



# Table Top Type Robot TT

## First Step Guide Second Edition

Thank you for purchasing our product. Make sure to read the Safety Guide and detailed Instruction Manual (CD) included with the product in addition to this First Step Guide to ensure correct use. This Instruction Manual is original.

**Warning :** Operation of this equipment requires detailed installation and operation instructions which are provided on the CD included in the box this device was packaged in. It should be retained with this device at all times. A copy of the CD Manual can be requested by contacting your nearest IAI Sales Office listed at the back cover of the Instruction Manual or on the First Step Guide.

- Using or copying all or part of this Instruction Manual without permission is prohibited.
- The company names, names of products and trademarks of each company shown in the sentences are registered trademarks.

### Product Check

This product is comprised of the following parts if it is of standard configuration. If you find any fault in the contained model or any missing parts, contact us or our distributor.

#### 1. Parts (The option is excluded.)

| No.                | Part Name                                    | Model   |
|--------------------|--|---|
| 1                  | Robot Main Body (with a built-in controller) | Refer to "How to read the model plate" and "How to read the model of the controller." |
| <b>Accessories</b> |  |   |
| 2                  | Power Supply Plug                            | AP-400-C (Manufacturer: Yamate Electric Co., Ltd.)                                    |
| 3                  | I/O Flat Cable                               | CB-DS-PIO020  |
| 4                  | First Step Guide                             |   |
| 5                  | Operation Manual (CD)                        |   |
| 6                  | Safety Guide                                 |   |

#### 2. Optional Components

| No. | Part Name  | Model |
|-----|--|-------|
| 1   | Main Body Mounting Bracket (with set bolts and nuts) | TT-FT |

#### 3. Teaching Tool (Option)

The personal computer application software or teaching pendant is required for the operations including program creation and setup such as position setting and parameter setting with teaching. Use either of them.

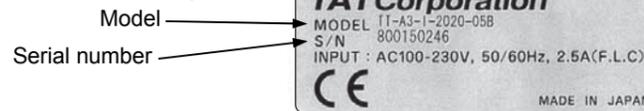
| No. | Part Name  | Model           | Remarks         |
|-----|--|-----------------|-----------------|
| 1   | PC Software (with RS232C Cable + Emergency Stop Box)                               | IA-101-X-MW     | RS232C→RS232C*1 |
| 2   | PC Software (USB conversion adapter + RS232C cable + Cable and Emergency Stop Box) | IA-101-X-USB MW | USB→RS232C*1    |
| 3   | PC Software (with USB Cable + Dummy Plug)  | IA-101-TT-USB   | USB→USB*1       |
| 4   | Teaching pendant   | SEL-T           | —               |
| 5   | Teaching pendant (with deadman switch)   | SEL-TD          | —               |
| 6   | Teaching pendant   | IA-T-X          | —               |
| 7   | Teaching pendant (with deadman switch)   | IA-T-XD         | —               |

\*1 The communication port on the left is for the personal computer and on the right is for the TT.

#### 4. Operation manuals related to this product, which are contained in the CD.

| No. | Name  | Manual No. |
|-----|---|------------|
| 1   | Table Top Type Robot TT Operation Manual    | ME0149     |
| 2   | PC software IA-101-X-MW Operation Manual    | ME0154     |
| 3   | Teaching pendant SEL-T/TD Operation Manual  | ME0183     |
| 4   | Teaching pendant IA-T-X/XD Operation Manual | ME0160     |
| 5   | DeviceNet Operation Manual                  | ME0124     |
| 6   | CC-Link Operation Manual                    | ME0123     |
| 7   | PROFIBUS-DP Operation Manual                | ME0153     |
| 8   | X-SEL Ethernet Operation Manual             | ME0140     |

