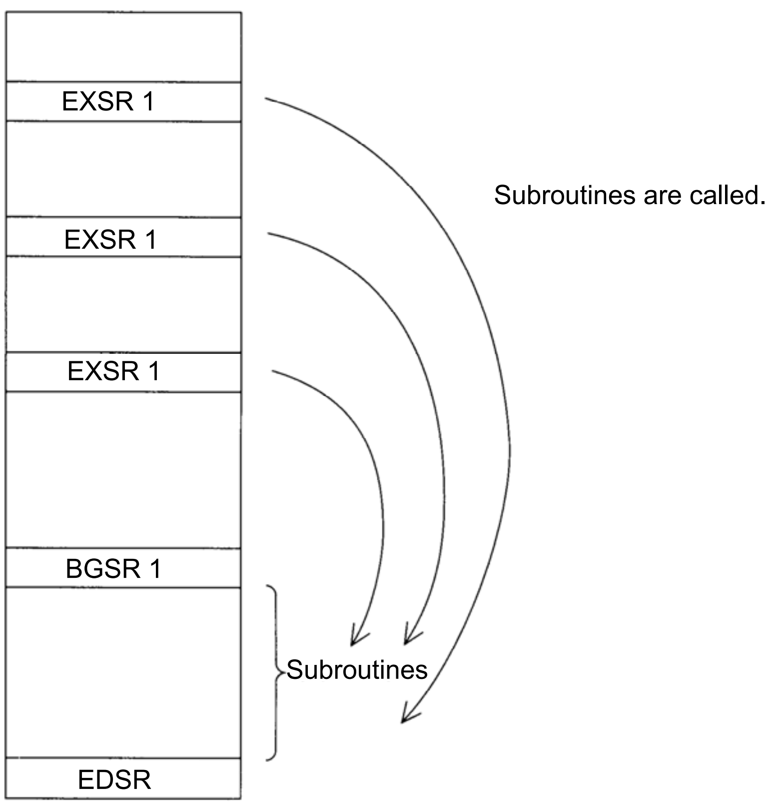
Subroutine execution command

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 X-SEL Controller, values such as variable numbers and flag numbers can be handled as symbols.

(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 1000

(4) Number of times symbol can be used in commands: Maximum 5000 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	Axis number
X-axis	1
Y-axis	2
Z-axis	3
R-axis	4

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
PPUT, PGET, ACHZ, AXST, PASE, PCHZ, ACHZ, PARG

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

	(Upper)			(Lower)
Axis	R-axis	Z-axis	Y-axis	X-axis
Used	1	1	1	1
Not used	0	0	0	0

[Example] When the X and Y-axes are used

Y-axis

0011 --- (The two 0s in front are not necessary. With the 0s removed, the expression reads “11.”)

X-axis

[Example] When the X and R-axes are used

R-axis

1001 --- (In this case, the 0s are needed to indicate the position of axis 4.)

X-axis

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.

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, STOP, PTST, PRED, PBND

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

2. Position Part

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

	X-axis	Y-axis	Z-axis	R-axis	*	*	*
No.	Axis 1	Axis 2	Axis 3	Axis 4	Vel	Acc	Dcl
1							
2							
3							
:							
:							
:							
2998							
2999							
3000							

* Varies depending on the actuator model.
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.

Values pertaining to the R-axis (rotating axis) are processed in degrees instead of millimeters.

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

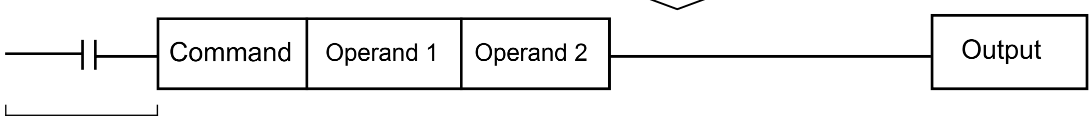
The primary feature of SEL language is its very simple command structure. Since the structure is simple, there is no need for a compiler (to translate into computer language) and high-speed operation is possible via an interpreter (the program runs as commands are translated).

3-1 SEL language Structure

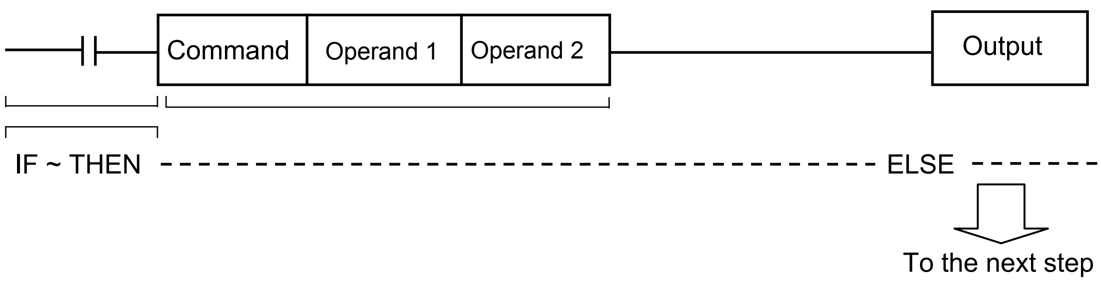
The table below shows the structure of one command step.

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

Using a ladder diagram, this is expressed as follows:

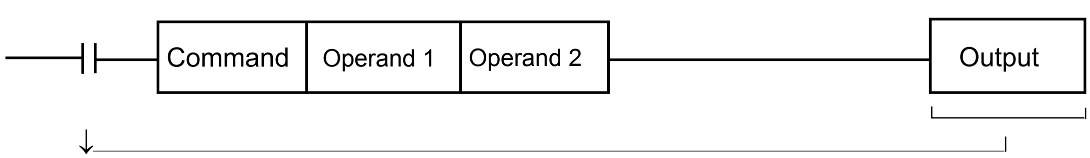


(1) The condition before the command is equivalent to “IF ~ THEN…” in BASIC.



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:

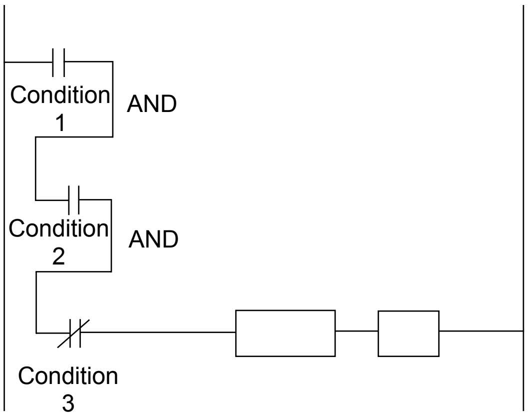


1. In the case of a control command relating to actuator operation, etc., the output will turn OFF the moment the execution of command is started, and turn ON when the execution is completed. In the case of a calculation operation command, etc., the output will turn ON if the result corresponds to a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

3-2 Extension Condition

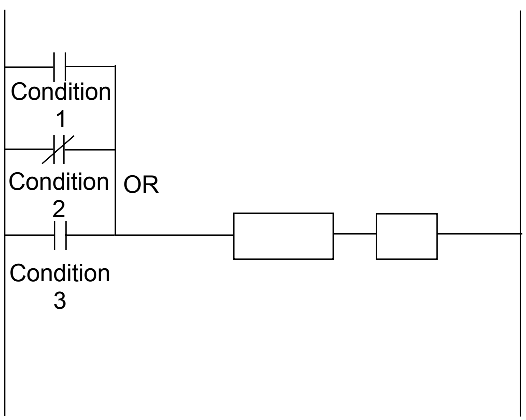
Conditions can be combined in a complex manner.

AND extension (Ladder diagram) (SEL language)



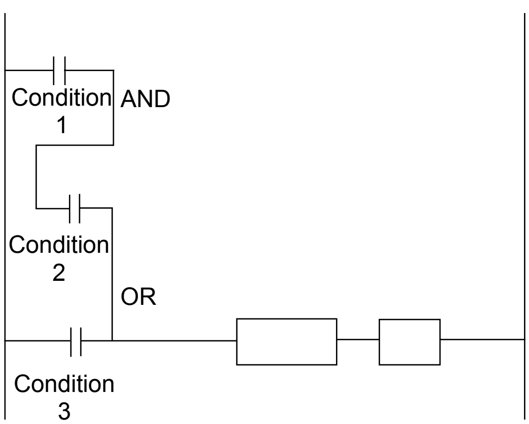
Extension condition	Input condition		Command, declaration	Operand 1	Operand 2	Output
E	N	CND				Pst
		Condition 1				
A		Condition 2				
A	N	Condition 3	Command	Operand 1	Operand 2	

OR extension



Extension condition	Input condition		Command, declaration	Operand 1	Operand 2	Output
E	N	CND				Pst
		Condition 1				
O	N	Condition 2				
O		Condition 3	Command	Operand 1	Operand 2	

AND extension and OR extension



Extension condition	Input condition		Command, declaration	Operand 1	Operand 2	Output
E	N	CND				Pst
		Condition 1				
A		Condition 2				
O		Condition 3	Command	Operand 1	Operand 2	

Part 4 Commands

Chapter 1 List of SEL Language Command Codes 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.

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 CPXX (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	72
	Optional	TRAN	Copy-destination variable	Copy-source variable	ZR	Copy	72
	Optional	CLR	Start-of-clear variable	End-of-clear variable	ZR	Clear variable	73
Arithmetic operation	Optional	ADD	Augend variable	Addend	ZR	Add	74
	Optional	SUB	Minuend variable	Subtrahend	ZR	Subtract	74
	Optional	MULT	Multiplicand variable	Multiplier	ZR	Multiply	75
	Optional	DIV	Dividend variable	Divisor	ZR	Divide	75
	Optional	MOD	Remainder assignment variable	Divisor	ZR	Calculate remainder	76
Function operation	Optional	SIN	Sine assignment variable	Operand [radian]	ZR	Sine	77
	Optional	COS	Cosine assignment variable	Operand [radian]	ZR	Cosine	78
	Optional	TAN	Tangent assignment variable	Operand [radian]	ZR	Tangent	79
	Optional	ATN	Inverse-tangent assignment operation	Operand	ZR	Inverse tangent	80
	Optional	SQR	Root assignment variable	Operand	ZR	Root	81
Logical operation	Optional	AND	AND operand variable	Operand	ZR	Logical AND	82
	Optional	OR	OR operand variable	Operand	ZR	Logical OR	83
	Optional	EOR	Exclusive-OR operand variable	Operand	ZR	Logical exclusive-OR	84
Comparison	Optional	CPXX	Comparison variable	Comparison value	EQ, NE, GT, GE, LT, LE	Compare	85
Timer	Optional	TIMW	Wait time (sec)	Prohibited	TU	Wait	86
	Optional	TIMC	Program number	Prohibited	CP	Cancel waiting	87
	Optional	GTTM	Time assignment variable	Prohibited	CP	Get time	88
I/O, flag operation	Optional	BTXX	Start output, flag	(End output, flag)	CP	Output, flag [ON, OF, NT]	89
	Optional	BTPN	Output port, flag	Timer setting	CP	Output ON pulse	90
	Optional	BTPF	Output port, flag	Timer setting	CP	Output OFF pulse	91
	Optional	WTXX	I/O, flag	(Wait time)	TU	Wait for I/O, flag [ON, OF]	92
	Optional	IN	Head I/O, flag	End I/O, flag	CC	Input binary (32 bits max.)	93
	Optional	INB	Head I/O, flag	Conversion digits	CC	Input BCD (8 digits max.)	94
	Optional	OUT	Head output, flag	End output, flag	CC	Output binary (32 bits max.)	95
	Optional	OUTB	Head output, flag	Conversion digits	CC	Output BCD (8 digits max.)	96
	Optional	FMIO	Format type	Prohibited	CP	Set IN (B)/OUT (B) command format	97
Program control	Optional	GOTO	Jump-destination tag number	Prohibited	CP	Jump	100
	Prohibited	TAG	Declaration tag number	Prohibited	CP	Declare jump destination	100
	Optional	EXSR	Execution subroutine number	Prohibited	CP	Execute subroutine	101
	Prohibited	BGSR	Declaration subroutine number	Prohibited	CP	Start subroutine	101
	Prohibited	EDSR	Prohibited	Prohibited	CP	End subroutine	102

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
Task management	Optional	EXIT	Prohibited	Prohibited	CP	End program	103
	Optional	EXPG	Execution program number	(Execution program number)	CC	Start program	104
	Optional	ABPG	Stop program number	(Stop program number)	CC	Stop other program	105
	Optional	SSPG	Pause program number	(Pause program number)	CC	Pause program	106
	Optional	RSPG	Resumption program number	(Resumption program number)	CC	Resume program	107
Position operation	Optional	PGET	Axis number	Position number	CC	Assign position to variable 199	108
	Optional	PPUT	Axis number	Position number	CP	Assign value of variable 199	109
	Optional	PCLR	Start position number	End position number	CP	Clear point data	110
	Optional	PCPY	Copy-destination position number	Copy-source position number	CP	Copy point data	111
	Optional	PRED	Read axis pattern	Save-destination position number	CP	Read current axis position	112
	Optional	PRDQ	Axis number	Variable number	CP	Read current axis position (1 axis direct)	113
	Optional	PTST	Confirmation axis pattern	Confirmation position number	CC	Confirm position data	114
	Optional	PVEL	Speed [mm/sec]	Assignment-destination position number	CP	Assign position speed	115
	Optional	PACC	Acceleration [G]	Assignment-destination position number	CP	Assign position acceleration	116
	Optional	PDCL	Deceleration [G]	Assignment-destination position number	CP	Assign position deceleration	117
	Optional	PAXS	Axis-pattern assignment variable number	Position number	CP	Read axis pattern	118
	Optional	PSIZ	Size assignment variable number	Prohibited	CP	Confirm position size	119
	Optional	GVEL	Variable number	Position number	CP	Get speed data	120
	Optional	GACC	Variable number	Position number	CP	Get acceleration data	121
	Optional	GDCL	Variable number	Position number	CP	Get deceleration data	122

Category	Condition	Command	Operand 1	Operand 2	Output	Function	Page
Actuator control declaration	Optional	VEL	Speed [mm/sec]	Prohibited	CP	Set speed for CP operation	123
	Optional	VELS	Ratio [%]	Prohibited	CP	Set speed ratio for PTP operation	124
	Optional	OVRD	Speed ratio [%]	Prohibited	CP	Set speed coefficient	125
	Optional	ACC	Acceleration [G]	Prohibited	CP	Set acceleration for CP operation	126
	Optional	ACCS	Ratio [%]	Prohibited	CP	Set acceleration ratio for PTP operation	127
	Optional	DCL	Deceleration [G]	Prohibited	CP	Set deceleration for CP operation	128
	Optional	DCLS	Ratio [%]	Prohibited	CP	Set deceleration ratio for PTP operation	129
	Optional	SCRV	Ratio [%]	Prohibited	CP	Set sigmoid motion ratio	130
	Optional	OFST	Setting axis pattern	Offset value [mm]	CP	Set offset	131
	Optional	DEG	Division angle [deg]	Prohibited	CP	Set division angle	132
	Optional	GRP	Valid axis pattern	Prohibited	CP	Set group axes	133
	Optional	HOLD	(Input port to pause)	(HOLD type)	CP	Declare port to pause	134
	Optional	CANC	(Input port to abort)	(CANC type)	CP	Declare port to abort	135
	Optional	DIS	Distance	Prohibited	CP	Set spline division distance	136
	Optional	POTP	0 or 1	Prohibited	CP	Set PATH output type	137
	Optional	PAPR	Distance	Speed	CP	Set PUSH command distance, speed	138
	Optional	DFTL	Tool coordinate system number	Position number	CP	Define tool coordinate system	139
	Optional	SLTL	Tool coordinate system number	Prohibited	CP	Select tool coordinate system	140
	Optional	GTTL	Tool coordinate system number	Position number	CP	Get tool coordinate system definition data	141
	Optional	DFWK	Work coordinate system number	Position number	CP	Define work coordinate system	142
	Optional	SLWK	Work coordinate system number	Prohibited	CP	Select work coordinate system	143
	Optional	GTWK	Work coordinate system number	Position number	CP	Get work coordinate system definition data	144
	Optional	DFIF	Interference check zone number	Position number	CP	Define coordinates of simple interference check zone	151
	Optional	SOIF	Interference check zone number	Output/global flag number	CP	Specify output for simple interference check zone	152
	Optional	SEIF	Interference check zone number	0 or 1 or 2 (Error type)	CP	Specify error type for simple interference check zone	153
	Optional	GTIF	Interference check zone number	Position number	CP	Get definition coordinates of simple interference check zone	154
	Optional	PTPD	Prohibited	Prohibited	CP	Specify current arm as PTP target arm system (Movement of the opposite arm system is prohibited when the target value cannot be achieved) (No arm operation)	149
	Optional	PTPE	Prohibited	Prohibited	CP	Specify current arm as PTP target arm system (Movement of the opposite arm system is permitted when the target value cannot be achieved) (No arm operation)	150
	Optional	PTPR	Prohibited	Prohibited	CP	Specify right arm as PTP target arm system (Movement of the opposite arm system is prohibited when the target value cannot be achieved) (No arm operation)	147
	Optional	PTPL	Prohibited	Prohibited	CP	Specify left arm as PTP target arm system (Movement of the opposite arm system is prohibited when the target value cannot be achieved) (No arm operation)	148
	Optional	RIGH	Prohibited	Prohibited	PE	Change current arm system to right arm (Arm 2 may operate if the current arm system is the opposite arm)	145
	Optional	LEFT	Prohibited	Prohibited	PE	Change current arm system to left arm (Arm 2 may operate if the current arm system is the opposite arm)	146

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 $\neq$ Operand 2,
GT: Operand 1 > Operand 2, GE: Operand 1 $\geq$ Operand 2,
LT: Operand 1 < Operand 2, LE: Operand 1 $\leq$ Operand 2

Category	Condition	Command	Operand 1	Operand 2	Output	Function	Page
Actuator control command	Optional	SVXX	Operation axis pattern	Prohibited	PE	Servo [ON, OF]	155
	Optional	MOVP	Destination position number	Prohibited	PE	Move to specified position	156
	Optional	MOVL	Destination position number	Prohibited	PE	Move to specified position via interpolation	157
	Optional	MVPI	Travel position number	Prohibited	PE	Move to relative position	158
	Optional	MVLI	Travel position number	Prohibited	PE	Move to relative position via interpolation	159
	Optional	PATH	Start position number	End position number	PE	Move along path	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	CIR2	Passing position 1 number	Passing position 2 number	PE	Move along circle 2 (arc interpolation)	165
	Optional	ARC2	Passing position number	End position number	PE	Move along arc 2 (arc interpolation)	166
	Optional	CIRS	Passing position 1 number	Passing position 2 number	PE	Move three-dimensionally along circle	167
	Optional	ARCS	Passing position number	End position number	PE	Move three-dimensionally along arc	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 width	171
	Optional	TMPI	Position number	Prohibited	PE	Move relatively between positions on tool coordinate system	172
	Optional	TMLI	Position number	Prohibited	PE	Move relatively between positions on tool coordinate system with interpolation	173
	Optional	PTRQ	Axis pattern	Ratio [%]	CC	Change push torque limit parameter	174
	Optional	CIR	Passing position 1 number	Passing position 2 number	PE	Move along circle (CIR2 is recommended)	175
	Optional	ARC	Passing position number	End position number	PE	Move along arc (ARC2 is recommended)	176
	Refer to the page on palletizing for commands relating to arch motion.						
	Optional	ARCH	Position number	Position number	PE	Arch motion	222
	Optional	ACHZ	Axis number	Prohibited	CP	Declare arch-motion Z-axis	213
	Optional	ATRG	Position number	Position number	CP	Set arch trigger	214
	Optional	AEXT	(Position number)	Prohibited	CP	Set arch-motion composition	215
	Optional	OFAZ	Offset value	Prohibited	CP	Set arch-motion Z-axis offset	215
Structural IF	Optional	IFXX	Comparison variable	Comparison value	CP	Compare [EQ, NE, GT, GE, LT, LE]	177
	Optional	ISXX	Column number	Column number, character literal	CP	Compare strings	178
	Prohibited	ELSE	Prohibited	Prohibited	CP	Declare execution destination when IF command condition is not satisfied	179
	Prohibited	EDIF	Prohibited	Prohibited	CP	Declare end of IF	179
Structural DO	Optional	DWXX	Comparison variable	Comparison value	CP	Loop [EQ, NE, GT, GE, LT, LE]	180
	Optional	LEAV	Prohibited	Prohibited	CP	Pull out from DO	180
	Optional	ITER	Prohibited	Prohibited	CP	Repeat DO	181
	Prohibited	EDDO	Prohibited	Prohibited	CP	Declare end of DO	181
Multi-branching	Optional	SLCT	Prohibited	Prohibited	CP	Declare start of multi-branching	182
	Prohibited	WHXX	Comparison variable	Comparison value	CP	Branch value [EQ, NE, GT, GE, LT, LE]	183
	Prohibited	WSXX	Column number	Column number, character literal	CP	Branch character string [EQ, NE]	184
	Prohibited	OTHE	Prohibited	Prohibited	CP	Declare branching destination when condition is not satisfied	185
	Prohibited	EDSL	Prohibited	Prohibited	CP	Declare end of SLCT	185

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	186
	Optional	PGST	Variable number	Program number	CP	Get program status	187
	Optional	SYST	Variable number	Prohibited	CP	Get system status	188
	Optional	GARM	Variable number	Prohibited	CP	Get current arm system	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	TMRD	Timer setting	Prohibited	CP	Set READ timeout value	192
	Optional	WRIT	Channel number	Column number	CP	Output to channel	193
	Optional	SCHA	Character code	Prohibited	CP	Set end character	194
String operation	Optional	SCPY	Column number	Column number, string literal	CC	Copy character string	195
	Optional	SCMP	Column number	Column number, string literal	EQ	Compare character strings	196
	Optional	SGET	Variable number	Column number, string literal	CP	Get character	197
	Optional	SPUT	Column number	Data	CP	Set character	198
	Optional	STR	Column number	Data	CC	Convert character string; decimal	199
	Optional	STRH	Column number	Data	CC	Convert character string; hexadecimal	200
	Optional	VAL	Variable number	Column number, string literal	CC	Convert character string data; decimal	201
	Optional	VALH	Variable number	Column number, string literal	CC	Convert character string data; hexadecimal	202
	Optional	SLEN	Character string length	Prohibited	CP	Set length	203

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	BGPA	Palletizing number	Prohibited	CP	Declare start of palletizing setting	204
	Prohibited	EDPA	Prohibited	Prohibited	CP	Declare end of palletizing setting	204
	Optional	PAPI	Count	Count	CP	Set palletizing counts	205
	Optional	PAPN	Pattern number	Prohibited	CP	Set palletizing pattern	205
	Optional	PASE	Axis number	Axis number	CP	Set palletizing axes	206
	Optional	PAPT	Pitch	Pitch	CP	Set palletizing pitches	206
	Optional	PAST	(Position number)	Prohibited	CP	Set palletizing reference point	207
	Optional	PAPS	Position number	Prohibited	CP	Set 3 palletizing points for teaching	208
	Optional	PSLI	Offset amount	(Count)	CP	Set zigzag	209
	Optional	PCHZ	(Axis number)	Prohibited	CP	Set palletizing Z-axis	210
	Optional	PTRG	Position number	Position number	CP	Set palletizing arch triggers	211
	Optional	PEXT	(Position number)	Prohibited	CP	Set palletizing composition	212
	Optional	OFPZ	Offset amount	Prohibited	CP	Set palletizing Z-axis offset	212
	Optional	ACHZ	Axis number	Prohibited	CP	Declare arch-motion Z-axis	213
	Optional	ATRG	Position number	Position number	CP	Set arch triggers	214
	Optional	AEXT	(Position number)	Prohibited	CP	Set arch-motion composition	215
	Optional	OFAZ	Offset amount	Prohibited	CP	Set arch-motion Z-axis offset	215
	Optional	PTNG	Palletizing number	Variable number	CP	Get palletizing position number	216
	Optional	PINC	Palletizing number	Prohibited	CC	Increment palletizing position number by 1	216
	Optional	PDEC	Palletizing number	Prohibited	CC	Decrement palletizing position number by 1	217
	Optional	PSET	Palletizing number	Data	CC	Set palletizing position number directly	217
	Optional	PARG	Palletizing number	Axis number	CP	Get palletizing angle	218
	Optional	PAPG	Palletizing number	Position number	CP	Get palletizing calculation data	218
	Optional	PMVP	Palletizing number	(Position number)	PE	Move to palletizing points via PTP	219
	Optional	PACH	Palletizing number	Position number	PE	Palletizing-point arch motion	220
	Optional	ARCH	Position number	Position number	PE	Arch motion	222
	Extension conditions LD (LOAD), A (AND), O (OR), AB (AND BLOCK) and OB (OR BLOCK) are supported.						
Building of pseudo-ladder task	Optional	CHPR	0 or 1	Prohibited	CP	Change task level	224
	Prohibited	TPCD	0 or 1	Prohibited	CP	Specify processing to be performed when input condition is not specified	224
	Optional	TSLP	0 or 1	Prohibited	CP	Task sleep	225
	Optional	OUTR	Output, flag number	Prohibited	CP	Output relay for ladder	See 267
	Optional	TIMR	Local flag number	Timer setting	CP	Timer relay for ladder	See 267

Chapter 2 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	1	2	Assign 2 to variable 1.
	LET	3	10	Assign 10 to variable 3.
	LET	*1	*3	Assign the content of variable 3 (10) to the variable of the content of variable 1 (variable 2).

● 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	1	2	Assign 2 to variable 1.
	LET	2	3	Assign 3 to variable 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 the variable of the content of variable 1 (variable 2).

The variables change as follows:

1	2	3	4		1	2	3	4
2	3	4	10	→	2	10	4	10

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

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

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

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

- MOD (Remainder of division)

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

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	1	100	Assign 100 to variable 1.
	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	*1	*101	Assign the sine of the content of variable 101 (0.5) to the content of variable 1 (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	1	100	Assign 100 to variable 1.
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	*1	*101	Assign the cosine of the content of variable 101 (0.5) to the content of variable 1 (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 $\pi \div 180$

[Example 1]

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

[Example 2]

LET	1	100	Assign 100 to variable 1.
LET	101	45	45 x $\pi \div 180$ (radian) (45° will be converted to radian and assigned to variable 101.)
MULT	101	3.141592	
DIV	101	180	
TAN	*1	*101	Assign the tangent of the content of variable 101 (1) to the content of variable 1 (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.

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

[Example 1]	ATN	100	1	Assign the inverse tangent of 1 (0.785398) to variable 100.
[Example 2]	LET	1	100	Assign 100 to variable 1.
	LET	101	1	Assign 1 to variable 101.
	ATN	*1	*101	Assign the inverse tangent of the content of variable 101 (0.785398) to the content of variable 1 (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	1	10	Assign 10 to variable 1.
	LET	2	4	Assign 4 to variable 2.
	SQR	*1	*2	Assign the root of the content of variable 2 (4) to the content of variable 1 (variable 10).

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	2	Assign 2 to variable 1.
LET	2	204	Assign 204 to variable 2.
LET	3	170	Assign 170 to variable 3.
AND	*1	*3	Assign the logical AND operation result (136) of the content of variable 1 (which is variable 2, or 204) and the content of variable 3 (170), to the content of variable 1 (variable 2).

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	2	Assign 2 to variable 1.
	LET	2	204	Assign 204 to variable 2.
	LET	3	170	Assign 170 to variable 3.
	OR	*1	*3	Assign the logical OR operation result (238) of the content of variable 1 (which is variable 2, or 204) and the content of variable 3 (170), to the content of variable 1 (variable 2).

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	2	Assign 2 to variable 1.
	LET	2	204	Assign 204 to variable 2.
	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 (which is variable 2, or 204) and the content of variable 3 (170), to the content of variable 1 (variable 2).

Decimal	Binary
204	11001100
EOR 170	EOR 10101010
102	01100110

1-5 Comparison Operation

- CPXX (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	CPXX	Variable number	Data	<u>EQ</u> <u>NE</u> <u>GT</u> <u>GE</u> <u>LT</u> <u>LE</u>

[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<u>XX</u>

- 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	2		Assign 2 to variable 1.
LET	2	10		Assign 10 to variable 2.
LET	3	10		Assign 10 to variable 3.
CPNE	*1	*3	310	Turn ON output 310 if the content of variable 1 (variable 2) is not 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

- BTXX (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	BTXX	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.

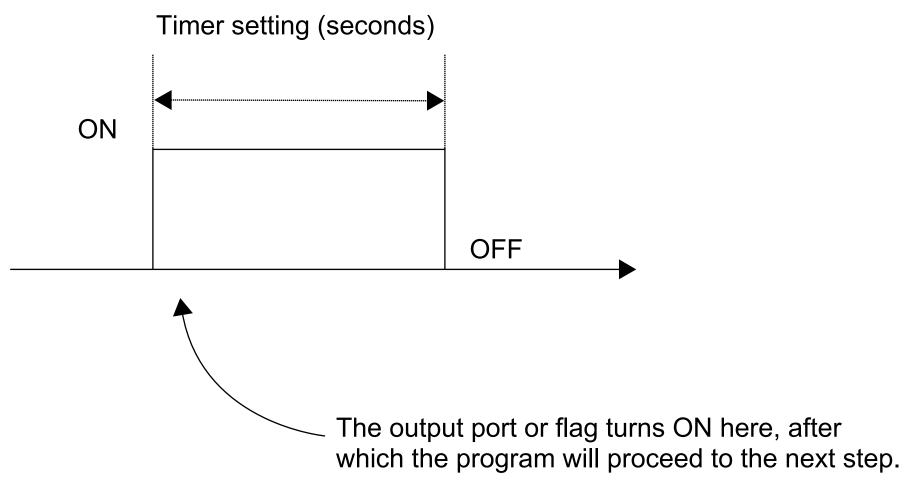
```
BTXX
  ├ON ............ Switch the status to ON.
  ├OF ............ Switch the status to OFF.
  └NT ............ Reverse the status.
```

[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.)

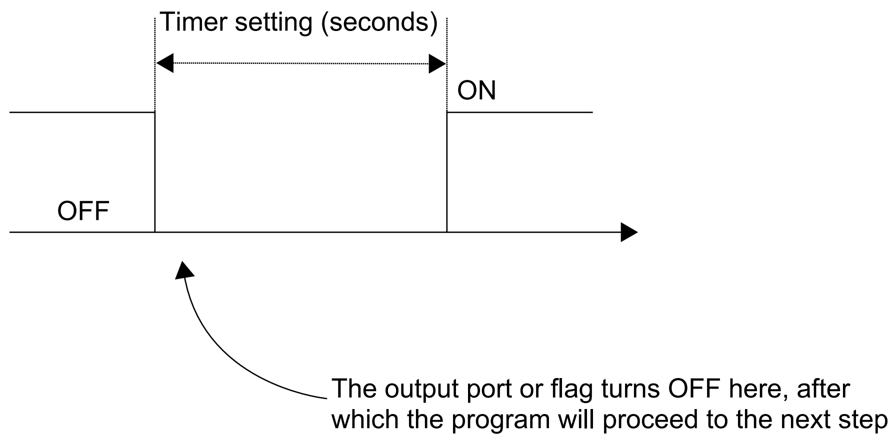
[Example]

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

- 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.