#### 5. How to read the model plate



#### 6. How to read the Model No.

Model No. Example **TT - A3 - I - 2020 - 05B - DV**  
 ① ② ③ ④ ⑤ ⑥

| ①Series     | ②Type                          | ③Encoder type  | ④XY Stroke        | ⑤Z Stroke    | ⑥Option                                 |
|-------------|--------------------------------|----------------|-------------------|--------------|---|
| TT (Normal) | A2:Gate Type with 2-axis       | I: Incremental | 2020<br>200×200mm | -            | DV:DeviceNet Specifications             |
|             | C2:Cantilever Type with 2-axis |                |                   |              | CC:CC-Link Specifications               |
|             | A3:Gate Type with 3-axis       |                |                   |              | PR:PROFIBUS Specifications              |
|             | C3:Cantilever Type with 2-axis |                | 4040<br>400×400mm | 05B<br>50mm  | ET :Ethernet Specifications             |
|             |                                |                |                   | 10B<br>100mm | FT :Main Body Mounting Bracket included |
|             |                                |                |                   |              | P :I/O PNP Specifications               |

### Basic Specifications

#### [Common Specifications]

| Item                                   | Specifications                                      |
|--|---|
| Surrounding air temperature · humidity | 0 to 40°C, Room Humidity 20% to 85% or less         |
| Motor Type                             | Pulse Motor (Servo Control)                         |
| Position detection method              | Incremental Encoder                                 |
| Driving System                         | Ball Screw (φ10mm, Rolled C10), Ball Screw Lead 6mm |
| Positioning Repeatability              | ± 0.02mm  |
| Backlash                               | 0.1mm or less                                       |
| Guide                                  | Direct Driven Infinite Circulation Type             |
| Allowable Load Moment*1                | Ma:6.5N·m Mb:9.3N·m Mc:16.4N·m                      |

\*1 Value found on the assumption of the life of 5000 km run

#### [Individual Mechanism Specifications]

| Type                        | Stroke (mm) | Max. Speed for each axes (mm/sec) | Acceleration/Deceleration (G) | Max. Load Capacity (kg)*2 | Weight (kg) | Model |        |        |        |                  |
|-----------------------------|-------------|-----------------------------------|-------------------------------|---------------------------|-------------|-------|--------|--------|--------|------------------|
|                             |             |                                   |                               |                           |             |       | X Axis | Y Axis | Z Axis |                  |
| Gate Type with 2-axis       | 200         | 200                               | -                             | 300                       | 0.3         | 10    | 5      | -      | 14.8   | TT-A2-I-2020     |
|                             | 400         | 400                               | -                             | 300                       | 0.3         | 10    | -      | 2      | 33     | TT-A2-I-4040     |
| Gate Type with 3-axis       | 200         | 200                               | 50                            | 300                       | 280         | 0.3   | 10     | -      | 16.5   | TT-A3-I-2020-05B |
|                             | 400         | 400                               | 100                           | 300                       | 280         | 0.3   | 10     | -      | 35     | TT-A3-I-2020-10B |
| Cantilever Type with 2-axis | 200         | 200                               | -                             | 300                       | 0.2         | -     | 4      | -      | 16.3   | TT-C2-I-2020     |
|                             | 400         | 400                               | -                             | 300                       | 0.2         | -     | 4      | -      | 35     | TT-C2-I-4040     |
| Cantilever Type with 3-axis | 200         | 200                               | 50                            | 300                       | 280         | 0.2   | -      | -      | 18     | TT-C3-I-2020-05B |
|                             | 400         | 400                               | 100                           | 300                       | 280         | 0.2   | -      | -      | 37     | TT-C3-I-2020-10B |
|                             |             |                                   |                               |                           |             |       |        |        |        | TT-C3-I-4040-05B |
|                             |             |                                   |                               |                           |             |       |        |        |        | TT-C3-I-4040-10B |

\*2 The load capacity in the case of rated acceleration is shown (Gate Type: 0.3G, Cantilever Type: 0.2G)