● WTXX (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	WTXX	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).

```
WTXX
  ├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	308	10	Wait for 10 seconds for output port 308 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.

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	.... Binary
15	14	13	12	11	10	9	8	.... Input port number
ON	OFF	OFF	OFF	OFF	ON	OFF	ON	

1	0	0	0	0	1	0	1	.... Binary

$2^7 + 0 + 0 + 0 + 0 + 2^2 + 0 + 2^0$

$128 + 0 + 0 + 0 + 0 + 4 + 0 + 1 = 133$

133 .... Variable 99

(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.

[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	Output, 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.

Upper digit　　Lower digit

15	14	13	12	11	10	9	8
ON	OFF	OFF	OFF	OFF	ON	OFF	ON

.... Input port number

8　　5

85 .... Variable 99

(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).

[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.

133 .... Variable 99

Upper							Lower	
1	0	0	0	0	1	0	1	.... Binary

307	306	305	304	303	302	301	300	.... Output port number
ON	OFF	OFF	OFF	OFF	ON	OFF	ON	

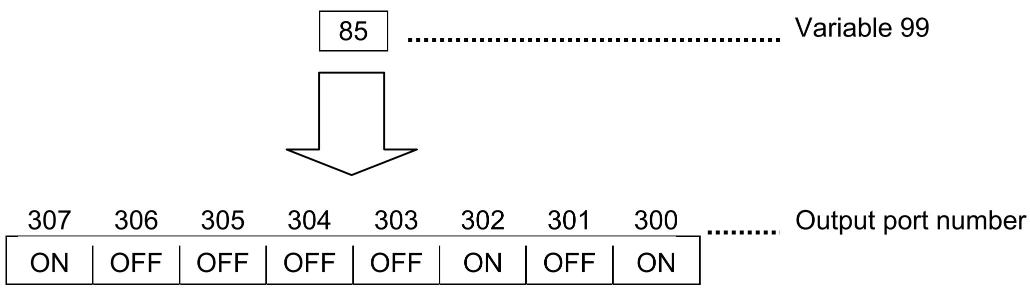
(Note) A maximum of 32 bits can be output.

[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).

[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.

(I/O, flag number upper) (I/O, flag number lower)

01234567h	⇔	01h	23h	45h	67h	⇔	0000 0001	0010 0011	0100 0101	0110 0111

Variable 99 Temporary data I/O port, flag status (0 = OFF, 1 = ON)

OUT(B) command →

← IN(B) command

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

(I/O, flag number upper) (I/O, flag number lower)

01234567h	⇔	23h	01h	67h	45h	⇔	0010 0011	0000 0001	0110 0111	0100 0101

Variable 99 I/O port, flag status (0 = OFF, 1 = ON)

Temporary data

OUT(B) command →

← IN(B) command

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

(I/O, flag number upper) (I/O, flag number lower)

01234567h	⇔	45h	67h	01h	23h	⇔	0100 0101	0110 0111	0000 0001	0010 0011

Variable 99 I/O port, flag status (0 = OFF, 1 = ON)

Temporary data

OUT(B) command →

← IN(B) command

(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.

Variable 99		Temporary data					(I/O, flag number upper)		(I/O, flag number lower)	
01234567h	⇔	67h	45h	23h	01h	⇔	0110 0111	0100 0101	0010 0011	0000 0001

I/O port, flag status (0 = OFF, 1 = ON)

Temporary data
OUT(B) command →
← IN(B) command

(Note) FMIO command is supported in main application version 0.56 or later, PC software version 2.0.45 or later and teaching pendant version 1.13 or later.

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

00123456h ← OUT(B) command / IN(B) command → Variable 99 1193046 (IN/OUT command)
123456 (INB/OUTB command)

OUT(B) command ↓ ↑ IN(B) command

	Temporary data					(I/O, flag number upper)		(I/O, flag number lower)	
FMIO = 0	00h	12h	34h	56h	⇔	0000 0000	0001 0010	0011 0100	0101 0110
FMIO = 1	12h	00h	56h	34h	⇔	0001 0010	0000 0000	0101 0110	0011 0100
FMIO = 2	34h	56h	00h	12h	⇔	0011 0100	0101 0110	0000 0000	0001 0010
FMIO = 3	56h	34h	12h	00h	⇔	0101 0110	0011 0100	0001 0010	0000 0000

Temporary data OUT(B) command → I/O port, flag status (0 = OFF, 1 = ON)
← IN(B) command

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

00001234h $\xrightarrow{\text{OUT(B) command}}$ / $\xleftarrow{\text{IN(B) command}}$ Variable 99 4660 (IN/OUT command)
1234 (INB/OUTB command)

OUT(B) command ↓ ↑ IN(B) command

	Temporary data					(I/O, flag number upper)		(I/O, flag number lower)	
FMIO = 0	00h	00h	12h	34h	⇔	0000 0000	0000 0000	0001 0010	0011 0100
FMIO = 1	00h	00h	34h	12h	⇔	0000 0000	0000 0000	0011 0100	0001 0010
FMIO = 2	12h	34h	00h	00h	⇔	0001 0010	0011 0100	0000 0000	0000 0000
FMIO = 3	34h	12h	00h	00h	⇔	0011 0100	0001 0010	0000 0000	0000 0000

Temporary data $\xrightarrow{\text{OUT(B) command}}$ / $\xleftarrow{\text{IN(B) command}}$ I/O port, flag status (0 = OFF, 1 = ON)

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

00000012h $\xrightarrow{\text{OUT(B) command}}$ / $\xleftarrow{\text{IN(B) command}}$ Variable 99 18 (IN/OUT command)
12 (INB/OUTB command)

OUT(B) command ↓ ↑ IN(B) command

	Temporary data					(I/O, flag number upper)		(I/O, flag number lower)	
FMIO = 0	00h	00h	00h	12h	⇔	0000 0000	0000 0000	0000 0000	0001 0010
FMIO = 1	00h	00h	12h	00h	⇔	0000 0000	0000 0000	0001 0010	0000 0000
FMIO = 2	00h	12h	00h	00h	⇔	0000 0000	0001 0010	0000 0000	0000 0000
FMIO = 3	12h	00h	00h	00h	⇔	0001 0010	0000 0000	0000 0000	0000 0000

Temporary data $\xrightarrow{\text{OUT(B) command}}$ / $\xleftarrow{\text{IN(B) command}}$ I/O port, flag status (0 = OFF, 1 = ON)

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 1) A GOTO command is valid only within the same program.
(Note 2) Do not create a program containing an indefinite loop of continuous movement commands (refer to Chapter 4 in Part 4) using the TAG-GOTO syntax. It will result in an accumulation of coordinate conversion errors.

[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 EDXX 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) IFXX or ISXX and EDIF syntax
(2) DWXX 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 (Note))	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.

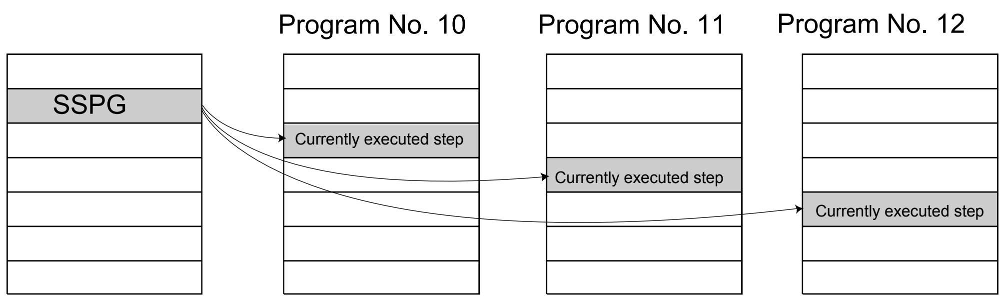
- SSPG (Pause program)

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.

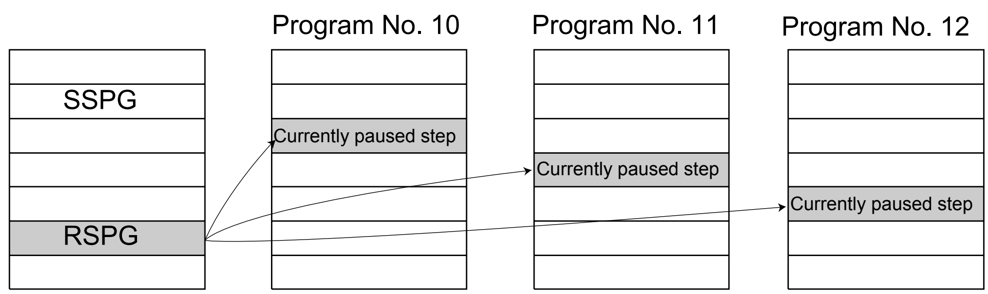
● RSPG (Resume program)

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 (this command will not be executed) if the data being read is XXX.XX.
Axis No. 1: X-axis, Axis No. 2: Y-axis, Axis No. 3: Z-axis, Axis No. 4: R-axis

[Example 1]	PGET	2	3	Read to variable 199 the data of the Y-axis (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 (Y-axis (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.
Axis No. 1: X-axis, Axis No. 2: Y-axis, Axis No. 3: Z-axis, Axis No. 4: R-axis

[Example 1]	LET	199	150	Assign 150 to variable 199.
	PPUT	2	3	Write the content of variable 199 (150) to the Y-axis (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 (Y-axis (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 will be expressed as XX.XXX (not 0.000).

[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	1011	10	Read the current positions of the X, Y and R-axes to position 10.
[Example 2]	LET	1	10	Assign 10 to variable 1.
	PRED	11	*1	Read the current positions of the X and Y-axes 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.
Axis No. 1: X-axis, Axis No. 2: Y-axis, Axis No. 3: Z-axis, Axis No. 4: R-axis

[Example] PRDQ 2 100 Read the current position of the Y-axis (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 when all of the data specified by the axis pattern is invalid (XX.XXX).
"0" is treated as valid data.

[Example 1]	PTST	11	10	300	Turn ON output 300 if there are no valid values of the X and Y-axes at position 10. Output 300 will turn OFF if the position data is given as follows:
[Example 2]	LET	1	11		Assign 11 to variable 1.
	PTST	1011	*1	600	Turn ON flag 600 if there are no valid values in the data of the X, Y and R-axes at the content of variable 1 (position 11). Flag 600 will turn ON if the position data is given as follows:

No.(Name)	Axis1	Axis2	Axis3	Axis4	Vel	Acc	Dcl
9( )							
10( )	200.000	100.000					
11( )			150.000				
12( )							

- 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 for CP operation specified in operand 1 to the position number specified in operand 2. The speed in operand 1 is set in mm/sec.

(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 for CP operation specified in operand 1 to the position number specified in operand 2.
The acceleration in operand 1 is set in G and may include up to two decimal places.

(Note) Range check is not performed for a PACC command.

[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] Write the deceleration for CP operation specified in operand 1 to the position number specified in operand 2.
The deceleration in operand 1 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	98	Read the axis pattern at position 98 to variable 1. If the position is given as follows, “3” (binary 0011) 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, “8” (binary 1000) will be stored in variable 3.

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

No.(Name)	Axis1	Axis2	Axis3	Axis4
98( )	200.000	100.000		
99( )	350.000		120.000	
100( )				
101( )				180.000

0 0 1 1 = 2 + 1 = 3
0 1 0 1 = 4 + 1 = 5
0
1 0 0 0 = 8

- 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.

No.(Name)	Axis1	Axis2	Axis3	Axis4	Vel	Acc	Dcl
9( )							
10( )	250.000	100.000	100.000	30.000	100	0.80	0.80
11( )							

If the position data is set as above when the command is executed, 100 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 1] GACC 100 10 Set the acceleration data at position No. 10 in variable 100.

No.(Name)	Axis1	Axis2	Axis3	Axis4	Vel	Acc	Dcl
9( )							
10( )	250.000	100.000	100.000	30.000	100	0.80	0.80
11( )							

If the position data is set as above when the command is executed, 0.8 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.

No.(Name)	Axis1	Axis2	Axis3	Axis4	Vel	Acc	Dcl
9( )							
10( )	250.000	100.000	100.000	30.000	100	0.80	0.80
11( )							

If the position data is set as above when the command is executed, 0.8 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 travel speed for CP operation in the value specified in operand 1.
The unit is mm/sec.

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

[Example 1]	VEL	100	Set the speed to 100 mm/sec.
	MOVL	1	Move to point 1 at 100 mm/sec.
[Example 2]	VEL	500	Set the speed to 500 mm/sec.
	MOVL	2	Move to point 2 at 500 mm/sec.

- VELS (Set speed 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	VELS	Ratio	Prohibited	CP

[Function] Set the travel speed for PTP operation command (angular velocity for axes other than the Z-axis) as a ratio of the maximum PTP speed to be specified in operand 1. The ratio in operand 1 is set as an integer (unit: %).

(Note 1) When a RIGH or LEFT command is used, the speed must be set via VELS even when PTP operation commands are not used.

[Example 1]	VELS	50	Set the travel speed for PTP operation command to 50% of the maximum speed.
	MOVP	1	The axes will move to position No. 1 in PTP mode at 50% of the maximum speed.

- 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. (Smoothness of actual operation cannot be guaranteed. Movement must be checked on the actual machine.)

[Example 1]

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

- 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 for CP operation in the value specified in operand 1.
The acceleration in operand 1 is set in G and may include up to two decimal places.

(Note) During CP operation, 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 CP 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.

- ACCS (Set acceleration 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	ACCS	Ratio	Prohibited	CP

[Function] Set the travel acceleration for PTP operation command (angular acceleration for axes other than the Z-axis) as a ratio of the maximum PTP acceleration to be specified in operand 1. The ratio in operand 1 is set as an integer (unit: %).

(Note 1) When setting the acceleration ratio, always refer to 5, "Precautions for Use," in the operation manual for the IX-Series Horizontal Articulated Robot.

[Example] ACCS 50 Set the travel acceleration for PTP operation command to 50% of the maximum acceleration.

- 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 for CP operation in the value specified in operand 1.
The deceleration in operand 1 is set in G and may include up to two decimal places.

(Note) During CP operation, 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 CP 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.

- DCLS (Set deceleration 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	DCLS	Ratio	Prohibited	CP

[Function] Set the travel deceleration for PTP operation command (angular deceleration for axes other than the Z-axis) as a ratio of the maximum PTP deceleration to be specified in operand 1. The ratio in operand 1 is set as an integer (unit: %).

(Note 1) When setting the deceleration ratio, always refer to 5, "Precautions for Use," in the operation manual for the IX-Series Horizontal Articulated Robot.

[Example] DCLS 50 Set the travel deceleration for PTP operation command to 50% of the maximum deceleration.

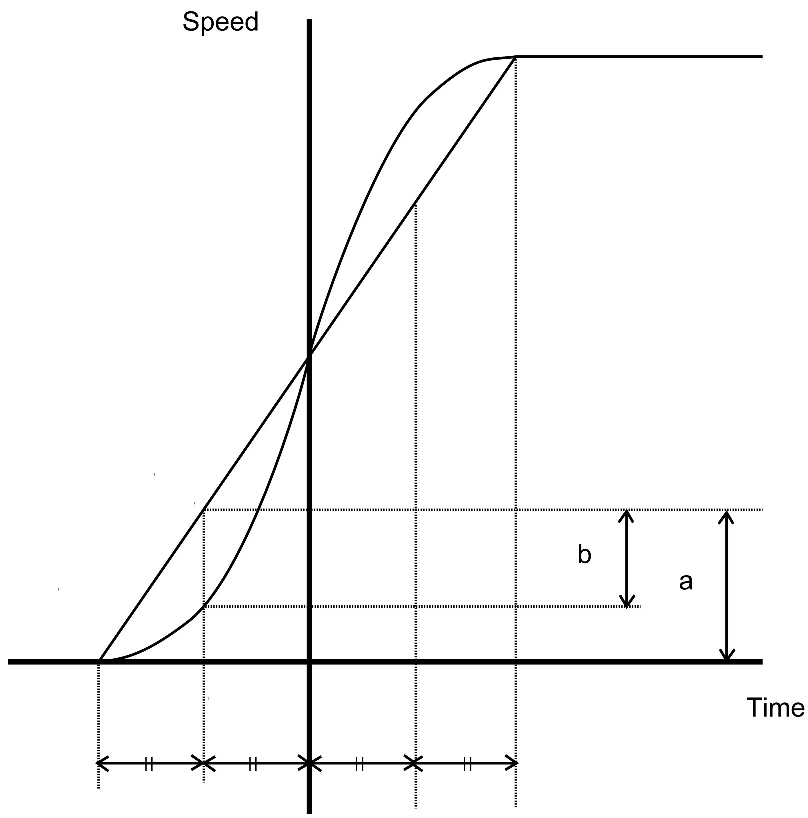
- SCRV (Set sigmoid motion ratio)

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, TMPI, TMLI



[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.

(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, TMLI, and TMPI commands.

[Example 1]	OFST	110	50	Add 50 mm to the specified positions of the Y and Z-axes.
[Example 2]	LET	1	30	Assign 30 to variable 1.
	OFST	1000	*1	Add the content of variable 1 (30°) to the specified position of the R-axis.

- 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.

- 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 OFST, DFTL, DFWK, and DFIF commands, as well as with servo operation commands using position data.

[Example 1]	GRP	11		Data of the X and Y-axes become valid.
	CIR2	1	2	Axis-pattern error will not generate even if data is set for the Z and R-axes.

- 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

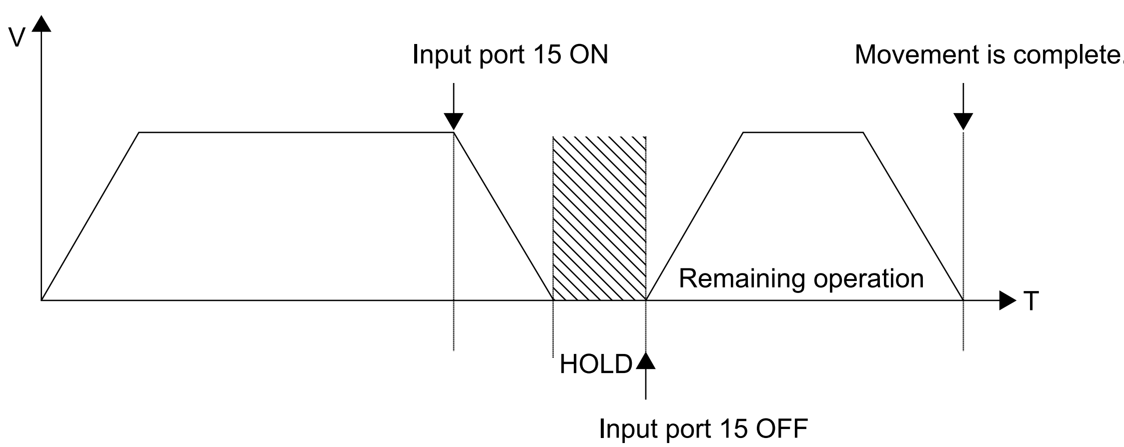
[Function] Declare an input port or global flag to pause while a servo command is being executed. When operation is performed on the input port or global flag specified in operand 1, the current servo processing will pause. (If the axes are moving, they will decelerate to a stop.) If nothing is specified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

[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 PATH operation.)

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



- CANC (Cancel: Declare axis port to abort)

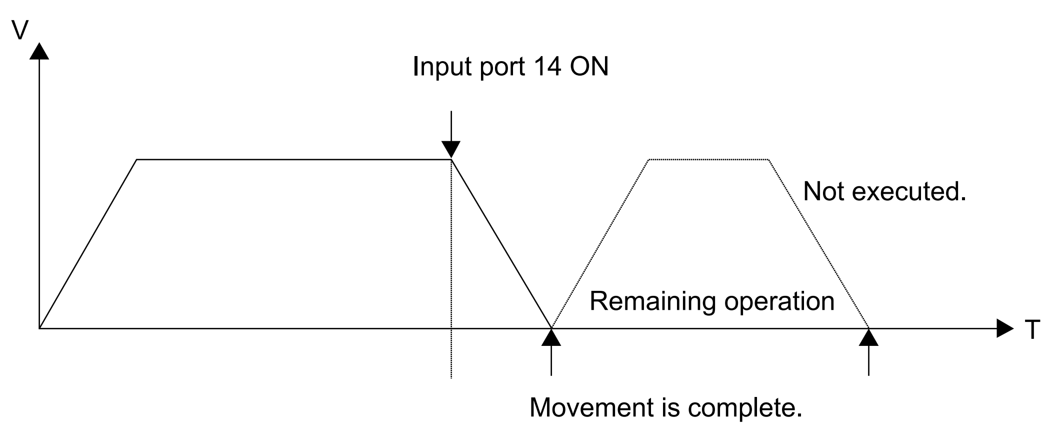
Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	CANC	(Input port, global flag)	(CANC type)	CP

[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.

[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 PATH operation.)

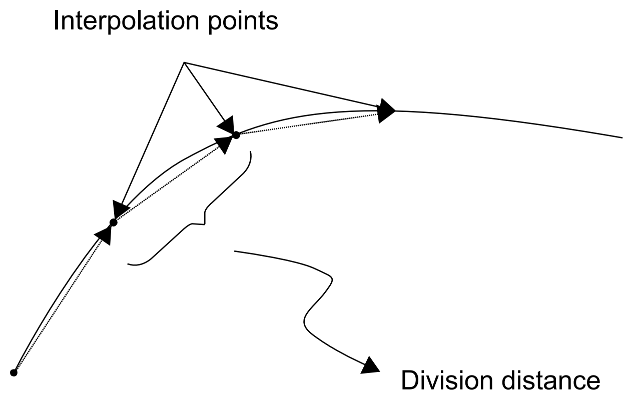
[Example] CANC 14 0 The axes will decelerate to a stop when input port 14 turns ON.



- 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

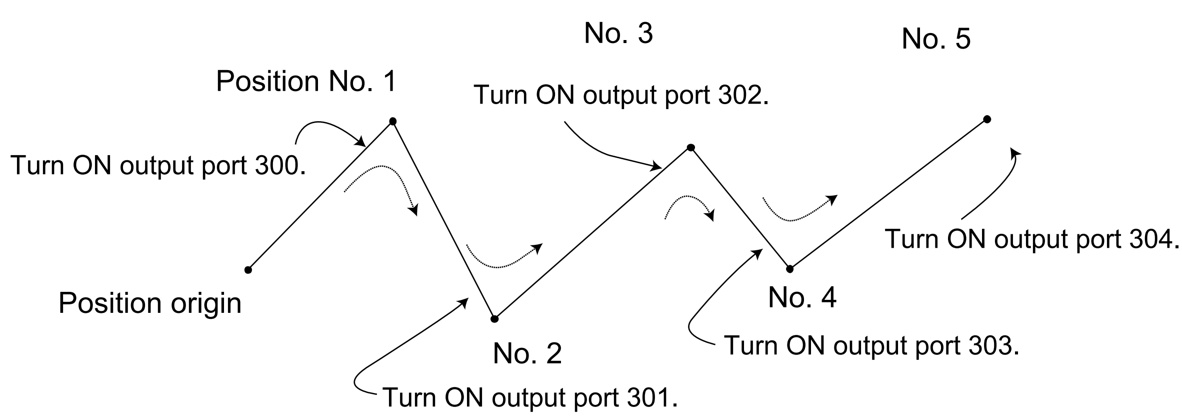
[Function] Set the output type in the output field to be used when a PATH or PSPL command is executed.
When a PATH or PSPL command is executed, the output will operate as follows in accordance with the setting of the POTP command.

(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]

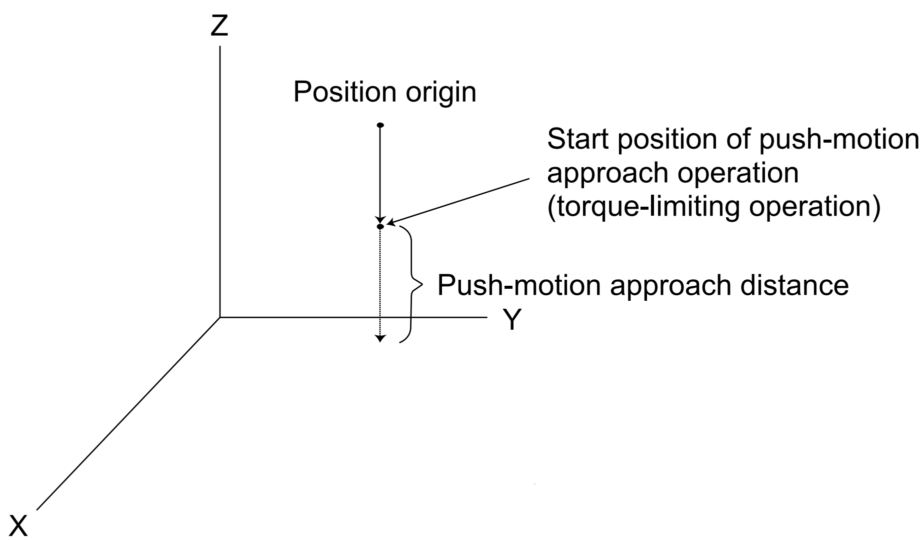
POTP	1			
PATH	1	5	300	Turn ON output port Nos. 300 through 304 sequentially each time a specified position approaches during a pass movement from position Nos. 1 through 5, starting from the first position.



- 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.

- DFTL (Define tool coordinate system)

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	DFTL	Tool coordinate system number	Position number	CP

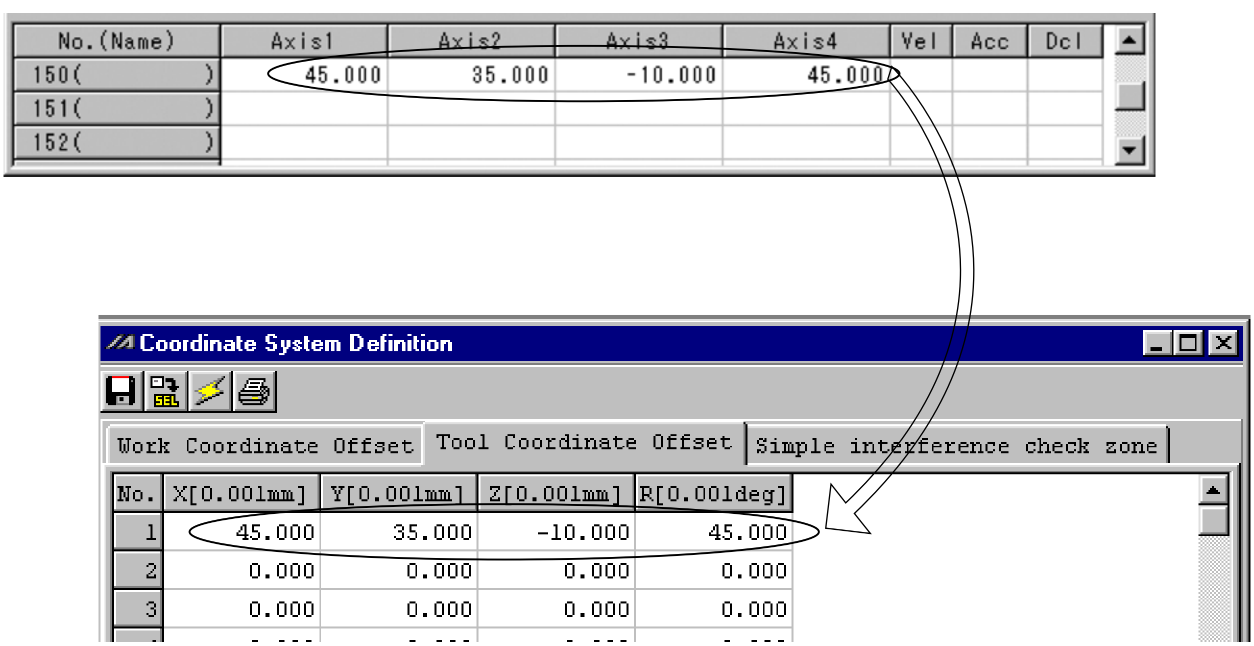
[Function] Set the position data specified in operand 2 as the offset data for the tool coordinate system specified in operand 1. The offset data for tool coordinate system will include the specified position data corresponding to all axes, but the position data for invalid axes will be set as "zero offset."

(Note 1) Tool coordinate system No. 0 is reserved for a condition where no tool offset is applicable. Therefore, setting this coordinate system number will generate an "Error No. B71: Coordinate system number error."

(Note 2) GRP commands are invalid with respect to this command.

[Example] DFTL 1 150

Position data



- SLTL (Select tool coordinate system)

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	SLTL	Tool coordinate system number	Prohibited	CP

[Function] Set the value specified in operand 1 as the selected tool coordinate system number.
Refer to 3, "Coordinate System," in Chapter 3 of Part 4.

(Note 1) The number declared last in the system becomes valid.
The selected tool coordinate system number will remain valid after the program ends, or even after reconnection of power if a system-memory backup battery is installed.

(Note 2) Only one tool coordinate system number can be selected in the system.

(Note 3) Expressly declare SLTL in the program to prevent problems that may occur when the coordinate system number changed via the PC software or teaching pendant was not returned to the original setting.
(Set SLCT = 0, if tool coordinate system is not used.)

- GTTL (Get tool coordinate system definition data)

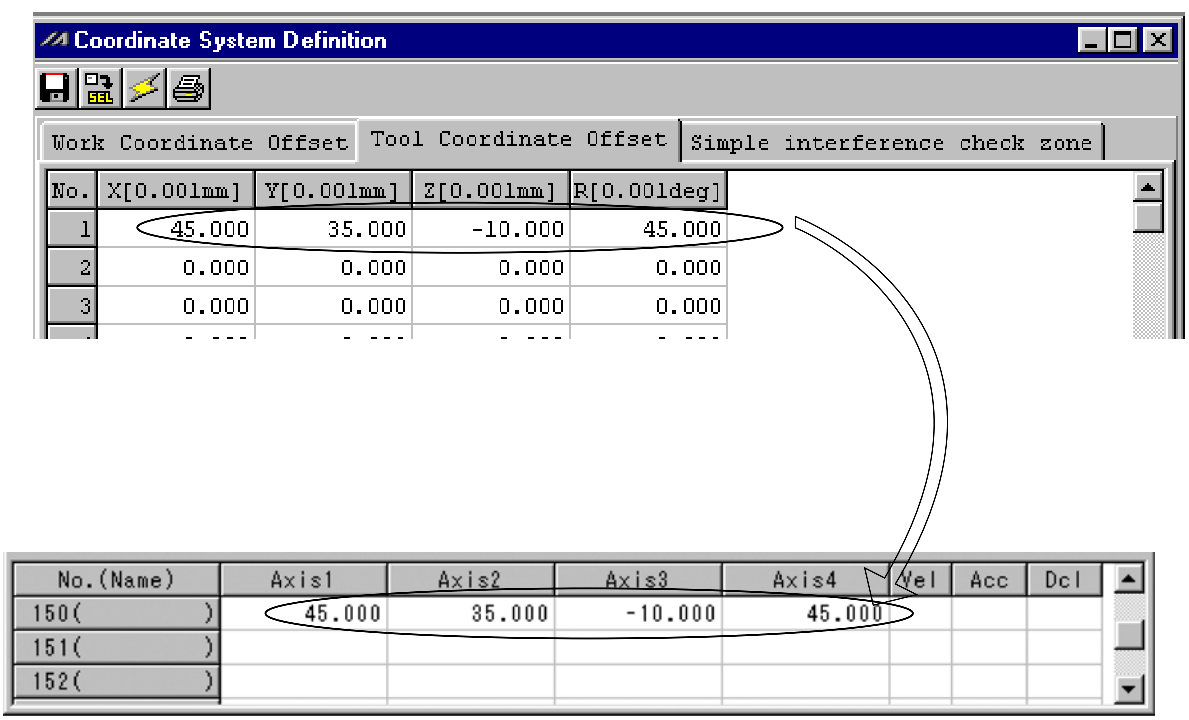
Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	GTTL	Tool coordinate system number	Position number	CP

[Function] Set in the position data specified in operand 2 the offset data for the tool coordinate system specified in operand 1. The position data will include the specified offset data for tool coordinate system corresponding to all axes.

(Note 1) Tool coordinate system No. 0 is reserved for a condition where no tool offset is applicable. Therefore, setting this coordinate system number will generate an "Error No. B71: Coordinate system number error."

(Note 2) GRP commands are invalid with respect to this command.

[Example 1]	OFST	110	50	The specified Y and Z-axis positions will be incremented by 50 mm.
[Example 2]	LET	1	30	Assign 30 to variable 1.
	OFST	1000	*1	The specified R-axis position will be incremented by the content of variable 1 (30°).
[Example]	GTTL	1	150	



● DFWK (Define work coordinate system)

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	DFWK	Work coordinate system number	Position number	CP

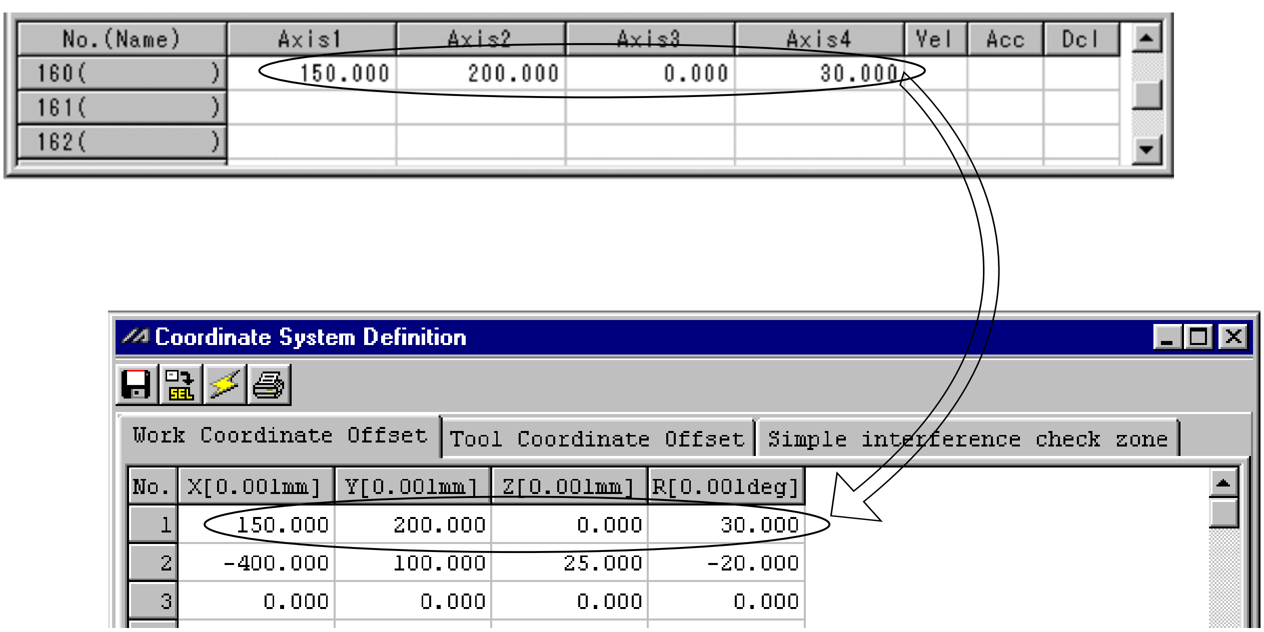
[Function] Set the position data specified in operand 2 as the offset data for the work coordinate system specified in operand 1. The offset data for work coordinate system will include the specified position data corresponding to all axes, but the position data for invalid axes will be set as "zero offset."

(Note 1) Work coordinate system No. 0 is reserved as the base coordinate system. Therefore, setting this coordinate system number will generate an "Error No. B71: Coordinate system number error."

(Note 2) GRP commands are invalid with respect to this command.

[Example] DFWK 1 160

Position data



● SLWK (Select work coordinate system)

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	SLWK	Work coordinate system number	Prohibited	CP

[Function] Set the value specified in operand 1 as the selected work coordinate system number.
Refer to 3, "Coordinate System," in Chapter 3 of Part 4.

(Note 1) The number declared last in the system becomes valid.
The selected work coordinate system number will remain valid after the program ends, or even after reconnection of power if a system-memory backup battery is installed.

(Note 2) Only one work coordinate system number can be selected in the system.

(Note 3) Expressly declare SLWK in the program to prevent problems that may occur when the coordinate system number changed via the PC software or teaching pendant was not returned to the original setting.
(Set SLWK = 0, if work coordinate system is not used.)

- GTWK (Get work coordinate system definition data)

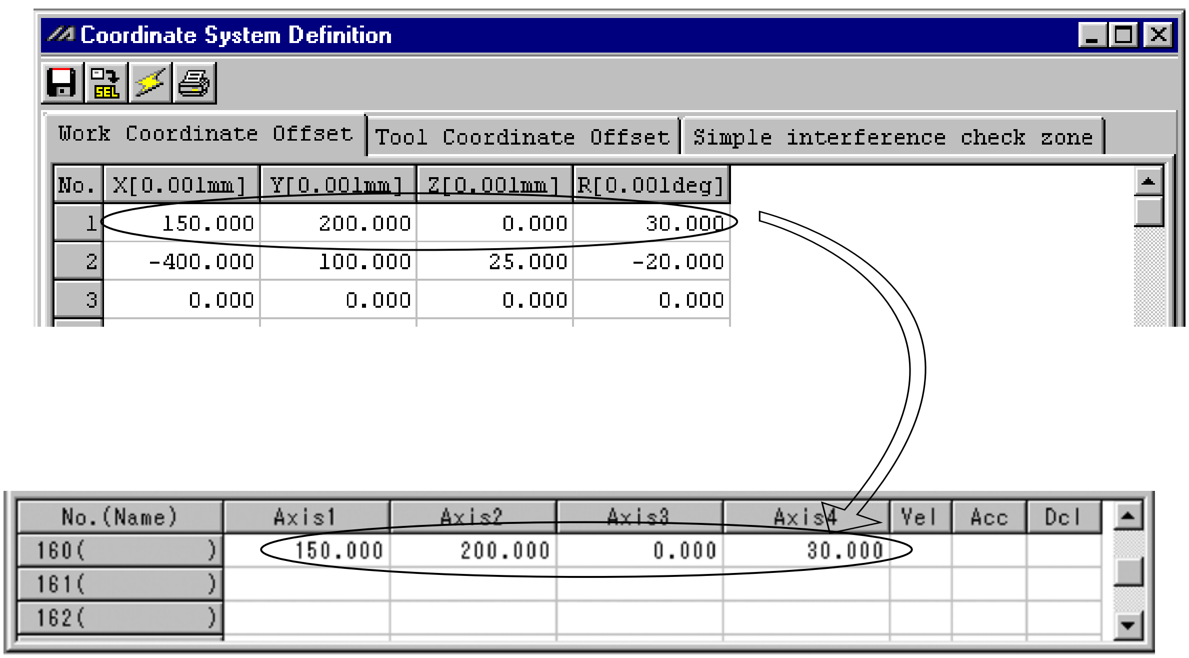
Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	GTWK	Work coordinate system number	Position number	CP

[Function] Set in the position data specified in operand 2 the offset data for the work coordinate system specified in operand 1. The position data will include the specified offset data for work coordinate system corresponding to all axes.

(Note 1) Work coordinate system No. 0 is reserved as the base coordinate system. Therefore, setting this coordinate system number will generate an "Error No. B71: Coordinate system number error."

(Note 2) GRP commands are invalid with respect to this command.

[Example] GTWK 1 160



- RIGH (Change current arm system to right arm (Arm 2 may operate if the current arm system is the opposite arm))

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	RIGH	Prohibited	Prohibited	PE

[Function] Change the current arm system to the right arm system. If the current arm system is the left arm system, arm 2 will be operated to change the arm system to the right arm system. After this operation, arms 1 and 2 will form a straight line. If the current arm system is the right arm system, no arm operation will take place.
(For details, refer to 2, "Arm System," in Chapter 3 of Part 4.)

(Note 1) When a RIGH or LEFT command is used, the speed must be set via VELS even when PTP operation commands are not used.

- LEFT (Change current arm system to left arm (Arm 2 may operate if the current arm system is the opposite arm))

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	LEFT	Prohibited	Prohibited	PE

[Function] Change the current arm system to the left arm system. If the current arm system is the right arm system, arm 2 will be operated to change the arm system to the left arm system. After this operation, arms 1 and 2 will form a straight line. If the current arm system is the left arm system, no arm operation will take place.
(For details, refer to 2, "Arm System," in Chapter 3 of Part 4.)

(Note 1) When a RIGH or LEFT command is used, the speed must be set via VELS even when PTP operation commands are not used.

- PTPR (Specify right arm as PTP target arm system (Movement of the opposite arm system is prohibited when the target value cannot be achieved) (No arm 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	PTPR	Prohibited	Prohibited	CP

[Function] Specify the right arm system as the target arm system for PTP operation command. Once a PTPR command is executed, the target arm system for PTP operation command will become the right arm system and any target value that cannot be achieved with the right arm system will generate an error. Executing this command itself will not accompany any arm operation. (For details, refer to 2, "Arm System," in Chapter 3 of Part 4.)

- PTPL (Specify left arm as PTP target arm system (Movement of the opposite arm system is prohibited when the target value cannot be achieved) (No arm 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	PTPL	Prohibited	Prohibited	CP

[Function] Specify the left arm system as the target arm system for PTP operation command. Once a PTPL command is executed, the target arm system for PTP operation command will become the left arm system and any target value that cannot be achieved with the left arm system will generate an error. Executing this command itself will not accompany any arm operation.
(For details, refer to 2, "Arm System," in Chapter 3 of Part 4.)

- PTPD (Specify current arm as PTP target arm system (Movement of the opposite arm system is prohibited when the target value cannot be achieved) (No arm 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	PTPD	Prohibited	Prohibited	CP

[Function] Specify the current arm system as the target arm system for PTP operation command. Once a PTPD command is executed, the target arm system for PTP operation command will become the current arm system and any target value that cannot be achieved with this arm system will generate an error. Executing this command itself will not accompany any arm operation.
(For details, refer to 2, "Arm System," in Chapter 3 of Part 4.)

- PTPE (Specify current arm as PTP target arm system (Movement of the opposite arm system is permitted when the target value cannot be achieved) (No arm 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	PTPE	Prohibited	Prohibited	CP

[Function] Specify the current arm system as the target arm system for PTP operation command. Once a PTPE command is executed, the target arm system for PTP operation command will become the current arm system and any target value that cannot be achieved with this arm system will be processed by changing the target arm system to the opposite arm system. Target values that cannot be achieved with either the right or left arm system will generate an error. Executing this command itself will not accompany any arm operation.
(For details, refer to 2, "Arm System," in Chapter 3 of Part 4.)

● DFIF (Define coordinates of simple interference check zone)

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	DFIF	Interference check zone number	Position number (Consecutive two positions will be used)	CP

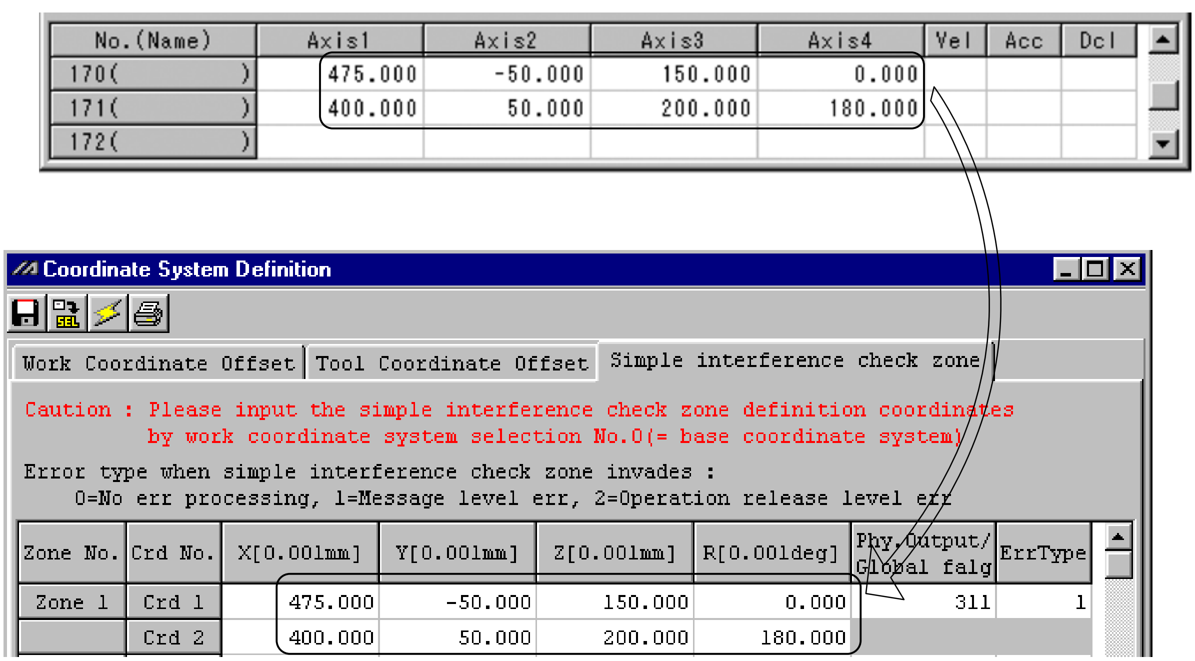
[Function] Set the consecutive two position data starting from the position number specified in operand 2 as the coordinate data defining the simple interference check zone specified in operand 1. The position data specified in operand 2 will be set as definition coordinates 1 of the simple interference check zone, while the next position data will be set as definition coordinates 2. If the axis patterns of the consecutive two position data do not match, an "Error No. C30: Axis pattern error" will generate.

(Note 1) The definition coordinates of simple interference check zone are always treated as data on the base coordinate system (work coordinate system No. 0). Therefore, to provide position data for valid definition coordinates for the purpose of executing a DFIF command, the data must be set on the base coordinate system beforehand.

(Note 2) After the definition coordinates of simple interference check zone are changed, it will take 5 msec before the check result reflects the new settings.

(Note 3) GRP commands are invalid with respect to this command.

[Example 1] DFIF 1 170



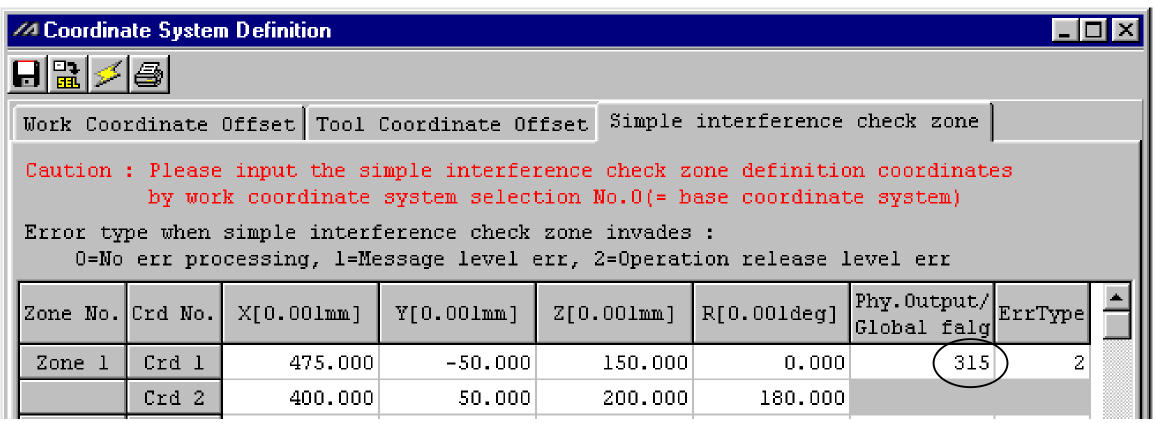
- SOIF (Specify output for simple interference check zone)

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	SOIF	Interference check zone number	Output/global flag number	CP

[Function] Set the output number/global flag number specified in operand 2 as the output to be turned on upon entry into the simple interference check zone specified in operand 1.

(Note 1) Duplicate specifications of physical output numbers or global flag numbers will cause chattering and the result will become indeterminable.

[Example] SOIF 1 315



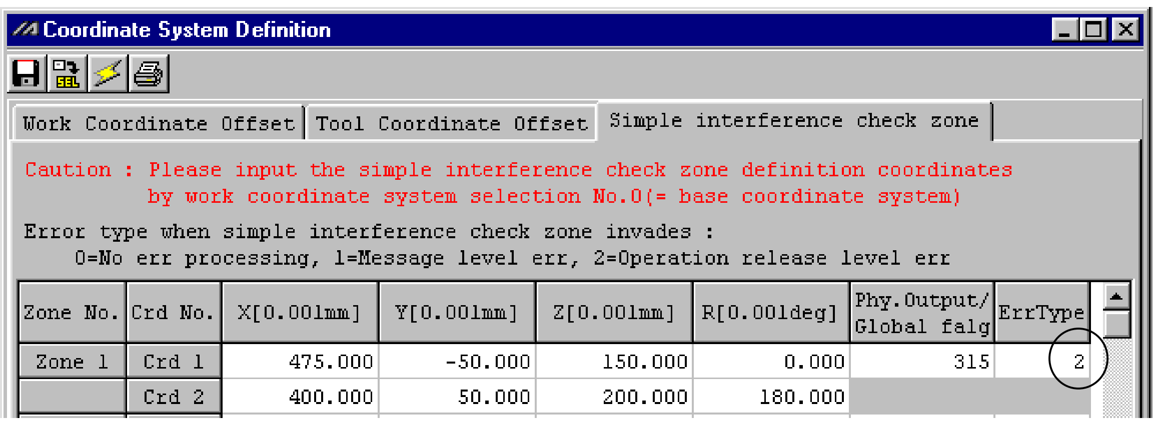
● SEIF (Specify error type for simple interference check zone)

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	SEIF	Interference check zone number	0 or 1 or 2 (Error type)	CP

[Function] Set the error type specified in operand 2 (see below) as the type of error generated upon entry into the simple interference check zone specified in operand 1.

Type of error generated upon entry into the simple interference check zone
0: No error
1: Message level error
2: Operation-cancellation level error

[Example 1] SEIF 1 2



- GTIF (Get definition coordinates of simple interference check zone)

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	GTIF	Interference check zone number	Position number (Consecutive two positions will be used)	CP

[Function] Set the definition coordinate data for the simple interference check zone specified in operand 1 in the consecutive two position data starting from the position number specified in operand 2.
Definition coordinates 1 of the simple interference check zone will be set in the position data specified in operand 2, while definition coordinates 2 will be set in the next position data. The coordinate data in the position data will include the specified definition coordinate data for simple interference check zone after all axes are set invalid.

(Note 1) The definition coordinates of simple interference check zone are always treated as data on the base coordinate system (work coordinate system No. 0). Therefore, position data set via a GTIF command must be handled on the base coordinate system.

(Note 2) GRP commands are invalid with respect to this command.

1-12 Actuator Control Command

- SVXX (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	SVXX	Axis pattern	Prohibited	PE

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

SVXX
- ON .... Turn ON the servo.
- OF .... Turn OFF the servo.

The arm system is set in local variable No. 99 upon successful completion of SVON.
Right arm system = 1
Left arm system = -1
Indeterminable = 0
Judgment is made on the basis of the angle of arm 2 after the arm 2 servo is turned ON.
This command sets the arm system immediately after servo ON and will not monitor the arm system continuously.

[Example 1] SVON 1100 Turn ON the servos of axes 3 and 4. Nothing will occur if the axis servos are already ON.

- MOVP (Move by specifying position data in PTP operation)

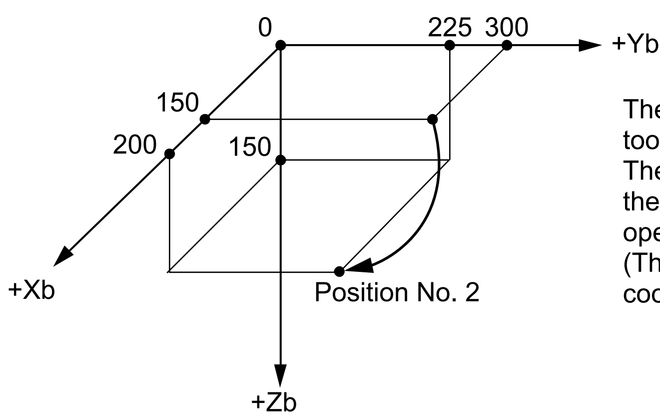
Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	MOVP	Position number	Prohibited	PE

[Function] Move the actuator in PTP mode to the position 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.

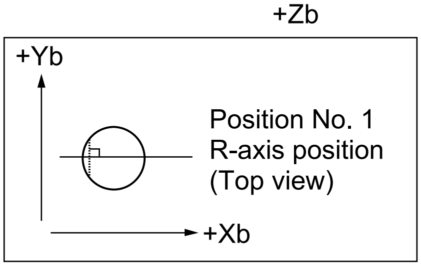
[Example 1] MOVP 2 Move the axes to the position corresponding to position No. 2 (200, 225, 150, 30).

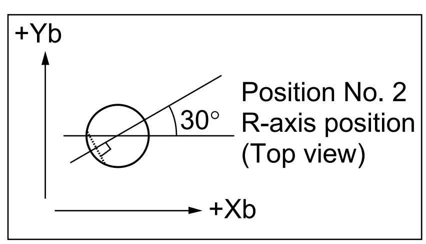
No.(Name)	Axis1	Axis2	Axis3	Axis4	Vel	Acc	Dcl
1( )	150.000	300.000	0.000	0.000			
2( )	200.000	225.000	150.000	30.000			
3( )							
4( )							

Travel path from position No. 1 to position No. 2



The center of the tool-mounting surface or the tool tip will move in PTP mode (not linearly). The movement locus will vary depending on the starting position and end position of operation, arm system, etc.
(The figure at left shows positions on the base coordinate system.)





- MOVL (Move by specifying position data in CP operation)

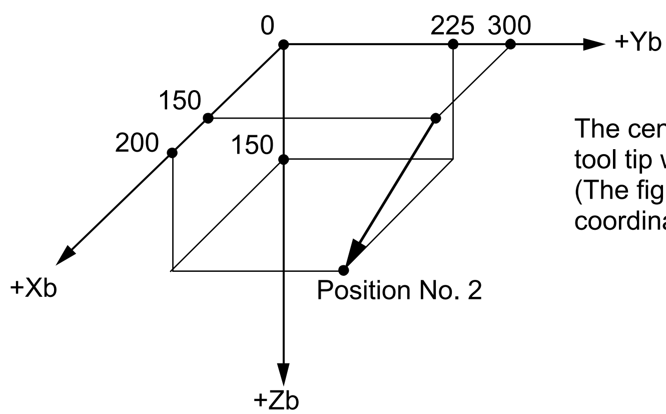
Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	MOVL	Position number	Prohibited	PE

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

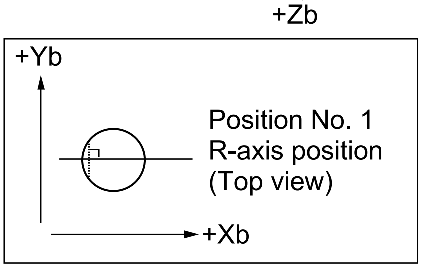
[Example 1] MOVL 2 Move the axes to the position corresponding to position No. 2 (200, 225, 150, 30), with interpolation.

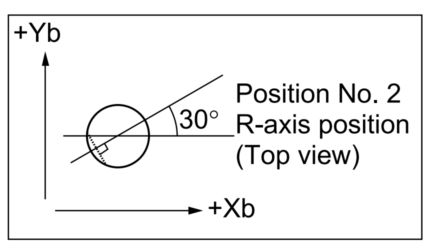
Travel path from position No. 1 to position No. 2

No.(Name)	Axis1	Axis2	Axis3	Axis4	Vel	Acc	Dcl
1( )	150.000	300.000	0.000	0.000			
2( )	200.000	225.000	150.000	30.000			
3( )							
4( )							



The center of the tool-mounting surface or the tool tip will move linearly.
(The figure at left shows positions on the base coordinate system.)





- MVPI (Move incrementally in PTP 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	MVPI	Position number	Prohibited	PE

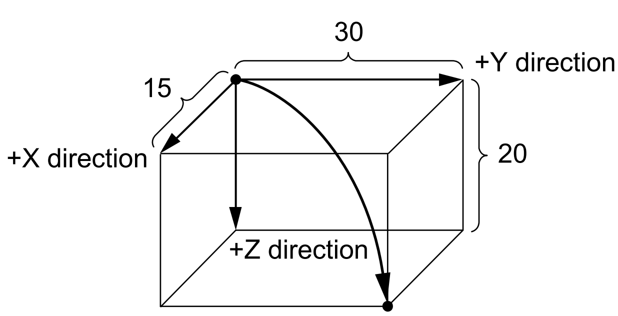
[Function] Move the actuator in PTP mode from the current position by the travel distance corresponding to the position number specified in operand 1.
The output will turn OFF at the start of axis movement, and turn ON when the movement is complete.

(Note) Repeated use of incremental (relative) movement commands (MVPI, MVLI, TMPI and TMLI) will accumulate coordinate-conversion rounding errors. To eliminate these errors, execute an absolute movement command (MOVP, MOVL, etc.).

[Example 1] MVPI 6 Each axis will move from the current position by the travel amount specified in position No. 6.
If the current positions are (200, 150, 50, 45) as specified in position No. 5 and the travel amounts are (15, 30, 20, 30) as specified in position No. 6, the positions after movement will become (215, 180, 70, 75).

No.(Name)	Axis1	Axis2	Axis3	Axis4	Vel	Acc	Dcl
5( )	200.000	150.000	50.000	45.000			
6( )	15.000	30.000	20.000	30.000			
7( )							

Travel path from position No. 5 by the travel distance corresponding to position No. 6



The center of the tool-mounting surface or the tool tip will move in PTP mode.
The movement locus will vary depending on the starting position and end position of operation, arm system, etc.

- MVLI (Move via incremental interpolation in CP 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	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.

(Note) Repeated use of incremental (relative) movement commands (MVPI, MVLI, TMPI and TMLI) will accumulate coordinate-conversion rounding errors. To eliminate these errors, execute an absolute movement command (MOVP, MOVL, etc.).

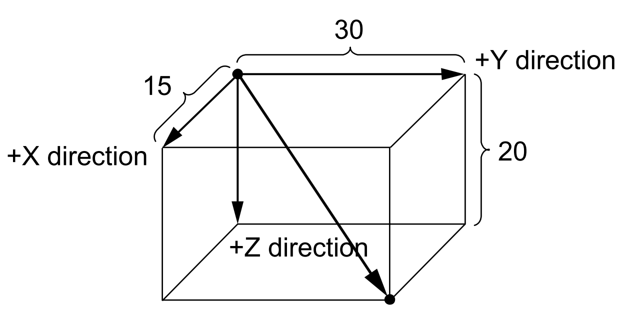
[Example 1] MVLI 6

Each axis will move from the current position by the travel amount specified in position No. 6.
If the current positions are (200, 150, 50, 45) as specified in position No. 5 and the travel amounts are (15, 30, 20, 30) as specified in position No. 6, the positions after movement will become (215, 180, 70, 75).

No.(Name)	Axis1	Axis2	Axis3	Axis4	Vel	Acc	Dcl
5( )	200.000	150.000	50.000	45.000			
6( )	15.000	30.000	20.000	30.000			
7( )							
8( )							

Travel path from position No. 5 by the travel distance corresponding to position No. 6

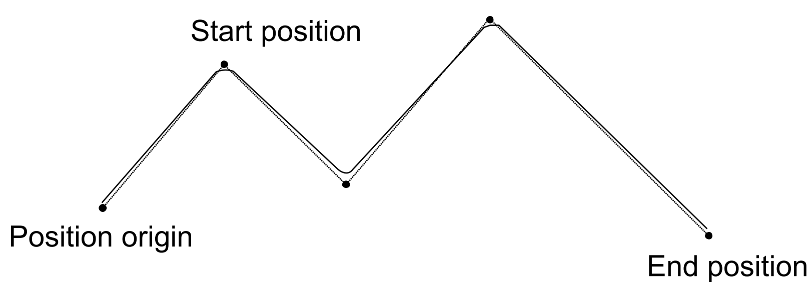
The center of the tool-mounting surface or the tool tip will move linearly.



- PATH (Move along path in CP 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	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.)

[Example 1] PATH 100 120 Move continuously from position Nos. 100 to 120.

- 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.

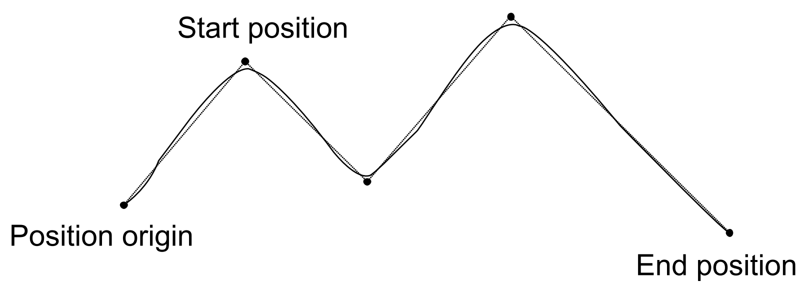
(Note 2) A STOP command only issues a deceleration-stop command (operation stop) to a specified axis pattern and does not wait for stopping to complete. Issuing other servo commands to a decelerating axis will either become invalid or generate an “axis duplication error,” etc.
Set a timer, etc., in the 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 is issued to an axis currently stopped, provide a minimum interval of 0.1 second before the next servo command is issued.

[Example 1] STOP 1100 Decelerate and stop the Z and R-axes.

- PSPL (Move along spline in CP 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	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.

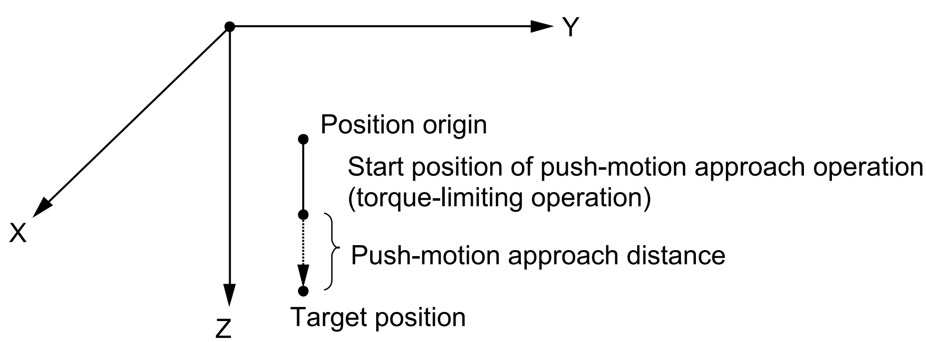
In this case, input in operand 1 the point number of the next target, instead of the predicted current position upon execution of the applicable command.
(Inputting a point number corresponding to the predicted current position will trigger movement to the same point during continuous movement, thereby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

[Example] PSPL 100 120 Continuously move from position Nos. 100 to 120 along a spline-interpolation curve.

● PUSH (Move by push motion in CP 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	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-card parameter No. 33, Push torque limit at positioning" (default value: 70%).

(Note 1) A PUSH command only moves the Z-axis. If multiple axes are specified, an "Error No. C91, Multiple push-axes specification error" will generate.

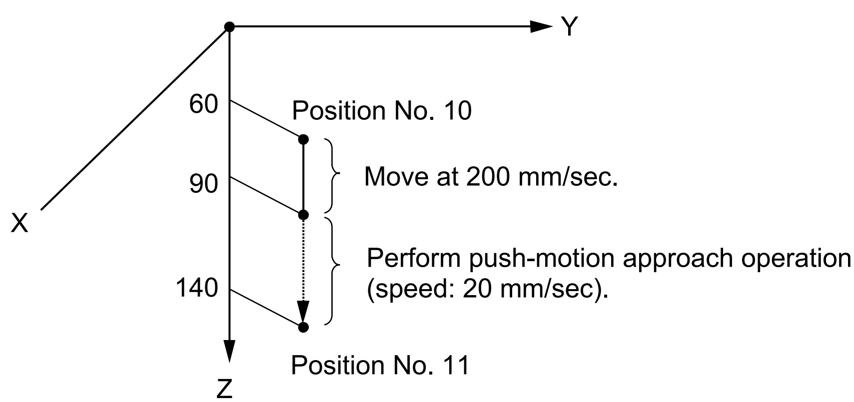
(Note 2) A push-motion approach speed exceeding the maximum speed permitted by the system will be clamped at the maximum speed. (The maximum system speed is not the maximum practical speed. Determine a practical speed by considering the impact upon contact, etc.)

[Example]	PAPR	50	20
	MOVP	10	
	PUSH	11	

Set the push-motion approach distance to 50 mm and push-motion approach speed to 20 mm/sec.
Move from the current position to position No. 10.
Perform push-motion movement from position Nos. 10 to 11.

The diagram below describes a push-motion movement based on the position data shown in the table below:

No.(Name)	Axis1	Axis2	Axis3	Axis4	Vel	Acc	Dcl
10( )	250.000	100.000	60.000	0.000	200	0.80	0.80
11( )			140.000				
12( )							



- CIR2 (Move along circle 2 (CP operation))

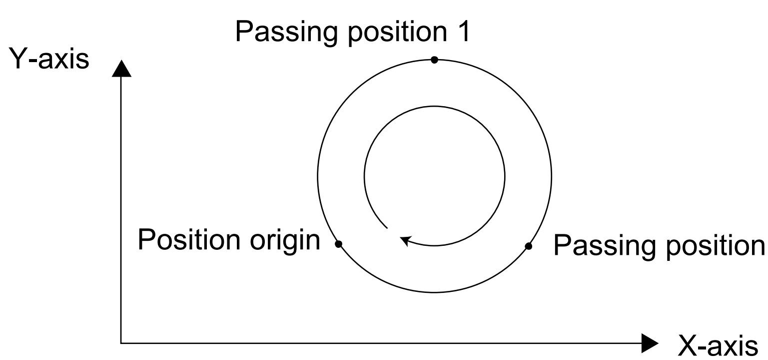
Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	CIR2	Passing position 1 number	Passing position 2 number	PE

[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:

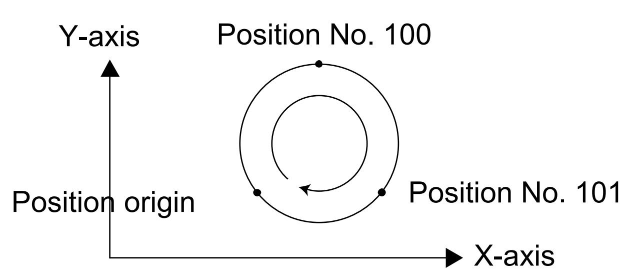
Priority	Speed	Acceleration (deceleration)
1	Setting in the position data specified in operand 1	Setting in the position data specified in operand 1
2	Setting by VEL command	Setting by ACC (DCL) command
3		Default CP acceleration in all-axis parameter No. 11 (Default CP deceleration in all-axis parameter No. 12)

If neither speed is set, a "C88 speed specification error" will generate.
If neither acceleration/deceleration is valid, a "C89 acceleration/deceleration specification error" will generate.



(Note) This command is valid only on the XY plane.

[Example] CIR2 100 101 Move along a circle (circular interpolation) passing position Nos. 100 and 101.



- ARC2 (Move along arc 2 (CP operation))

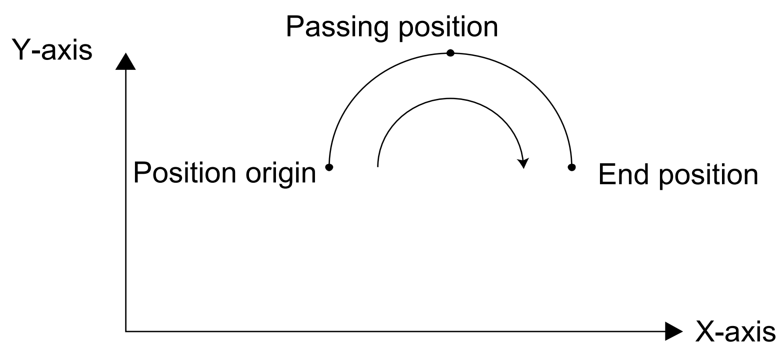
Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	ARC2	Passing position number	End position number	PE

[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:

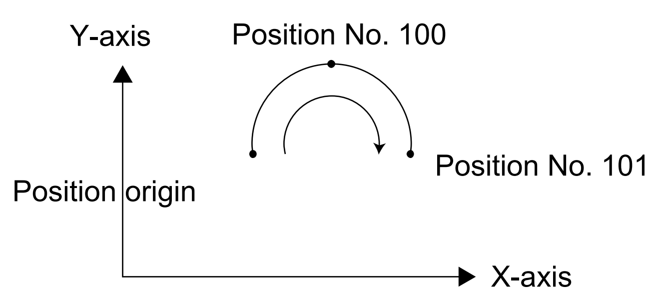
Priority	Speed	Acceleration (deceleration)
1	Setting in the position data specified in operand 1	Setting in the position data specified in operand 1
2	Setting by VEL command	Setting by ACC (DCL) command
3		Default CP acceleration in all-axis parameter No. 11 (Default CP deceleration in all-axis parameter 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 only on the XY plane.

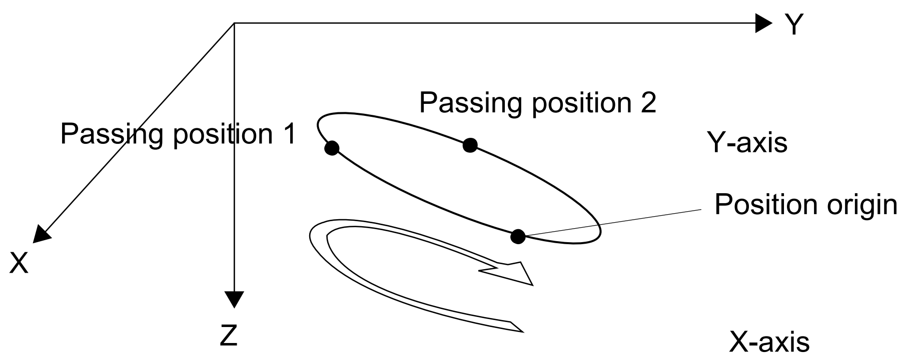
[Example] ARC2 100 101 Move along an arc (circular interpolation) from the current position to position No. 101 by passing position No. 100.



- CIRS (Move three-dimensionally along circle (CP 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	CIRS	Passing position 1 number	Passing position 2 number	PE

[Function] Move along a circle (three-dimensional movement) originating from the current position and passing positions 1 and 2 sequentially.
The rotating direction of the circle is determined by the given position data.
The movement in the diagram below will be performed in the reverse direction if passing positions 1 and 2 are reversed.



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	Same as the valid acceleration value
2	Setting by VEL command	Setting by ACC command	
3		Default CP acceleration in all-axis parameter No. 11	

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 1) This command is valid on arbitrary planes in a three-dimensional space. (Axis 2 (if there are only two valid axes) or axis 3 may be selected automatically prior to axis 1 in accordance with the position data.)

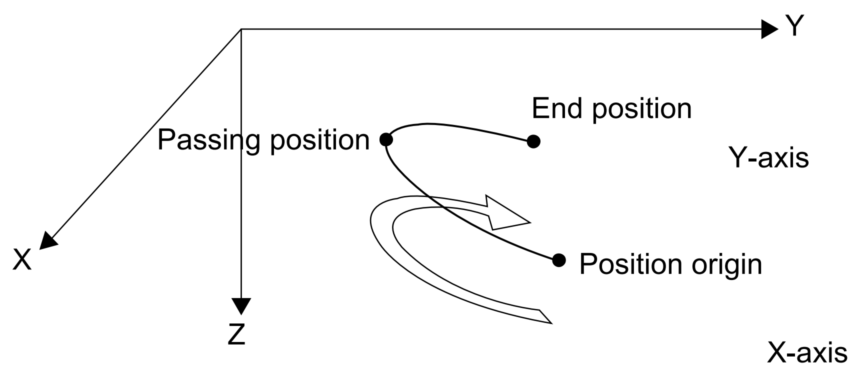
(Note 2) The locus tends to shift inward as the speed increases. Minor adjustment, such as setting the position data slightly outward, may be required.

(Note 3) If the circle diameter is small with respect to the set speed, the speed may be limited.

- ARCS (Move three-dimensionally along arc (CP 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	ARCS	Passing position number	End position number	PE

[Function] Move along an arc (three-dimensional movement) originating from the current position, passing the specified position and terminating at the end position.



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	Same as the valid acceleration value
2	Setting by VEL command	Setting by ACC command	
3		Default CP acceleration in all-axis parameter No. 11	

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 1) This command is valid on arbitrary planes in a three-dimensional space. (Axis 2 (if there are only two valid axes) or axis 3 may be selected automatically prior to axis 1 in accordance with the position data.)

(Note 2) The locus tends to shift inward as the speed increases. Minor adjustment, such as setting the position data slightly outward, may be required.

(Note 3) If the arc diameter is small with respect to the set speed, the speed may be limited.

- ARCD (Move along arc via specification of end position and center angle (CP operation))

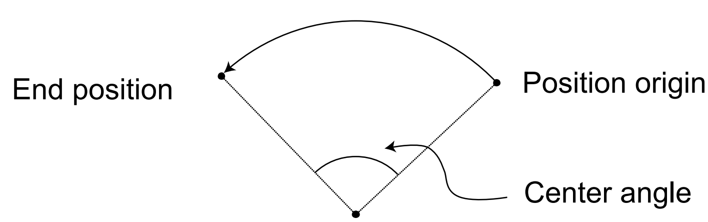
Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	ARCD	End position number	Center angle	PE

[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. 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 CP acceleration in all-axis parameter No. 11 (Default CP 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 only on the XY plane.

[Example] 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 (CP operation))

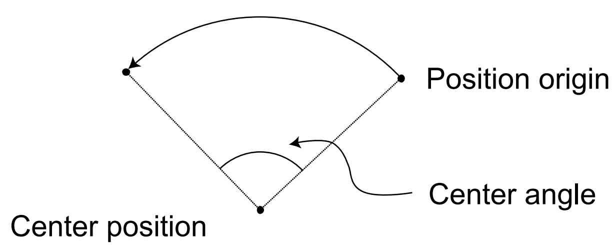
Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	ARCC	Center position number	Center angle	PE

[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). 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 CP acceleration in all-axis parameter No. 11 (Default CP 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.



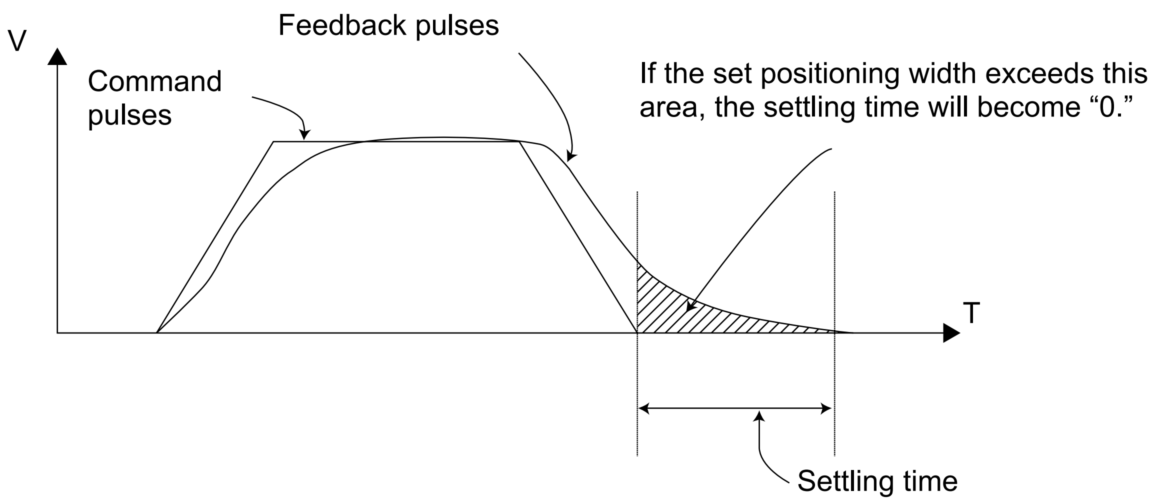
(Note) This command is valid only on the XY plane.

[Example] 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 width)

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 positioning completion width for the axes in the axis pattern specified in operand 1. The distance in operand 2 is set in degrees for the X, Y and R-axes and in mm for the Z-axis. As a rule, positioning is deemed complete when all command pulses have been output and the current position is inside the positioning width. 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° will have effect, but the effect must be confirmed on the actual machine.)



(Note 1) If positioning width is not set with a PBND command, the value set in “Axis-specific parameter No. 58, Positioning width” will be used.

(Note 2) If the positioning width is changed, the new setting will remain valid even after the program ends. Therefore, to build a system using PBND commands, a positioning width must be expressly specified with a PBND command before operation of each program. An assumption that the positioning width will be reset to the original value when the operation ends in other program may lead to an unexpected problem, because the positioning width 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 width” will not be written by a PBND command.

[Example] PBND 11 5 Set the positioning width for the X and Y-axes to 5° after this command.

- TMPI (Move relatively between positions on tool coordinate system)

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	TMPI	Position number	Prohibited	PE

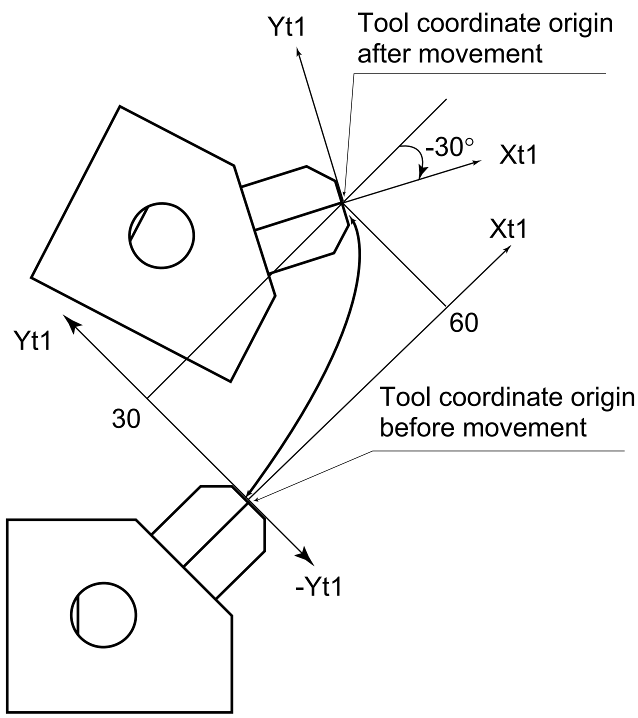
[Function] Each axis will move relatively on the tool coordinate system without interpolation (= PTP operation) based on the position data specified in operand 1 setting the travel amount from the current position.

(Note 1) Repeated use of incremental (relative) movement commands will accumulate coordinate-conversion rounding errors.

[Example] TMPI 120

Position data

No.	Axis1	Axis2	Axis3	Axis4	Vel	Acc	Dcl
120	60.000	30.000	0.000	-30.000			
121							
122							



- TMLI (Move relatively between positions on tool coordinate system with 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	TMLI	Position number	Prohibited	PE

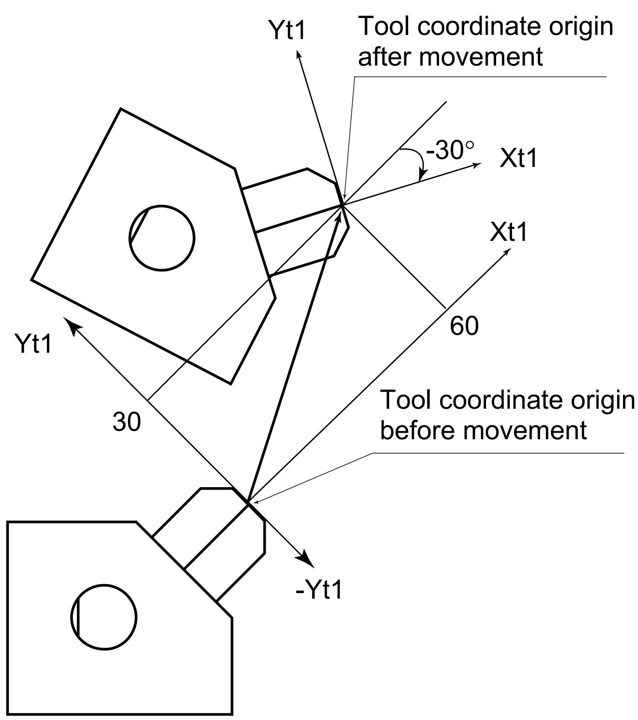
[Function] Each axis will move relatively on the tool coordinate system with interpolation (= CP operation) based on the position data specified in operand 1 setting the travel amount from the current position.

(Note 1) Repeated use of incremental (relative) movement commands will accumulate coordinate-conversion rounding errors.

[Example] TMLI 120

Position data

No.	Axis1	Axis2	Axis3	Axis4	Vel	Acc	Dcl
120	60.000	30.000	0.000	-30.000			
121							
122							



- 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 for the axis pattern specified in operand 1 (axis pattern can be specified only for the Z-axis) to the value specified in operand 2. The ratio in operand 2 is set as an integer (unit: %).
"Driver-card parameter No. 33: Push torque limit at positioning" can be rewritten temporarily using a PTRQ command.

(Note 1) If push torque limit is not set with a PTRQ command, the value set in "Driver-card parameter No. 33: Push torque limit at positioning" will be used.

(Note 2) If the push torque limit is changed, the new setting will remain valid even after the program ends. Therefore, to build a system using PTRQ commands, a push torque limit must be expressly specified with a PTRQ command before operation of each program. An assumption that the push torque limit will be reset to the original value when the operation ends in other program may lead to an unexpected problem, because the push torque limit will become different from what is anticipated in case the applicable program is aborted due to error, etc.

(Note 3) The new value set with a PTRQ command will become invalid after a power-ON reset or software reset.

(Note 4) The value set in "Driver-card parameter No. 33: Push torque limit at positioning" (inside the driver EEPROM (non-volatile memory) will not be written by a PTRQ command.

[Example] PTRQ 3 50 Change the push torque limit parameter for the Z-axis to 50%.

- CIR (Move along circle (CP 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	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 CP 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 the XY plane.

[Example 1]	CIR	100	101	Move along a circle from the current position by passing positions 100 and 101 sequentially.
[Example 2]	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 (CP 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	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 CP 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 the XY plane.

[Example 1]	ARC	100	101	Move along an arc from the current position to position 101 by passing position 100.
[Example 2]	LET	1	5	Assign 5 to variable 1.
	LET	2	6	Assign 6 to variable 2.
	ARC	*1	*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

- IFXX (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	IFXX	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 IFXX command is not executed, the program will proceed to the step next to the corresponding EDIF.
A maximum of 15 nests are supported when ISXX and DWXX are combined.

IFXX

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]

	SVON	1111			Set the current arm system in variable 99.
	PRDQ	1	100		Read the current X coordinate into variable 100.
	CPNE	99	0	600	If the arm system is indeterminable, the arm system whose flag 600 is turned OFF will be determined. If this arm system is also indeterminable, the operation will end.
600	IFEQ	99	1		
	IFGE	100	0		If the X coordinate is 0 or greater:
	MOVP	1			Move the axis to position No. 1 in PTP mode.
	ELSE				
	MOVP	2			Move the axis to position No. 2 in PTP mode.
	EDIF				
	ELSE				
	IFGE	100	0		If the X coordinate is 0 or greater:
	MOVP	3			Move the axis to position No. 3 in PTP mode.
	ELSE				
	MOVP	4			Move the axis to position No. 4 in PTP mode.
	EDIF				
	EDIF				
	EXIT				

If the current arm system is the right arm and the X coordinate is 0 or greater, the axis will move to position No. 1; if the X coordinate is below 0, the axis will move to position No. 2. If the current arm system is the left arm and the X coordinate is 0 or greater, the axis will move to position No. 3; if the X coordinate is below 0, the axis will move to position No. 4.

(Note) Using a GOTO command to branch out of or into an IFXX-EDIF syntax is prohibited.

- ISXX (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	ISXX	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 ISXX command is not executed, the program will proceed to the step next to the EDIF.
A maximum of 15 nests are supported when IFXX and DWXX are combined.

```
ISXX
  ├ EQ  .............  Operand 1 = Operand 2
  └ NE  .............  Operand 1 ≠ Operand 2
```

[Example 1]

```
            SCPY   10   'GOFD' (Move forward)
            SCPY   14   'GOBK' (Move backward)
            SLEN   4                          Set the number of comparing characters to 4.
┌─ 600      ISEQ   1    "AXSX" (X-axis)       Select an axis.
│       ┌── ISEQ   5    10                    Select a moving direction.
│       │   MOVL   1                          Move the axis to position No. 1 in CP mode.
│       ├── ELSE
│       │   MOVL   2    5                     Move the axis to position No. 2 in CP mode.
│       └── EDIF
├────────── ELSE
│       ┌── ISNE   5    14                    Select a moving direction.
│       │   MOVL   3                          Move the axis to position No. 3 in CP mode.
│       ├── ELSE
│       │   MOVL   4                          Move the axis to position No. 4 in CP mode.
│       └── EDIF
└────────── EDIF
```

Move in CP mode by selecting position Nos. 1 and 2 by columns 1 to 4 and position Nos. 3 and 4 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, the axis will be moved to position No. 1 in CP mode.

1 2	3 4	5 6	7 8	
AX	SX	GO	FD	

(Note) Using a GOTO command to branch out of or into an ISXX-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 IFXX or ISXX command to declare the command part to be executed when the condition is not satisfied.

[Example 1] Refer to the sections on IFXX and ISXX.

- EDIF (End IFXX)

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 IFXX or ISXX command.

[Example 1] Refer to the sections on IFXX and ISXX.

1-14 Structural DO

- DWXX (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	DWXX	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 DWXX command is not executed, the program will proceed to the step next to the corresponding EDDO.
A maximum of 15 nests are supported when IFXX and ISXX are combined.

DWXX

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 Repeat the command up to an EDDO command while variable 1 contains "0."

```
:
:
EDDO
```

If DWXX 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 DWXX-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 DOXX loop and proceed to the step next to EDDO.

[Example 1] DWEQ 1 0 Repeat the commands up to an EDDO command while variable 1 contains '0.'

```
      :
600   LEAV      Forcibly end the loop if flag 600 is ON and proceed to
                the step next to an EDDO command.
      :
      EDDO
```

- 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 DOXX loop.

[Example 1]

```
      DWEQ   1   0    Repeat the commands up to an EDDO command while
                      variable 1 contains "0."
        :
600   ITER            Forcibly switch the control to an EDDO command and
        :             perform end judgment, if flag 600 is ON.
      EDDO
```

- 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 DWXX.
If the DWXX condition is not satisfied, the program will proceed to the step next to this command.

[Example 1] Refer to the section on DWXX.

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 WHXX or WSXX 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 WHXX, WSXX 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 WXXX 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.

- WHXX (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	
Optional	Optional	WHXX	Variable number	Data	CP

[Function] This command is used between SLCT and EDSL commands to execute the subsequent commands up to the next WXXX 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.

```
WHXX
  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
:			was 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 WXXX will become valid and any subsequent commands will not be executed. Therefore, state from the command with the most difficult condition or highest priority.

- WSXX (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	WSXX	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 WXXX 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.

```
WSXX
  ├─ 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 command will not be executed.
:			
(1)			
:			
WSEQ	2	*1	(2) will be executed if the content of the number of characters specified by SLEN after column 2 is the same as the content of the column specified in variable 1.
:			
(2)			
:			
OTHE			This command will be executed if none of the conditions are satisfied. In this example, since (2) was executed, (3) will not be executed.
:			
(3)			
:			
EDSL			The processing will move here if any of the conditions was satisfied and the applicable command executed. In this example, (2) and (4) will be executed.
:			
(4)			
:			

* If multiple conditions are likely to be satisfied, remember that the first WXXX 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, WHXX and WSXX.

- 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, WHXX and WSXX.

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

$3188 = 12\ (= C) \times 16^2 + 7 \times 16 + 4$
$= C74$ (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.

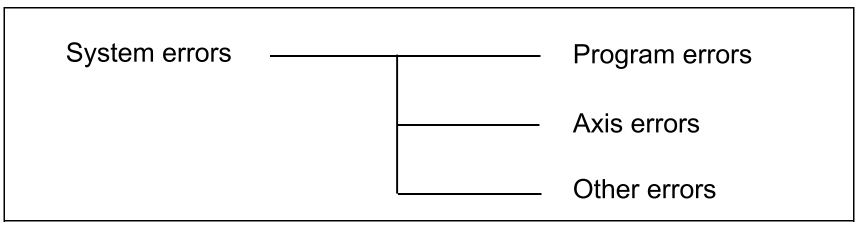
[Example] PGST 1 2 Read the error number for program No. 2 to variable 1.

- 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.

- GARM (Get current arm system)

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	GARM	Variable number	Prohibited	CP

[Function] Obtain the current arm system and set in the variable specified in operand 1 one of the following values corresponding to this arm system:
Arm system is indeterminable = 0
Right arm system = 1
Left arm system = -1

(Note 1) This command sets the arm system immediately after command execution. The arm system will not be monitored continuously.

[Example] GARM 200 Set "1" in variable No. 200 if the current arm system is the right arm system, or "-1" if the current arm system is the left arm system.

1-17 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    1
        Specify 10 (= LF) as the end character.
        Open channel 1.
```

> Note: If "OPEN 1" is executed, the teaching-pendant connector (D-sub, 25-pin) will be disconnected. (This is because channel 1 is shared by the teaching pendant/PC software.)

- 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    1
Close channel 1.

LET     1       2
CLOS    *1
        Assign 2 to variable 1.
        Close the content of variable 1 (channel 2).
```

- 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.

Specifying “0” in operand 2 will trigger a dummy read (clear the receive buffer and disable reception) (the return code will be "successful completion").

[Example]

SCHA	10		Set LF (= 10) as the end character.
OPEN	1		Open channel 1.
READ	1	2	Read a character string from channel 1 to column 2 until LF is received.
CLOS	1		Close the channel.

- Return code of the READ command
  The return code is stored in a local variable. Variable number can be set by “Other parameter No. 24.” The default variable number is 99.
  The variable number is fixed to 99 in main application version 0.20 and earlier.
  0: READ completed successfully (Receive complete)
  1: READ timeout (the timeout value is set by a TMRD command) (Continue to receive)
  2: READ cancelled due to timer (the waiting status was 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 expanded-SIO overrun error (Receive disabled)
  9: READ expanded-SIO parity error (Receive disabled)
  10: READ expanded-SIO framing error (Receive disabled)
  11: READ expanded-SIO buffer overflow error (Receive disabled)
  12: READ expanded-SIO receive error due to other factor (Receive disabled)

- TMRD (Set READ 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	TMRD	Timer setting	Prohibited	CP

[Function] Set the timeout to be applied to a READ command.
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 a local variable immediately after the READ command is executed.)
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.

(Note) TMRD is set to “0” in the default condition before TMRD setting is performed.

[Example]

SCHA	10		Set LF (=10) as the end character.
TMRD	30		Set the READ timeout value to 30 seconds.
OPEN	1		Open channel 1.
READ	1	2	Read the character string from channel 1 to column 2 until LF is read.
TRAN	1	99	Assign the return code to variable 1.
CLOS	1		Close the channel.

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	CP

[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	1		Open channel 1.
WRIT	1	2	Write the character string in column 2 to channel 1 until LF is written.
CLOS	1		Close the channel.

(Note 1) The “Forcibly enable receive immediately before send” function is available when expanded SIOs (channel 2 and onward; optional) are used via RS232C or RS422 (this function can be cancelled via parameter). If this function is used, the X-SEL can receive a response from the other side without delay after a send.

(Note 2) If expanded SIOs (channel 2 and onward; optional) are used via RS232C or RS422, once OPEN is executed those tasks other than the open ones can also be used to execute WRIT (send). Therefore, the X-SEL can receive a response from the other side without delay after a send, by executing READ with an open task and then WRIT with other task.

(Note 3) If expanded SIOs (channel 2 and onward; optional) are used via RS485, or when standard SIOs (channel 1; also used by the teaching pendant) are used, the “Forcibly enable receive immediately before send” function is available. However, using this function will generate a delay time (the function can be cancelled via parameter). A minimum delay of 3 msec is required after the X-SEL executes a send before it receives a response from the other side.

● 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.

1-18 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
			E	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:

1-19 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.

(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-point 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 for the PX-axis (preferential axis), PY-axis and PZ-axis (when palletizing Z-axis declaration is valid) for use in palletizing calculation.
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, the reference point will be set to X = 0 and Y = 0.
Palletizing positions are calculated as points on the palletizing plane consisting of the reference point, PX-axis and PY-axis. Therefore, the position data defining the reference point must include valid coordinate components for the PX-axis, PY-axis and PZ-axis (when palletizing Z-axis declaration is valid). If the coordinate components for these axes are invalid, an error will generate during palletizing-position coordinate calculation accompanying a PAPG command (get palletizing calculation data) or any palletizing movement command. The coordinate components for other axes will be ignored during palletizing-position coordinate calculation.

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.

(Note 1) If this command is not set while work coordinate system No. 0 (base coordinate system) is selected, executing a palletizing movement command will generate an error because the palletizing start point becomes (0, 0) and the axes are unable to move.

(Note 2) If the R-axis is set in the position data, exclude the R-axis from the valid axes using a GRP command. (This is not necessary if the R-axis column is empty.)
The R-axis data of a given palletizing position is set using a PEXT command.

- 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	Prohibited	CP

Specify the first position number among a series of position numbers that contain the point data for three points specifying the reference point, PX-axis (preferential axis) and PY-axis for use in palletizing calculation.
If “n” is set as the position number in operand 1, point n will represent the reference point, point n+1 will represent the end point in the PX-axis direction and point n+2 will represent the end point in the PY-axis direction.

If a PAPS (set palletizing points) command is executed after specifying the axes to be used with a GRP command, the portions applicable to the palletizing axes in the above position data of n, n+1 and n+2 will be used as the palletizing position data. Even if a GRP command is executed in other setting thereafter, no effects will be felt.

As for the point data for 3-point teaching specifying the reference point, end point in the PX-axis direction and end point in the PY-axis direction, there must be two or three valid axes and the valid axis pattern must match. If the valid axis pattern does not match, an error “CB0, Mismatched valid axes and palletizing 3-point teaching data” will generate.
If a PZ-axis (palletizing Z-axis) is already declared, there must be two valid axes excluding the PZ-axis. If there are not enough valid axes, an error “CAE, Insufficient valid axes for palletizing 3-point teaching data” will generate. If there are too many valid axes, an error “CAF, Excessive valid axes for palletizing 3-point teaching data” will generate.
This command cannot be used with PASE (set palletizing axes). Whichever is set later will be given priority.
A single PAPS command can substitute PASE, PAPT and PAST.
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.

(Note 1) If R-axis data is set in position No. n, n+1 or n+2, exclude the R-axis from the valid axes using a GRP command. (This is not necessary if the R-axis data column is empty.) The R-axis data of a given palletizing position is set using a PEXT command.

- 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.

- PCHZ (Declare palletizing 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	PCHZ	(Axis number)	Prohibited	CP

Specify the axis number representing the palletizing Z direction.
The axis number specified in operand 1 will be set as the axis number representing the palletizing Z direction.
If operand 1 is not specified, the specification of palletizing Z-axis that was already declared will become invalid.
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.

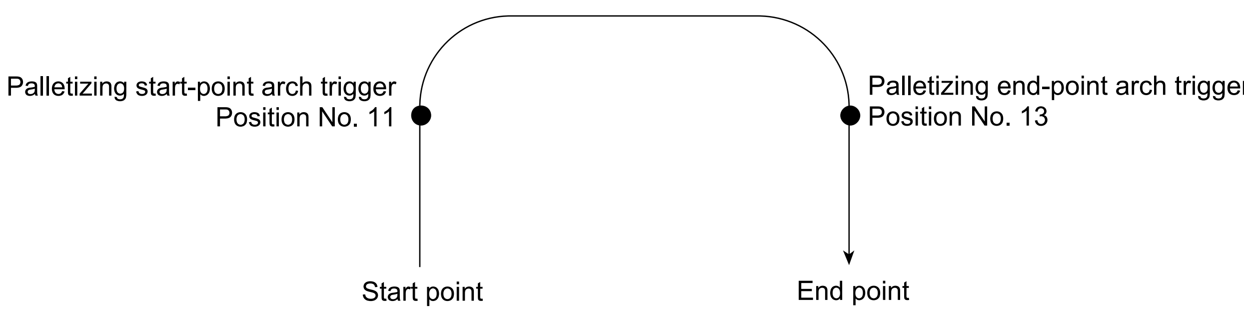
(Note 1) The palletizing Z-axis can be specified only as the Z-axis on the work coordinate system (axis No. 3).

[Example] PCHZ 3

- PTRG (Set palletizing 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	PTRG	Position number	Position number	CP

Set the arch triggers to be used for arch motion along the palletizing points.
(This setting becomes valid when a PACH command is executed.)
Set the PZ-axis (palletizing Z-axis) position data in the point data specified in operand 1 as the palletizing start-point arch trigger, and set the PZ-axis position data in the point data specified in operand 2 as the palletizing end-point arch trigger.



PTRG 11 13

(Refer to “Palletizing Setting” – “Palletizing arch triggers” under "How to Use.")
As for the point data, the PZ-axis data specified by a PCHZ command must be valid.
For an arch-motion operation along the palletizing points, 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 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.

- PEXT (Set palletizing composition)

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	PEXT	(Position number)	Prohibited	CP

Set palletizing composition.
The position number specified in operand 1 will be set for use in composition.
The R-axis coordinate of a given palletizing position is set using this command.
When a palletizing movement command is executed, the data of any valid axes other than the PX, PY (and PZ)-axes in the specified point data will comprise the end-point coordinates of the composite axis.
If operand 1 is not specified, the position number for composition setting that was already declared will become invalid.
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.

- OFPZ (Set palletizing 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	OFPZ	Offset value	Prohibited	CP

Set the offset in the palletizing Z-axis direction.
The value specified in operand 1 will be set as the offset in the PZ-axis (palletizing 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 PACH (palletizing-point arch motion) operation.
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.

- 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.

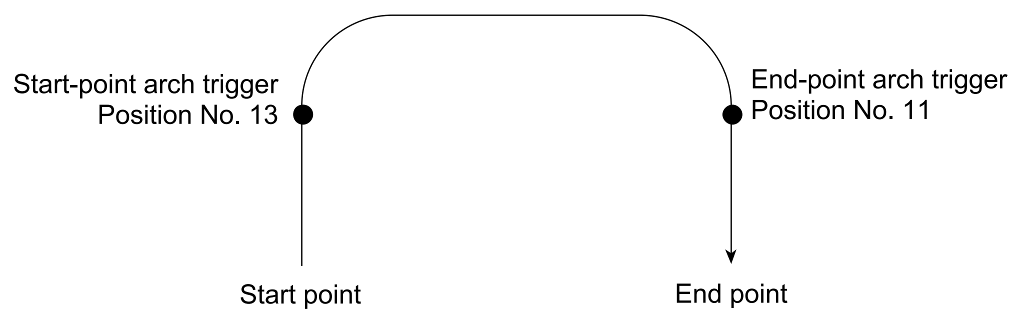
(Note 1) The arch-motion Z-axis can be specified only as the Z-axis on the work coordinate system (axis No. 3).

[Example] ACHZ 3

- 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.

- AEXT (Set arch-motion composition)

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	AEXT	(Position number)	Prohibited	CP

Set arch-motion composition.
The position number specified in operand 1 will be set for use in composition.
When an arch motion is executed, the data of valid axes in the point data specified in this command, except for the data of valid axes in the arch-motion end-point data as well as the arch-motion Z-axis data, will comprise the end-point coordinates of the composite axis.
If operand 1 is not specified, the position number for composition setting that was already declared will become invalid.
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 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 will 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 work coordinate system 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 work coordinate system 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.
The axes to be used can be specified with a GRP command before PAPS is executed (refer to the detailed explanation of PAPS). If the valid axis pattern of the 3-point teaching data does not match, an error “CB0, Mismatched valid axes and palletizing 3-point teaching data” will generate.
If the number of valid point-data axes (the number of valid axes excluding the PZ-axis, if a PZ-axis (palletizing Z-axis) has already been declared) is less than two, an error “CAE, Insufficient valid axes for palletizing 3-point teaching data” will generate. If the number of valid point-data axes is more than two, an error “CB9, PX/PY-axes indeterminable when obtaining palletizing angle” will generate.
If the axis number specified in operand 2 is neither of the two valid axes in the point data excluding the PZ-axis, an error “CBA, Reference axis and PX/PY-axes mismatch when obtaining palletizing angle” will generate.
If the reference point among the three teaching points is the same as the point data at the PX-axis end point other than the PZ-axis component, an error “Reference point and PX-axis end point identical when obtaining palletizing angle” will generate, and angle calculation will be disabled.
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.

1-22 Palletizing Movement Command

- PMVP (Move to palletizing points via PTP)

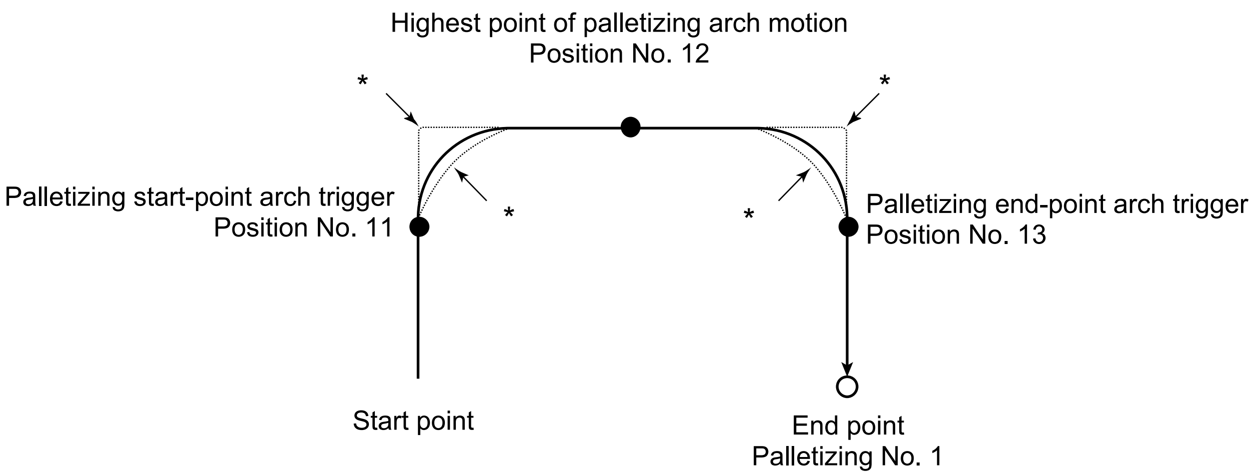
Extension condition (LD, A, O, AB, OB)	Input condition (I/O, flag)	Command, declaration			Output (Output, flag)
		Command, declaration	Operand 1	Operand 2	
Optional	Optional	PMVP	Palletizing number	(Position number)	PE

Move to the calculated palletizing points via PTP.
The axes will move to the palletizing points specified in operand 1, via PTP.
If the palletizing points are valid only for the PX/PY-axes (when a PZ-axis (palletizing Z-axis) is not specified, etc.), movement in directions other than the PX/PY-axis directions will not be performed. If the PZ-axis coordinates of the palletizing points are also valid, movement in the PZ-axis direction will also be performed.
However, if a position number is specified in operand 2, the Z-direction position will move to the height of the specified position number by ignoring the palletizing calculation.
Any data other than palletizing Z-axis data contained in the position number specified in operand 2 will be ignored. Absence of PZ-axis data will be handled as an error.
If palletizing composition is set, any axes other than the PX, PY (and PZ)-axes will also be operated if data is available for such axes.
Executing this command will not increment the palletizing position number by 1.
Before specifying operand 2, a palletizing Z-axis must have been declared (PCHZ) in the palletizing setting.
If palletizing Z-axis has not been declared, an error will generate.

- PACH (Palletizing-point 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	PACH	Palletizing number	Position number	PE

Perform arch motion from the current point and move to the palletizing points.
- Move to the palletizing points specified in operand 1, via arch motion.
- Movements in the PX/PY-axis directions will begin after rising from the current point to the palletizing start-point arch trigger. After the Z point specified in operand 2 (as the highest point) is passed and movements in the PX/PY-axis directions are complete, the axes will come down to the palletizing end-point arch trigger and reach the calculated palletizing point.
- Palletizing arch triggers must have been set using a PTRG command.




```
PCHZ    3
PTRG    11    13
  |
  |
  |
PACH    1     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 PZ-axis coordinate of the end point will become the PZ-axis component of the position coordinates of the palletizing point, if any, plus the palletizing Z-axis offset. If there is no PZ component, the PZ-axis coordinate of the end point will become the PZ-axis coordinate of the start point plus the palletizing Z-axis offset. (Normally the offset is added to all palletizing positions, such as the arch triggers and Z point.)
- An error will generate if the palletizing start-point arch trigger is set below the start point or the palletizing end-point arch trigger is set below the end point. (Note: Up/down has nothing to do with +/– on the coordinate system.)
- The PZ-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 PZ-axis will come down after a rise-process command value is output. Therefore, the operation may follow the locus shown below depending on the settings of palletizing arch-trigger points and Z point:

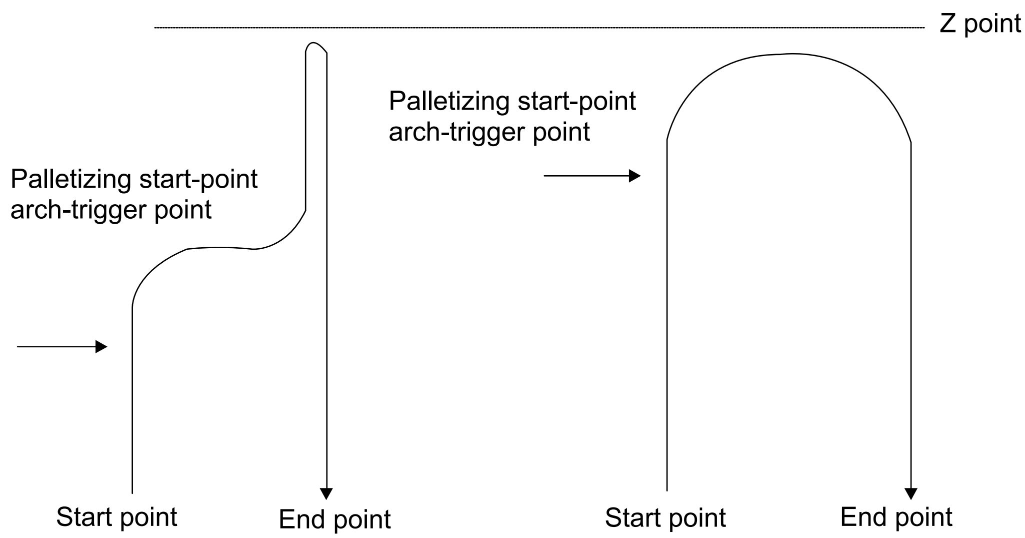


Fig. 5

In this case, change the palletizing arch triggers and PZ point to increase the operation efficiency.
- If palletizing composition (PEXT) is set, axes other than the PX, PY and PZ-axes will also be operated if data is available for such axes. However, the composite axis will start/end operation at positions above the arch triggers. If the R-axis is set with a PEXT command, R-axis operation will start and end above the arch triggers.