#### [Controller Specifications]

| Item  | Specifications  |
|---|---|
| Number of axes                                    | 2-axis  |
| Supply voltage                                    | Single-phase 100 to 115VAC, 200 to 230VAC ± 10%   |
| Power frequency                                   | 50Hz/60Hz   |
| Power-source capacity                             | 100VAC 150VA 210VA<br>200VAC 155VA 215VA  |
| Maximum Current*3                                 | 3A (100VAC), 1.6A (200VAC) 4.2A (100VAC), 2.2A (200VAC)   |
| Rush Current*4                                    | 15A (100VAC), 30A (200VAC)  |
| Leakage Current                                   | 0.75mA  |
| Insulation Strength                               | 2000VAC for 1 min.  |
| Momentary Power Interruption Tolerance            | 500μs or more   |
| Speed Setting                                     | 1 to 300mm/sec  |
| Acceleration Setting                              | 0.01G to 0.3G   |
| Program language                                  | Super SEL language  |
| Number of programs (Number of multitask programs) | 64 programs (16 programs)   |
| Number of program steps                           | 6000 steps (Total)  |
| Number of positions                               | 3000 positions (Total)  |
| Program Startup                                   | Special Digital Switch + Special Start Switch   |
| Data storage device                               | Flash ROM + SRAM*5  |
| Standard I/O Board                                | 16 Input Points / 16 Output Points  |
| Applicable Field Bus                              | DeviceNet / CC-Link / PROFIBUS / Ethernet   |
| Protective functions                              | Over-voltage, motor over current, motor overload, driver temperature abnormality, encoder abnormality, etc. |

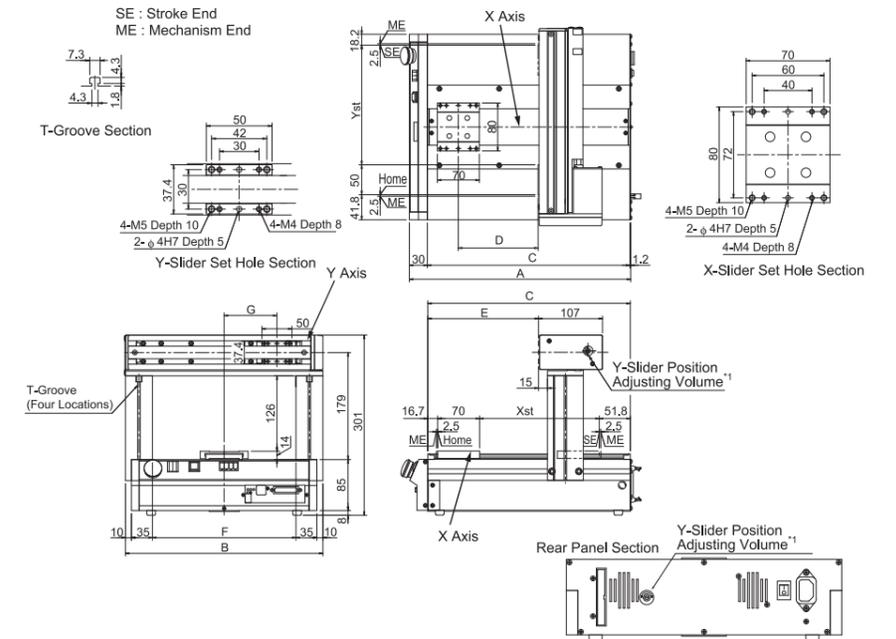
\*3 The current reaches its maximum level when the servo-motor exciting phase is detected which is to be performed in the first servo-motor turning ON processing after the power injection. (Normal: Approx. 1 to 2 sec, Max.: 10 sec)

\*4 Rush current at the power connection continues for about 20 msec. Consider the safety rate at the time when rush current passes. The rush current value varies depending on the impedance of the power line.

\*5 The SRAM data is not battery backed up. Accordingly, when the power is turned off, the data of flags and variables used in the program, are not saved. Take the greatest care. The same procedure is applied when the program or position data is not written on the Flash ROM.

### External Dimensions

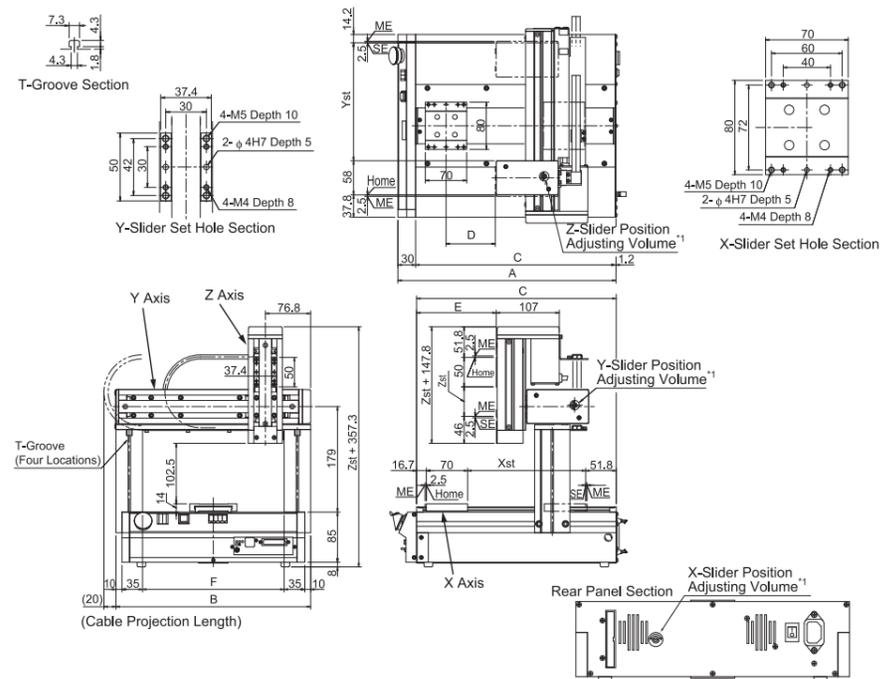
#### [Gate Type with 2-axis TT-A2]



\*1 Fine slider adjustment (for manual tweaking of position).

| Model        | A     | B   | C     | D     | E   | F   | G     | Xst | Yst |
|--------------|-------|-----|-------|-------|-----|-----|-------|-----|-----|
| TT-A2-I-2020 | 369.7 | 330 | 338.5 | 133.3 | 185 | 240 | 88.2  | 200 | 200 |
| TT-A2-I-4040 | 569.7 | 530 | 538.5 | 333.3 | 385 | 440 | 188.2 | 400 | 400 |

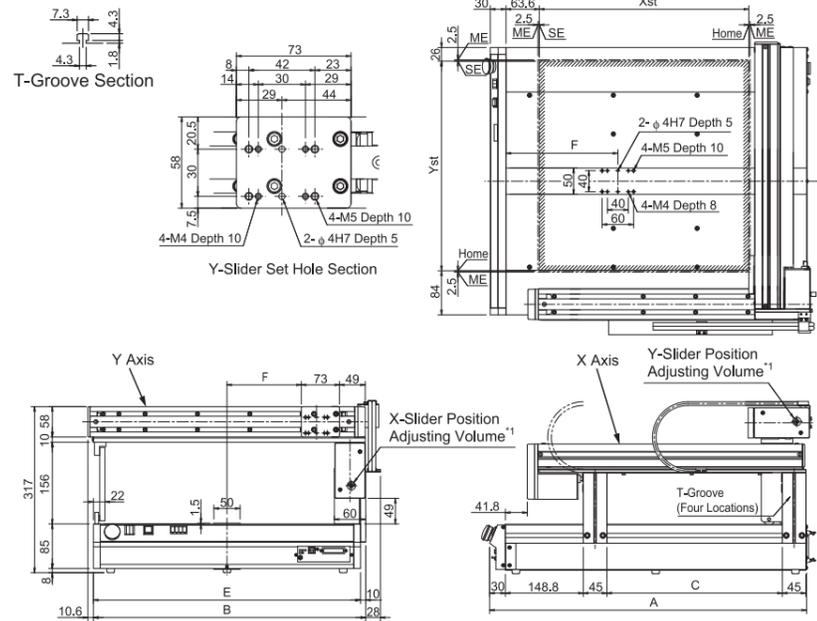
#### [Gate Type with 3-axis TT-A3]



\*1 Fine slider adjustment (for manual tweaking of position).

| Model            | A     | B   | C     | D     | E   | F   | Xst | Yst | Zst |
|------------------|-------|-----|-------|-------|-----|-----|-----|-----|-----|
| TT-A3-I-2020-05B | 369.7 | 330 | 338.5 | 83.3  | 135 | 240 | 200 | 200 | 50  |
| TT-A3-I-2020-10B | 369.7 | 330 | 338.5 | 83.3  | 135 | 240 | 200 | 200 | 100 |
| TT-A3-I-4040-05B | 569.7 | 530 | 538.5 | 283.3 | 335 | 440 | 400 | 400 | 50  |
| TT-A3-I-4040-10B | 569.7 | 530 | 538.5 | 283.3 | 335 | 440 | 400 | 400 | 100 |