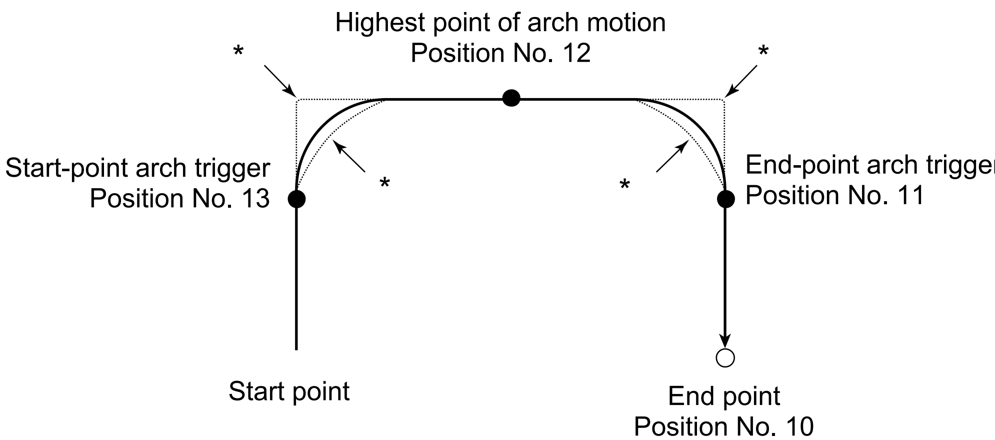
- Executing this command will not increment the palletizing position number by 1.

(Note 1) The PACH command executes PTP operation.

- 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   3
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, the operation may follow the locus in Fig. 5 given in the aforementioned explanation of PACH command, depending on the settings of arch-trigger points and Z point. In this case, change the arch triggers and Z point to increase the operation efficiency.

- As for the arch-trigger end-point data, if there is any valid axis data other than the data of the arch-motion Z-axis, then operation will be started/ended for the applicable axes in the same manner—but above the arch triggers.

- If R-axis data is included in the end-point data, R-axis operation will start and end above the arch triggers.

- If arch-trigger composition is set, any valid axes other than those set in the end-point data or the arch-motion Z-axis will also be operated as long as data is available for such axes. In this case, operation of the applicable axes will also be started/ended above the arch triggers.

(Note 1) The ARCH command executes PTP operation.

1-23 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.

Chapter 3 Key Characteristics of Horizontal Articulated Robot Operation

This chapter explains how to set the key characteristics of horizontal articulated robot operation, such as commands and operations, arm systems, various coordinate systems and simple interference check zones.

1. CP Operation and PTP Operation

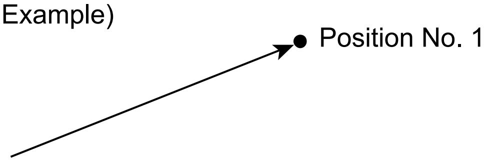
A horizontal articulated robot performs CP operation and PTP operation.

1-1 CP Operation

(1) Locus

The axes move to the target position via mutual interpolation. The locus of axis tip during movement can be specified using commands (linear, circular, arc, path movement, etc.).

Example)



MOVL 1
Move from the current position to position No. 1 linearly.

The arm system will not change during CP operation.
CP operation commands: MOVL, MVLI, TMLI, PATH, PSPL, PUSH, CIR2, ARC2, ARCD, ARCC, CIRS, ARCS, CIR, ARC
For details on these commands, refer to Chapter 2, "Explanation of Commands."

(2) Speed and acceleration/deceleration settings for CP operation

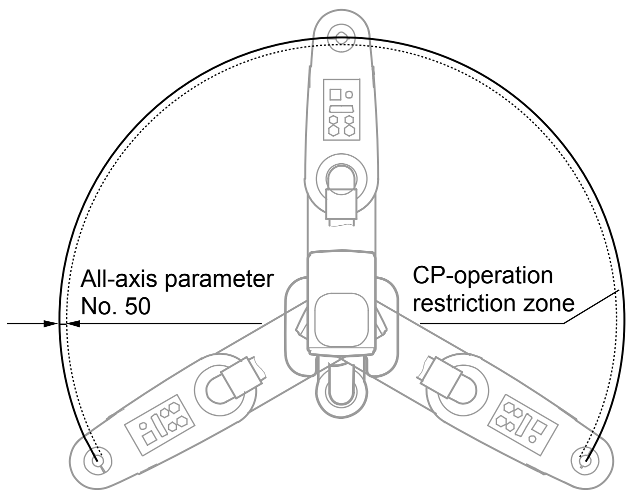
The speed and acceleration/deceleration for CP operation are predefined in a program using control declaration commands.
Speed setting command "VEL"; unit [mm/sec]
Acceleration setting command "ACC"; unit [G]
Deceleration setting command "DCL"; unit [G]

Example)

ACC	0.5	Set the acceleration for CP operation to 0.5 G.
DCL	0.5	Set the deceleration for CP operation to 0.5 G.
VEL	500	Set the speed for CP operation to 500 mm/sec.
MOVL	2	Move to position No. 2 linearly.

The speed and acceleration/deceleration for CP operation can also be set in the VEL, ACC and DCL columns of position data.
If the speed and acceleration/deceleration are set in position data, they must be set for each position number. If they are set in the VEL, ACC and DCL columns of a given position number, movement to that position number will be given priority over the commands "VEL," "ACC" and "DCL" in the program.

(3) Notes on CP operation
The singular point refers to a position where arms 1 and 2 form a straight line.
Performing CP operation along a path near the singular point may reduce locus accuracy, cause vibration (noise) or generate errors. The errors that may occur include the following:
"D09: Driver overspeed error," "B91: Main overspeed error," "C64: Invalid servo acceleration/deceleration error," "B74: CP-operation restriction zone entry error," and "C6B: Deviation overflow error"
These problems may be prevented by lowering the speed or acceleration/deceleration.



The CP-operation restriction zone is defined as the area between the singular point locus and the locus of the value set in all-axis parameter No. 50. CP operation is prohibited inside this area. (In the figure shown at left, the area between the solid line and dotted line is the CP-operation restriction zone.)

The controller will generate an error upon detecting that the target locus used in locus calculation or the actual movement locus has entered the CP-operation restriction zone. If the target movement locus has entered the CP-operation restriction zone during locus calculation, a "B7C: Error due to target locus inside CP-operation restriction zone (PTP/jogging of each axis enabled)" will generate.
When the actual movement locus has entered the CP-operation restriction zone, a "B74: CP-operation restriction zone entry error (PTP/jogging of each axis enabled)" or "C74: Actual-position soft limit over error" will generate.
The width of the CP-operation restriction zone (distance between the solid line and dotted line) will vary depending on the robot arm length. (If the arm length is 500/600, the restriction zone will become approx. 0.5 mm wide (All-axis parameter No. 50: Width of CP-operation restriction zone near arm 1/2 straight-line point)).

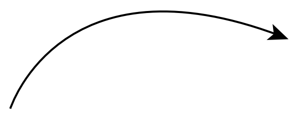
Avoid creating a program that will cause the axes to pass the CP-operation restriction zone during CP operation. Once inside, the axes cannot be pulled out of the CP-operation restriction zone via CP operation. Move the axes via PTP operation. Exercise caution when the arm condition is not recognized at the start of program, etc. As for CP operation, always perform test operation at low speed and confirm absence of problem beforehand. Then, gradually raise the speed to an appropriate level.

1-2 PTP Operation

(1) Movement locus

The axes move to the target position at the specified speed. The locus of axis tip during movement cannot be specified using commands.

Example)



Position No. 1

MOVP 1
Move from the current position to position No. 1 via PTP operation.

The arm system may change during movement depending on the operation area or upon execution of an arm-system control command.
PTP operation commands: MOVP, MVPI, TMPI, PACH, PMVP, ARCH
For details on these commands, refer to Chapter 2, "Explanation of Commands."

(2) Speed and acceleration/deceleration settings for PTP operation

The speed and acceleration/deceleration for PTP operation are predefined in a program using control declaration commands.
Speed setting command "VELS"; unit [% (ratio to the value set in "Axis-specific parameter No. 28: Maximum PTP speed")]
Acceleration setting command "ACCS"; unit [% (ratio to the value set in "Axis-specific parameter No. 134: Maximum PTP acceleration”)]
Deceleration setting command "DCLS"; unit [% (ratio to the value set in "Axis-specific parameter No. 135: Maximum PTP deceleration”)]

Example)

ACCS	50	Set the acceleration for PTP operation to 50% of the maximum PTP acceleration.
DCLS	50	Set the deceleration for PTP operation to 50% of the maximum PTP deceleration.
VELS	50	Set the speed for PTP operation to 50% of the maximum PTP speed.
MOVP	2	Move to position No. 2 via PTP operation.

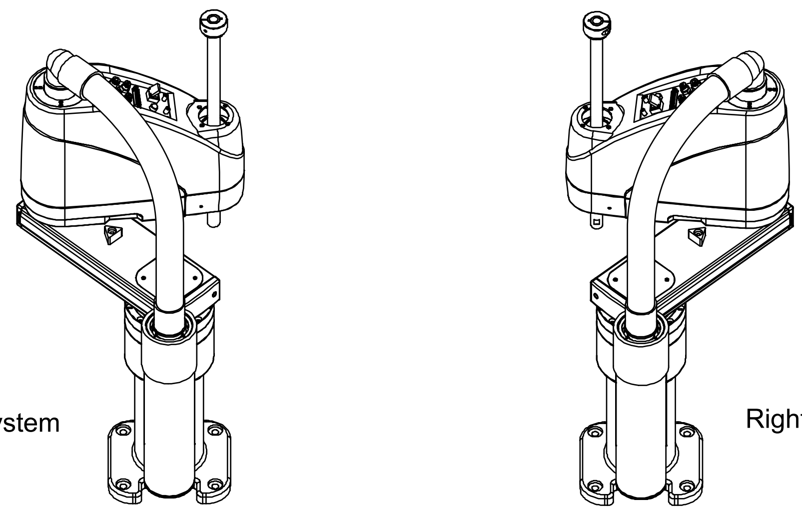
(3) Notes on PTP operation

The arm system may change during movement depending on the operation area or upon execution of an arm-system control command.
Refer to 2, "Arm System," on the following page.

2. Arm System

2-1 Right/Left Arm Systems

The robot position has two patterns based on the right arm system and the left arm system, respectively.



Left arm system

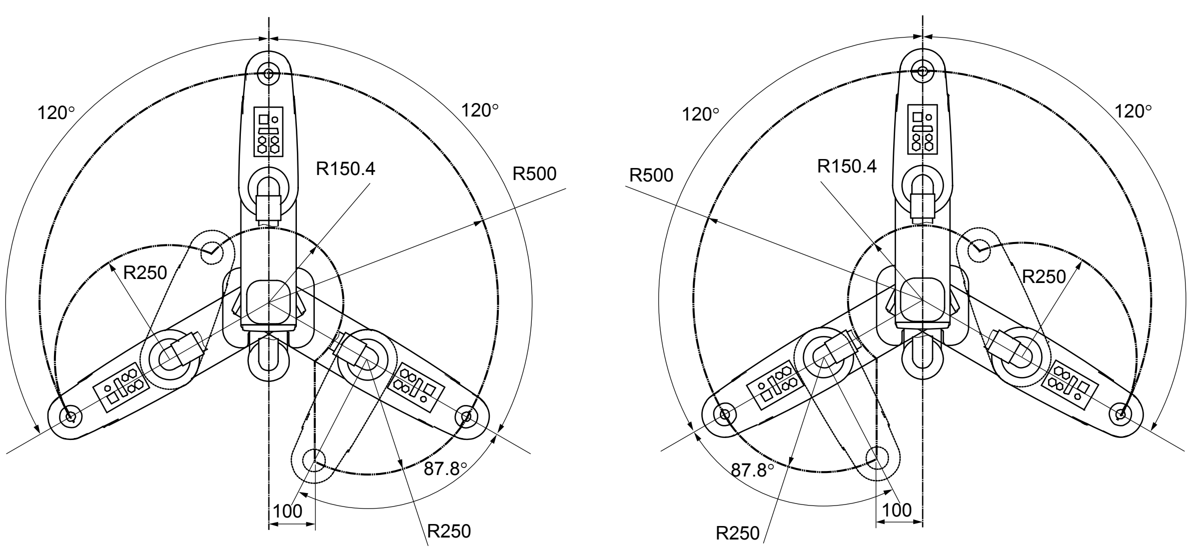
Right arm system

Right arm system: Arm 2 is located at a point away in the CCW direction from the position where arms 1 and 2 form a straight line.

Left arm system: Arm 2 is located at a point away in the CW direction from the position where arms 1 and 2 form a straight line.

Both terms express a robot arm condition by drawing a parallel to human arms.

The operation area is different between the right arm system and the left arm system.
The figure below shows the operation area of each arm system on a robot with an arm length of 500 mm:



Operation area of the left arm system

Operation area of the right arm system

2-2 Arm-System Control Commands

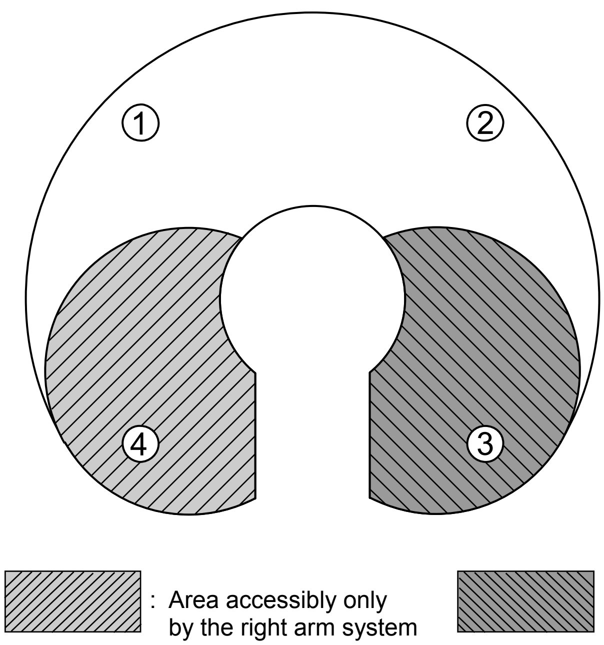
The right and left arm systems are defined as the opposite arm systems to the left and right arm systems, respectively.
The actual arm system that is currently effective is defined as the current arm system.
The arm system to be used for positioning to the target using a movement command is defined as the target arm system.

The commands used to control the robot's arm system include PTPD, PTPE, PTPR, PTPL, RIGH and LEFT.
PTPD, PTPE, PTPR and PTPL are control declaration commands for the target arm system in PTP operation. Therefore, once executed these commands will remain valid while the program is running. CP operation commands do not accompany change of arm systems during command execution, so they are not affected by the above commands and the relevant operations will be performed by the current arm system.
Only one of PTPD, PTPE, PTPR and PTPL, whichever is executed last, will become valid.
RIGH and LEFT are control commands for the current arm system.

2-3 Arm-System Control Commands and Change of Arm Systems

This section explains the arm-system control commands and how the arm system changes in PTP operation after declaration of each command.

Position Nos. 1 to 4 are set as illustrated below (① to ④).
Movement in the order of 1 → 2 → 3 → 2 → 1 → 4 will be attempted using MOVP commands (PTP operation).
The robot is initially resting at position No. 1.
Position No. 3 exists inside an area accessible only by the left arm system. (Positioning to this point cannot be performed with the right arm system.)
Position No. 4 exists inside an area accessible only by the right arm system. (Positioning to this point cannot be performed with the left arm system.)



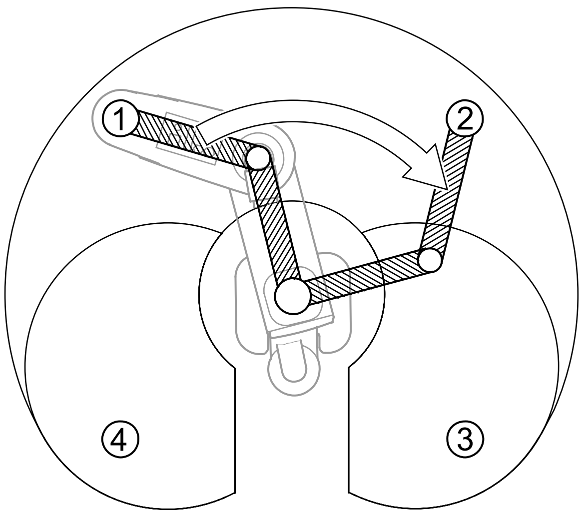

```
:
:
:
Arm-system control commands
MOVP    2
MOVP    3
MOVP    2
MOVP    1
MOVP    4
EXIT
```

How the arm system will change is explained for each arm-system control command.

In the figure, a black arrow indicates a movement involving change of arm systems.
A white arrow indicates a movement not involving change of arm systems.
The striped arm represents the right arm system, while the white arm represents the left arm system.

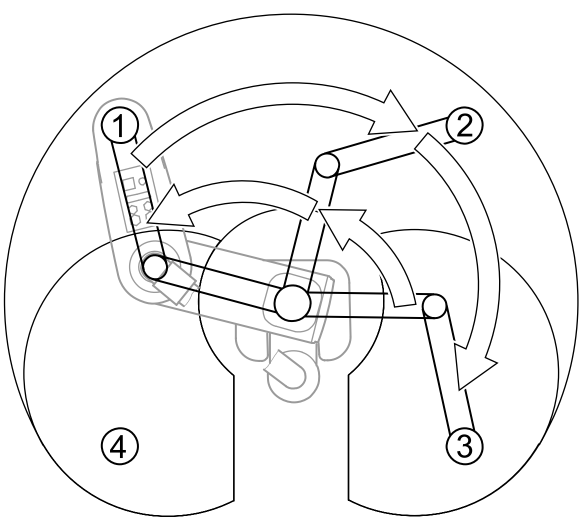
(1) PTPD
After a PTPD command is executed, the robot will move the current arm system to perform positioning. The PTPD command prohibits the current arm system and target arm system from becoming the opposite arm systems. Attempting a movement to an area where positioning is possible only with the opposite arm system will generate an error (C73: Target-locus soft limit over error).
When a program is started, the robot is already in a PTPD-declared mode even before executing a PTPD command.

a. Starting with the right arm system




```
    :
    :
    :
PTPD
MOVP    2
MOVP    3 ⇒ C73 error will generate.
```

b. Starting with the left arm system



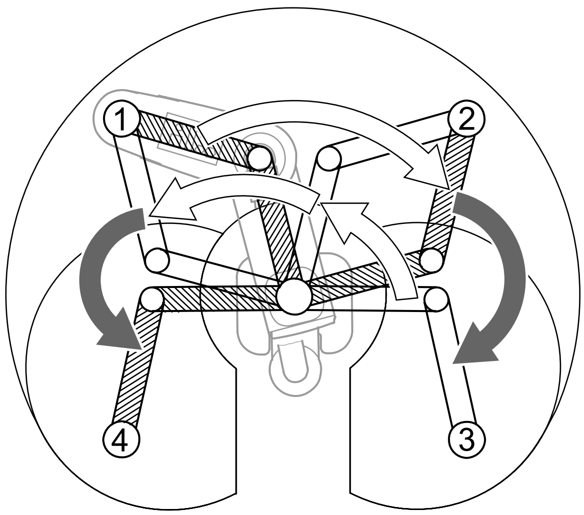

```
    :
    :
    :
PTPD
MOVP    2
MOVP    3
MOVP    2
MOVP    1
MOVP    4 ⇒ C73 error will generate.
```

(2) PTPE

After a PTPE command is executed, the robot will give priority to movements and positioning operations using the current arm system. The PTPE command permits the current arm system and target arm system to become the opposite arm systems. Therefore, movements to an area accessible only by the opposite arm system will also be enabled.

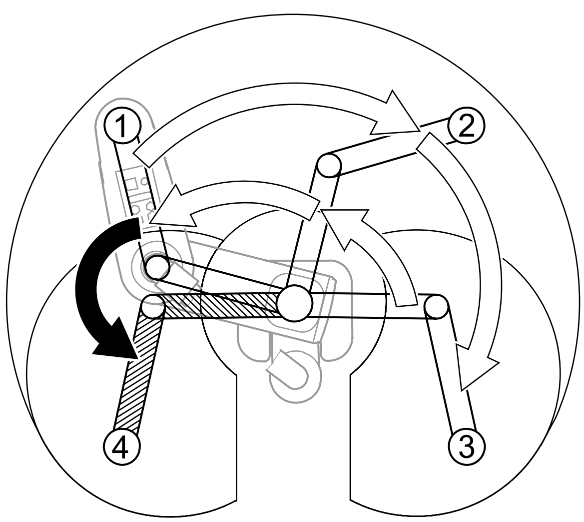
After permitting movements to an area accessible only by the opposite arm system, prohibition of such movements can be effectuated by executing a PTPD command.

a. Starting with the right arm system




```
   :
   :
   :
PTPE
MOVP      2
MOVP      3
MOVP      2
MOVP      1
MOVP      4
EXIT
```

b. Starting with the left arm system

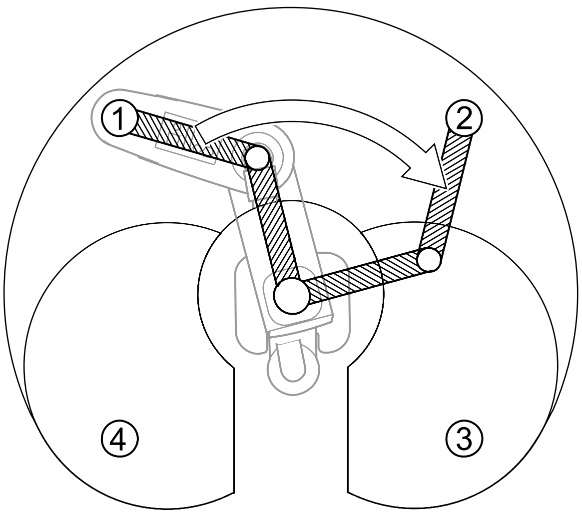



```
   :
   :
   :
PTPE
MOVP      2
MOVP      3
MOVP      2
MOVP      1
MOVP      4
EXIT
```

(3) PTPR

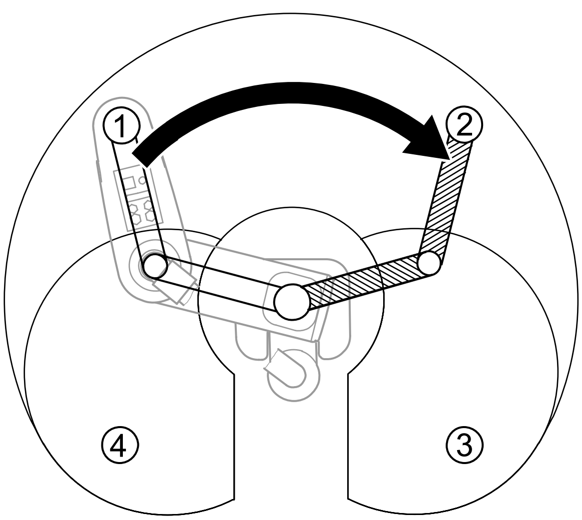
After a PTPR command is executed, the robot will perform positioning using the right arm system. The PTPR command limits the target arm system to the right arm system. Therefore, attempting a movement to an area where positioning is possible only with the left arm system will generate an error (C73: Target-locus soft limit over error).
Executing a PTPR command itself will not trigger any arm operation.
If a PTP movement command is executed following a PTPR command when the current arm system is the left arm system, the axes will move as the arm system changes from left to right and the positioning will be performed using the right arm system.

a. Starting with the right arm system




```
:
:
:
PTPR
MOVP     2
MOVP     3 ⇒ C73 error will generate.
```

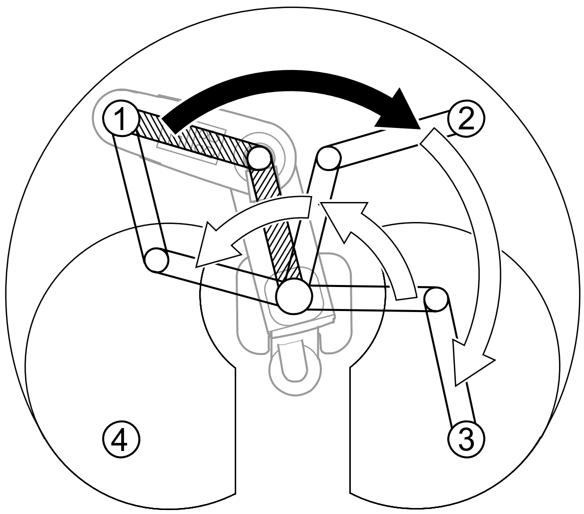
b. Starting with the left arm system




```
:
:
:
PTPR
MOVP     2
MOVP     3 ⇒ C73 error will generate.
```

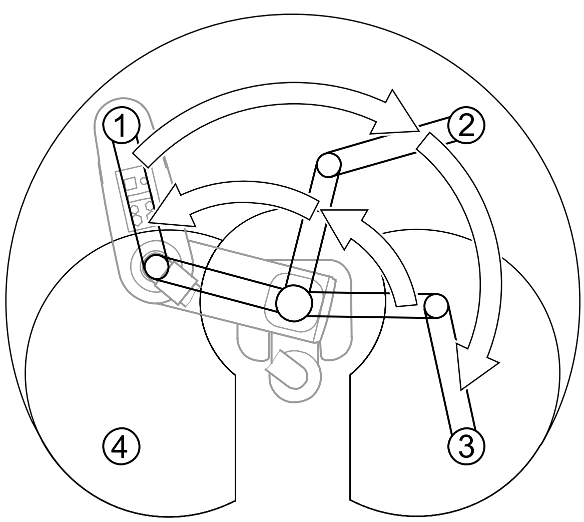
(4) PTPL
After a PTPL command is executed, the robot will perform positioning using the left arm system.
The PTPL command limits the target arm system to the left arm system. Therefore, attempting a movement to an area where positioning is possible only with the right arm system will generate an error (C73: Target-locus soft limit over error).
Executing a PTPL command itself will not trigger any arm operation.
If a PTP movement command is executed following a PTPL command when the current arm system is the right arm system, the axes will move as the arm system changes from right to left and the positioning will be performed using the left arm system.

a. Starting with the right arm system




```
  :
  :
  :
PTPL
MOVP        2
MOVP        3
MOVP        2
MOVP        1
MOVP        4 ⇒ C73 error will generate.
```

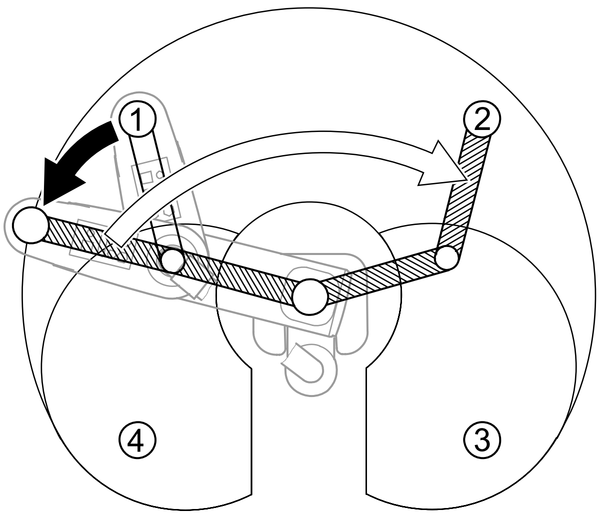
b. Starting with the left arm system




```
  :
  :
  :
PTPL
MOVP        2
MOVP        3
MOVP        2
MOVP        1
MOVP        4 ⇒ C73 error will generate.
```

(5) RIGH
The RIGH command changes the current arm system to the right arm system.
If a RIGH command is executed when the current arm system is the left arm system, arm 2 will move until arms 1 and 2 form a straight line.
Executing a RIGH command when the current arm system is the right arm system will not trigger any arm operation.

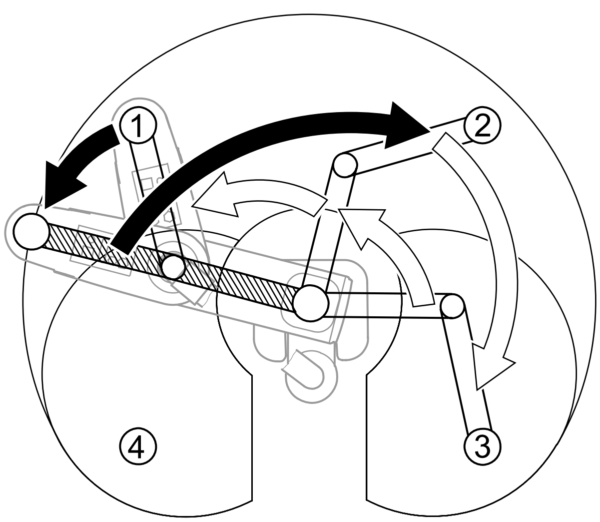
a. Starting with the left arm system




```
    :
    :
    :
RIGH
MOVP      2
MOVP      3 ⇒ C73 error will generate.
```

In the above example, no arm-system control command is set except for a RIGH command and therefore a PTPD command is valid.
The RIGH command controls only the current arm system. It does not limit the positioning arm in PTP operation to the right arm system. Which arm system is used in a given positioning operation will depend on which control declaration (PTPD, PTPE, PTPR or PTPL) is valid for the target arm system. Therefore, the operation to follow a RIGH command execution will vary depending on the valid control declaration for the target arm system.

b. RIGH command when a PTPL command is valid

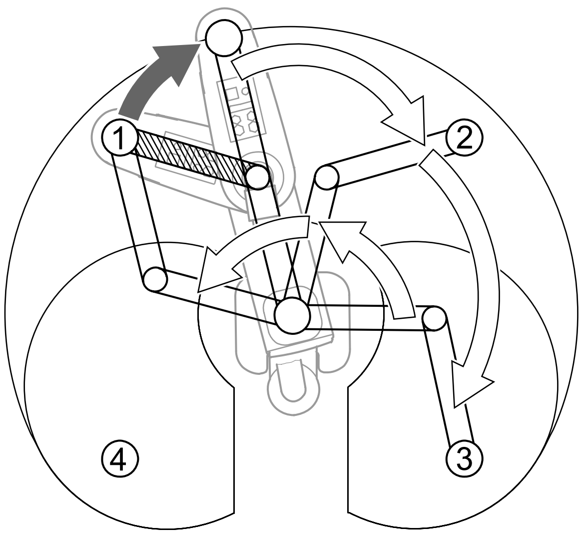



```
    :
    :
    :
PTPL
    :
    :
    :
RIGH
MOVP      2
MOVP      3
MOVP      2
MOVP      1
MOVP      4 ⇒ C73 error will generate.
```

(6) LEFT

The LEFT command changes the current arm system to the left arm system.
If a LEFT command is executed when the current arm system is the right arm system, arm 2 will move until arms 1 and 2 form a straight line.
Executing a LEFT command when the current arm system is the left arm system will not trigger any arm operation.

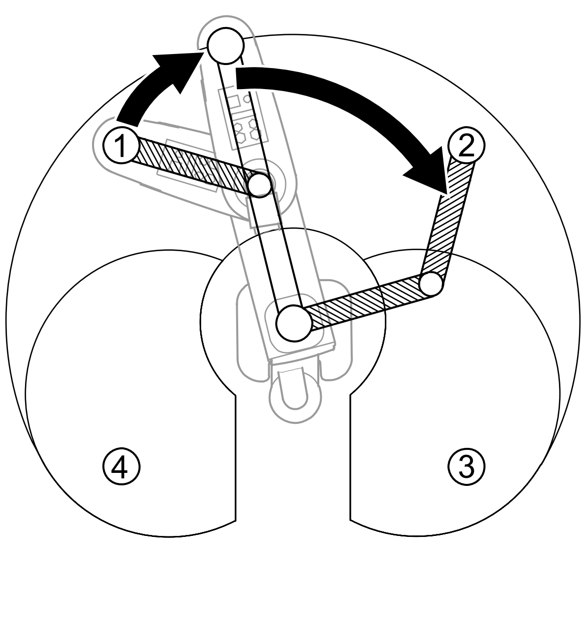
a. Starting with the right arm system




```
   :
   :
   :
LEFT
MOVP      2
MOVP      3
MOVP      2
MOVP      1
MOVP      4 ⇒ C73 error will generate.
```

In the above example, no arm-system control command is set except for a LEFT command and therefore a PTPD command is valid.
The LEFT command controls only the current arm system. It does not limit the positioning arm in PTP operation to the left arm system. Which arm system is used in a given positioning operation will depend on which control declaration (PTPD, PTPE, PTPR or PTPL) is valid for the target arm system. Therefore, the operation to follow a LEFT command execution will vary depending on the valid control declaration for the target arm system.

b. LEFT command when a PTPR command is valid



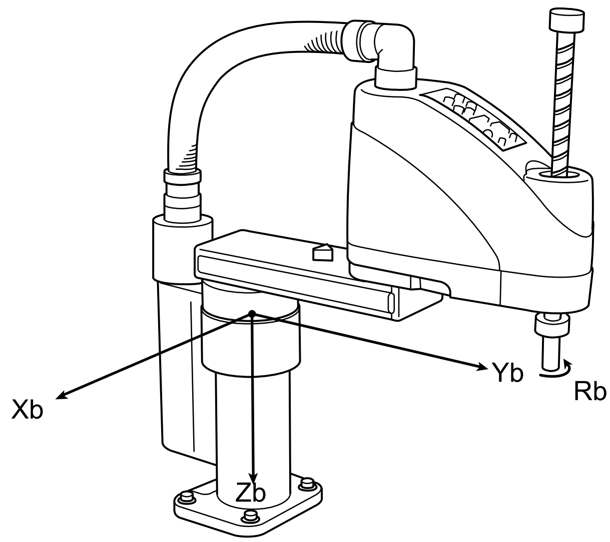

```
   :
   :
   :
PTPR
   :
   :
   :
LEFT
MOVP      2
MOVP      3 ⇒ C73 error will generate.
```

3. Coordinate System

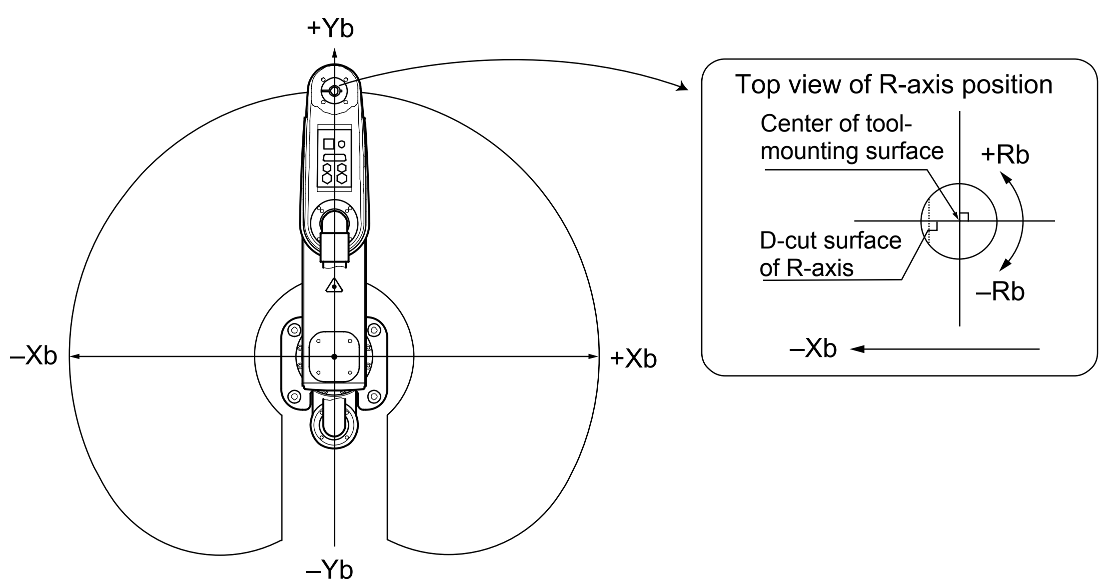
A horizontal articulated robot uses three types of coordinate systems: base coordinate system, work coordinate system and tool coordinate system.
When tool coordinate system No. 0 (= tool coordinate system offsets are 0) is selected, normally the robot will position the center of the tool-mounting surface on the selected work coordinate system. If any of tool coordinate system Nos. 1 to 127 (= tool coordinate system offsets are valid) is selected, the robot will position the tool tip on the selected work coordinate system. Note that the SEL commands TMPI and TMLI as well as jog commands on XY (tool) coordinates will be executed on the tool coordinate system.

3-1 Base Coordinate System (= Work Coordinate System No. 0)

This coordinate system covers the three-dimensional orthogonal coordinates and rotating-axis coordinates factory-defined in the robot.
The base coordinate system corresponds to work coordinate system No. 0 (work coordinate system offsets are 0).



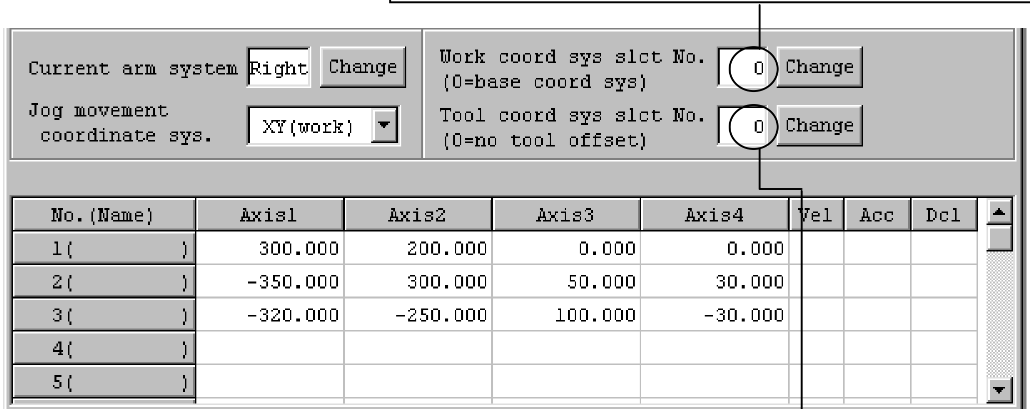
The XY-axis origin is located at the base center (rotating center of arm 1).
The Z-axis origin is located at the upper end of the valid Z-axis stroke.
The R-axis origin is where the D-cut surface faces the –Xb direction.
The X-axis, Y-axis, Z-axis and R-axis on the base coordinate system are expressed as Xb, Yb, Zb and Rb, respectively.



(1) Positioning on the base coordinate system
Perform positioning after selecting work coordinate system No. 0.
Use a SLWK command to select a work coordinate system number in a SEL program. The selected work coordinate system number will remain valid after the program ends, and even after reconnection of power if a system-memory backup battery is installed.

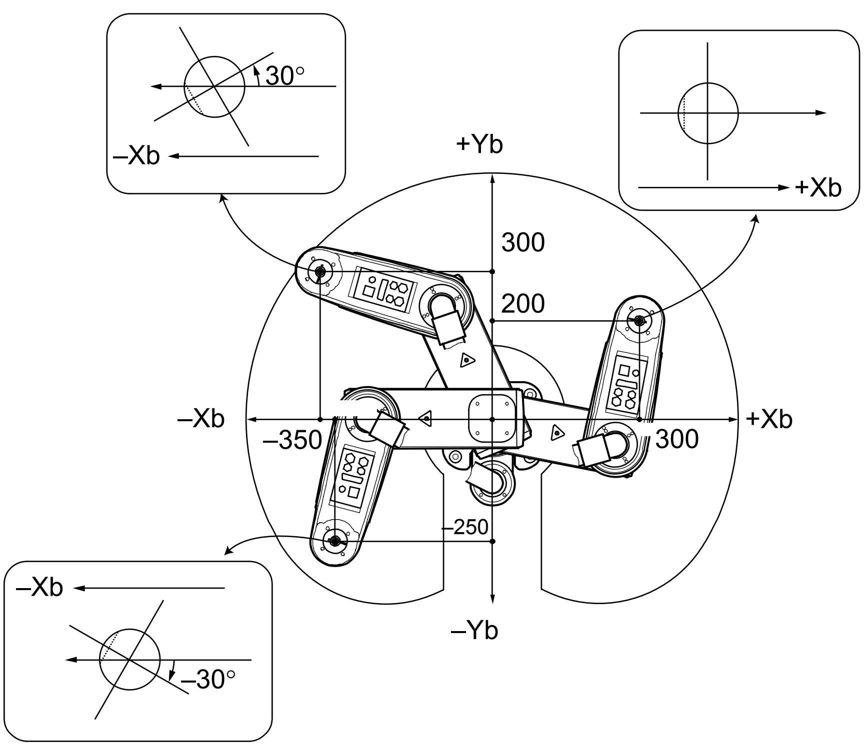
The figure below shows a part of the position data edit screen on the PC software for horizontal articulated robot. Sample teaching data comprising the following contents have been entered:
X = 300, Y = 200, Z = 0, R = 0 as the position data of position No. 1
X = -350, Y = 300, Z = 50, R = 30 as the position data of position No. 2
X = -320, Y = -250, Z = 100, R = -30 as the position data of position No. 3

The selected work coordinate system number is displayed. Work coordinate system No. 0 = Base coordinate system



The selected tool coordinate system number is displayed. Tool coordinate system No. 0 = Positioning of the center of the tool-mounting surface

When poisoning to the above position data in PTP mode:

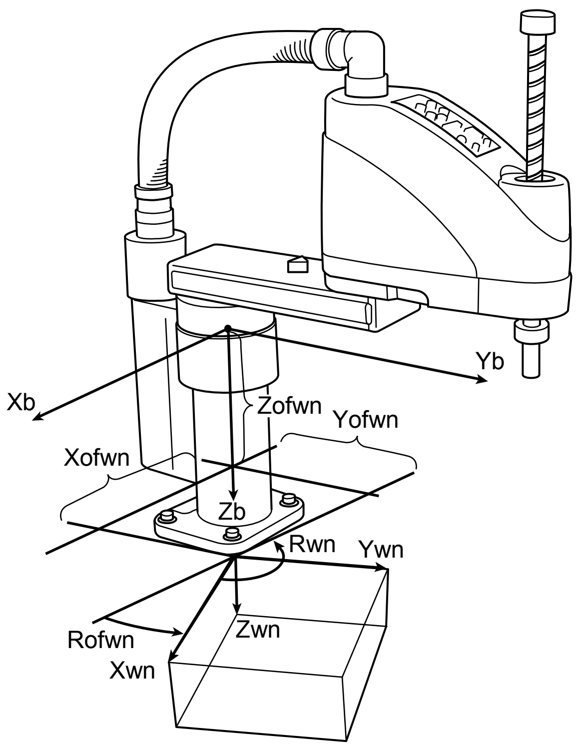


Program example

VELS	50	
ACCS	50	
SLWK	0	Select work coordinate system No. 0.
SLTL	0	Select tool coordinate system No. 0.
PTPR		Specify the right arm as the PTP target arm system.
MOVP	1	
MOVP	2	
MOVP	3	
EXIT		

3-2 Work Coordinate System

This coordinate system provides 32 sets of three-dimensional orthogonal coordinates and rotating-axis coordinates as defined by the offset of each axis with respect to the base coordinate system. Note that work coordinate system No. 0 is reserved by the system as the base coordinate system (= work coordinate system offsets are 0).



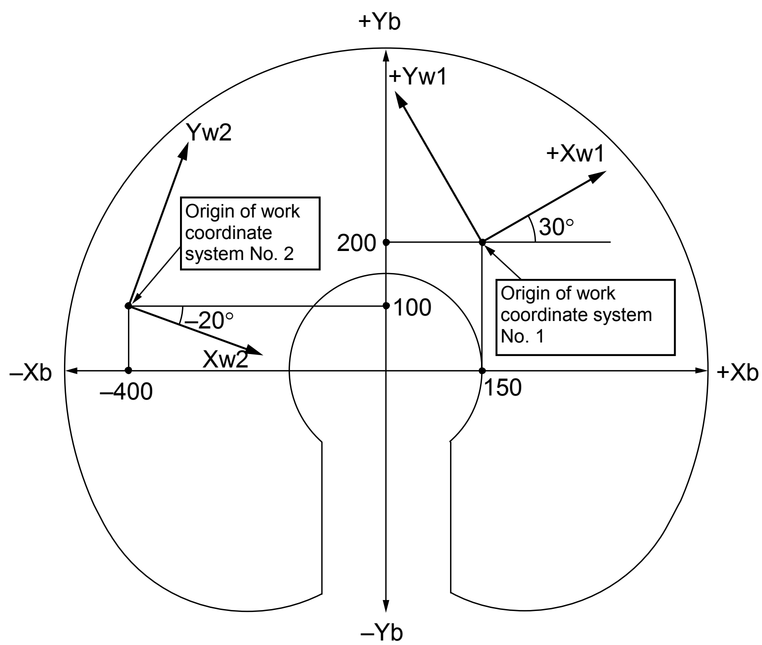
Xofwn: X work coordinate offset
Yofwn: Y work coordinate offset
Zofwn: Z work coordinate offset
Rofwn: R work coordinate offset

Xwn: Work coordinate system, X-axis
Ywn: Work coordinate system, Y-axis
Zwn: Work coordinate system, Z-axis
Rwn: Work coordinate system, R-axis

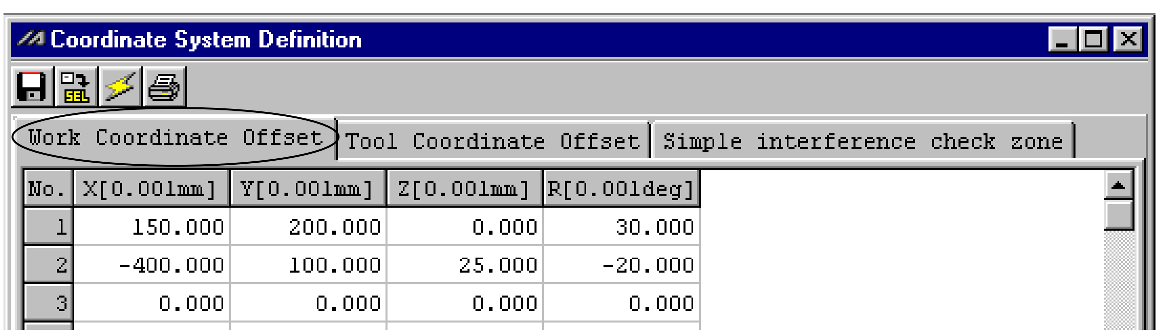
(“n” indicates work coordinate system number.)

(1) Setting the work coordinate system
Set the offsets with respect to the base coordinate system.

• Setting example of work coordinate system
When defining work coordinate system Nos. 1 and 2 as shown below:



The offsets of work coordinate system No. 1 are set as Xofw1 = 150, Yofw1 = 200, Zofw1 = 0 and Rofw1 = 30.
The offsets of work coordinate system No. 2 are set as Xofw2 = -400, Yofw2 = 100, Zofw2 = 25 and Rofw2 = -20.
The figure below shows the edit screen for work coordinate system definition data on the PC software for horizontal articulated robot, where work coordinate system Nos. 1 and 2 are set:

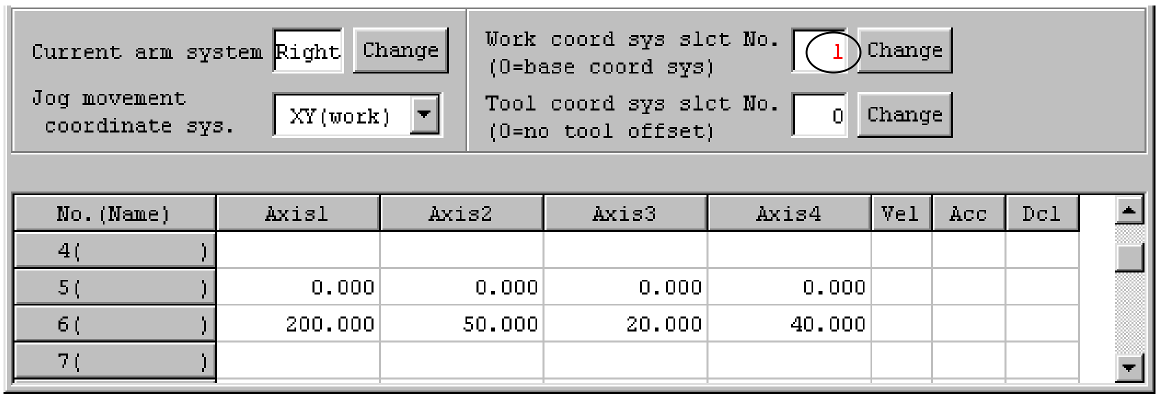


No.	X[0.001mm]	Y[0.001mm]	Z[0.001mm]	R[0.001deg]
1	150.000	200.000	0.000	30.000
2	-400.000	100.000	25.000	-20.000
3	0.000	0.000	0.000	0.000

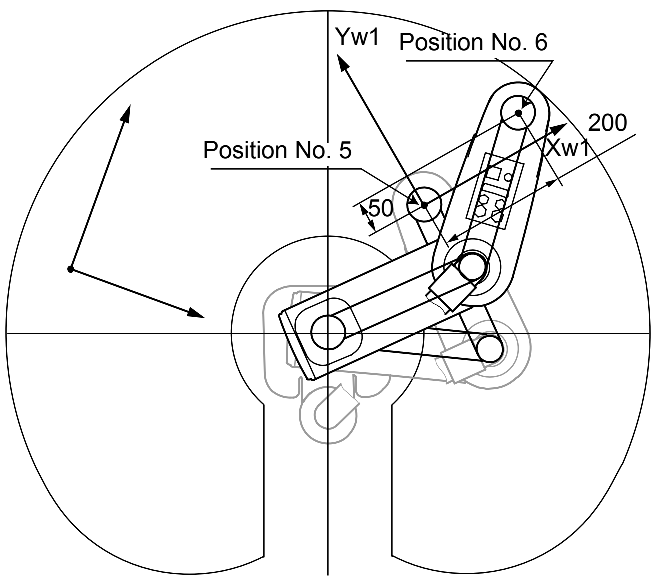
* Use a DFWK command to set work coordinate system offsets in a SEL program.

(2) Positioning on the work coordinate system
Perform positioning after selecting a desired work coordinate system.
Use a SLWK command to select a work coordinate system number in a SEL program.
The selected work coordinate system number will remain valid after the program ends, and even after reconnection of power if a system-memory backup battery is installed.

a. When positioning to position Nos. 5 and 6 in PTP mode on work coordinate system No. 1



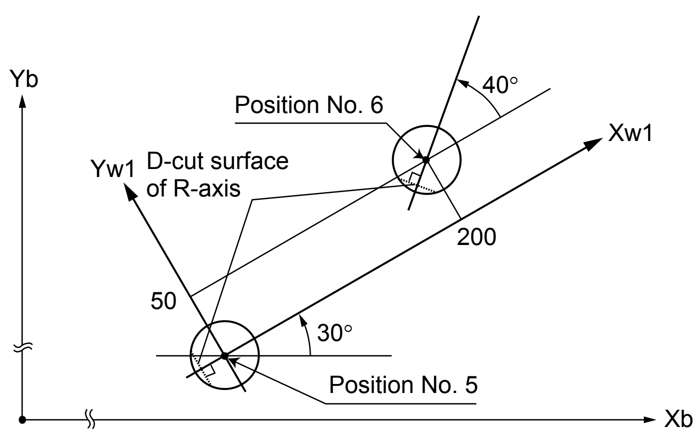
No.(Name)	Axis1	Axis2	Axis3	Axis4	Vel	Acc	Dcl
4( )							
5( )	0.000	0.000	0.000	0.000			
6( )	200.000	50.000	20.000	40.000			
7( )							



Program example
:
:
:

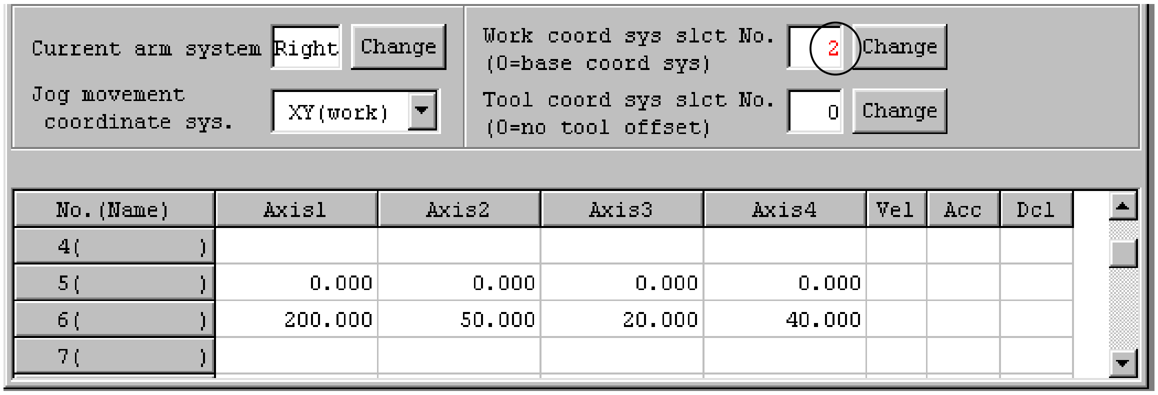
SLWK	1	Select work coordinate system No. 1.
SLTL	0	Select tool coordinate system No. 0.
PTPR		Specify the right arm as the PTP target arm system.
MOVP	5	Move to position No. 5.
MOVP	6	Move to position No. 6.

:
:
:



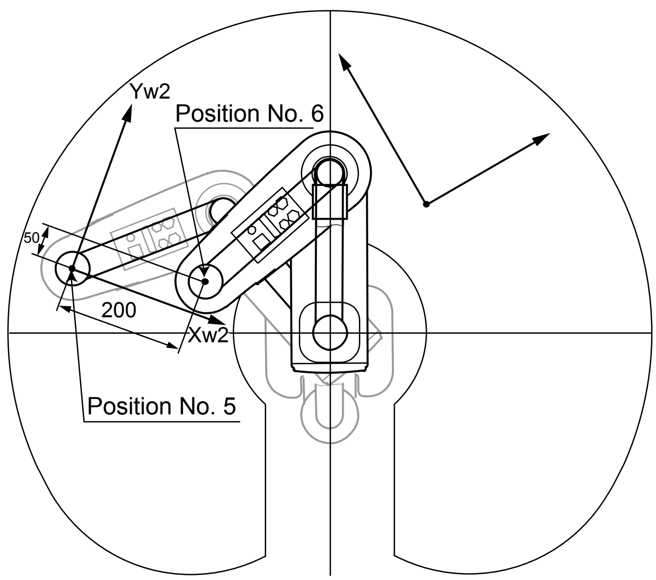
The R-axis position will be as shown in the figure at left (top view).
The Z-axis position will be as follows:
Position No. 5: Zb = 0
Position No. 6: Zb = 20

b. When positioning to position Nos. 5 and 6 in PTP mode on work coordinate system No. 2



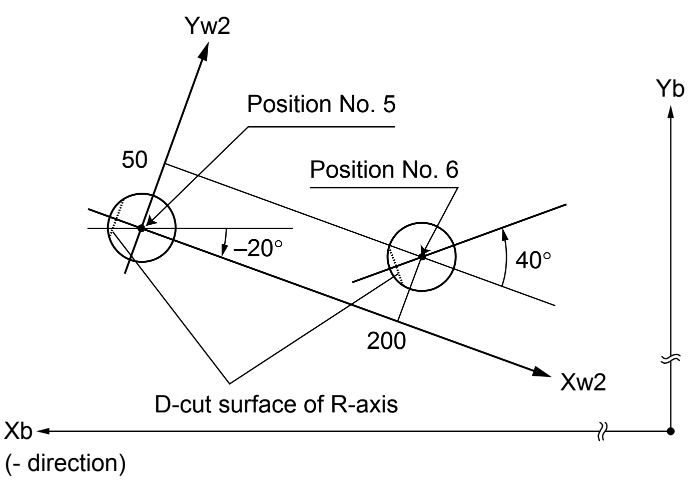
Current arm system: Right [Change]
Jog movement coordinate sys.: XY(work)
Work coord sys slct No. (0=base coord sys): 2 [Change]
Tool coord sys slct No. (0=no tool offset): 0 [Change]

No.(Name)	Axis1	Axis2	Axis3	Axis4	Vel	Acc	Dcl
4( )							
5( )	0.000	0.000	0.000	0.000			
6( )	200.000	50.000	20.000	40.000			
7( )							



Program example

```
:
:
:
SLWK    2    Select work coordinate
             system No. 2.
SLTL    0    Select tool coordinate
             system No. 0.
PTPR         Specify the right arm as the
             PTP target arm system.
MOVP    5    Move to position No. 5.
MOVP    6    Move to position No. 6.
:
:
:
```

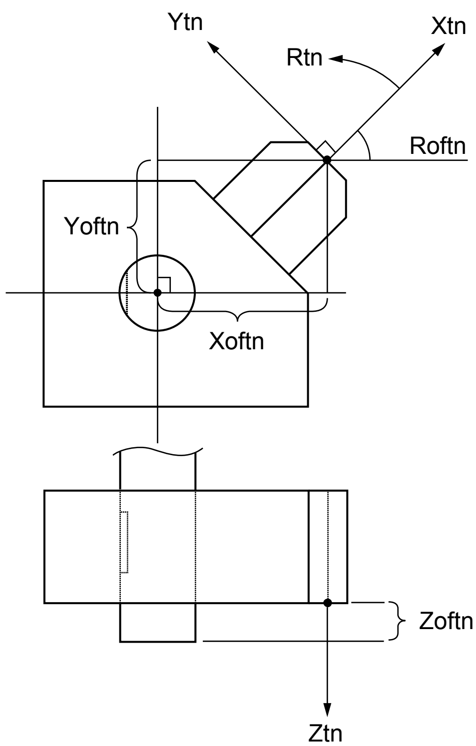


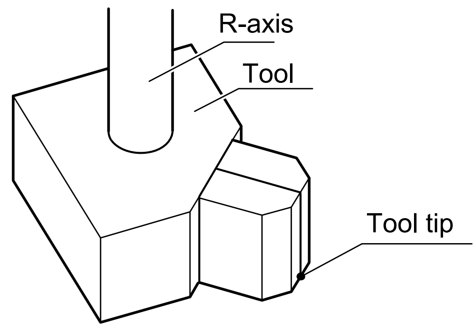
The R-axis position will be as shown in the figure at left (top view).
The Z-axis position will be as follows:
Position No. 5: Zb = 25
Position No. 6: Zb = 45

3-3 Tool Coordinate System

This coordinate system provides 128 sets of three-dimensional orthogonal coordinates and rotating-axis coordinates as defined by the dimensions (offsets) of a tool (hand, etc.) installed on the tool-mounting surface. Note that tool coordinate system No. 0 is reserved by the system as a tool coordinate system with zero offsets.

When any of the defined tool coordinate system numbers is selected, the tool tip, rather than the center of the tool-mounting surface, will be used as the reference point in moving to the target position.



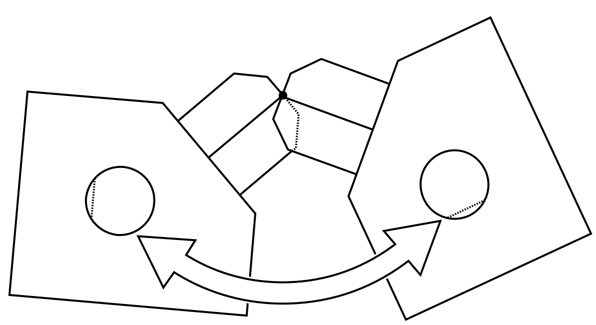


Xoftn: X tool coordinate offset
Yoftn: Y tool coordinate offset
Zoftn: Z tool coordinate offset
Roftn: R tool coordinate offset

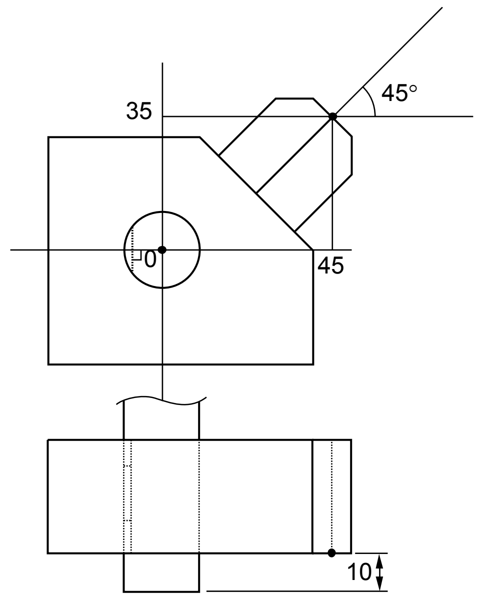
Xtn: Tool coordinate system, X-axis
Ytn: Tool coordinate system, Y-axis
Ztn: Tool coordinate system, Z-axis
Rtn: Tool coordinate system, R-axis

("n" indicates tool coordinate system number.)

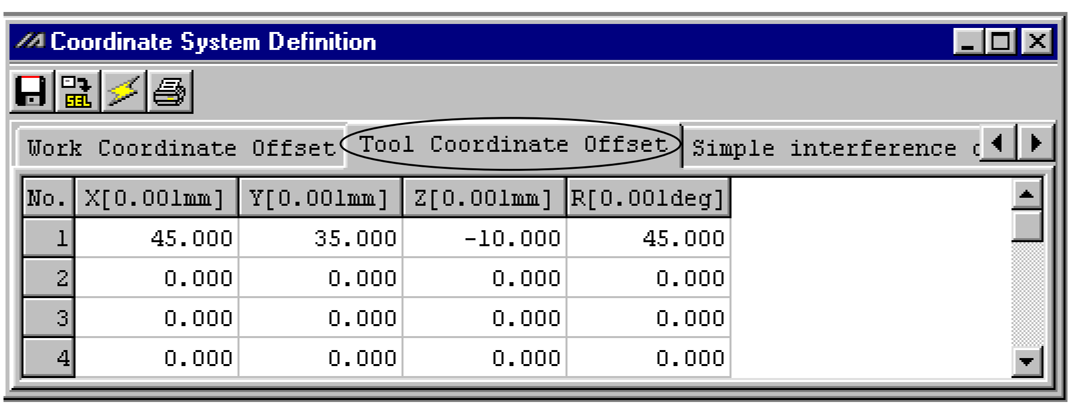
Selecting a defined tool coordinate system and executing a jog command for the R-axis will result in the following operation:



(1) Setting the tool coordinate system
Set the offsets from the center of the tool-mounting surface to the tool tip.
• Setting example of tool coordinate system
When defining tool coordinate system No. 1 as shown below:



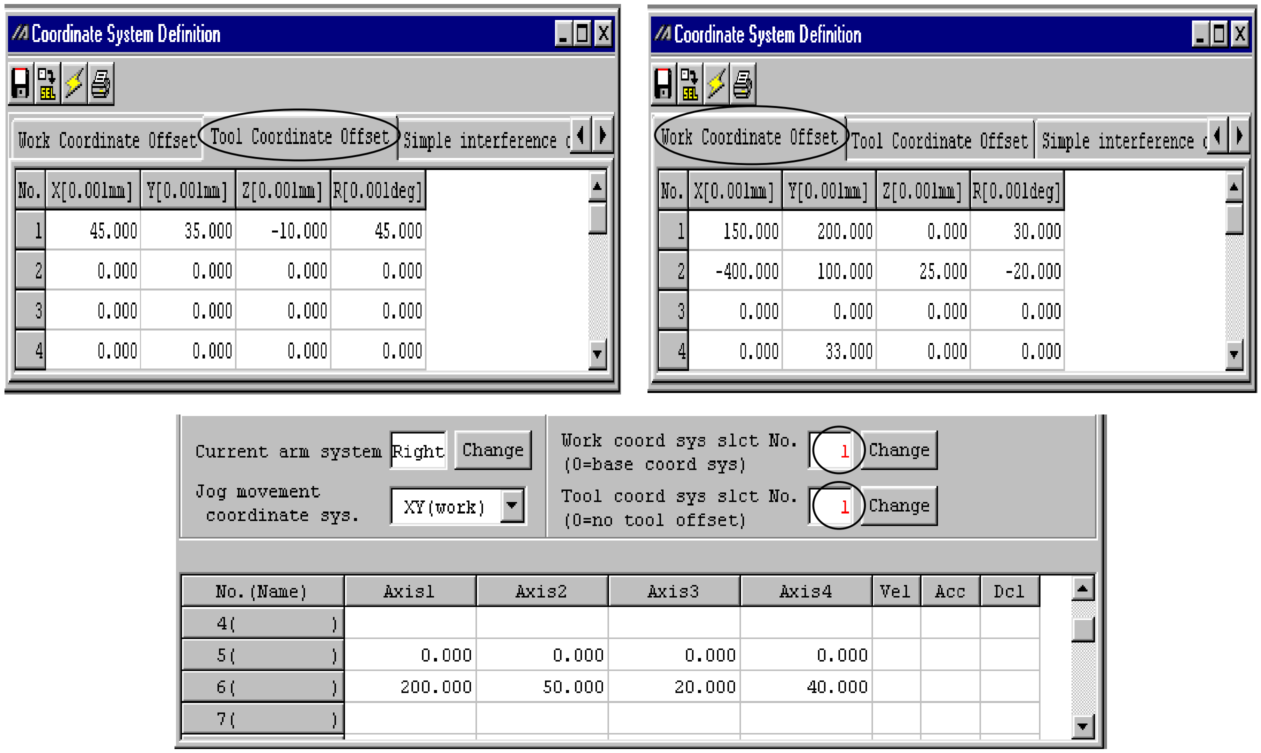
The offsets of tool coordinate system No. 1 are set as Xoft1 = 45, Yoft1 = 35, Zoft1 = -10 and Roft1 = 45.
The figure below shows the edit screen for tool coordinate system definition data on the PC software for horizontal articulated robot, where tool coordinate system No. 1 is set:

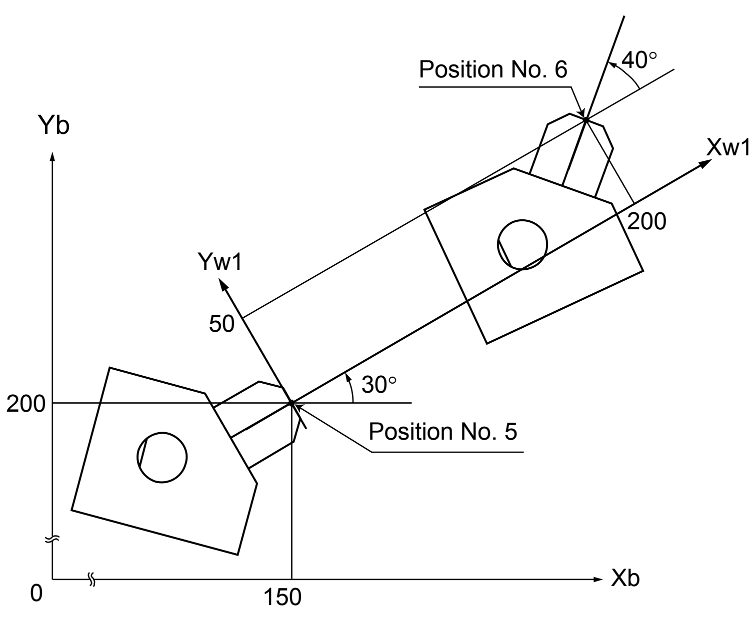


* Use a DFTL command to set tool coordinate system offsets in a SEL program.

(2) Positioning using tool coordinate system offsets
Perform positioning after selecting a desired tool coordinate system.
Use a SLTL command to select a tool coordinate system number in a SEL program.
The selected tool coordinate system number will remain valid after the program ends, and even after reconnection of power if a system-memory backup battery is installed.

a. When positioning the tool tip on tool coordinate system No. 1 to position Nos. 5 and 6 on work coordinate system No. 1 in PTP mode





Program example

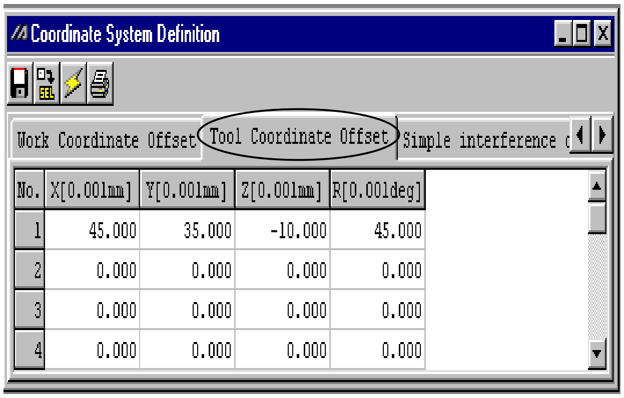
:
:
:

SLWK	1	Select work coordinate system No. 1.
SLTL	1	Select tool coordinate system No. 1.
PTPR		Specify the right arm as the PTP target arm system.
MOVP	5	Move to position No. 5.
MOVP	6	Move to position No. 6.

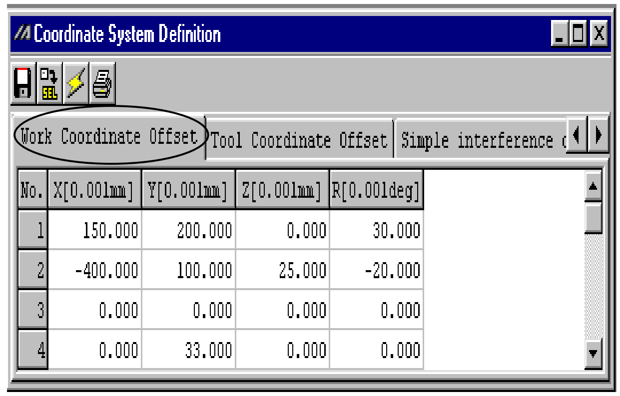
:
:
:

The Z-axis position of tool tip will be as follows:
Position No. 5: Zb = 0
Position No. 6: Zb = 20
The figure shown at left is a top view.

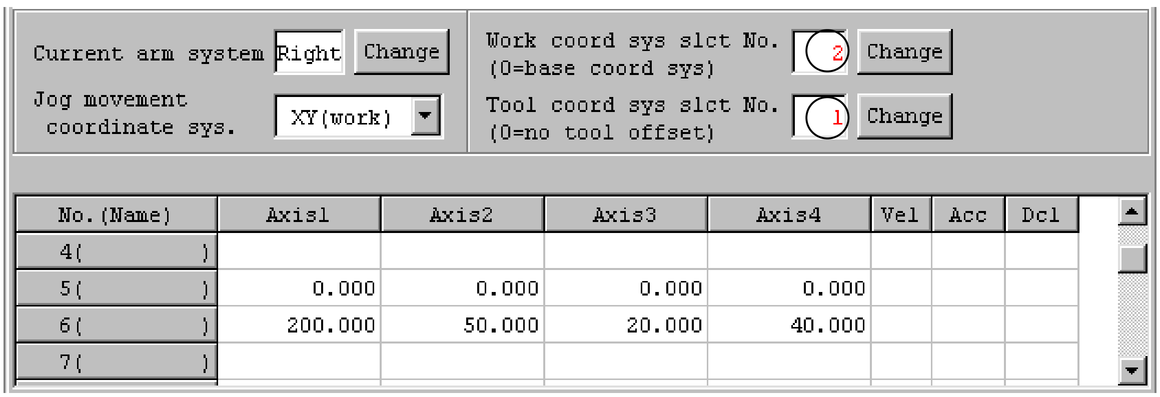
b. When positioning the tool tip on tool coordinate system No. 2 to position Nos. 5 and 6 on work coordinate system No. 1 in PTP mode



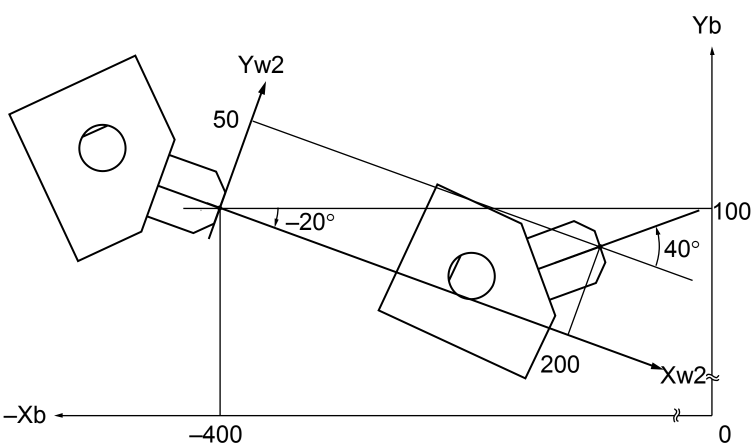
No.	X[0.001mm]	Y[0.001mm]	Z[0.001mm]	R[0.001deg]
1	45.000	35.000	-10.000	45.000
2	0.000	0.000	0.000	0.000
3	0.000	0.000	0.000	0.000
4	0.000	0.000	0.000	0.000



No.	X[0.001mm]	Y[0.001mm]	Z[0.001mm]	R[0.001deg]
1	150.000	200.000	0.000	30.000
2	-400.000	100.000	25.000	-20.000
3	0.000	0.000	0.000	0.000
4	0.000	33.000	0.000	0.000



No.(Name)	Axis1	Axis2	Axis3	Axis4	Vel	Acc	Dcl
4( )							
5( )	0.000	0.000	0.000	0.000			
6( )	200.000	50.000	20.000	40.000			
7( )							



Program example

:
:
:

SLWK	2	Select work coordinate system No. 2.
SLTL	1	Select tool coordinate system No. 1.
PTPR		Specify the right arm as the PTP target arm system.
MOVP	5	Move to position No. 5.
MOVP	6	Move to position No. 6.

:
:
:

The Z-axis position of tool tip will be as follows:
Position No. 5: Zb = 25
Position No. 6: Zb = 45

4. Simple Interference Check Zone

The simple interference check zone is an area set for the purpose of checking possible interference between the robot and peripherals.
In the case of tool coordinate system No. 0 (= tool coordinate system offsets are 0), entry into the simple interference check zone can be detected based on the center of the tool-mounting surface. In the case of tool coordinate system Nos. 1 to 127 (= tool coordinate offsets are valid), entry into the check zone can be detected based on the tool tip.

(1) Notes on use of simple interference check zone

Entry into the simple interference check zone of the center of the tool-mounting surface (when tool coordinate system No. 0 is selected) or the tool tip (when any of tool coordinate system Nos. 1 to 127 is selected) is detected. Entry of the R-axis periphery or parts other than the tool tip will not be detected.

This function does not prevent entry into the simple interference check zone. It only detects that the specified part has entered the zone.

Entry into the simple interference check zone cannot be detected reliably unless the entered part stays in the zone for at least 5 msec. The function is designed as a simple check during low-speed operation.

The locus is different between high-speed operation (operation at the actual operating speed) and low-speed operation. Provide a sufficient margin to prevent interference. (During high-speed operation, the locus tends to shift inward compared with during low-speed operation.)

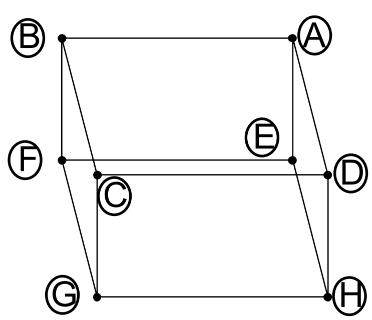
The definition coordinates of simple interference check zone are always treated as data on the base coordinate system (work coordinate system No. 0). Therefore, changing the work coordinate system will not change the position of the simple interference check zone. Exercise caution.

After changing the definition coordinates of simple interference check zone, it will take at least 5 msec before the new settings are reflected in the check result.

During PTP operation, movements will not follow specified paths. When moving near an interfering object (including the robot itself), always perform test operation at low speed and confirm absence of interference, and then gradually raise the speed to an appropriate level.

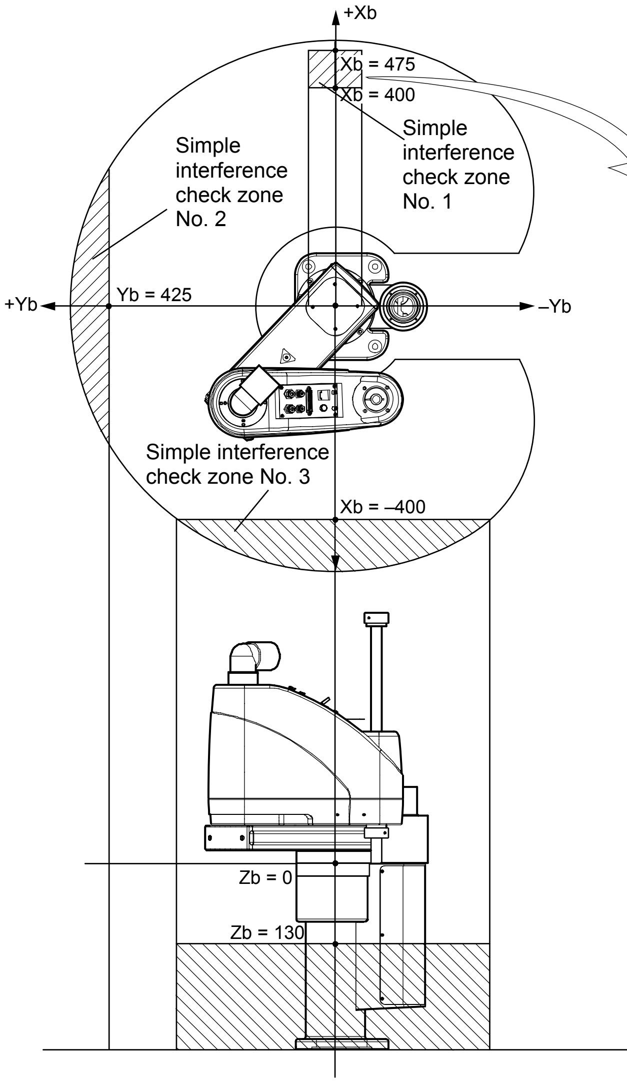
(2) Setting the simple interference check zone

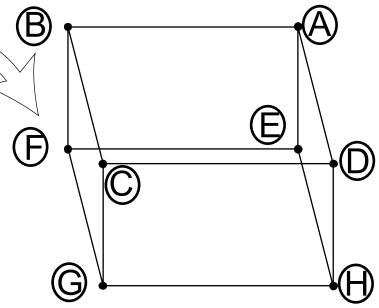
The simple interference check zone is set using position data on the base coordinate system.
Enter the maximum and minimum values defining the simple interference check zone.
Set the boundaries of the simple interference check zone in parallel with the base coordinate axes.



To set the rectangular solid shown at left as a simple interference check zone, enter coordinates of two points corresponding to one of the combinations of (A)-(G), (B)-(H), (C)-(E) and (D)-(F).

Setting example of simple interference check zone
Define simple interference check zone Nos. 1, 2 and 3 as follows:



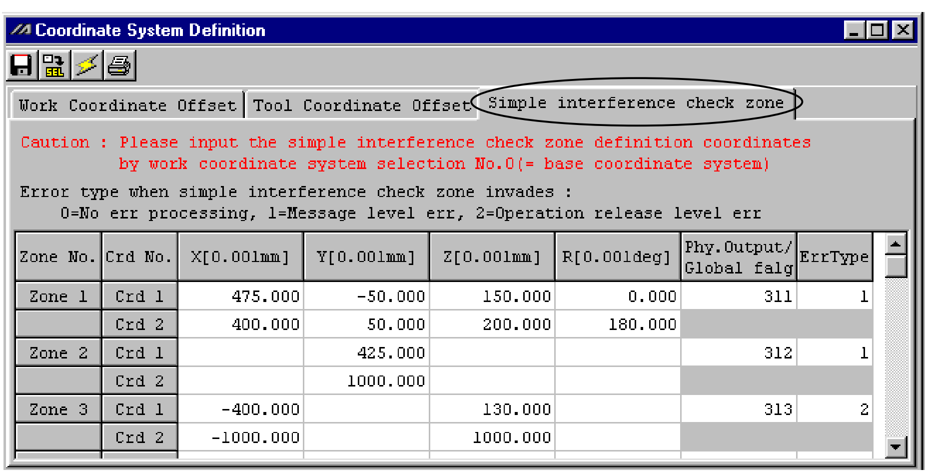


1. Set the area inside a rectangular solid as simple interference check zone No. 1.
   (A): Xb = 475, Yb = –50, Zb = 150, Rb = 0
   (G): Xb = 400, Yb = 50, Zb = 200, Rb = 180
   If Rb is outside the range of 0 to 180°, entry into this rectangular solid area will not be detected.

2. Set an area with Yb of 425 mm or more as simple interference check zone No. 2.

3. Set an area with Xb of –400 mm or less and Zb of 130 mm or more as simple interference check zone No. 3.

The figure below shows the edit screen for definition data of simple interference check zone on the PC software for horizontal articulated robot, where simple interference check zone Nos. 1, 2, and 3 are set:



Zone No.	Crd No.	X[0.001mm]	Y[0.001mm]	Z[0.001mm]	R[0.001deg]	Phy.Output/ Global falg	ErrType
Zone 1	Crd 1	475.000	-50.000	150.000	0.000	311	1
	Crd 2	400.000	50.000	200.000	180.000		
Zone 2	Crd 1		425.000			312	1
	Crd 2		1000.000				
Zone 3	Crd 1	-400.000		130.000		313	2
	Crd 2	-1000.000		1000.000			

As for simple interference check zone No. 1, entry into this rectangular solid area will not be detected if Rb is outside the range of 0 to 180°. To enable detection regardless of the R-axis coordinate, do not enter anything in coordinates 1 and 2 in the R column for zone 1.

If either the maximum value or minimum value needs not be limited, as in the case of simple interference check zone No. 2 or 3, enter a value outside the operation area (1000 in zone 2, 1000 or -1000 in zone 3).

The maximum/minimum value may be set in either coordinate 1 or 2.

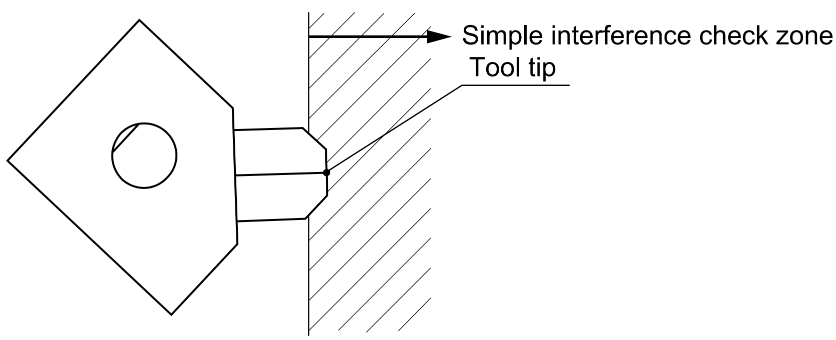
Entry into simple interference check zone No. 1, 2 or 3 will turn ON output port No. 311, 312 or 313, respectively.

Duplicate specifications of physical output numbers or global flag numbers will cause chattering and the result will become indeterminable. Do not set duplicate numbers.

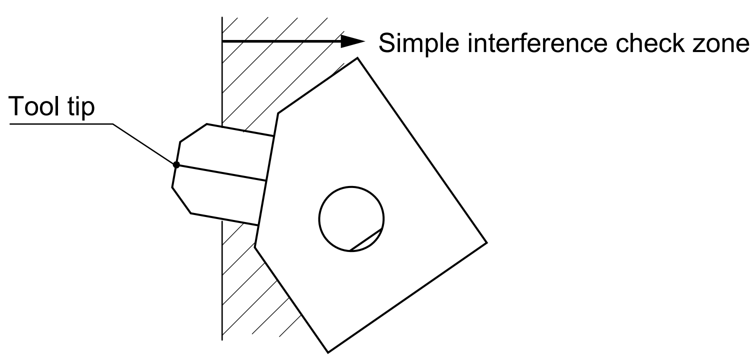
Use of simple interference check zones will reduce the CPU performance significantly. When simple interference check zones are not used, set “0” in "physical output port number/global flag number" and "error type" to disable the function.

* Use a DFIF command to set a simple interference check zone in a SEL program.

(3) Notes on operation when a tool coordinate system is selected
When a tool coordinate system is selected, entry of the tool tip into the simple interference check zone, not entry of the center of the mounting surface, will be detected.



Depending on the movement locus, a part other than the tool tip may enter the simple interference check zone, as shown below. In this case, however, detection will not occur until the tool tip enters the simple interference check zone. Exercise due caution.

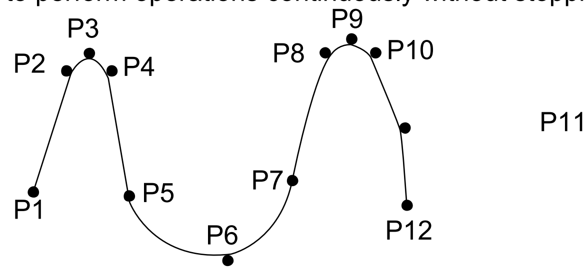


Chapter 4 Key Characteristics of Actuator Control Commands and Points to Note

1. Continuous Movement Commands [PATH, PSPL, CIR2, ARC2, CIRS, ARCS, ARCD, ARCC, CIR, ARC]

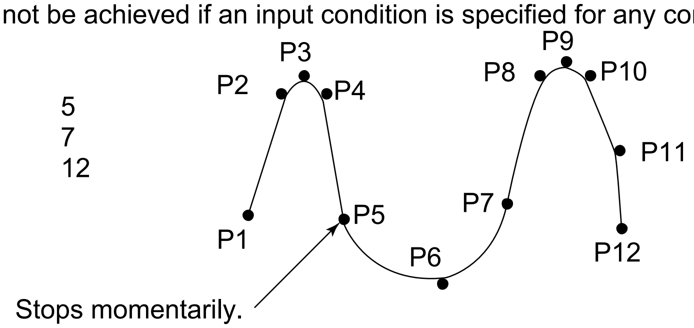
(1) By running a program with continuous movement commands input in a series of continuous program steps, you can allow the actuators to perform operations continuously without stopping between steps.

```
        PATH    1    5
        ARC2    6    7
        PATH    8    12
```

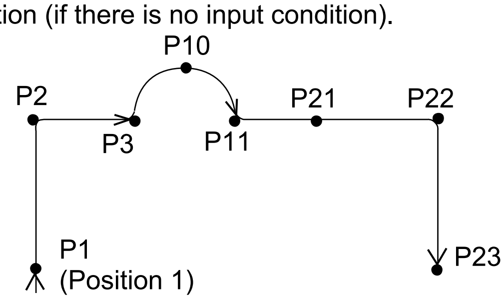


(2) Continuous movement will not be achieved if an input condition is specified for any continuous movement command.

```
        PATH    1    5
20      ARC2    6    7
        PATH    8    12
```



(3) 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     308
ARC2    10    11    311
PATH    21    23    312
  |
  |
  |
```

Output field	Timing
308	Turn ON as P1 approaches.
309	Turn ON as P2 approaches.
310	Turn ON as P3 approaches.
311	Turn ON as P11 approaches.
312	Turn ON as P21 approaches.
313	Turn ON as P22 approaches.
314	Turn ON when P23 operation is complete.

[Example 2] (POTP = 0)

```
PATH    1     3     308
ARC2    10    11    311
PATH    21    23    312
```

Output field	Timing
308	Turn ON as P3 approaches.
311	Turn ON as P11 approaches.
312	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     308
20      ARC2   10    11    311
        PATH   21    23    312
```

Output field	Timing
308	Turn ON as P1 approaches.
309	Turn ON as P2 approaches.
310	Turn ON when P3 operation is complete.
311	Turn ON as P11 approaches.
312	Turn ON as P21 approaches.
313	Turn ON as P22 approaches.
314	Turn ON when P23 operation is complete.

(4) 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
  |
  |
  |
```

(5) 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                   Do not let outputs 305 through 308 to duplicate, as in
                           the example shown at left.
PATH   1     5     305 ┐
  |                    │
  |                    ├ Continuous operation section executed by continuous
  |                    │ movement commands
PATH   11    15    304 ┘
```

The final output status of duplicate 305 through 308 is indeterminable, because it is affected by the positioning calculation time and the relationship of durations of actual operations.

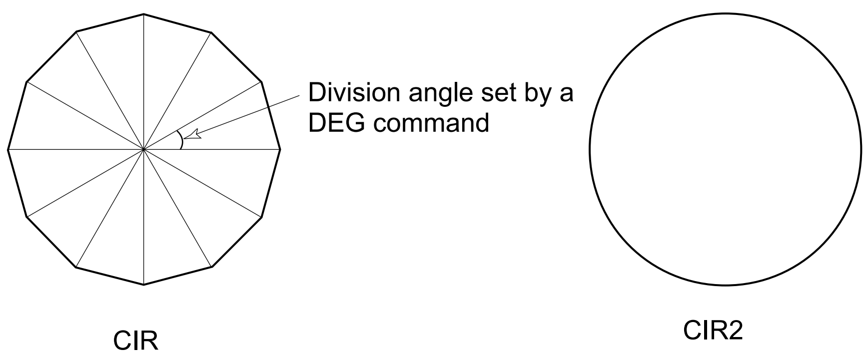
Do not create a program containing an indefinite loop of continuous movement commands using the TAG-GOTO syntax. (It will result in an accumulation of coordinate conversion errors.)

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.
This command is valid only on the XY plane.



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 CP acceleration in all-axis parameter No. 11 (Default CP deceleration in all-axis parameter No. 12)

This command is valid only on the XY plane.

Chapter 5 Palletizing Function

The SEL language used by the X-SEL 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.

BGPA	1	Declare the start of setting for palletizing No. 1.
		Set palletizing in these steps.
EDPA		Declare the end of palletizing setting at the end.

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.”

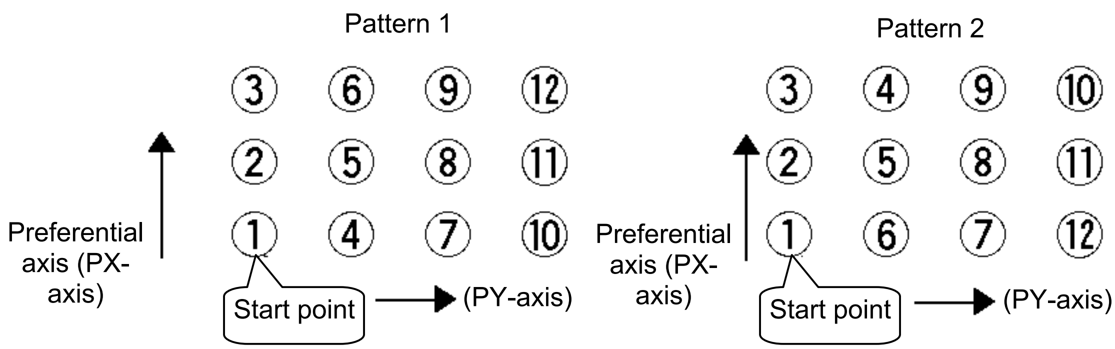


Fig. 1

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.

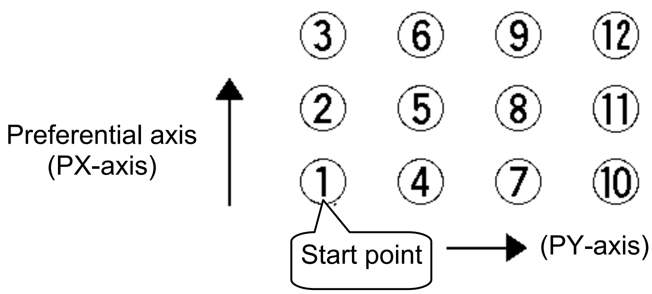
	Setting method	Commands
A	3-point teaching method Set three position-data points specifying the palletizing positions.	PAPS
B	Method to set palletizing positions in parallel with the actuators Set from the palletizing axes, palletizing reference point and palletizing pitches.	PASE, PAST, PAPT

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 work coordinate system axes 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: Reference point
Position No. 12 ③: The end point in the PX-axis direction
Position No. 13 ⑩: 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.



The pitches are calculated automatically from the count set for each axis.
In 3-point teaching, you can specify position data for two axes or three axes. If data are specified for three axes, the palletizing plane will become a three-dimensional plane.

Fig. 1

```
PAPS    11
PEXT    14
```

Do no enter anything in the R-axis data column of the position data specified by a PAPS command. (Alternatively, disable the R-axis using a GRP command.)
Use a PEXT command to set the R-axis coordinate of a given palletizing position.

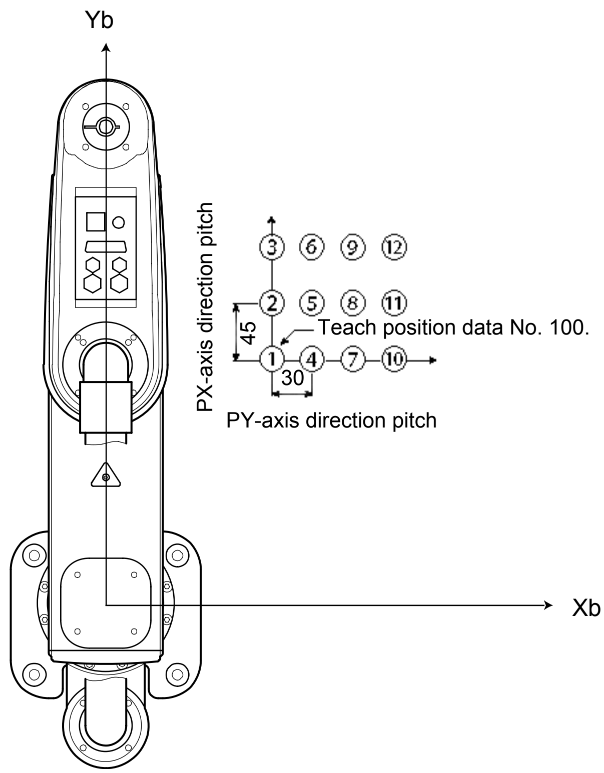
B. Method to set palletizing positions in parallel with the work coordinate system axes

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.
Use a PEXT command to set the R-axis coordinate of a given palletizing position.

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)



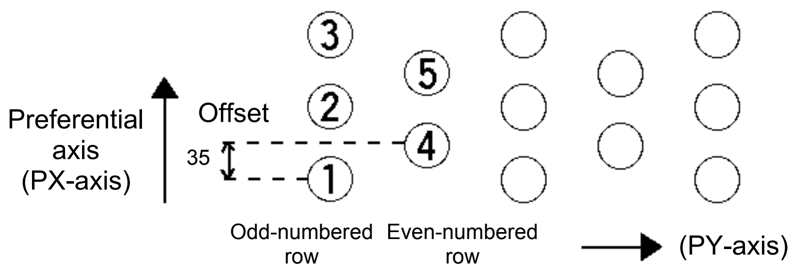
PAST	100		Teach position data No. 100 as the start point.
PAPT	45	30	The PX-axis direction pitch is 45 mm and the PY-axis direction pitch is 30 mm.
PASE	2	1	Set axis 2 (Y-axis) as the preferential axis (PX-axis) and axis 1 (X-axis) as the axis perpendicular to the preferential axis.

(Note) When the palletizing axes, palletizing pitches and palletizing reference point are used, the PX-axis and PY-axis must be parallel with the work coordinate system axes and crossing with each other. In the example shown above, work coordinate system No. 0 (base coordinate system) is selected.

Select either method A or B for each palletizing setting.

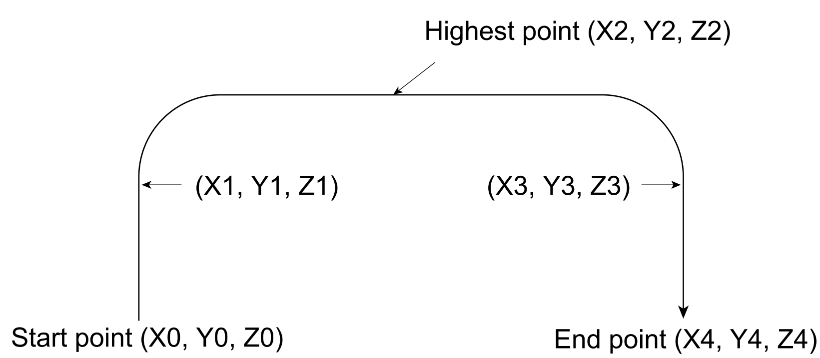
(5) Zigzag setting --- Command: PSLI
Use a PSLI command to set a zigzag layout as shown below.
Zigzag offset: Offset amount in the preferential-axis direction, which will be applied when even-numbered rows are placed.
"Even-numbered rows" refer to the rows occurring at the even numbers based on the row placed first representing the first row.

Zigzag count: Number in the even-numbered rows. Two in the diagram below.



PSLI 35 2

(6) Arch-motion setting
(a) Arch-motion Z-axis number --- Command: ACHZ
(b) Arch-motion Z-axis offset --- Command: OFAZ
(c) Arch-motion composition --- Command: AEXT
Composition data refers to position data of any additional axis you wish to use in arch-motion operation, other than the valid end-point axes or arch-motion Z-axis. Examples include rotation angle.
Note that operation of the composite axis will start and end above the arch triggers.
In an arch-motion composition setting command, specify a position number storing arch-motion composition data.
(d) Arch triggers --- Command: ATRG
The arch-trigger settings used for arch motion include the items specified below.
In an arch-trigger setting command, specify position numbers storing arch-trigger coordinate data.
(d-1) Start-point arch trigger
Specify when to start moving in other axis direction after the start of arch motion from the start point, as an arch-motion Z-direction coordinate position reached.
Start-point arch trigger = Z1
(d-2) End-point arch trigger
Specify when to end moving in other axis direction during downward arch motion, as an arch-motion Z-direction coordinate position reached.
End-point arch trigger = Z3



(7) Palletizing arch-motion setting
- (a) Palletizing Z-direction axis number --- Command: PCHZ
- (b) Palletizing Z-axis offset --- Command: OFPZ
- (c) Palletizing composition --- Command: PEXT
  Composition data refers to position data of any additional axis you wish to use with palletizing movement commands, other than the PX, PY (and PZ)-axes. Use a PEXT command to set the R-axis position coordinates of a given palletizing position.
  Note that operation of the composite axis will start and end above the palletizing arch triggers.
  In a palletizing-composition setting command, specify a position number storing palletizing composition data.

- (d) Palletizing arch triggers --- Command: PTRG
  If the end point is a palletizing point, a palletizing arch trigger must be set just like an arch trigger.
  In a palletizing arch-trigger setting command, specify position numbers storing palletizing arch-trigger coordinate data.
  (d-1) Palletizing start-point arch trigger
  (d-2) Palletizing end-point arch trigger

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 (excluding ARCH) --- 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
Angle formed by the physical axis and the palletizing preferential axis (PX-axis) (θ in the figure below).
θ indicates an angle calculated by ignoring the coordinate in the palletizing Z-axis direction.
In the figure below, θ will become a positive value if axis 1 is used as the reference for angle calculation.

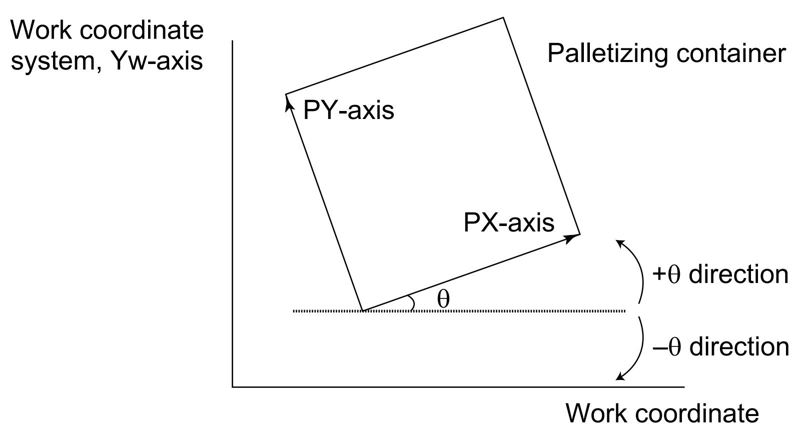


Fig. 4

If the composite axis is a rotating axis, obtaining the palletizing angle and adding it to the composite-axis operation as an offset will allow correction of the composite axis against positional shift of the palletizing container.
Executing a “get palletizing angle” command following a palletizing setting via 3-point teaching will automatically obtain the palletizing angle.
If the setting by 3-point teaching was done three-dimensionally, a palletizing Z-axis must be specified (PCHZ).

(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.

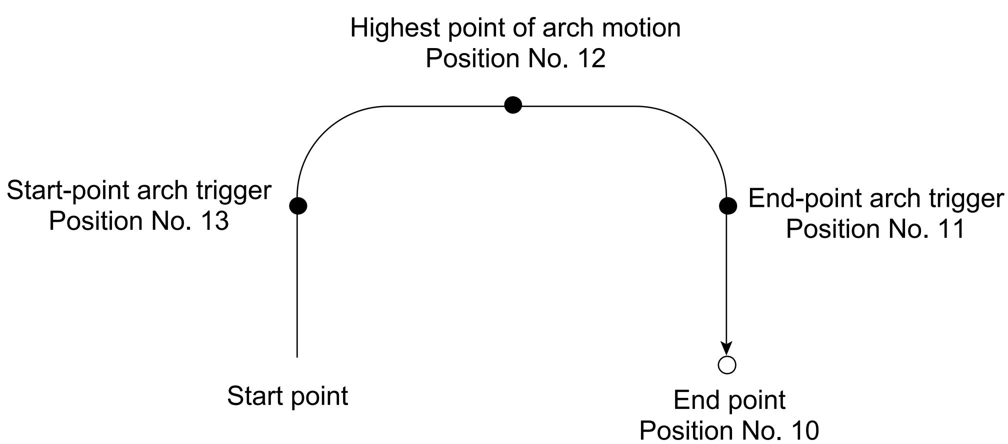
4. Palletizing Movement

Palletizing movement commands include those used to move to a palletizing point and one used to move to an end point specified by position data.

(1) Movement commands to palletizing point --- PMVP, PMVL, PACH

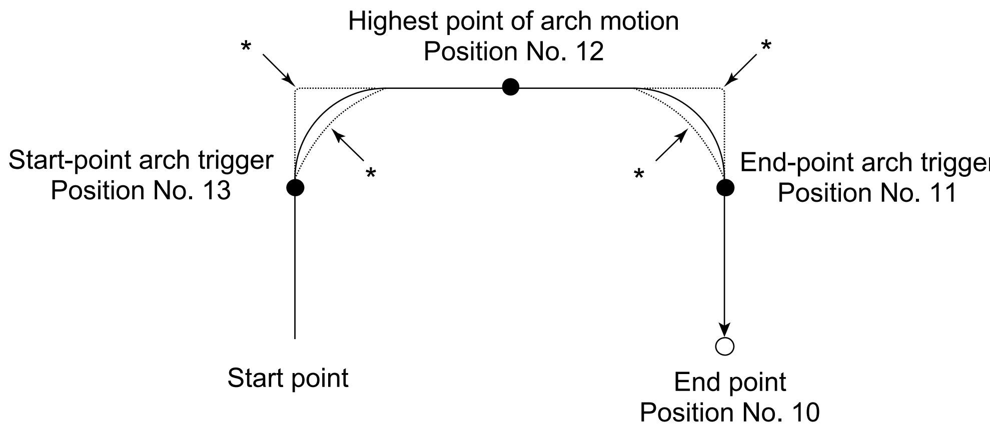
Position coordinates of a two-dimensionally or three-dimensionally placed palletizing point are calculated and movement is performed using the calculated point as the end point. (The axes will move to the palletizing point of the palletizing position number specified in the executed command.)

PMVP: Move from the current position to a palletizing point via PTP.
PMVL: Move from the current position to a palletizing point via interpolation.
PACH: Move from the current position to a palletizing point via arch motion.
Palletizing arch motion must be set in a palletizing setting.




```
PCHZ    3
PTRG    11    13
 |
 |
 |
PACH    1     12
```

(2) Movement comment based on end point specified by point data --- ARCH
Perform arch motion using an end point specified by position data.
In the case of a linear movement in parallel with an actuator, operation can be performed only with two axes including the applicable axis and the PZ-axis.
Arch motion must be set.




```
ACHZ    3
ATRG    13    11
 |
 |
 |
ARCH    10    12
```

5. Program Examples

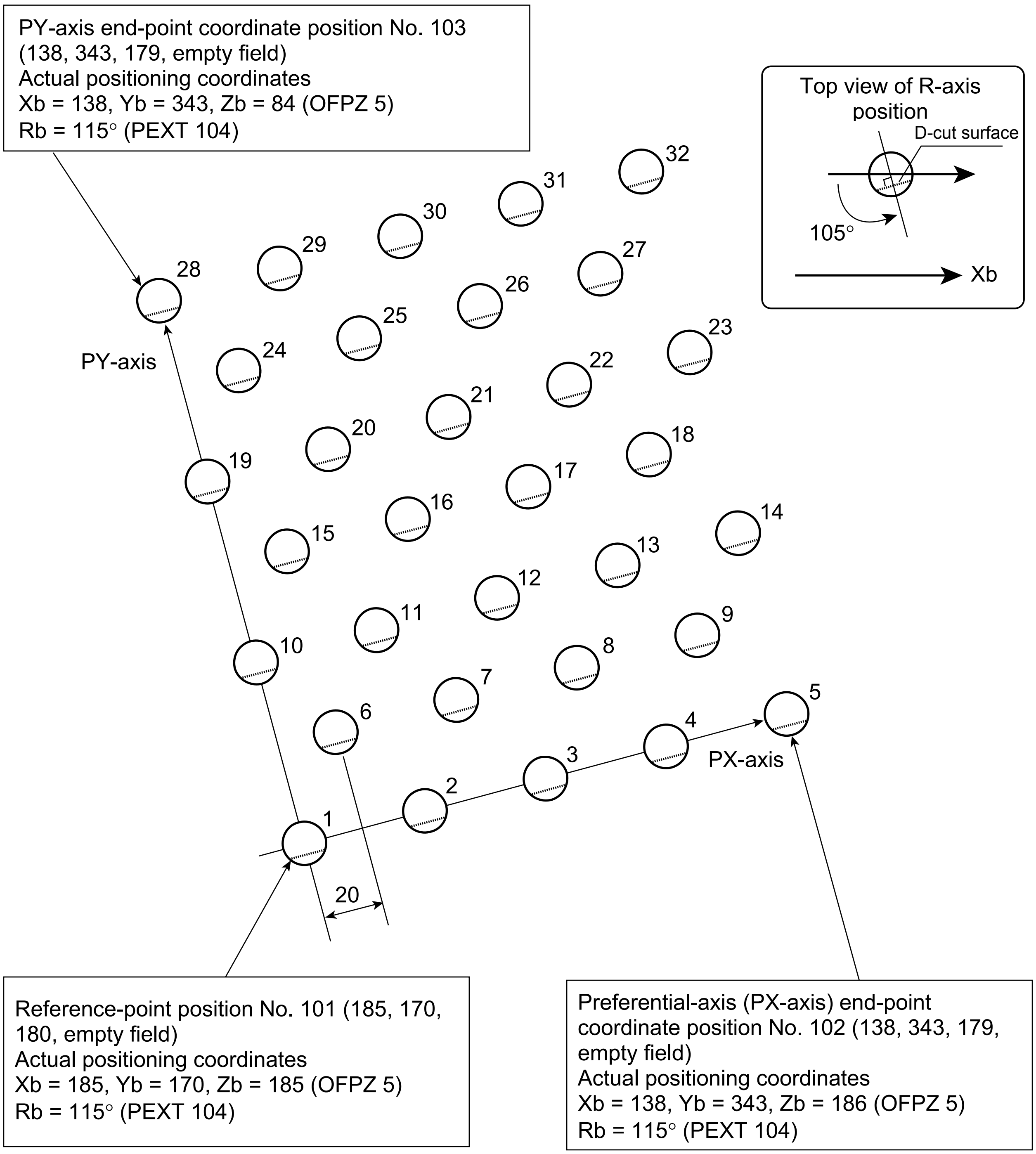
(1) Program example using PAPS (set by 3-point teaching)
The example below specifies movement only and does not cover picking operation.

Step	E	N	Cnd	Cmnd	Operand 1	Operand 2	Pst	Comment
1				VELS	80			PTP travel speed: 80%
2				ACCS	50			PTP travel acceleration: 50%
3				DCLS	50			PTP travel deceleration: 50%
4				VEL	100			CP travel speed: 100 mm/sec
5				ACC	0.3			CP travel acceleration: 0.3 G
6				DCL	0.3			CP travel deceleration: 0.3 G
7				SLWK	0			Select work coordinate system No. 0.
8				SLTL	0			Select tool coordinate system No. 0.
9								
10				BGPA	1			Start setting palletizing No. 1.
11				PAPI	5	7		Palletizing counts: 5 x 7
12				PAPS	101			Set by 3-point teaching.
13				PEXT	104			Set palletizing R-axis coordinate.
14				PSLI	20	4		Zigzag offset = 20 mm
15				PAPN	1			Palletizing pattern 1
16				PCHZ	3			Palletizing Z-axis = Axis 3
17				PTRG	105	105		Set palletizing arch triggers.
18				OFPZ	5			PZ-axis offset = 5 mm
19				EDPA				
20								
21				ATRG	105	105		Set arch triggers.
22				ACHZ	3			Arch-motion Z-axis = Axis 3
23								
24				PTPL				Perform positioning in PTP mode using left arm.
25				MOVP	110			Move to picking position in PTP mode.
26				PSET	1	1		Set palletizing position number to 1.
27				TAG	1			Beginning of loop processing
28				PACH	1	106		Palletizing arch motion
29				ARCH	110	106		Arch motion
30				PINC	1		600	Increment palletizing position number by 1.
31			600	GOTO	1			Beginning of loop when PINC is successful.
32				MOVL	109			Move to standby position in CP mode.
33				EXIT				End

Position data (Stroke with arm length 500)

No.(Name)	Axis1	Axis2	Axis3	Axis4	
101( )	185.000	170.000	180.000		Reference-point position
102( )	340.000	211.000	181.000		PX-axis end point
103( )	138.000	343.000	179.000		PY-axis end point
104( )				105.000	Palletizing R-axis position
105( )			100.000		Arch/palletizing trigger position
106( )			80.000		Highest position (Z point)
107( )					
108( )					
109( )	0.000	160.000	0.000	0.000	Standby position
110( )	-200.000	330.000	180.000	0.000	Pickup position

Schematic diagram of palletizing positions based on the above program



The number shown at top right of each cycle indicates the corresponding palletizing position number.
Count in PX-axis direction = 5, count in PY-axis direction = 7
Zigzag offset: 20, zigzag count: 4

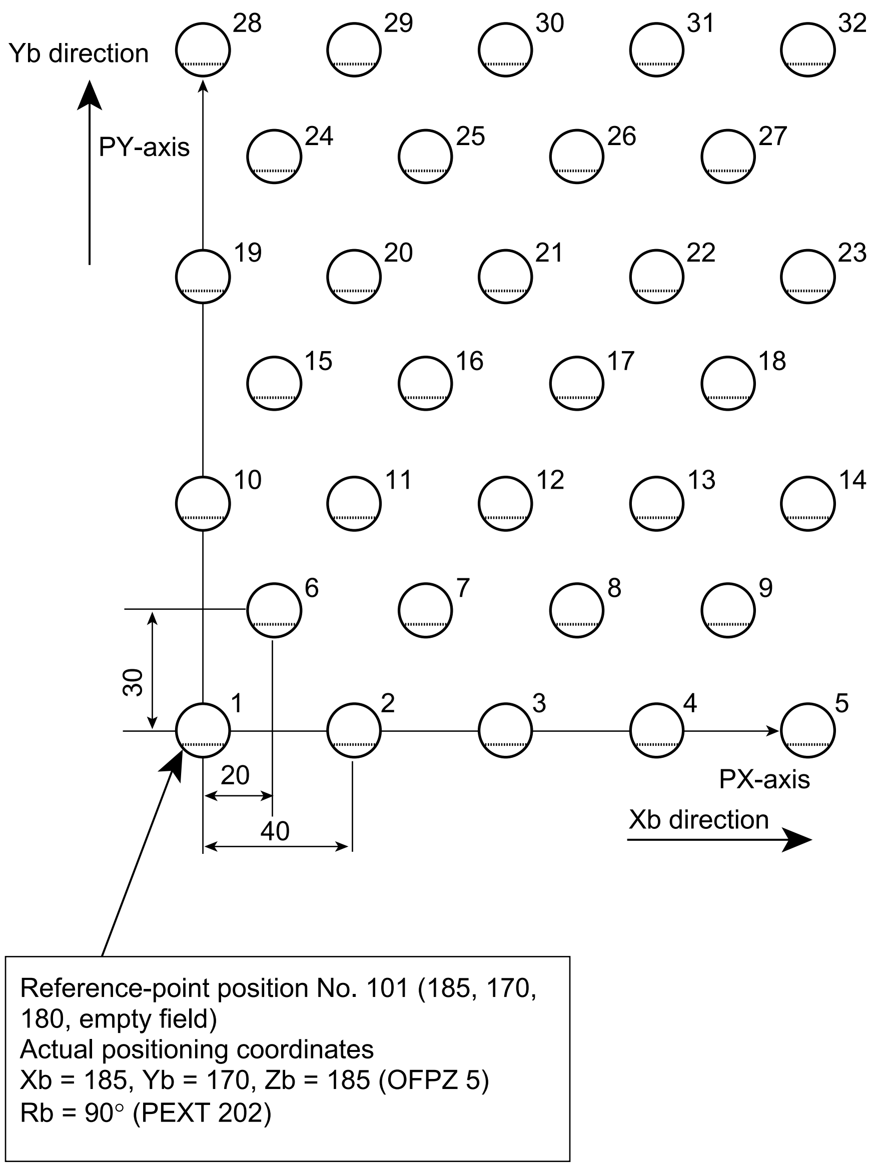
(2) Program example using PASE, PAPT and PAST
The example below specifies movement only and does not cover picking operation.

Step	E	N	Cnd	Cmnd	Operand 1	Operand 2	Pst	Comment
1				VELS	80			PTP travel speed: 80%
2				ACCS	50			PTP travel acceleration: 50%
3				DCLS	50			PTP travel deceleration: 50%
4				VEL	100			CP travel speed: 100 mm/sec
5				ACC	0.3			CP travel acceleration: 0.3 G
6				DCL	0.3			CP travel deceleration: 0.3 G
7				SLWK	0			Select work coordinate system No. 0.
8				SLTL	0			Select tool coordinate system No. 0.
9								
10				BGPA	1			Start setting palletizing No. 1.
11				PAST	201			Set reference-point data.
12				PASE	1	2		PX-axis = X-axis, PY-axis = Y-axis
13				PAPT	40	30		Pitch PX: 40, PY: 30
14				PAPI	5	7		Palletizing counts: 5 x 7
15				PSLI	20	4		Zigzag offset = 20 mm, count = 4
16				PEXT	202			Set palletizing R-axis coordinate.
17				PCHZ	3			Palletizing Z-axis = Axis 3
18				PTRG	203	203		Set palletizing arch triggers.
19				OFPZ	5			PZ-axis offset = 5 mm
20				EDPA				
21								
22				ATRG	203	203		Set arch triggers.
23				ACHZ	3			Arch-motion Z-axis = Axis 3
24								
25				PTPL				Perform positioning in PTP mode using left arm.
26				MOVP	208			Move to picking position in PTP mode.
27				PSET	1	1		Set palletizing position number to 1.
28				TAG	1			Beginning of loop processing
29				PACH	1	204		Palletizing arch motion
30				ARCH	208	204		Arch motion
31				PINC	1		600	Increment palletizing position number by 1.
32			600	GOTO	1			Beginning of loop when PINC is successful.
33				MOVL	207			Move to standby position in CP mode.
34				EXIT				End

Position data (Stroke with arm length 500)

No.(Name)	Axis1	Axis2	Axis3	Axis4	
201( )	185.000	170.000	180.000		Reference-point position
202( )				90.000	Palletizing R-axis position
203( )			100.000		Arch/palletizing trigger position
204( )			80.000		Highest position (Z point)
205( )					
206( )					
207( )	0.000	160.000	0.000	0.000	Standby position
208( )	-200.000	330.000	180.000	0.000	Pickup position

Schematic diagram of palletizing positions based on the above program
(The PX and PY-axes are parallel with Xb and Yb (base coordinates), respectively.)



The number shown at top right of each cycle indicates the corresponding palletizing position number.
Count in PX-axis direction = 5, count in PY-axis direction = 7
Pitch in PX-axis direction: 40
Pitch in PY-axis direction: 30
Zigzag offset: 20, zigzag count: 4

Chapter 6 Pseudo-Ladder Task

With the X-SEL Controller, a pseudo-ladder task function can be used depending on the command and extension condition.
The input format is shown below.

1. Basic Frame

Extension condition	Input condition		Command	Operand 1	Operand 2	Output	
E	N	Cnd				Pst	
LD		7001	CHPR	1			
			TPCD	1			
			TAG	1			
⋮		⋮	⋮	⋮			Ladder statement field
LED		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.

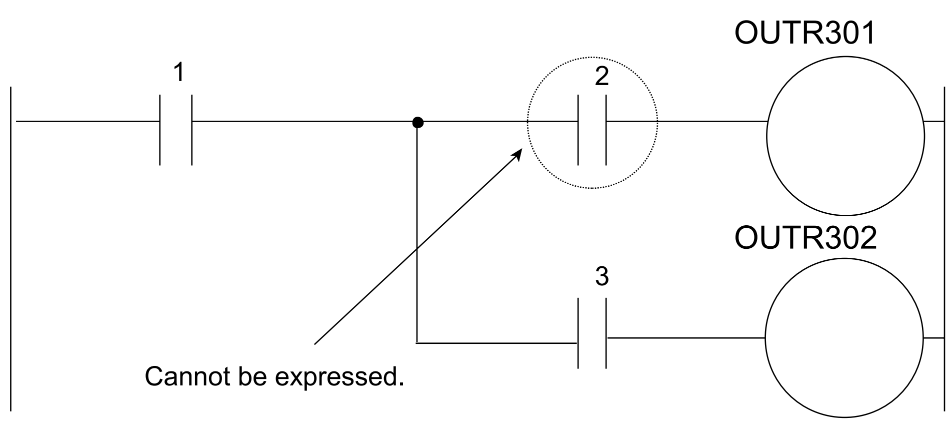
(2) Ladder commands

OUTR	....	Ladder output relay (Operand 1 = Output, flag number)
TIMR	....	Ladder timer relay (Operand 1 = Local flag number, Operand 2 = Timer setting (sec))

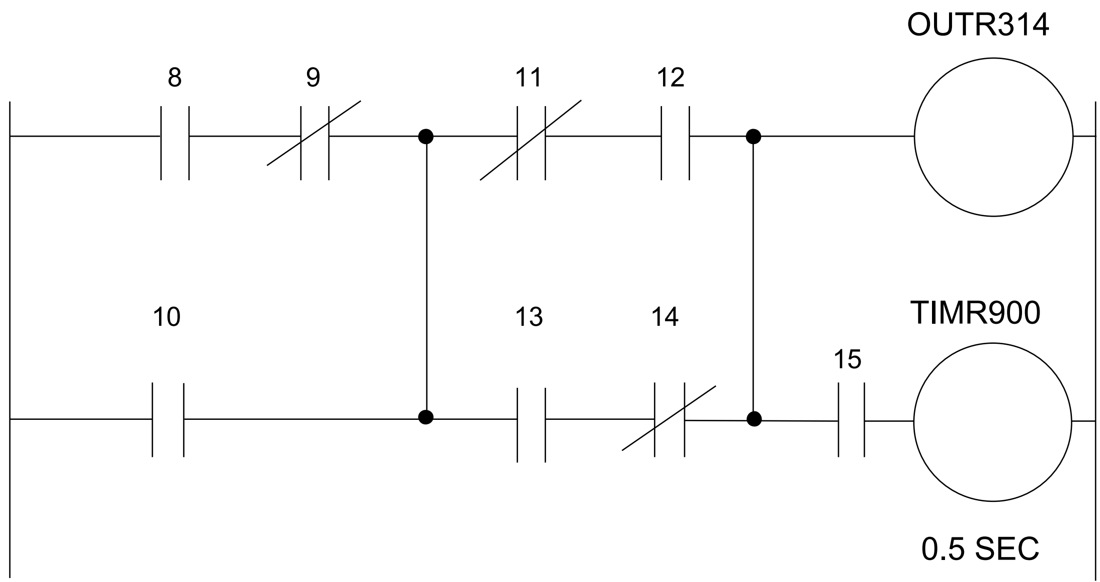
3. Points to Note

- 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	Input condition		Command	Operand 1	Operand 2	Output
No.	E	N	Cnd				Pst
1	LD		7001	CHPR	1		
2				TPCD	1		
3				TAG	1		
4							
5	LD		8				
6	A	N	9				
7	O		10				
8	LD	N	11				
9	A		12				
10	LD		13				
11	A	N	14				
12	OB						
13	AB			OUTR	314		
14	A		15	TIMR	900	0.5	
15							
16	LD		7001	TSLP	3		
17	LD		7001	GOTO	1		
18	LD		7001	EXIT			

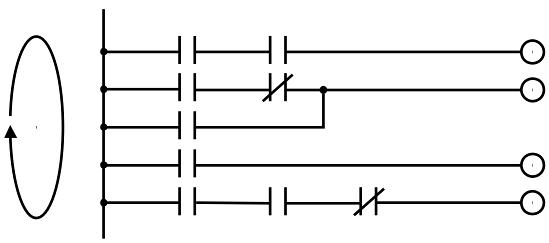
Chapter 7 Multi-Tasking

"Multi-tasking" operation means running several programs in parallel.

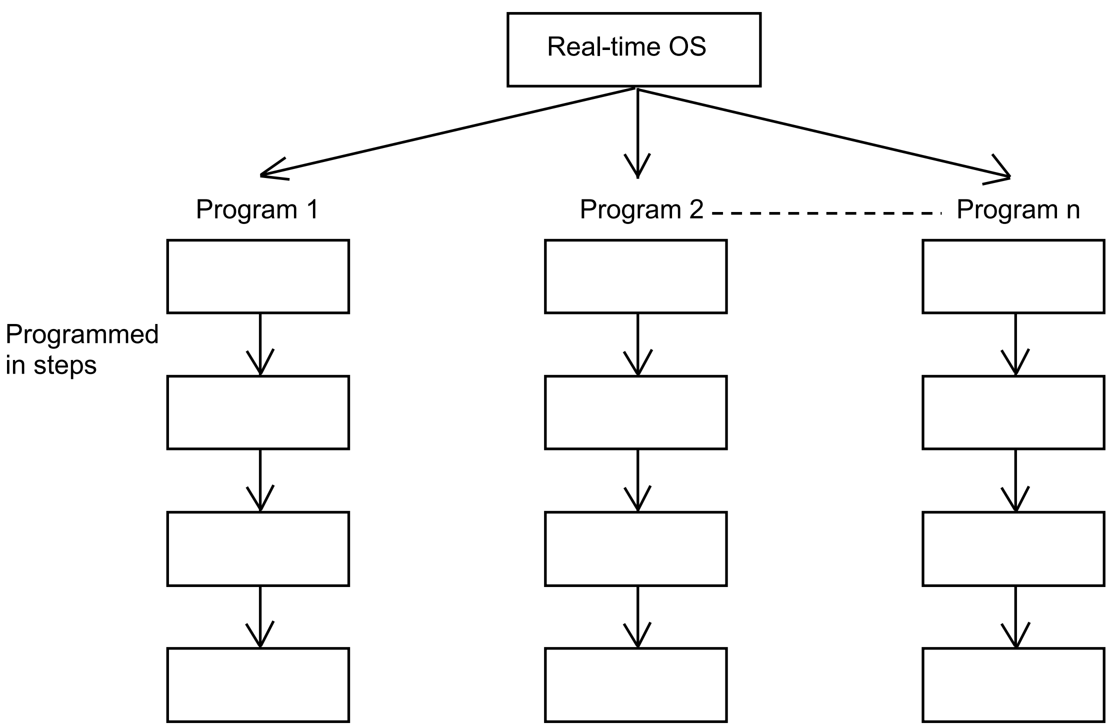
1. Difference from a Sequencer

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.



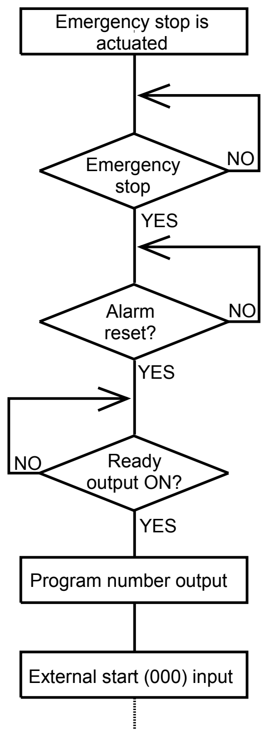
The programmer need not worry about running all programs in parallel, which is controlled by the real-time operating system.

2. 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

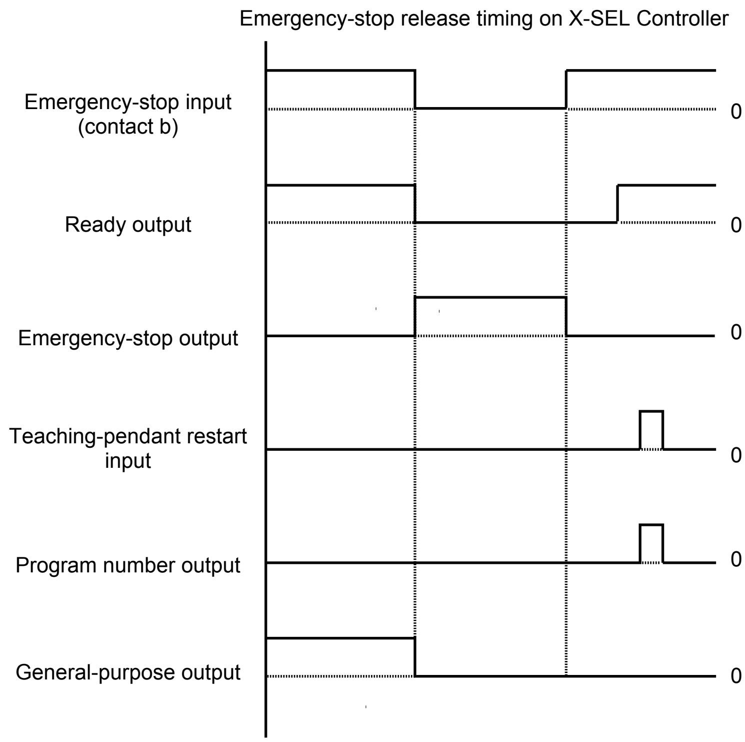
An emergency stop is actuated by turning the emergency-stop contact b input to OFF, and released by turning the input to ON.

(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.

3. 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 1, “Operation” (Starting via External Signal Selection) in Part 2, “Operation.”

(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 origin 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.

Appendix

◎ Expansion Board (Optional)

Only one optional board can be installed on the compact type.

Type: IA-103-X-32

Pin No.	Category	Port No.	Function
1	Input	-	General-purpose: NC, Compact: +24-V input
2			General-purpose input
3			General-purpose input
4			General-purpose input
5			General-purpose input
6			General-purpose input
7			General-purpose input
8			General-purpose input
9			General-purpose input
10			General-purpose input
11			General-purpose input
12			General-purpose input
13			General-purpose input
14			General-purpose input
15			General-purpose input
16			General-purpose input
17			General-purpose input
18			General-purpose input
19			General-purpose input
20			General-purpose input
21			General-purpose input
22			General-purpose input
23			General-purpose input
24			General-purpose input
25			General-purpose input
26			General-purpose input
27			General-purpose input
28			General-purpose input
29			General-purpose input
30			General-purpose input
31			General-purpose input
32			General-purpose input
33			General-purpose input
34	Output		General-purpose output
35			General-purpose output
36			General-purpose output
37			General-purpose output
38			General-purpose output
39			General-purpose output
40			General-purpose output
41			General-purpose output
42			General-purpose output
43			General-purpose output
44			General-purpose output
45			General-purpose output
46			General-purpose output
47			General-purpose output
48			General-purpose output
49			General-purpose output
50		-	General-purpose: NC, Compact: 0 V

Type: IA-103-X-16

Pin No.		Port No.	Function
1	Input	-	General-purpose: NC, Compact: +24-V input
2			General-purpose input
3			General-purpose input
4			General-purpose input
5			General-purpose input
6			General-purpose input
7			General-purpose input
8			General-purpose input
9			General-purpose input
10			General-purpose input
11			General-purpose input
12			General-purpose input
13			General-purpose input
14			General-purpose input
15			General-purpose input
16			General-purpose input
17			General-purpose input
18	Output		General-purpose output
19			General-purpose output
20			General-purpose output
21			General-purpose output
22			General-purpose output
23			General-purpose output
24			General-purpose output
25			General-purpose output
26			General-purpose output
27			General-purpose output
28			General-purpose output
29			General-purpose output
30			General-purpose output
31			General-purpose output
32			General-purpose output
33			General-purpose output
34			General-purpose output
35			General-purpose output
36			General-purpose output
37			General-purpose output
38			General-purpose output
39			General-purpose output
40			General-purpose output
41			General-purpose output
42			General-purpose output
43			General-purpose output
44			General-purpose output
45			General-purpose output
46			General-purpose output
47			General-purpose output
48			General-purpose output
49			General-purpose output
50		-	General-purpose: NC, Compact: 0 V

◎ Expanded SIO Specification (Optional)

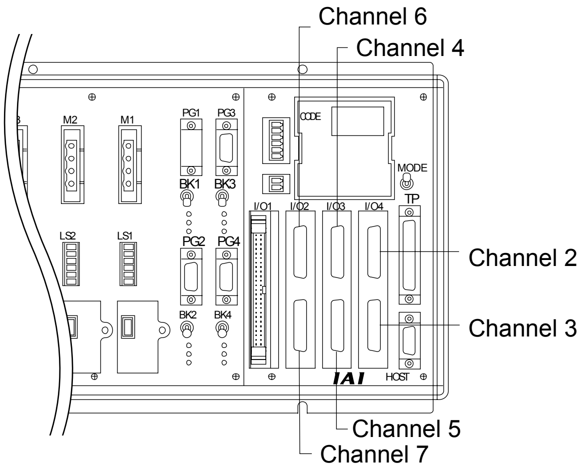
Type	IA-105X-MW-A	(for RS232 connection)
	IA-105X-MW-B	(for RS422 connection)
	IA-105X-MW-C	(for RS485 connection)

(1) Specification
An SIO expansion board is a D-sub, 15-pin interface used for both RS232 and RS422 connection. The terminal assignments allow selection of RS232, RS422 or RS485 in accordance with the wiring. (The controller is shipped with a joint cable appropriate for the selected communication method.)

Specifications

<table>
<tr><th>Item</th><th colspan="4">Description</th></tr>
<tr><td>Connector</td><td colspan="4">D-sub, 15-pin (male D-sub connector)</td></tr>
<tr><td>Connector name</td><td colspan="4">CN3 connector</td></tr>
<tr><td>Maximum baud rate</td><td colspan="4">115.2K</td></tr>
<tr><td>Maximum connection distance</td><td colspan="3">10M (RS232) / 600M (RS422)</td><td>38.4K (RS232C), 115.2K (RS422)</td></tr>
<tr><td>Interface standards</td><td colspan="4">RS232 / RS422 / RS485</td></tr>
<tr><td>Communication method</td><td colspan="4">Start-stop synchronous full/half-duplex communication</td></tr>
<tr><td>Character length</td><td colspan="4">7/8 bits</td></tr>
<tr><td>Start bit</td><td colspan="4">1 bit</td></tr>
<tr><td>Stop bit</td><td colspan="4">1/2 bits</td></tr>
<tr><td>Parity</td><td colspan="4">None, even, odd</td></tr>
<tr><td>Connection cable</td><td colspan="4">Special 15-pin cable</td></tr>
<tr><td rowspan="15">Terminal assignments</td><td>1</td><td>Out</td><td>SD (TXD)</td><td>Transmitted data</td></tr>
<tr><td>2</td><td>In</td><td>RD (RXD)</td><td>Received data</td></tr>
<tr><td>3</td><td>Out</td><td>RS (RTS)</td><td>Request to send</td></tr>
<tr><td>4</td><td>In</td><td>CS (CTS)</td><td>Clear to send</td></tr>
<tr><td>5</td><td>Out</td><td>DR (DTR)</td><td>Equipment ready</td></tr>
<tr><td>6</td><td>In</td><td>ER (DSR)</td><td>Data set ready</td></tr>
<tr><td>7</td><td></td><td>SG</td><td>Signal ground</td></tr>
<tr><td>8</td><td></td><td>N.C.</td><td>Not used</td></tr>
<tr><td>9</td><td></td><td>N.C.</td><td>Not used</td></tr>
<tr><td>10</td><td></td><td>N.C.</td><td>Not used</td></tr>
<tr><td>11</td><td>In</td><td>RDA (RXD +)</td><td>Received data</td></tr>
<tr><td>12</td><td>In</td><td>RDB (RXD –)</td><td>Received data</td></tr>
<tr><td>13</td><td>In</td><td>TRM</td><td>Termination connection</td></tr>
<tr><td>14</td><td>Out</td><td>SDB (TXD –)</td><td>Transmitted data</td></tr>
<tr><td>15</td><td>Out</td><td>SDA (TXD +)</td><td>Transmitted data</td></tr>
</table>

(2) Parameter setting
The channel numbers and specifications of the expanded SIOs are set as follows prior to shipment using parameters:



Specifications	
Baud rate:	38.4 kbps
Data length:	8
Stop bit length:	1
Parity type:	None
Communication mode:	RS232

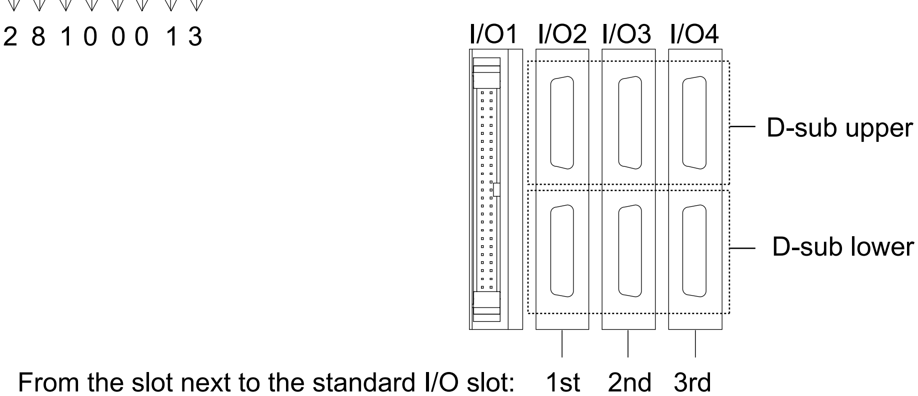
Channel numbers are determined by the corresponding parameter numbers set.

Channel 2 → I/O parameter No. 100	Channel 3 → I/O parameter No. 102
Channel 4 → I/O parameter No. 104	Channel 5 → I/O parameter No. 106
Channel 6 → I/O parameter No. 108	Channel 7 → I/O parameter No. 110

1. Parameter content
Example) When '28100013' is set in I/O parameter No. 100
The specification of channel 2 is shown below (the boxed portions indicate the current settings):

- Baud rate: 0: 9.6 kbps, 1: 19.2 kbps, $\boxed{\text{2: 38.4 kbps}}$, 3: 57.6 kbps, 4: 76.8 kbps, 5: 115.2 kbps
- Data length: 7, $\boxed{8}$
- Stop bit length: $\boxed{1}$, 2
- Parity type: $\boxed{\text{0:None}}$, 1: Odd, 2: Even
- Communication mode: $\boxed{\text{0: RS232}}$, 1: RS422, 2: RS485
- Half-duplex receive operation type (valid only when RS485 is selected): $\boxed{\text{0: Enable receive after send}}$, 1: Do not enable receive after send
- Channel assignment on board: $\boxed{\text{1: D-sub upper}}$, 2: D-sub lower (See the figure below)
- Expansion I/O slot assignment: $\boxed{\text{3rd}}$ from next to standard I/O (See the figure below)

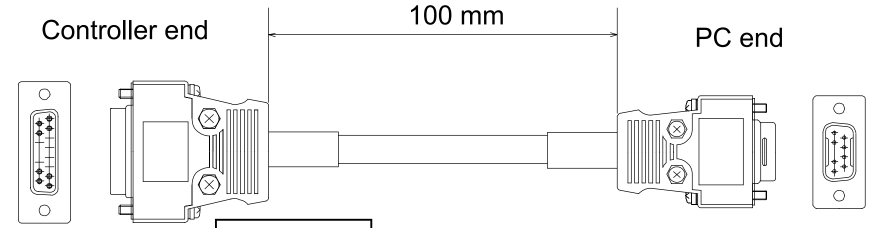
2 8 1 0 0 0 1 3



(3) Joint cable

1. For RS232C connection

Type: CB-ST-232J001



Wiring diagram

XM2D-1501

Wire type	Color	Signal	No.
AWG24 x 7 cores	Orange/black dot	SD	1
	Orange/black dot	RD	2
	Light gray/black dot	RS	3
	Light gray/red dot	CS	4
	White/black dot	ER	5
	White/red dot	DR	6
	Yellow/black dot	SG	7
			8
			9
			10
			11
			12
			13
			14
			15

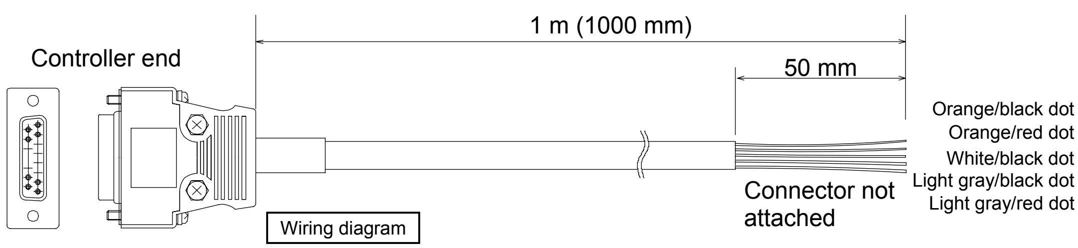
XM2A-0901

No.	Signal	Color	Wire type
3	SD	Orange/black dot	AWG24 x 7 cores
2	RD	Orange/black dot	
7	RS	Light gray/black dot	
8	CS	Light gray/red dot	
4	ER	White/black dot	
6	DR	White/red dot	
5	SG	Yellow/black dot	
1			
9			

* The user must provide connection cables with external equipment.

2. For RS422/RS485 connection

Type: CB-ST-422J010



Wiring diagram

XM2D-1501

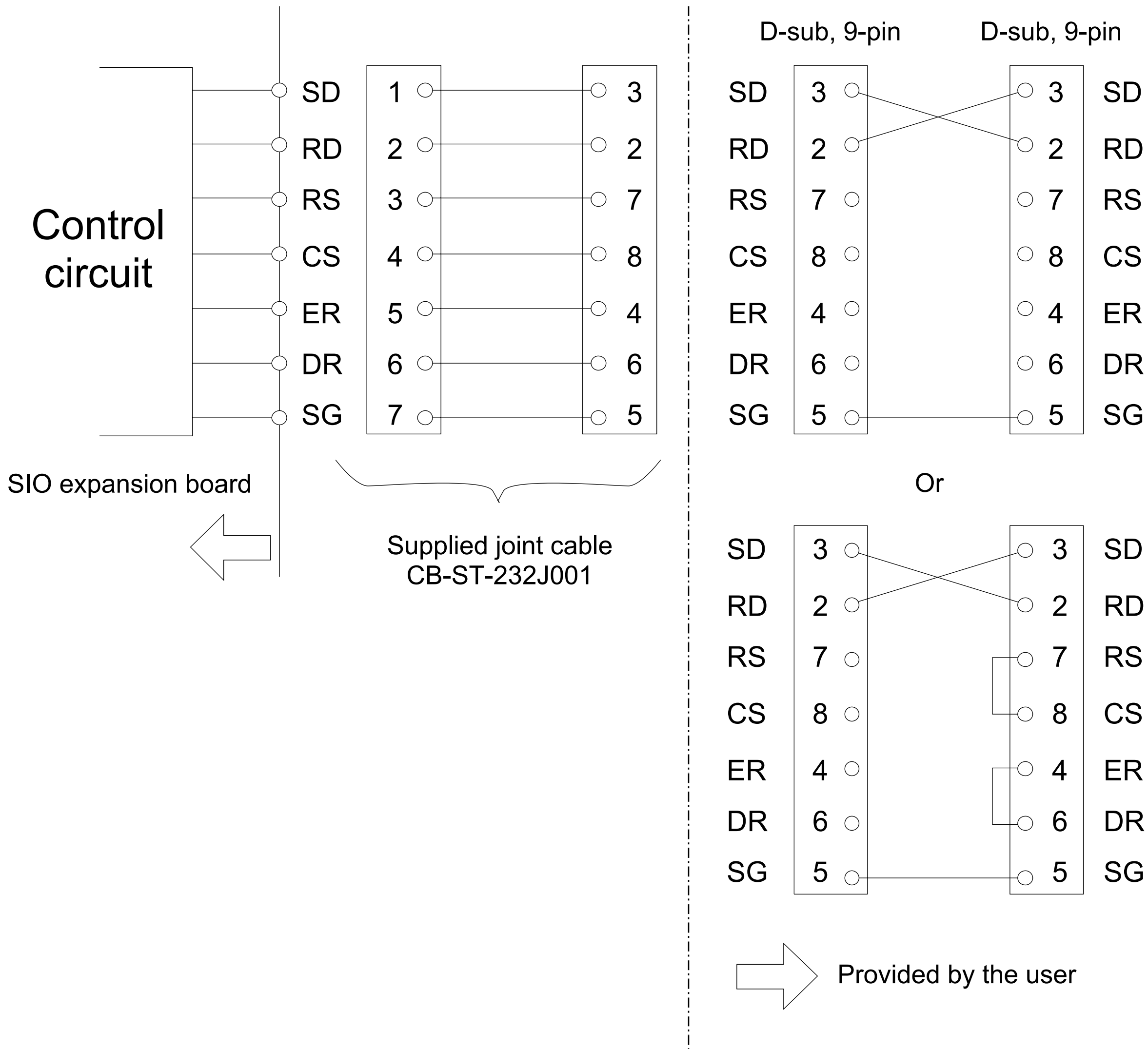
Wire type	Color	Signal	No.
AWG24 x 7 cores			1
			2
			3
			4
			5
			6
			7
			8
			9
			10
	Orange/black dot	RD +	11
	Orange/red dot	RD -	12
	White/black dot	TRM	13
	Light gray/black dot	SD -	14
	Light gray/red dot	SD +	15

* Connect to a terminal block, etc.

* Termination: The user must terminate the RS422/RS485 transmission lines.

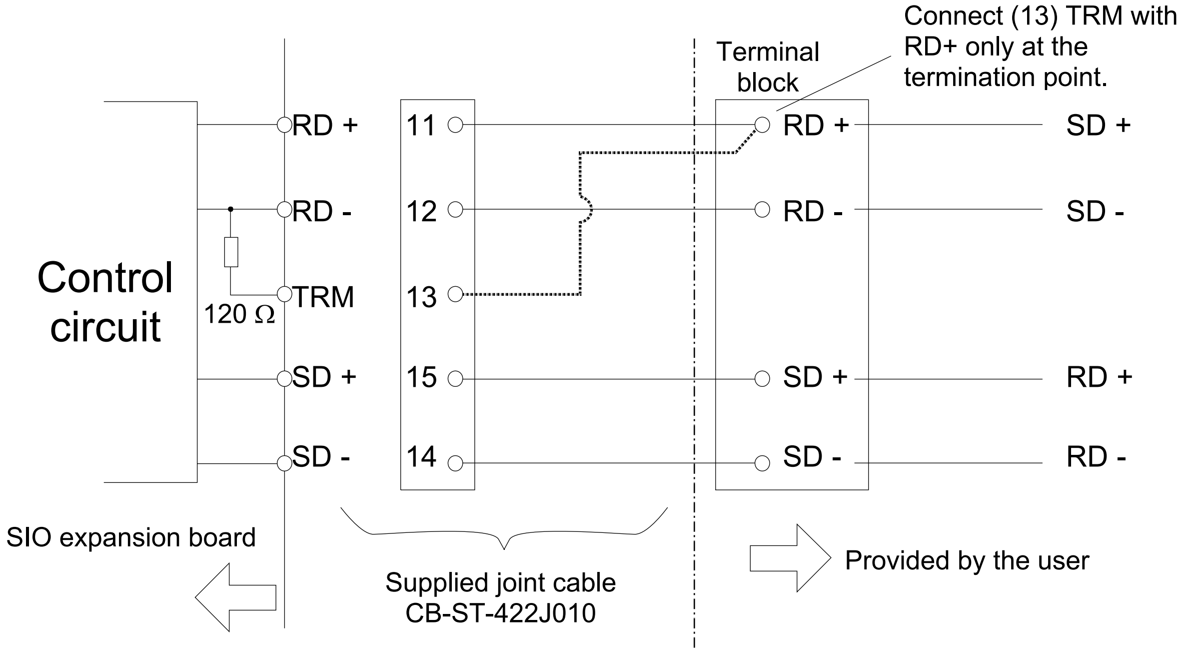
(4) Wiring

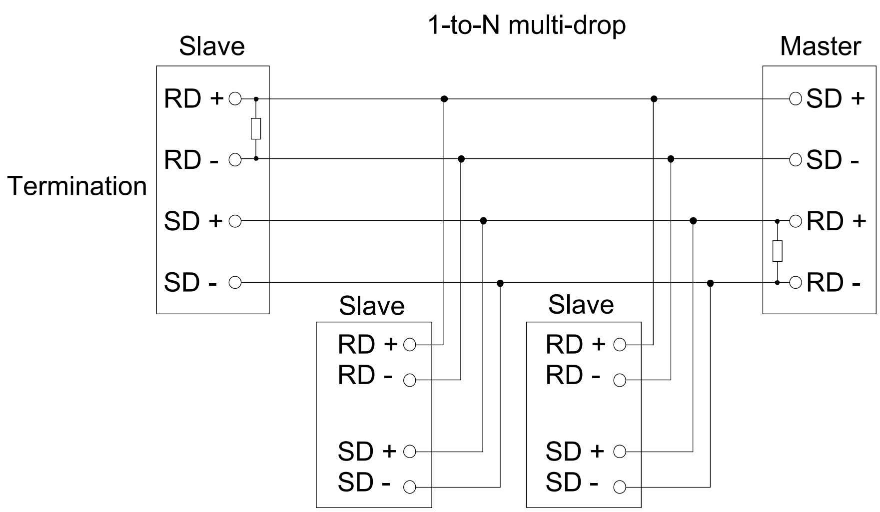
1. RS232C wiring



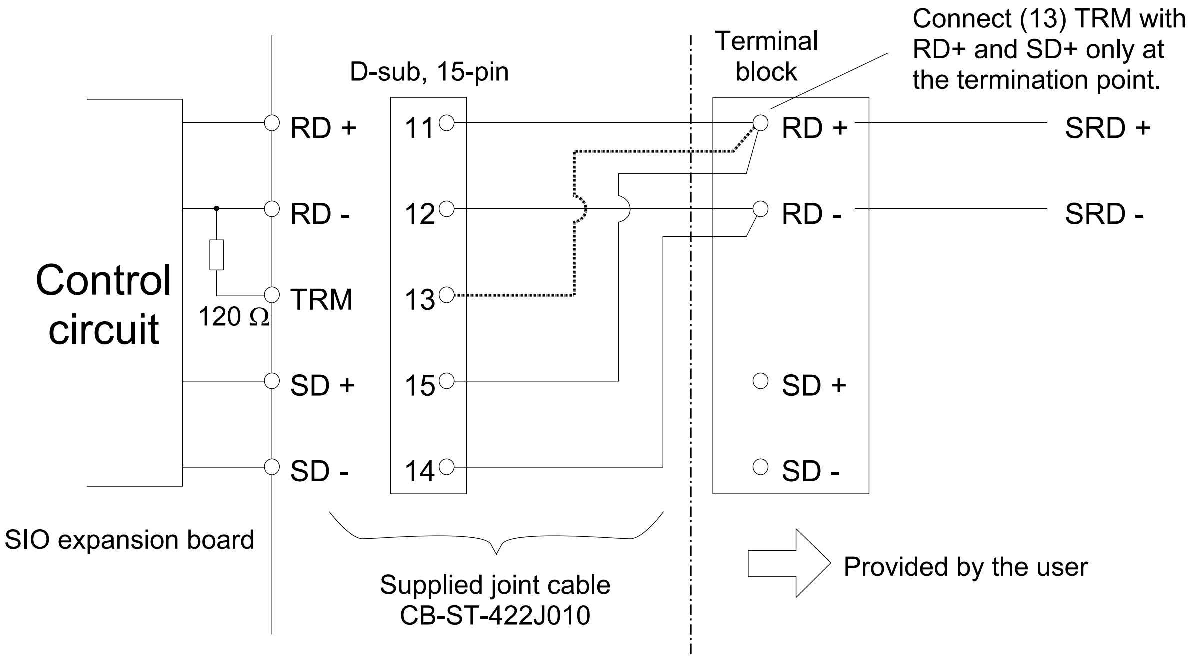
Flow control (software flow, hardware flow) is not provided. If the other side does not use flow control, commercial RS232C cross cables can be used.

2. RS422 wiring

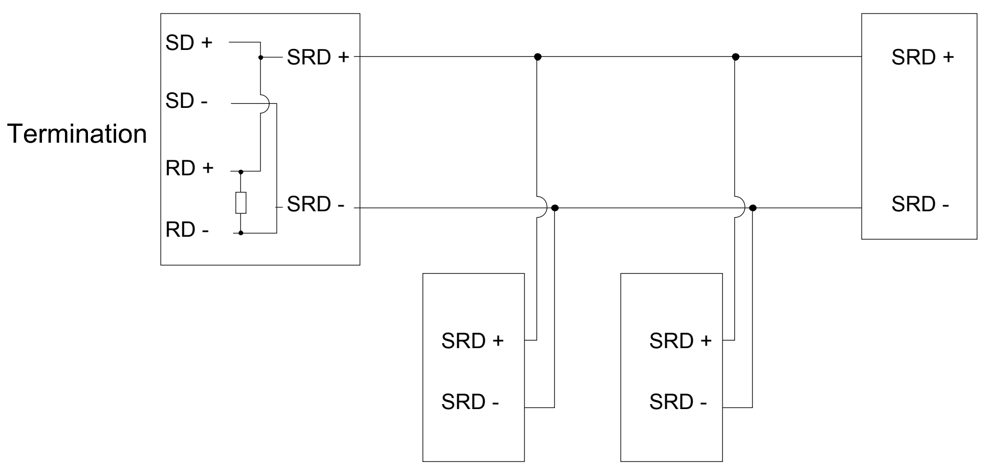




3. RS485 wiring



N-to-N multi-drop



◎ 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.

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: Slot 1 (standard I/O) ~ * Ports are assigned only for the installed adjoining slots, starting from slot 1 = 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	Input port start number with fixed expanded I/O1 assignments (I/O2)	-1	-1 ~ 599		0 + (Multiple of 8) (Invalid if a negative value is set) (Slot next to the standard I/O slot)
5	Output port start number with fixed expanded I/O1 assignments (I/O2)	-1	-1 ~ 599		300 + (Multiple of 8) (Invalid if a negative value is set)
6	Input port start number with fixed expanded I/O2 assignments (I/O3)	-1	-1 ~ 599		0 + (Multiple of 8) (Invalid if a negative value is set)
7	Output port start number with fixed expanded I/O2 assignments (I/O3)	-1	-1 ~ 599		300 + (Multiple of 8) (Invalid if a negative value is set)
8	Input port start number with fixed expanded I/O3 assignments (I/O4)	-1	-1 ~ 599		0 + (Multiple of 8) (Invalid if a negative value is set)
9	Output port start number with fixed expanded I/O3 assignments (I/O4)	-1	-1 ~ 599		300 + (Multiple of 8) (Invalid if a negative value is set)
10	Standard I/O error monitor (I/O1)	1	0 ~ 5		0: Do not monitor, 1: Monitor * Some exceptions apply.
11	Expanded I/O1 error monitor (I/O2)	1	0 ~ 5		0: Do not monitor, 1: Monitor (Slot next to the standard I/O slot) * Some exceptions apply.
12	Expanded I/O2 error monitor (I/O3)	1	0 ~ 5		0: Do not monitor, 1: Monitor * Some exceptions apply.
13	Expanded I/O3 error monitor (I/O4)	1	0 ~ 5		0: Do not monitor, 1: Monitor * Some exceptions apply.
14	Number of ports using network-I/F-card remote input	0	0 ~ 256		Multiple of 8
15	Number of ports using network-I/F-card remote output	0	0 ~ 256		Multiple of 8
16	(For future extension = Change prohibited)	0	0 ~ 256		Multiple of 8
17	(For future extension = Change prohibited)	0	0 ~ 256		Multiple of 8

I/O Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
18~ 19	(For extension)	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	Register input filtering periods	2	1 ~ 9	msec	Input signal is recognized when the status is held for twice the period set by this parameter.
22	Remote-I/O-card fieldbus ready timeout value	2000	0 ~ 99999	msec	Timeout check is not performed if "0" is set.
23	Overcurrent detection input specification for multi-point DIO external terminal block	0H	0H ~ FFFFFFFFH		Bits 12 to 15: Standard I/O (I/O1) input specification (0: Do not input, 1: Input) Bits 8 to 11: Expanded I/O1 (I/O2) input specification (0: Do not input, 1: Input) Bits 4 to 7: Expanded I/O2 (I/O3) input specification (0: Do not input, 1: Input) Bits 0 to 3: Expanded I/O3 (I/O4) input specification (0: Do not input, 1: Input) * If "Input" is specified, the two input ports, namely the 24th and 48th ports on the applicable card (023 and 047 on the card), will be disabled.
24	I/O setting bit pattern 1	0	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 be actuated and hardware error is not present) 1: Error of operation-cancellation level or higher is not present * This setting can be selected only when the main CPU board's hardware versions are MREV = 0/FREV = c or later. 2: Error of cold-start level or higher is not present * This setting can be selected only when the main CPU board's hardware versions are MREV = 0/FREV = c or later.) (Main application version 0.25 or later) 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) (Main application version 0.25 or later)
25~ 29	(For extension)	0			
30	Input function selection 000	1	0 ~ 5		0: General-purpose input 1: Program start signal (ON edge) (007 to 013: BCD-specified program number) 2: Program start signal (ON edge) (007 to 013: Binary-specified program number) * If this parameter is used as a program start signal, turn ON the signal for at least 100 msec so that the program will start without fail.
31	Input function selection 001	0	0 ~ 5		0: General-purpose input 1: Software reset (restart) signal (1 second ON) * If continued operation is specified as the action upon emergency stop, enable the software reset signal (to provide a means of canceling the operation).
32	Input function selection 002	0	0 ~ 5		0: General-purpose input 1: Servo ON * ON edge: Equivalent to the all-valid-axis servo ON command, OFF edge: Equivalent to the all-valid-axis servo OFF command (A minimum interval of 1.5 seconds is required) (Must be executed in non-operating condition)

I/O Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
33	Input function selection 003	1	0 ~ 5		0: General-purpose input 1: General-purpose input (Start the auto-start program upon power-ON reset/software reset in the AUTO mode) 2: Auto-start program start signal (ON edge: Start, OFF edge: Abort all operations/programs (excluding the I/O processing program at operation/program abort)) * If this parameter is used as an auto-start program start signal, turn ON the signal for at least 100 msec so that the program will start without fail.
34	Input function selection 004	0	0 ~ 5		0: General-purpose input 1: All servo axis soft interlock (OFF level) (Valid for all commands other than the servo OFF command) (Operation is held upon interlock actuation during automatic operation; operation is terminated upon interlock in non-AUTO mode)
35	Input function selection 005	0	0 ~ 5		0: General-purpose input, 1: Operation-pause reset signal (ON edge)
36	Input function selection 006	0	0 ~ 5		0: General-purpose input 1: Operation-pause reset signal (OFF level) (Valid only during automatic operation) * Cancel pause when an operation-pause reset signal is received.
37	Input function selection 007	1	0 ~ 5		0: General-purpose input, 1: Program number specified for program start (least significant bit)
38	Input function selection 008	1	0 ~ 5		0: General-purpose input, 1: Program number specified for program start
39	Input function selection 009	1	0 ~ 5		0: General-purpose input, 1: Program number specified for program start
40	Input function selection 010	1	0 ~ 5		0: General-purpose input, 1: Program number specified for program start
41	Input function selection 011	1	0 ~ 5		0: General-purpose input, 1: Program number specified for program start
42	Input function selection 012	1	0 ~ 5		0: General-purpose input, 1: Program number specified for program start
43	Input function selection 013	1	0 ~ 5		0: General-purpose input 1: Program number specified for program start 2: Error reset (ON edge)
44	Input function selection 014	0	0 ~ 5		0: General-purpose input (Cancel cutoff when the drive-source cutoff factor is removed) 1: Drive-source cutoff reset input (ON edge) (Valid when the factor has been removed) * Drive-source cutoff reset control is not available 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.
45	Input function selection 015	0	0 ~ 5		0: General-purpose input
46	Output function selection 300	2	0 ~ 20		0: General-purpose output 1: Output error of operation-cancellation level or higher (ON) 2: Output error of operation-cancellation level or higher (OFF) 3: Output error of operation-cancellation level or higher + emergency stop (ON) 4: Output error of operation-cancellation level or higher + emergency stop (OFF)
47	Output function selection 301	3	0 ~ 20		0: General-purpose output 1: READY output (PIO trigger program can be run) 2: READY output (PIO trigger program can be run and error of operation-cancellation level or higher is not present) 3: READY output (PIO trigger program can be run and error of cold-start level or higher is not present)
48	Output function selection 302	2	0 ~ 20		0: General-purpose output 1: Emergency-stop output (ON) 2: Emergency-stop output (OFF)
49	Output function selection 303	0	0 ~ 5		0: General-purpose output 1: AUTO mode output 2: Output during automatic operation (Other parameter No. 12)
50	Output function selection 304	0	0 ~ 5		0: General-purpose output

I/O Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
51	Output function selection 305	0	0 ~ 5		0: General-purpose output 1: For future extension 2: Output when axis-1 servo is ON (F-ROM 16-Mbit version only) 3: For future extension
52	Output function selection 306	0	0 ~ 5		0: General-purpose output 1: For future extension 2: Output when axis-1 servo is ON (F-ROM 16-Mbit version only) 3: For future extension
53	Output function selection 307	0	0 ~ 5		0: General-purpose output 1: For future extension 2: Output when axis-1 servo is ON (F-ROM 16-Mbit version only) 3: For future extension
54	Output function selection 308	0	0 ~ 5		0: General-purpose output 1: For future extension 2: Output when axis-1 servo is ON (F-ROM 16-Mbit version only) 3: For future extension
55	Output function selection 309	0	0 ~ 5		0: General-purpose output
56	Output function selection 310	0	0 ~ 5		0: General-purpose output
57	Output function selection 311	0	0 ~ 5		0: General-purpose output
58	Output function selection 312	0	0 ~ 5		0: General-purpose output
59	Output function selection 313	0	0 ~ 5		0: General-purpose output 1: System-memory backup battery voltage-low warning level or lower
60	Output function selection 314	0	0 ~ 5		0: General-purpose output 1: Absolute-data backup battery voltage-low warning level or lower (OR check of all axes. Upon detection of abnormal level, the output will be latched until a power-ON reset or software reset is executed.)
61	Output function selection 315	0	0 ~ 5		0: General-purpose output
62	For future extension (Change prohibited)	0	0H ~ FFFFFFFFH		
63	For future extension (Change prohibited)	0	0H ~ FFFFFFFFH		
64	Physical input port number for axis-3 brake forced release	0	0 ~ 299		Forcibly unlock the brake when the applicable port is ON (be aware of a falling load). * Invalid if "0" is set (Invalid if input port No. 0 is specified)
65	Physical input port number for axis-4 brake forced release	0	0 ~ 299		Forcibly unlock the brake when the applicable port is ON (be ware of a falling load). * Invalid if "0" is set (Invalid if input port No. 0 is specified)
66~69	(For extension)	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)

I/O Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
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) (Valid with TP application version 1.05 or later)
75	TP user output port start number (hand, etc.)	0	0 ~ 599		Referenced by TP. (Valid with TP application version 1.05 or later)
76~ 78	(For extension)	0			
79	Input port number for remote mode control	0	0 ~ 299		System mode = AUTO if specified DI = OFF AND AUTO/MANU-SW = AUTO. (Invalid if “0” is set) (F-ROM 16-Mbit version only) * Must be combined with specified hardware.
80	(PC/TP SIO usage)	1	1 ~ 1		Switching of DIP switches
81	(PC/TP SIO station code)	153	Reference only		Fixed to 153 (99H).
82	(PC/TP SIO reservation)	0			
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 1 opened to user	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 1 opened to user	153	0 ~ 255		Valid only with IAI protocol.
92	Baud rate type of SIO channel 1 opened to user	0	0 ~ 2		0: 9.6 1: 19.2 2: 38.4 kbps
93	Data length of SIO channel 1 opened to user	8	7 ~ 8		
94	Stop bit length of SIO channel 1 opened to user	1	1 ~ 2		
95	Parity type of SIO channel 1 opened to use	0	0 ~ 2		0: None 1: Odd 2: Even
96	Receive operation type of SIO channel 1 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 1 opened to user	0	0 ~ 999	msec	Valid only with IAI protocol.
98	(Reservation of SIO channel 1 opened to user) 0	0			

I/O Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
99	(Reservation of SIO channel 1 opened to user) 0	0			
100	Attribute 1 of SIO channel 2 opened to user (expanded)	28100010H	0H ~ FFFFFFFFH		Bits 28 to 31: Baud rate type (0: 9.6, 1: 19.2, 2: 38.4, 3: 57.6, 4: 76.8, 5: 115.2 kbps) Bits 24 to 27: Data length (7 or 8) Bits 20 to 23: Stop bit length (1 or 2) Bits 16 to 19: Parity type (0: None, 1: Odd, 2: Even) Bits 12 to 15: Communication mode (0: RS232C, 1: RS422, 2: RS485) * Only “0” can be selected for board channels other than Nos. 1 and 2 Bits 8 to 11: Receive operation type (0: RS485 = Forcibly enable receive immediately after send, RS232C/RS422 = Forcibly enable receive immediately before send 1: Do not forcibly enable receive at send) Bits 4 to 7: Board channel assignment number (1: D-sub upper, 2: D-sub lower, 3: Flat connector upper, 4: Flat connector lower) Bits 0 to 3: Expanded I/O slot assignment number (Expanded I/O slots 1 to 3 from the slot next to the standard IO (I/O1) slot. * “0” means no slots are used)
101	(Reservation of SIO channel 2 opened to user (expanded))	0	0H ~ FFFFFFFFH		
102	Attribute 1 of SIO channel 3 opened to user (expanded)	28100020H	0H ~ FFFFFFFFH		(Same as with I/O parameter No. 100)
103	(Reservation of SIO channel 3 opened to user (expanded))	0	0H ~ FFFFFFFFH		
104	Attribute 1 of SIO channel 4 opened to user (expanded)	28100010H	0H ~ FFFFFFFFH		(Same as with I/O parameter No. 100)
105	(Reservation of SIO channel 4 opened to user (expanded))	0	0H ~ FFFFFFFFH		
106	Attribute 1 of SIO channel 5 opened to user (expanded)	28100020H	0H ~ FFFFFFFFH		(Same as with I/O parameter No. 100)
107	(Reservation of SIO channel 5 opened to user (expanded))	0	0H ~ FFFFFFFFH		
108	Attribute 1 of SIO channel 6 opened to user (expanded)	28100010H	0H ~ FFFFFFFFH		(Same as with I/O parameter No. 100)
109	(Reservation of SIO channel 6 opened to user (expanded))	0	0H ~ FFFFFFFFH		
110	Attribute 1 of SIO channel 7 opened to user (expanded)	28100020H	0H ~ FFFFFFFFH		(Same as with I/O parameter No. 100)
111	(Reservation of SIO channel 7 opened to user (expanded))	0	0H ~ FFFFFFFFH		
112	SIO system use (SP8)	28100030H	0H ~ FFFFFFFFH		(Same as with I/O parameter No. 100)
113	(SIO system reserve (SP8))	0	0H ~ FFFFFFFFH		
114	SIO system use (SP9)	28100040H	0H ~ FFFFFFFFH		(Same as with I/O parameter No. 100)
115	(SIO system reserve (SP9))	0	0H ~ FFFFFFFFH		

I/O Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
116~ 119	(For extension)	0			
120	Network attribute 1	1	0H ~ FFFFFFFFH		Bits 0 to 3: CC-Link remote register area H/L byte swap selection (0: Do not swap, 1: Swap) * The number of used ports and number of occupied stations in I/O parameter Nos. 14 and 15 must match. (Main application version 0.25 or later)
121	Network attribute 2	0	0H ~ FFFFFFFFH		
122	Network attribute 3	0	0H ~ FFFFFFFFH		
123	Network attribute 4	0H	0H ~ FFFFFFFFH		Bits 0 to 3: Ethernet TCP/IP message communication IP address of connection destination on server Whether to permit 0.0.0.0 (IP address of connection destination can be ignored) (0: Do not permit 1: Permit (not recommended)) * Note: Number of clients that can be connected simultaneously to one server port channel = 1
124	Network attribute 5	0H	0H ~ FFFFFFFFH		Ethernet TCP/IP message communication attribute Ethernet client/server type (0: Not in use 1: Client (Automatic assignment of own port number) (2: Client (Specification of own port number) → This setting is not recommended because of device limitations, such as an error generation when the port is opened for approx. 10 minutes after disablement of close response check due to a power failure at the connection destination, etc.) 3: Server (Specification of own port number)) * Note: Number of clients that can be connected simultaneously to one server port channel = 1 Bits 0 to 3: IAI protocol B/TCP (MANU mode) * PC software can be connected only in the case of a client. Bits 4 to 7: IAI protocol B/TCP (AUTO mode) * PC software can be connected only in the case of a client. Bits 8 to 11: Channel 31 opened to user Bits 12 to 15: Channel 32 opened to user Bits 16 to 19: Channel 33 opened to user Bits 20 to 23: Channel 34 opened to user * If the parameter settings for own port number, client/server type, IP address of connection destination and port number of connection destination do not match completely in the IAI protocol B/TCP MANU or AUTO mode, the connection will be cut off when the MANU/AUTO mode is switched.

I/O Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
125	Network attribute 6	A32H	0H ~ FFFFFFFFH		Bits 0 to 7: Module-initialization check timer setting when Ethernet is used (100 msec) Bits 8 to 15: Module-initialization check timer setting when Ethernet is not used (100 msec) Bits 16 to 23: Increment of "PC/TP reconnection delay time upon software reset" when Ethernet is used (sec)
126	Network attribute 7	7D007D0H	0H ~ FFFFFFFFH		Ethernet TCP/IP message communication attribute Bits 0 to 15: Min timeout value (msec) Bits 16 to 31: Mout timeout value (msec)
127	Network attribute 8	5050214H	0H ~ FFFFFFFFH		Ethernet TCP/IP message communication attribute Bits 0 to 7: CONNECT_TIMEOUT (Change is prohibited) (Setting of “0” is prohibited) (sec) Bits 8 to 15: Connection retry interval (IAI protocol B/TCP) (sec) Bits 16 to 23: Send timeout value (sec) Bits 24 to 31: IAI protocol B-SIO non-communication check timer setting (sec) (IAI protocol B/TCP connection trigger)
128	Network attribute 9	0H	0H ~ FFFFFFFFH		Ethernet TCP/IP message communication attribute Bits 0 to 15: SEL server open timeout value (sec) (No timeout check when “0” is set)
129	Network attribute 10	0	0H ~ FFFFFFFFH		Ethernet operation requirement Bits 0 to 3: Modbus/TCP (Remote I/O) (0: Not in use 1: Use (Disable exception status) (Main application version 0.25 or later) 2: Use (Enable exception status (upper two digits of error number))) * Refer to the explanation of error levels in the operation manual and perform processing appropriate for each error level. Bits 4 to 7: TCP/IP message communication (0: Not in use, 1: Use) Bits 8 to 31: Reserved (Operation requirement)
130	Own MAC address (H)	0H	Reference only (HEX)		Only the lower two bytes are valid.
131	Own MAC address (L)	0H	Reference only (HEX)		
132	Own IP address (H)	192	1 ~ 255		* Setting of “0” and “127” is prohibited.
133	Own IP address (MH)	168	0 ~ 255		
134	Own IP address (ML)	0	0 ~ 255		
135	Own IP address (L)	1	2 ~ 254		* Setting of “0” and “255” is prohibited.
136	Subnet mask (H)	255	0 ~ 255		
137	Subnet mask (MH)	255	0 ~ 255		
138	Subnet mask (ML)	255	0 ~ 255		
139	Subnet mask (L)	0	0 ~ 255		

I/O Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
140	Default gateway (H)	0	0 ~ 255		
141	Default gateway (MH)	0	0 ~ 255		
142	Default gateway (ML)	0	0 ~ 255		
143	Default gateway (L)	0	0 ~ 255		
144	IAI protocol B/TCP: Own port number (MANU mode)	64511	1025 ~ 65535		* Important note: Always set a unique number for each own port number. (Duplication of own port numbers is permitted only in the IAI protocol B/TCP MANU/AUTO modes.)
145	Channel 31 opened to user (TCP/IP): Own port number	64512	1025 ~ 65535		
146	Channel 32 opened to user (TCP/IP): Own port number	64513	1025 ~ 65535		
147	Channel 33 opened to user (TCP/IP): Own port number	64514	1025 ~ 65535		
148	Channel 34 opened to user (TCP/IP): Own port number	64515	1025 ~ 65535		
149	IAI protocol B/TCP: IP address of connection destination (MANU mode) (H)	192	0 ~ 255		* Setting of “0” and “127” is prohibited.
150	IAI protocol B/TCP: IP address of connection destination (MANU mode) (MH)	168	0 ~ 255		
151	IAI protocol B/TCP: IP address of connection destination (MANU mode) (ML)	0	0 ~ 255		
152	IAI protocol B/TCP: IP address of connection destination (MANU mode) (L)	100	0 ~ 254		* Setting of “0” and “255” is prohibited.
153	IAI protocol B/TCP: Port number of connection destination (MANU mode)	64611	0 ~ 65535		* “0” can be set in the case of a server. 0 = Port number of connection destination is ignored (only the IP address is checked) * “0” cannot be set in the case of a client.
154	IAI protocol B/TCP: IP address of connection destination (AUTO mode) (H)	192	0 ~ 255		* Setting of “0” and “127” is prohibited.
155	IAI protocol B/TCP: IP address of connection destination (AUTO mode) (MH)	168	0 ~ 255		
156	IAI protocol B/TCP: IP address of connection destination (AUTO mode) (ML)	0	0 ~ 255		* Setting of “0” and “255” is prohibited.
157	IAI protocol B/TCP: IP address of connection destination (AUTO mode) (L)	100	0 ~ 254		
158	IAI protocol B/TCP: Port number of connection destination (AUTO mode)	64611	0 ~ 65535		* “0” can be set in the case of a server. 0 = Port number of connection destination is ignored (only the IP address is checked) * “0” cannot be set in the case of a client.
159	IAI protocol B/TCP: Own port number (AUTO mode)	64516	1025 ~ 65535		* Important note: Always set a unique number for each own port number. (Duplication of own port numbers is permitted only in the IAI protocol B/TCP MANU/AUTO modes.)
160~ 169	(For network extension)	0			

I/O Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
170~ 200	(For extension)	0			
			~		
			~		

PC: PC software
TP: Teaching pendant

2. Parameters Common to All Axes

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
1	For future extension (Change prohibited)	1111B	Reference only		
2	Default override	100	1 ~ 100		Used if not specified in program. (Invalid for SIO operation)
3~8	(For extension)	0			
9	For future extension (Change prohibited)		11111111B	Reference only	
10	(For extension)	0			
11	CP Default acceleration	10	1 ~ 200	0.01 G	Used if not specified in position data, program or SIO message, etc.
12	CP Default deceleration	10	1 ~ 200	0.01 G	Used if not specified in position data, program or SIO message, etc.
13	CP 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	For future extension (Change prohibited)	~	0H ~ FFFFFFFFH		
16~19	(For extension)	0	~		
20	For future extension (Change prohibited)	2000	0H ~ FFFFFFFFH		
21	CP Maximum speed	200	1 ~ 9999	mm/s	
22	CP Maximum acceleration	200	1 ~ 999	0.01 G	
23	CP Maximum deceleration	50	1 ~ 999	0.01 G	
24	CP Minimum emergency deceleration	0	1 ~ 999	0.01 G	
25	For future extension (Change prohibited)	0	0H ~ FFFFFFFFH		
26	For future extension (Change prohibited)	0	0H ~ FFFFFFFFH		
27	For future extension (Change prohibited)	0	0H ~ FFFFFFFFH		
28	For future extension (Change prohibited)	0	0H ~ FFFFFFFFH		
29	All-axis setting bit pattern 1	0	0H ~ FFFFFFFFH	F	Bits 0 to 3: (For future extension) 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) (Main application version 0.25 or later) 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) (Main application version 0.25 or later)
30	Default division angle	150	0 ~ 1200	0.1 deg	
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 (PTP-A1c/A2c) 1: Check operation amount only

Parameters Common to All Axes

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
33	CP Safety speed in manual mode	250	1 ~ 250	mm/s	
34	PTP Safety speed in manual mode	3	1 ~ 10	%	
35	Maximum jog speed of each axis system	5	1 ~ 10	%	
36	Maximum jog speed of each axis system when coordinate is not confirmed	3	1 ~ 10	%	
37~ 43	(For extension)	0	~		
44	PTP SM control ratio	3	0 ~ 50	%	
45	Radius of circle prohibiting entry of tool reference point	150000	0 ~ 999999	0.001 mm	For simple check. (Radius of a circle centered around the axis of arm 1)
46	CPxy check tolerance	2000	100 ~ 9999	0.001 mm	
47	Default PTP acceleration	20	1 ~ 100	%	
48	Default PTP deceleration	20	1 ~ 100	%	
49	Default PTP speed	2	1 ~ 100	%	
50	Width of CP-operation restriction zone near arm 1/2 straight-line point	500	Reference only	0.001 mm	* Change is prohibited unless instructed by the manufacturer.
51~ 100	(For extension)	0			
			~		
			~		
			~		

3. Axis-Specific Parameters

No	Parameter name	Default value (Reference)	Input range	Unit	Remarks
1	For future extension (Change prohibited)	1, 1, 0, 1	Reference only		
2~5	(For extension)	0	~		
6	For future extension (Change prohibited)	1, 1, 0, 0	Reference only		0: Motor CCW → Positive direction on the coordinate system 1: Motor CCW → Negative direction on the coordinate system
7	Soft limit +	210000, 145000, 200000, 720000	-99999999 ~ 99999999	0.001 deg, 0.001 mm	
8	Soft limit –	-30000, -145000, 0, -720000	-99999999 ~ 99999999	0.001 deg, 0.001 mm	
9	Soft-limit actual position margin	1000	0 ~ 9999	0.001 deg, 0.001 mm	
10	For future extension (Change prohibited)	0	Reference only		
11	For future extension (Change prohibited)	0	Reference only		0: Negative end of the coordinate system 1: Positive end of the coordinate system (Used only for axis 3 (Zc))
12	For future extension (Change prohibited)	9000, 0, 0, -90000	Reference only	0.001 deg, 0.001 mm	
13	For future extension (Change prohibited)	0	Reference only		
14	For future extension (Change prohibited)	0	Reference only		
15	For future extension (Change prohibited)	0	Reference only		
16	For future extension (Change prohibited)	0	Reference only		
17	For future extension (Change prohibited)	0, 0, 10, 0	Reference only		
18	For future extension (Change prohibited)	0, 0, 100, 0	Reference only		
19	For future extension (Change prohibited)	0, 0, 20, 0	Reference only	mm/sec	(Used only for axis 3 (Zc))
20	For future extension (Change prohibited)	0, 0, 3, 0	Reference only	mm/sec	Exercise caution, since limitations apply depending on the read/encoder pulse count. (Used only for axis 3 (Zc))
21	For future extension (Change prohibited)	0, 0, 0, 0	Reference only	0.001 mm	(Positive value = Direction of moving away from the end) (Used only for axis 3 (Zc). Fixed to "0" for axis 1 (A1c), axis 2 (A2c) and axis 4 (Rc).)
22	Phase-Z position at origin return	1000, 1000, 500, 1000	0 ~ 99999999	0.001 deg, 0.001 mm	Minimum allowable value of actual distance (angle) between [search end (axis 3 (Zc)) or reference position (eye mark) (axis 1 (A1c), axis 2 (A2c) or axis 4 (Rc))] and phase Z
23	For future extension (Change prohibited)	1	Reference only		
24	Push stop check time upon movement to absolute reset position	700	1 ~ 5000	msec	(Used only for axis 3 (Zc))
25	Push stop check time at positioning	500	1 ~ 5000	msec	
26	For future extension (Change prohibited)	0	0H ~ FFFFFFFFH		
27	Maximum motor speed	5000	Reference only	rpm	
28	Maximum PTP speed	480, 480, 1393, 1200	1 ~ 9999	deg/sec, mm/sec	

Axis-Specific Parameters

No	Parameter name	Default value (Reference)	Input range	Unit	Remarks
29	For future extension (Change prohibited)	0	0H ~ FFFFFFFFH		
30	Servo ON check time	150	0 ~ 5000	msec	
31	For future extension (Change prohibited)	3	Reference only	mm/sec	(Used only for axis 3 (Zc))
32	Actual distance between phase Z and end	-1	-1 ~ 99999	0.001 mm	Absolute distance from the search end. 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. (Used only for axis 3 (Zc))
33	Ideal distance between phase Z and end	0	0 ~ 99999	0.001 mm	Absolute distance from the search end (Used only for axis 3 (Zc))
34	Brake equipment specification	0, 0, 1, 1,	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	For future extension (Change prohibited)	0	Reference only		
38	For future extension (Change prohibited)	1	Reference only		
39	For future extension (Change prohibited)	0	0H ~ FFFFFFFFH		
40	For future extension (Change prohibited)	0	0H ~ FFFFFFFFH		
41	For future extension (Change prohibited)	0	0H ~ FFFFFFFFH		
42	For future extension (Change prohibited)	131072	Reference only	Pulse (rev)	Pulses (before division)/rev, in the case of a rotary encoder
43	For future extension (Change prohibited)	2	Reference only		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.) (Used only for axis 3 (Zc))
45~ 46	(For extension)	0			
47	For future extension (Change prohibited)	20000	Reference only	0.001 mm	Valid only for linear movement axes. (Used only for axis 3 (Zc))
48~ 49	(For extension)	0			
50	For future extension (Change prohibited)	1, 1, 1, 1	Reference only		
51	For future extension (Change prohibited)	50, 50, 10, 15	Reference only		
52	(For extension)	0		F	
53	Setting bit pattern 1 of each axis	0	0H ~ FFFFFFFFH	F	
54	Travel distance for push stop detection upon movement to absolute reset position	20	1 ~ 99999	0.001 mm	(Used only for axis 3 (Zc))
55	Travel distance for push stop detection at positioning	30	1 ~ 99999	0.001 mm	(Used only for axis 3 (Zc))
56	Push-abort deviation ratio upon movement to absolute reset position	3000	1 ~ 99999		Deviation is compared against “Steady-state deviation of push speed + Push-speed pulse speed x Abort deviation ratio.” (Used only for axis 3 (Zc))
57	Push-abort deviation ratio at positioning	3000	1 ~ 99999		Deviation is compared against “Steady-state deviation of push speed + Push-speed pulse speed x Abort deviation ratio.”
58	Positioning width	50, 50, 100, 150	1 ~ 9999	0.001 deg, 0.001 mm	
59	Allowable deviation error ratio (Maximum speed pulse ratio)	85	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	PSG	60			* Change is prohibited unless instructed by the manufacturer.

Axis-Specific Parameters

No	Parameter name	Default value (Reference)	Input range	Unit	Remarks
61	FAG	0			* Change is prohibited unless instructed by the manufacturer.
62	For future extension (Change prohibited)	0	0H ~ FFFFFFFFH		
63	Stop special output range	1	0 ~ 9999	Pulse	Invalid if "0" is set.
64	Stop special output value	1	0 ~ 999	DRVVR	
65	For future extension (Change prohibited)	0	0H ~ FFFFFFFFH		
66	For future extension (Change prohibited)	0	0H ~ FFFFFFFFH		
67	For future extension (Change prohibited)	0	0H ~ FFFFFFFFH		
68	For future extension (Change prohibited)	0	0H ~ FFFFFFFFH		
69	(For extension)	0			
70	For future extension (Change prohibited)	32767	Reference only	DRVVR	(Change prohibited)
71	For future extension (Change prohibited)	32767	Reference only	DRVVR	(Change prohibited)
72	For future extension (Change prohibited)	1	Reference only	DRVVR	(Change prohibited) To ensure symmetry of positive and negative sides.
73	For future extension (Change prohibited)	0	Reference only	DRVVR	(Change prohibited) To ensure symmetry of positive and negative sides.
74	For future extension (Change prohibited)	32436	Reference only	DRVVR	(Change prohibited)
75	For future extension (Change prohibited)	-32435	Reference only	DRVVR	(Change prohibited)
76	For future extension (Change prohibited)	0	0H ~ FFFFFFFFH		(Change prohibited)
77	For future extension (Change prohibited)	0	0H ~ FFFFFFFFH		
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. (This setting is valid only when a brake is equipped.)
79	Actual takeoff check distance	5	0 ~ 3000	0.001 mm	Absolute value input
80	For future extension (Change prohibited)	0	0 ~ 9999		
81	For future extension (Change prohibited)	0	0 ~ 9999		
82	For future extension (Change prohibited)	0	0 ~ 9999		
83	For future extension (Change prohibited)	0	0H ~ FFFFFFFFH		
84	For future extension (Change prohibited)	0	0H ~ FFFFFFFFH		
85	For future extension (Change prohibited)	0, 0, 15, 0	Reference only	0.01 G,	(Used only for axis 3 (Zc))
86	For future extension (Change prohibited)	0	-99999999 ~ 99999999		
87	For future extension (Change prohibited)	0	-99999999 ~ 99999999		
88	For future extension (Change prohibited)	0	0 ~ 899		
89	For future extension (Change prohibited)	0	-99999999 ~ 99999999		
90	For future extension (Change prohibited)	0	-99999999 ~ 99999999		
91	For future extension (Change prohibited)	0	0 ~ 899		
92	For future extension (Change prohibited)	0	-99999999 ~ 99999999		
93	For future extension (Change prohibited)	0	-99999999 ~ 99999999		
94	For future extension (Change prohibited)	0	0 ~ 899		

Axis-Specific Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
95	For future extension (Change prohibited)	0	-99999999 ~ 99999999		
96	For future extension (Change prohibited)	0	-99999999 ~ 99999999		
97	For future extension (Change prohibited)	0	0 ~ 899		
98~ 118	(For extension)	0	~		
119	FSG	50	0 ~ 100		* Change is prohibited unless instructed by the manufacturer.
120	FFF	10	0 ~ 100		* Change is prohibited unless instructed by the manufacturer.
121~ 130	(For extension)	0	~		
131	For future extension (Change prohibited)	0	0H ~ FFFFFFFFH		
132	For future extension (Change prohibited)	0	0H ~ FFFFFFFFH		
133	For future extension (Change prohibited)	0	0H ~ FFFFFFFFH		
134	Maximum PTP acceleration	2700, 5400, 160, 11000	1 ~ 99999999	deg/sec^2, 0.01 G	Set in units of 0.01 G only for axis 3 (Zb). Set in deg/sec^2 for other axes.
135	Maximum PTP deceleration	2700, 5400, 160, 11000	1 ~ 99999999	deg/sec^2, 0.01 G	Set in units of 0.01 G only for axis 3 (Zb). Set in deg/sec^2 for other axes.
136	Minimum PTP emergency deceleration	2700, 5400, 160, 11000	1 ~ 99999999	deg/sec^2, 0.01 G	Set in units of 0.01 G only for axis 3 (Zb). Set in deg/sec^2 for other axes.
137	For future extension (Change prohibited)	0	0H ~ FFFFFFFFH		
138	Arm length	250000, 250000, 0, 0	1 ~ 99999999	0.001 mm	(Used only for axis 1 (A1c) and axis 2 (A2c))
139	Rear entry prohibition area MAX (Xb)	100000, 0, 0, 0	-99999999 ~ 99999999	0.001 mm	For simple check. (Used only for axis 1 (Xb))
140	Rear entry prohibition area MIN (Xb)	-100000, 0, 0, 0	-99999999 ~ 99999999	0.001 mm	For simple check. (Used only for axis 1 (Xb))
141	Reference position coordinates at automatic update of origin preset value	90000, 0, 0, -90000	-99999999 ~ 99999999	0.001 mm	
142	Selection of R-axis→Z-axis correction coordinate direction	0	0 ~ 1		0: Correction from positive direction of R-axis coordinate to positive direction of Z-axis coordinate 1: Correction from positive direction of R-axis coordinate to negative direction of Z-axis coordinate (Used only for axis 3 (Zc))
143	Limit entry angle into CP-operation restriction zone based on actual position (A2c)	0, 0, 0, 0	Reference only	0.001 deg	Enter an absolute value (used only for axis 2 (A2c)). * Change is prohibited unless instructed by the manufacturer.
144	End offset travel at standby at tentative origin	0, 0, 5500, 0	-99999999 ~ 99999999	0.001 mm	(Positive value = Direction of moving away from the end) (Used only for axis 3 (Zc))
145	SIO current-arm-system change speed default value (A2c)	0, 3, 0, 0	1 ~ 10	%	(Used only for axis 2 (A2c))
146	For future extension (Change prohibited)	5000	Reference only	rpm	
147~ 170	(For extension)	0	~		

4. Driver Card 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 4 (Manufacturing information)	Space	Reference only		
5	Manufacturing data 5 (Manufacturing information)	Space	Reference only		
6	Manufacturing data 6 (Manufacturing information)	Space	Reference only		
7	Manufacturing data 7 (Manufacturing information)	Space	Reference only		
8	Board type (Function information)	30	Reference only		
9	Function information 01 (hard): Encoder support information (upper word)	0000H	Reference only	Encoder ID bit pattern	All encoders are supported when the most significant bit is ON (wildcard bit).
10	Function information 02 (hard): Encoder support information (lower word)	0003H	Reference only	Encoder ID bit pattern	
11	Function information 03 (hard): Hardware support information word	0001H	Reference only		Bit 0: Brake support specification bit (1: Supported, 0: Not supported)
12	Function information 04 (hard): For future extension	0000H	Reference only		
13	Function information 05 (hard): For future extension	0000H	Reference only		
14	Function information 06 (hard): For future extension	0000H	Reference only		
15	Function information 07 (soft): Motor support information (upper word)	0000H	Reference only	Motor ID bit pattern	All motors are supported when the most significant bit is ON (wildcard bit).
16	Function information 08 (soft): Motor support information (lower word)	003FH	Reference only	Motor ID bit pattern	
17	Function information 09 (soft): Encoder support information (upper word)	0000H	Reference only	Encoder ID bit pattern	All encoders are supported when the most significant bit is ON (wildcard bit).
18	Function information 10 (soft): Encoder support information (lower word)	0003H	Reference only	Encoder ID bit pattern	
19	Function information 11 (soft): Software support information word 0 (For future extension = Change prohibited)	0000H	Reference only		Bit 0: Auto-tuning support specification bit (1: Supported, 0: Not supported)
20	Function information 12 (soft): Software version information	0001H	Reference only		
21	Function information 13 (soft): For future extension	0000H	Reference only		

Driver Card Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
22	Function information 14 (soft): System log control word	0000H	Reference only		Area used for design [Driver version 0.10 or earlier] 1: The alarm LED blinks at high speed when the current command value reaches the limiter value (torque limit setting or three times) (Driver version 0.04 or later). 2: Fixed output on speed monitor 123H and current monitor 456H (Driver version 0.08 or later) [Driver version 0.11 or later] Bit 0: Indication of current command limit (torque limit setting or three times) (high-speed blinking of alarm LED); 0 = Indicate, 1 = Do not indicate Bits 1 to 3: Selection of monitor output data; other than the following = Speed monitor & current monitor output 1 = Speed monitor (123H) & current monitor (456H) output 2 = Phase-D voltage monitor & phase-Q voltage monitor output 3 = Speed monitor & speed command monitor output Bit 4: Non-detection of overspeed & overload (including compile switch); 0 = Detected, 1 = Not detected
23	Configuration information 01: Configured capacity (rated motor output)	003CH	Reference only	W	
24	Configuration information 02: Configured voltage (motor voltage)	00C8H	Reference only	V	The most significant bit is the AC/DC identification bit (OFF: AC, ON: DC).
25	Configuration information 03: Motor/encoder configuration information	0000H	Reference only	Motor/ encoder ID bit number	Upper: Motor identification bit number Lower: Encoder identification bit number [Motor identification bit number] EU8302NxxxExxx (ϕ 38-60W-200V) = 0 EU8306NxxxExxx (ϕ 60-100W-200V) = 1 EU8303NxxxExxx (ϕ 38-100W-200V) = 2 EU8304NxxxExxx (ϕ 38-150W-200V) = 3 EU8307NxxxExxx (ϕ 60-200W-200V) = 4 EU8309NxxxExxx (ϕ 60-400W-200V) = 5 EU8342NxxxExxx (ϕ 28-20W-200V) = 6 (Driver version 0.25 or later) EU8343NxxxExxx (ϕ 28-30W-200V) = 7 (Driver version 0.25 or later) EU8313NxxxExxx (ϕ 80-600W-200V) = 8 (Driver version 0.25 or later) EU8314NxxxExxx (ϕ 80-750W-200V) = 9 (Driver version 0.25 or later) [Encoder identification bit number] TS5668Nxx (Tamagawa wire-saving 17-bit incremental) = 0 TS5669Nxxx (Tamagawa wire-saving 17-bit absolute) = 1 No encoder = 2 (Driver version 0.13 or later) (* Note: “N”th power of 2 will be set if the driver version is 0.00) (As of current driver version 0.25)
26	Configuration information 04: For future extension	0000H	Reference only		
27	Configuration information 05: Encoder resolution (upper word)	0002H	Reference only		Rotary encoder pulses (before division)/rev
28	Configuration information 06: Encoder resolution (lower word)	0000H	Reference only		

Driver Card Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
29	Configuration information 07: Motor/encoder characteristics word	0004H	Reference only		Bit 0: Change prohibited (0: Rotary) Bit 1: ABS specification bit (1: ABS, 0: INC) (Used for consistency check with the MAIN parameter. When INC is specified, the installed encoder must behave like an incremental encoder to the MAIN application even when it is of absolute type.) Bit 2: Magnetic-pole sensor installation bit (1: Installed, 0: Not installed) Bit 3: Brake installation bit (1: Installed, 0: Not installed)
30	Configuration information 08: Encoder cable length	2	1 ~ 30	m	★ Be sure to change this parameter when retrofitting.
31	Configuration information 09: Control characteristics word	0018H	Reference only		Bit 0: Incremental operation specification bit when absolute data is indeterminable (1: Incremental operation, 0: Error stop) Bits 1 and 2: Control mode specification bits (2: Pickup operation (driver version 0.13 or later), 1: Torque control, 0: Speed control, Other: Reserved) Bits 3 to 6: Encoder division factor (Pulses are multiplied by the "n"th power of 1/2. Code extension.) Bit 7: DB operation specification bit at servo OFF (1: Use, 0: Do not use) Bits 8 to 11: Maximum torque limit specification bits (300-n (positive value) x 10%) (Driver version 0.22 to 0.25) Bits 8 to 12: Maximum torque limit specification bits (300-n (positive value) x 10%) (Any setting below 10% will be considered 10%.) (Driver version 0.26)
32	Configuration information 10: Push torque limit at origin return	100	0 ~ 150	%	When the setting is changed, whether or not the phase-Z detection position will shift due to change in the elastic deformation amount of the contacted mechanical end must be checked.
33	Configuration information 11: Push torque limit at positioning	70	0 ~ 70	%	
34	Configuration information 12: For future extension	0H	Reference only		
35	Configuration information 13: Speed-command underrun count	0	Reference only		
36	Configuration information 14: For future extension	0H	Reference only		
37	Configuration information 15: For future extension	0H	Reference only		
38	Speed-loop gain (Parameter list 1)	500	1 ~ 6000		Proportional gain
39	Speed-loop integral constant (Parameter list 1)	30	1 ~ 1000		Integral gain
40	Current-loop control range number (Parameter list 1)	4	0 ~ 4		
41	For future extension (Parameter list 1)	0H	Reference only		
42	Torque filter constant (Parameter list 1)	0	0 ~ 2500		
43	Rotor inertia ratio (Parameter list 1) (For future extension = Change prohibited)	10	1 ~ 100		
44	For future extension (Parameter list 1)	0H	Reference only		
45	For future extension (Parameter list 1)	0H	Reference only		

Driver Card Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
46	For future extension (Parameter list 1)	0H	Reference only		
47	For future extension (Parameter list 1)	0H	Reference only		
48	For future extension (Parameter list 1)	0H	Reference only		
49	For future extension (Parameter list 1)	0H	Reference only		
50	For future extension (Parameter list 1)	0H	Reference only		
51	Reversing time (for encoder adjustment)	0H	Reference only	SEC	Continuous rotation in the negative direction when “0” is set (Driver version 0.13 or later)
52	Self-propelling speed (for encoder adjustment)	0H	Reference only	rpm	DRVVR register is ignored when “1” or greater is set (Driver version 0.13 or later)
53	Speed-loop gain (Parameter list 2)	500	1 ~ 6000		Proportional gain
54	Speed-loop integral constant (Parameter list 2)	30	1 ~ 1000		Integral gain
55	Current-loop control range number (Parameter list 2)	4	0 ~ 4		
56	For future extension (Parameter list 2)	0H	Reference only		
57	Torque filter constant (Parameter list 2)	0	0 ~ 2500		
58	Rotor inertia ratio (Parameter list 2) (For future extension = Change prohibited)	10	1 ~ 100		
59	For future extension (Parameter list 2)	0H	Reference only		
60	For future extension (Parameter list 2)	0H	Reference only		
61	For future extension (Parameter list 2)	0H	Reference only		
62	For future extension (Parameter list 2)	0H	Reference only		
63	For future extension (Parameter list 2)	0H	Reference only		
64	For future extension (Parameter list 2)	0H	Reference only		
65	For future extension (Parameter list 2)	0H	Reference only		
66	Phase-U current sense offset adjustment	0	Reference only		Adjustment of offset for phase-U current detection value (Any value between –257 and –32768 is treated as –256. Any value between 257 and 32767 is treated as 256.) (Driver version 0.30 or later)

Driver Card Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
67	Phase-W current sense offset adjustment	0	Reference only		Adjustment of offset for phase-W current detection value (Any value between –257 and –32768 is treated as –256. Any value between 257 and 32767 is treated as 256.) (Driver version 0.30 or later)
68	Speed-loop gain (Parameter list 3)	500	1 ~ 6000		Proportional gain
69	Speed-loop integral constant (Parameter list 3)	30	1 ~ 1000		Integral gain
70	Current-loop control range number (Parameter list 3)	4	0 ~ 4		
71	For future extension (Parameter list 3)	0H	Reference only		
72	Torque filter constant (Parameter list 3)	0	0 ~ 2500		
73	Rotor inertia ratio (Parameter list 3) (For future extension = Change prohibited)	10	1 ~ 100		
74	For future extension (Parameter list 3)	0H	Reference only		
75	For future extension (Parameter list 3)	0H	Reference only		
76	For future extension (Parameter list 3)	0H	Reference only		
77	For future extension (Parameter list 3)	0H	Reference only		
78	For future extension (Parameter list 3)	0H	Reference only		
79	For future extension (Parameter list 3)	0H	Reference only		
80	For future extension (Parameter list 3)	0H	Reference only		
81	For future extension	0H	Reference only		
82	For future extension	0H	Reference only		
83	Speed-loop gain (Parameter list 4)	500	1 ~ 6000		Proportional gain
84	Speed-loop integral constant (Parameter list 4)	30	1 ~ 1000		Integral gain
85	Current-loop control range number (Parameter list 4)	4	0 ~ 4		
86	For future extension (Parameter list 4)	0H	Reference only		
87	Torque filter constant (Parameter list 4)	0	0 ~ 2500		
88	Rotor inertia ratio (Parameter list 4) (For future extension = Change prohibited)	10	1 ~ 100		

Driver Card Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
89	For future extension (Parameter list 4)	0H	Reference only		
90	For future extension (Parameter list 4)	0H	Reference only		
91	For future extension (Parameter list 4)	0H	Reference only		
92	For future extension (Parameter list 4)	0H	Reference only		
93	For future extension (Parameter list 4)	0H	Reference only		
94	For future extension (Parameter list 4)	0H	Reference only		
95	For future extension (Parameter list 4)	0H	Reference only		
96	For future extension	0H	Reference only		
97	For future extension	0H	Reference only		
98	Overrun error counter (Query information)	0H	Reference only		
99	FPGA detection error counter (Query information)	0H	Reference only		
100	Speed-command underrun-count error counter (Query information)	0H	Reference only		
101	For future extension (Query information)	0H	Reference only		
102	Overload error counter (Query information)	0H	Reference only		
103	Overspeed error counter (Query information)	0H	Reference only		
104	Overcurrent error counter (Query information)	0H	Reference only		
105	Overheat error counter (Query information)	0H	Reference only		
106	Encoder error counter (Query information)	0H	Reference only		
107	CPU error counter (Query information)	0H	Reference only		
108	Phase-U current sense adjustment value (Query information)	0H	Reference only		
109	Phase-W current sense adjustment value (Query information)	0H	Reference only		
110	For future extension (Query information)	0H	Reference only		
111	For future extension (Query information)	0H	Reference only		
112	For future extension (Query information)	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 4 (Manufacturing information)	Space	Reference only		
5	Manufacturing data 5 (Manufacturing information)	Space	Reference only		
6	Manufacturing data 6 (Manufacturing information)	Space	Reference only		
7	Manufacturing data 7 (Manufacturing information)	Space	Reference only		
8	Board type (Function information)	80	Reference only		
9	Function information 01: Configured capacity (rated motor output)	003CH	Reference only		
10	Function information 02: Configured voltage (motor voltage)	00C8H	Reference only	V	The most significant bit is the AC/DC identification bit (OFF: AC, ON: DC).
11	Function information 03: Motor/encoder configuration information	0000H	Reference only	Motor/ encoder ID bit number	Upper: Motor identification bit number Lower: Encoder identification bit number
12	Function information 04: Encoder resolution (upper word)	0002H	Reference only		Rotary pulses (before division)/rev
13	Function information 05: Encoder resolution (lower word)	0000H	Reference only		
14	Function information 06: Motor/encoder characteristics word	0004H	Reference only		Bit 0: Change prohibited (0: Rotary) Bit 1: ABS specification bit (1: ABS, 0: INC) Bit 2: Magnetic-pole sensor installation bit (1: Installed, 0: Not installed) Bit 3: Brake installation bit (1: Installed, 0: Not installed)
15	Function information 07: Rated thrust (For future extension = Change prohibited)	0000H	Reference only	0.1 N	
16	Function information 08: Repeatability (For future extension = Change prohibited)	0000H	Reference only	0.001 mm	
17	Function information 09: Actuator weight (For future extension = Change prohibited)	0000H	Reference only	0.1 kg	
18	Function information 10: Screw lead (For future extension = Change prohibited)	0000H	Reference only	0.001 mm/ rev	
19	Function information 11: Slider loading capacity (For future extension = Change prohibited)	0000H	Reference only	0.1 kg	
20	Function information 12: Slider stroke (For future extension = Change prohibited)	0000H	Reference only	mm	
21	Function information 13: (For future extension)	0000H	Reference only		
22	Function information 14: (For future extension)	0000H	Reference only		
23~ 30	Card parameter (by board type)	0000H	Reference only		

6. I/O-Slot Card 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 4 (Manufacturing information)	Space	Reference only		
5	Manufacturing data 5 (Manufacturing information)	Space	Reference only		
6	Manufacturing data 6 (Manufacturing information)	Space	Reference only		
7	Manufacturing data 7 (Manufacturing information)	Space	Reference only		
8	Board type (Function information)	0	Reference only		
9	Function information 01 (by board type)	0000H	Reference only		
10	Function information 02 (by board type)	0000H	Reference only		
11	Function information 03 (by board type)	0000H	Reference only		
12	Function information 04 (by board type)	0000H	Reference only		
13	Function information 05 (by board type)	0000H	Reference only		
14	Function information 06 (by board type)	0000H	Reference only		
15	Function information 07 (by board type)	0000H	Reference only		
16	Function information 08 (by board type)	0000H	Reference only		
17	Function information 09 (by board type)	0000H	Reference only		
18	Function information 10 (by board type)	0000H	Reference only		
19	Function information 11 (by board type)	0000H	Reference only		
20	Function information 12 (by board type)	0000H	Reference only		
21	Function information 13 (by board type)	0000H	Reference only		
22	Function information 14 (by board type)	0000H	Reference only		
23~ 112	Card parameter (by board type)	0000H	Reference only		

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, servo OFF or all-axis servo OFF, 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	11000	1 ~ 99999	msec	* The setting will become effective after the PC software/TP is shut down and restarted. (Main application version 0.25 or later)
7~8	(For extension)	0			
9	Deadman-switch recovery type	0	0 ~ 2		0: Abort operations/programs 2: Operation continued (Only during automatic operation. * In the PC software version is 1.0.0.5 or later or TP application version is 1.01 or later, operation commands from the PC software/TP will be aborted on the PC software/TP side.)

Other Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
10	Emergency-stop recovery type	0	0 ~ 4		0: Abort operations/programs 1: Recovery after reset 2: Operation continued (Only during automatic operation. * If the PC software version is 1.0.0.5 or later or TP application version is 1.01 or later, operation commands from the PC software/TP will be aborted on the PC software/TP side.) 3: Abort operations/programs (Software reset when the emergency stop is reset. The origin-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 I/O parameter No. 33 = 1 AND I/O parameter No. 44 ≠ 1 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 origin-return completion status of incremental-encoder axes will be retained.)
11	Safety-gate open recovery type	0	0 ~ 2		0: Abort operations/programs 1: Recovery after reset 2: Operation continued (Only during automatic operation * If the PC software version is 1.0.0.5 or later or TP application version is 1.01 or later, operation commands from the PC software/TP will be aborted on the PC software/TP side.)
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 extension)	0			
20	System-memory backup battery installation function type	2	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 operation 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)) (PC software version 1.1.0.7 or later and TP application version 1.06 or later) * Referenced by the PC/TP.

Other Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
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		
25~29	(For extension)	0			
30	Option Password 00	0H	0H ~ FFFFFFFFH		Reserved (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 extension)	0			
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 (PC software version 2.0.0.42 or later and TP application version 1.09 or later)
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 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 (PC software version 2.0.0.42 or later and TP application version 1.09 or later)

Other Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
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 (PC software version 2.0.0.42 or later and TP application version 1.09 or later)
39	PC/TP data protect setting (Coordinate system)	0H	0H ~ FFFFFFFFH		Bits 0 to 3: Protect type (Tool coordinate offset) (0: Read/write, 1: Read only, 2: No read/write) Bits 4 to 7: Protect release method (Tool coordinate offset) (0: Special operation) Bits 8 to 11: Protect type (Work coordinate offset) (0: Read/write, 1: Read only, 2: No read/write) Bits 12 to 15: Protect release method (Work coordinate offset) (0: Special operation) Bits 16 to 19: Protect type (Definition coordinates of simple interference check zone) (0: Read/write, 1: Read only, 2: No read/write) Bits 20 to 23: Protect release method (Definition coordinates of simple interference check zone) (0: Special operation) * Referenced by the PC/TP (PC software version 2.0.0.42 or later and TP application version 1.09 or later)
40	For future extension (Change prohibited)	3H	Reference only		0: Checksum invalid, 1: Checksum valid Bit 0 = Driver Bit 1 = Encoder Bit 2 = I/O board Bit 3 = Power stage Bit 4 = Motor drive power Bit 5 = Regenerative resistance Bit 6 = Control power

Other Parameters

No.	Parameter name	Default value (Reference)	Input range	Unit	Remarks
41	For future extension (Change prohibited)	E0H	Reference only		0: Not checked, 1: Checked Bit 0 = Standard I/O installation check Bit 1 = Power stage installation/consistency check Bit 2 = Motor drive power installation/consistency check Bit 3 = Regenerative resistance installation check Bit 4 = Control power installation check Bit 5 = Encoder support check Bit 6 = Motor support check Bit 7 = System parameter/hardware information consistency check
42	For future extension (Change prohibited)	7H	Reference only		0: Do not conduct test, 1: Conduct test – Bit 0 = Driver I/F register write/read test, Bit 1 = I/O-slot I/F register write/read test, Bit 2 = Driver installation/ready check upon reset
43	(For extension (HEX))	0H	0H ~ FFFFFFFFH		
44	(For extension)	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: For future extension Bits 8 to 11: Permission of auto program start when all-operation-cancellation factor is present (0: Do not permit, 1: Permit) (Main application version 0.29 or later) Bits 12 to 15: Permission of ON edge acceptance for PIO program start (input port 000) when all-operation-cancellation factor is present (0: Do not permit, 1: Permit) * 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. (Main application version 0.29 or later)
46	Other setting bit pattern 1	1	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 (Main application version 0.29 or later) (0: Do not round, 1: Round) Bits 8 to 11: For future extension
47~70	(For extension)	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.	O	O	O	O	O
Without safety speed	Not required.	O		O	O	O

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.	O	O	O		
SIO start and jog (safety speed)	1817 (*1)		O	O	O	
SIO start and jog	1818 (*1)			O	O	
SIO/PIO start and jog	1819 (*1)			O	O	O

(*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.	O	O	O	O	O
Disable	Not required.	O		O	O	O

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.	O	O	O	O	(*3)	
Disable	1818 (*1)	O		O	O	(*3)	

PIO start prohibition selection	Password	Functions					
		Edit	Safety speed	Jog, move, continuous move	SIO program start	PIO program start	*2
Prohibit	Not required.	O	(*4)	O	O		
Enable	1819 (*1)	O	(*4)	O	O	O	

(*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.

9. Use Examples of Key Parameters

You can add functions to those available under the factory settings or set dedicated functions to I/O ports, by changing the parameter values.
Before changing a parameter, be sure to read the corresponding section in the List of Parameters.

Description	Action	Parameter setting	Manipulation/operation
Want to prevent errors relating to the standard I/O board and optional boards (DeviceNet, CC-Link, etc.). (Want to perform trial operation when boards are not wired, etc.)	I/O-board error monitor can be disabled to prevent errors from occurring.	Set "0" in the I/O parameter corresponding to the I/O board whose error monitor you wish to disable. Standard I/O (I/O1): I/O parameter No. 10 = 0 Expanded I/O1 (I/O2): I/O parameter No. 11 = 0 Expanded I/O2 (I/O3): I/O parameter No. 12 = 0 Expanded I/O3 (I/O4): I/O parameter No. 13 = 0	Set "0" in I/O parameter Nos. 10 and 11 to disable error monitor for the standard I/O (I/O1) and expanded I/O1 (I/O2) boards, respectively. Note: To operate a disabled I/O board, be sure to revert the parameter setting to "1."
Want to execute restart (software reset) using an external input signal.	Input port No. 1 can be set as a restart input.	I/O parameter No. 31 = 1	Turning ON input port No. 1 for at least 1 second will execute restart.
Want to execute servo ON using an external input signal.	Input port No. 2 can be set as a servo ON input.	I/O parameter No. 32 = 1	Servo ON will be executed at the ON edge of input port No. 2. Servo OFF will be executed at the OFF edge.
Want to execute auto program start using an external input signal. (Under the default setting, the specified program will restart upon power ON or restart (software reset) in the AUTO mode.) (More steps will be required to execute auto program start.)	Input port No. 3 can be set as an auto program start input.	I/O parameter No. 33 = 2	The specified program will start at the ON edge of input port No.3. The program will be aborted at the OFF edge.
Want to execute pause using an external input signal.	Input port No. 6 can be set as a pause input. Input port No. 5 can be set as a pause reset input.	I/O parameter No. 36 = 1 I/O parameter No. 35 = 1	Turning OFF input port No. 6 will execute pause. Pause will be reset at the ON edge of input port No. 5 after turning ON input port No. 6. (Input port No. 6 is always ON.)
Want to reset errors using an external input signal (errors of operation-cancellation level or lower).	Input port No. 13 can be set as an error reset input.	I/O parameter No. 43 = 2	Errors will be reset at the ON edge of input port No 13.

<table>
<tr><th>Description</th><th>Action</th><th>Parameter setting</th><th>Manipulation/operation</th></tr>
<tr><td>Want to input program numbers from input ports in binary. (The default setting is BCD input.)</td><td>Program numbers can be input from input port Nos. 7 to 13 in binary.</td><td>I/O parameter No. 30 = 2</td><td></td></tr>
<tr><td>Want to check the level of the present error from an output port.
Want to check for emergency stop status from an output port.</td><td>Error level can be checked from the ON/OFF combination of output port Nos. 300 and 301.
Emergency stop status can be checked from ON/OFF of output port No. 302.</td><td>I/O parameter No. 46 = 2
I/O parameter No. 47 = 3
I/O parameter No. 48 = 2
(Parameter settings at shipment)</td><td>ON/OFF of output port Nos. 300 and 301 and corresponding error levels
<table>
<tr><td></td><td>300</td><td>301</td></tr>
<tr><td>Message level or lower</td><td>○</td><td>○</td></tr>
<tr><td>Operation-cancellation level</td><td>●</td><td>○</td></tr>
<tr><td>Cold-start level</td><td>●</td><td>●</td></tr>
</table>
○: ON ●: OFF

Output port No. 302 being OFF indicates an emergency stop status.
<table>
<tr><td></td><td>302</td></tr>
<tr><td>Emergency stop actuated</td><td>●</td></tr>
<tr><td>Emergency stop not actuated</td><td>○</td></tr>
</table>
Note) Parameter settings at shipment</td></tr>
<tr><td>Want to output signal during the AUTO mode.</td><td>Output port No. 303 can be set as an AUTO mode output signal.</td><td>I/O parameter No. 49 = 1</td><td>Output port No. 303 will turn ON during the AUTO mode.</td></tr>
<tr><td>Want to output signal during automatic operation.</td><td>Output port No. 303 can be set as an automatic operation output.</td><td>I/O parameter No. 49 = 2</td><td>Output port No. 303 will turn ON during automatic operation.</td></tr>
<tr><td>Recognition of automatic operation:
Recognition of automatic operation can be changed using the setting of other parameter No. 12.</td><td>• Recognize automatic operation if a program is running (either in the MANU or AUTO mode).
• Recognize automatic operation if a program is running OR in the AUTO mode (regardless of whether or not a program is running).
In either case, all-operation-cancellation factor must not be present.
One of the conditions is recognized as automatic operation.</td><td>• Other parameter No. 12 = 0
Recognize automatic operation if a program is running.
• Other parameter No. 12 = 1
Recognize automatic operation if a program is running OR in the AUTO mode.
• “All-operation-cancellation factor is not present” means errors of operation-cancellation level or higher are not present AND emergency-stop signal is not input AND safety-gate signal is not input AND deadman switch is ON (teaching-pendant option).</td><td></td></tr>
</table>

Description	Action	Parameter setting	Manipulation/operation
Want to output warning signal when the system-memory backup battery voltage is low.	Output port No. 313 can be set as a signal warning that the system-memory backup battery voltage is low.	I/O parameter No. 59 = 1	Output port No. 313 will turn ON when the system-memory backup battery voltage is low.
Want to output warning signal when the absolute-encoder backup battery voltage is low.	Output port No. 314 can be set as a signal warning that the absolute-encoder backup battery voltage is low. (This function is not supported on early units. The main application version must be 0.28 or later.)	I/O parameter No. 60 = 1	Output port No. 314 will turn ON when the absolute-encoder backup battery voltage is low. The output will remain ON until the power is reconnected or controller is restarted.
Want to retain output status while emergency-stop signal is input or the safety gate is open.	Minimum and maximum port numbers indicating the output ports you wish to retain can be set.	I/O parameter No. 70 = Output port number MIN I/O parameter No. 71 = Output port number MAX Setting example) To retain output ports from port Nos. 303 through 315, set as follows: I/O parameter No. 70 = 303 I/O parameter No. 71 = 315	← The status of output port Nos. 305 through 315 will be retained while emergency-stop signal is input or the safety gate is open.
Want to start programs while emergency-stop signal is input or the safety gate is open. Programs to be started are I/O processing or calculation programs that do not command actuator operation (PIO processing programs).	A PIO processing program to start can be set. Set in the applicable parameters a desired PIO processing program as well as minimum and maximum port numbers indicating the output ports at which the program will be processed.	Other parameter No. 2 = PIO processing program number I/O parameter No. 70 = Output port number MIN I/O parameter No. 71 = Output port number MAX Setting example) To start program No. 5 that involves processing at output port Nos. 303 through 315, set as follows: Other parameter No. 2 = 5 I/O parameter No. 70 = 303 I/O parameter No. 71 = 315	← Program No. 5 will start while emergency-stop signal is input or the safety gate is open. Output port Nos. 303 through 315 can be used for processing.

Description	Action	Parameter setting	Manipulation/operation
Want to switch between AUTO and MANU modes using an input port.	A general-purpose input port can be set as a mode switching input (dedicated input). Set a desired input port number in I/O parameter No. 79.	I/O parameter No. 79 = Input port number	Set the mode switch to the AUTO side. The AUTO mode will be enabled when the specified input port turns OFF, and the MANU mode will be enabled when the input port turns ON. If the mode switch is set to the MANU side, the MANU mode will be enabled regardless of ON/OFF of this input port. This function is available on controllers shipped in or after 2003.
Want to automatically execute restart (software reset) after the emergency stop is reset, and start the auto-start program.	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 I/O parameter No. 33 = 1	After the emergency-stop button is released, the system will automatically execute restart (software reset) and start the auto-start program.
Want to automatically execute error reset after the emergency stop is reset, and start the auto-start program.	The emergency-stop recovery type can be set to “Abort operations/programs (Error reset and auto program start when the emergency stop is reset).”	Other parameter No. 10 = 4 I/O parameter No. 33 = 1 I/O parameter No. 44 ≠ 1	After the emergency-stop button is released, the system will automatically execute error reset and start the auto-start program.
Want to continue actuator operation after the emergency stop is reset (want to resume actuator operation from the part stopped due to emergency stop input). Programs other than the one commanding actuator operation remain running while emergency-stop signal is input. (Programs not commanding actuator operation remain running while emergency-stop signal is input. The program commanding actuator operation will remain running until the execution step reaches an operation command.)	The emergency-stop recovery type can be set to “Operation continued.”	Other parameter No. 10 = 2 I/O parameter No 35 = 1 (Input port No. 5 is set as a pause reset input.) I/O parameter No. 31 = 1 (Input port No. 1 is set as a restart input. This is to provide a means of canceling the operation.)	After the emergency-stop button is released, actuator operation will continue at the ON edge of input port No. 5. To discontinue the operation, turn ON input port No. 1 for at least 1 second to execute restart, without executing ON-edge input to input port No. 5.

Description	Action	Parameter setting	Manipulation/operation
Do not want to use a system-memory backup battery.	The controller can be used without installing a system-memory backup battery.	Other parameter No. 20 = 0	In this setting, SEL global data will be cleared when the main power is turned off. In addition, even after running a program that rewrites position data, the previous position data will be restored once the main power is turned off or the application is restarted (software reset). To retain the new position data, the data must be written to the flash ROM in the MANU mode before turning off the main power or restarting the application. Be sure to refer to 2, "When the system-memory backup battery is not used," in Chapter 1 of Part 3.

◎ 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): Other parameter No. 4 = 0	Program run (Application only): Other parameter No. 4 = 1	Error reset (Application only)	Remarks
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	900 ~ 93F	○	△ (Battery and fieldbus errors will be registered in an error list.)				Enabled.	Status display, input error, etc.
	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	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. (Main application version 0.17 or later)	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	D00 ~ D8F	○	○	○ (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	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	FF0 ~ FBF	○	○	○	All programs will be cancelled.		Not enabled.	The controller power must be reconnected (MAIN only). (The CPU and OS will not run.)
	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.
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	Shutdown relay ER status	The motor-drive power ON status remains ON even after the controller has been shut down. The shutdown relay contact 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.)
80A	Expanded-SIO overrun status (SEL reception)	Communication failure. Check for noise, connected equipment and communication setting.
80B	Expanded-SIO parity ER status (SEL reception)	Communication failure. Check for noise, shorted/disconnected communication cable, connected equipment and communication setting.
80C	Expanded SIO framing ER status (SEL reception)	Communication failure. Check for noise, shorted/disconnected communication cable, connected equipment and communication setting.
80D	Expanded-SIO receive ER status due to other factor (SEL reception)	Communication failure. Take the same action specified for error No. 80A, 80B or 80C.
80E	Expanded-SIO receive buffer overflow status (SEL reception)	The receive buffer overflowed. Excessive data was received from outside.
80F	Ethernet control status 1	Ethernet control information (for analysis)
810	Ethernet control status 2	Ethernet control information (for analysis)

(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 extension)
911	Message conversion error (IAI protocol HT reception)	The transmitted message does not match the message format or contains invalid data. (For future extension)
930	Coordinate system number error	The coordinate system number is invalid.
931	Coordinate system type error	The coordinate system type is invalid.
932	Coordinate system definition data count-specification error	The specified number of coordinate system definition data is invalid.
933	Axis number error	The axis number is invalid.
934	Operation type error for SCARA ABS-reset special movement	The operation type for SCARA ABS-reset special movement is invalid.
935	Positioning operation type error	The positioning operation type is invalid.
936	Simple interference check zone number error	The simple interference check zone number is invalid.
938	Simple interference check zone data count-specification error	The specified number of simple interference check zone data is invalid.
939	Detection of entry into simple interference check zone (Message level specification)	Entry into the simple interference check zone was detected. (Message level specification)
93A	R-axis CP jog prohibition error when out of operation range (When tool XY offset is valid)	Move into the operation range by jogging each axis.
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 detection)	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.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
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
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)	The applicable axis is currently in use.

(In the panel window, the three digits after "E" indicate an error number.)

Error No.	Error name	Description, action, etc.
A20	Servo-control-right acquisition error (SIO)	The servo control right is not available.
A21	Servo-control-right duplicate-acquisition error (SIO)	The servo control right has already been acquired.
A22	Servo-control-right non-acquisition error (SIO)	An attempt to retain the servo control right has failed.
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.

(In the panel window, the three digits after "E" indicate an error number.)

Error No.	Error name	Description, action, etc.
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.
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.

(In the panel window, the three digits after "E" indicate an error number.)

Error No.	Error name	Description, action, etc.
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.
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.
A64	SCIF overrun error (SIO bridge)	Communication failure. Check for noise, connected equipment and communication setting.
A65	SCIF receive error (SIO bridge)	Communication failure. Check for noise, shorted/disconnected communication cable, connected equipment and communication setting.
A66	SCI overrun error (SIO bridge)	Communication failure. Check for noise, circuit failure and slave card.
A67	SCI framing error (SIO bridge)	Communication failure. Check for noise, shorting, circuit failure and slave card.
A68	SCI parity error (SIO bridge)	Communication failure. Check for noise, shorting, circuit failure and slave card.
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 support monitor type based on the Standard DIO Board Support Monitor Type/Main CPU Board FROM Procedure does not match the monitor type set in the PC software (monitor screen selection).

(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	Origin-return method error	The setting of “Axis-specific parameter No. 10, Origin-return method” is invalid. (Not incremental encoder AND current position 0 origin 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 origin return	The operation at origin return exceeded the estimate stroke. The origin sensor or creep sensor may be faulty, among other reasons.
B06	Expanded-SIO in-use error	An attempt was made to open a channel already opened by other task.
B07	Expanded-SIO unopen error	An attempt was made to use a channel not opened by own task.
B08	Expanded-SIO duplicate WRIT execution error	WRIT commands were executed simultaneously by multiple tasks for the same channel.
B09	Expanded-SIO RS485 WRIT/READ simultaneous execution error	WRIT and READ commands were executed simultaneously in the RS485 mode.
B0A	Expanded-SIO unassigned-channel use error	An attempt was made to use a channel not assigned properly. Check I/O parameter Nos. 100 to 111 and the statuses of I/O slots.
B10	Phase-Z search timeout error	Phase Z cannot be detected. Check for operation restriction, wiring, encoder, motor, etc.
B11	Origin-sensor pull-out timeout error	Pull-out from the origin sensor cannot be confirmed. Check for operation restriction, wiring, motor, origin 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. Check “Other parameter No. 24, Local variable number for storing READ command return code,” etc.
B13	Backup SRAM data checksum error	The backup SRAM data has been destroyed. Check the battery.
B14	Flash-ROM, 8-Mbit version unsupported function error	An attempt was made to use a function not supported in the flash-ROM, 8-Mbit board environment. (HT connection specification, etc.)
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 socket open-without-close error	An attempt was made to open the socket again without closing it.
B1C	Ethernet channel in-use error	An attempt was made to open a channel already opened by other task.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
B1D	Ethernet non-open error	An attempt was made to use a channel not yet 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.
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.
B70	Checksum error in coordinate system definition data	The flash ROM data is damaged.
B71	Coordinate system number error	The coordinate system number is invalid.
B72	Coordinate system type error	The coordinate system type is invalid.
B73	Error due to prohibition of change of coordinate system data used by servo	Changing of coordinate system data currently used by the servo system is prohibited.
B74	CP-operation restriction zone entry error (PTP/jogging of each axis enabled)	Entry into the CP-operation restriction zone was detected. PTP operation and jogging operation of each axis are enabled.
B75	Singular-point calculation error	CP calculation cannot be performed due to the singular point. Check for invalid coordinate caused by an inappropriate origin of arm 2, etc.
B77	Current arm system setting error	The target arm system to be set does not match the actual angle of arm 2, or coordinates are not yet determined.
B78	Current arm system indetermination error	The current arm system is indeterminable.
B79	R-axis servo OFF detection error during position control correction	R-axis servo OFF was detected during position control correction.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
B7A	Z-axis servo OFF detection error during RZ mechanism correction	Z-axis servo OFF was detected during RZ mechanism correction.
B7B	Error due to target locus inside rear entry prohibition area	The target position or movement locus is inside the rear entry prohibition area.
B7C	Error due to target locus inside CP-operation restriction zone (PTP/jogging of each axis enabled)	The target position or movement locus is inside CP-operation restriction zone. PTP operation and jogging operation of each axis are enabled.
B7D	Physically unrealizable target error	The specified target is unrealizable based on the arm length composition of axes 1 and 2. Check "Axis-specific parameter No. 138, Arm length" and the target value.
B7F	Servo use purpose error	The use purpose of servo is invalid.
B80	Specification-prohibited axis error	Specification of the axis is prohibited. Set an axis that can be specified.
B81	Axis-specific PTP multiple-axis specification error	Axis-specific PTP operation was specified for multiple axes. Axis-specific PTP operation can be specified only for a single axis.
B82	Jogging multiple-axis specification error	Jogging/inching was specified for multiple axes. Jogging/inching can be specified only for a single axis.
B83	Rc = 0 wait timeout error upon Zc-axis origin return	Timeout of R-axis 0 positioning has occurred. Check for operation restriction, wiring, encoder, motor, etc.
B84	Arm length error	The arm length is invalid. Check "Axis-specific parameter No. 138, Arm length."
B85	Operation start-position acquisition error inside work area using application servo	Operation start position cannot be obtained inside the work area using the application servo.
B86	SEL PTRQ command preparation error	An error equivalent to error No. A3A, A39 or A35 was detected.
B87	Error due to target locus error inside tool-center entry prohibition circle	The target position or movement locus is inside the circle where entry of the tool reference point is prohibited.
B88	Logic error during calculation of valid target data	An internal logic error generated during calculation of valid target data.
B89	SCARA CP logic error	An internal logic error was detected during SCARA CP processing.
B8C	Detection of entry into simple interference check zone (Operation-cancellation level specification)	Entry into the interference check zone was detected. (Operation-cancellation level specification)
B8D	SLPR parameter type specification error	The specified SLPR parameter type is invalid.
B8E	SEL STPR command preparation error	An error equivalent to error No. A3A, A39 or A35 was detected.
B8F	Positioning time calculation error	A positioning time calculation error occurred.
B90	Passing distance calculation error	A passing distance calculation error occurred.
B91	Main overspeed requirement error	An excessive speed is required. This error may also occur when passing near the singular point (where arms 1 and 2 form a straight line) during CP operation. Program CP operation by avoiding movements near the singular point. This error may be prevented by lowering the specified speed.
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.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
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.
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 stuck has failed.
C17	Extension-condition code error	Input program step error. The extension condition code is invalid.
C18	Extension-condition LD simultaneous processing over error	The number of LDs processed simultaneously exceeds the limit value.
C19	Extension-condition LD shortage error 1	There is not enough LD when extension condition A or O is used.
C1A	Extension-condition LD shortage error 2	There is not enough LD when extension 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 extension condition AB or OB.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
C1F	Input-condition CND shortage error	The necessary input condition is not found when an extension 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.
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.
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 channel number error	The channel number of the 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.

(In the panel window, the three digits after "E" indicate an error number.)

Error No.	Error name	Description, action, etc.
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.
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.
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 1 opened to the user does not match the parameter. Check "I/O parameter No. 90, Usage of SIO channel opened to user."
C4F	Program data/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

(In the panel window, the three digits after "E" indicate an error number.)

Error No.	Error name	Description, action, etc.
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.
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	Origin-return incomplete error	Origin return has not completed yet.
C70	Absolute coordinate non-confirmation error	Absolute coordinates have not been confirmed. The power must be reconnected.
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.
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 motion data packets overflowed.
C79	Pole sense operation error	Operation is disabled in the pole sense mode.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
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
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 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.
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, etc.).
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	Programs cannot be started from the “I/O processing program at operation/program abort.” (Applicable only to main application version 0.33 or earlier.)
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.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
C99	Origin sensor non-detection error	The origin 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 origin 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 origin 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.
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.

(In the panel window, the three digits after "E" indicate an error number.)C

Error No.	Error name	Description, action, etc.
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 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.
CBF	Positioning distance overflow error	The positioning distance is too large.
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.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
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.
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 (P0E2)	An error occurred in the axis sensor.
D0F	Power stage temperature error (P0E1)	The power stage board exceeded the upper temperature limit.
D10	IPM error (P0E0)	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.
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.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
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.
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.
D51	Fieldbus error (FBMIRQ reset)	A FBMIRQ reset error was detected.
D52	Fieldbus error (FBMBSY)	A FBMBSY was detected.
D53	Fieldbus error (BSYERR)	A BSYERR was detected.
D54	Window lock error (LERR)	A LERR was detected.
D55	Fieldbus error (Min busy)	A Min busy error was detected.
D56	Fieldbus error (MinACK timeout)	A Min ACK timeout was detected.
D57	Fieldbus error (MoutSTB timeout)	A Mout STB timeout was detected.
D58	Fieldbus error (INIT timeout)	An INIT timeout was detected.
D59	Fieldbus error (DPRAM write/read)	A DPRAM write/read error was detected.
D5A	Fieldbus error (TOGGLE timeout)	A TOGGLE timeout was detected.
D5B	Fieldbus error (Access-privilege retry over)	An access-privilege retry over error was detected.
D5C	Fieldbus error (Access-privilege open error)	An access-privilege open error was detected.
D5D	Fieldbus error (FBRS link error)	A FBRS link error was detected.
D5E	Fieldbus error (Mailbox response)	A mailbox response error was detected.
D60	Expanded-SIO 2/4 CH insulation power error	An Expanded-SIO insulation power error was detected.
D61	Expanded-SIO 1/3 CH insulation power error	An Expanded-SIO insulation power error was detected.
D62	Expanded-SIO baud-rate-generator clock oscillation error	An Expanded-SIO clock oscillation error was detected.
D63	Expanded-SIO UART paging error	An Expanded-SIO paging error was detected.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
D64	Expanded-SIO assignment error	The “board channel assignment number” or “expanded-I/O slot assignment number” in I/O parameter Nos. 100, 102, 104, 106, 108 or 110 may be outside the input range or duplicated, a serial communication expansion board may not be installed in the specified slot, or a “communication mode” other than RS232C may have been selected when the “board channel assignment number” is other than “1” or “2,” among other reasons.
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
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.
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.
D80	SCARA unsupported function error	An attempt was made to use a function not supported by SCARA.
D81	Parameter error during calculation of valid target data	An invalid parameter value was detected during calculation of valid target data. Check axis-specific parameter Nos. 7, 8, 138, etc.
D82	Simple interference check zone output-number specification error	A value other than an output port/global flag number (0 is allowed) may have been input, or the specified number may be already used as a system output number via the I/O parameter for output function selection.
D83	Simple interference check zone number error	The simple interference check zone number is invalid.
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.
E12	SIO-bridge SCIF send-queue overflow error	The send queue overflowed.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
E13	SIO-bridge SCI send-queue overflow error	The send queue overflowed.
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.
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.
E2E	No required power stage error	The required power stage is not installed for the valid axes.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
E2F	No required regenerative resistance error	The required regenerative resistance is not installed for the valid axes.
E30	No required motor-drive power error	The required motor-drive power is not installed for the valid axes.
E31	No standard I/O slot error	Standard I/O unit is not installed.
E32	No control power error	Control power unit is not installed.
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. (Check axis-specific parameter No. 38.)
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.
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

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
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.
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	Shutdown relay error	The shutdown relay may have been melted.
E6E	Power-stage rating (W) mismatch error	A power stage with inappropriate rated capacity (W) is installed.
E6F	Power-stage rating (V) mismatch error	A power stage with inappropriate rated voltage (V) is installed.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
E70	Motor-drive power rating (V) mismatch error	A motor-drive power source with inappropriate rated voltage (V) is installed.
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.
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.

(In the panel window, the three digits after "E" indicate an error number.)

Error No.	Error name	Description, action, etc.
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.
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.

◎ 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.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
CD0	Drive unit 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.
EAC	Servo control underrun error	A servo control underrun error was detected.

(In the panel window, the three digits after “E” indicate an error number.)

Error No.	Error name	Description, action, etc.
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.

Trouble Report Sheet

Trouble Report Sheet				Date:	
Company name		Department		Reported by	
TEL	(Ext)	FAX			
IAI agent		Purchase date			
Serial number		Manufacture date			

[1] 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. Work transfer
2. Push-motion operation
3. Load: Approx. ________ kg
4. Speed: Approx. _________ mm/sec

[7] Special specification (option, etc.)

Catalog No.: MJ0119-2A-E (September 2003)

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
New Jersey Office: 7 South Main St., Suite-F, Marlboro, NJ 07746
TEL (732) 683-9101 FAX (732) 683-9103

Home page: www.intelligentactuator.com

IAI Industrieroboter GmbH

Ober der Röth 4, D-65824 Schwalbach am Taunus, Germany
TEL 06196-88950 FAX 06196-889524