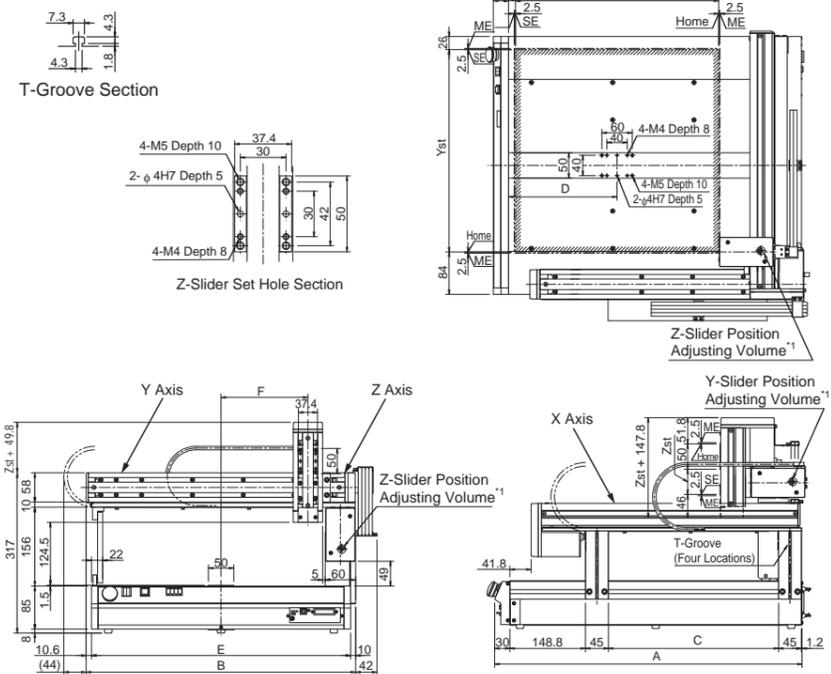
[Cantilever Type with 2-axis TT-C2]



\*1 Fine slider adjustment (for manual tweaking of position).

| Model        | A   | B   | C   | D     | E   | F   | Xst | Yst |
|--------------|-----|-----|-----|-------|-----|-----|-----|-----|
| TT-C2-I-2020 | 405 | 320 | 135 | 120   | 310 | 42  | 200 | 200 |
| TT-C2-I-4040 | 605 | 520 | 335 | 213.6 | 510 | 142 | 400 | 400 |

[Cantilever Type with 3-axis TT-C3]



(Cable Projection Length)

\*1 Fine slider adjustment (for manual tweaking of position).

| Model            | A   | B     | C   | D     | E   | F   | Xst | Yst | Zst |
|------------------|-----|-------|-----|-------|-----|-----|-----|-----|-----|
| TT-C3-I-2020-05B | 405 | 330.6 | 135 | 120   | 310 | 71  | 200 | 200 | 50  |
| TT-C3-I-2020-10B | 405 | 330.6 | 135 | 120   | 310 | 71  | 200 | 200 | 100 |
| TT-C3-I-4040-05B | 605 | 530.6 | 335 | 213.6 | 510 | 171 | 400 | 400 | 50  |
| TT-C3-I-4040-10B | 605 | 530.6 | 335 | 213.6 | 510 | 171 | 400 | 400 | 100 |

Installation Environment

Do not use this product in the following environment:

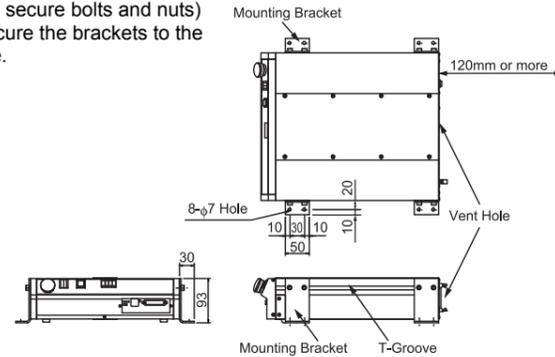
- Location where the surrounding air temperature exceeds the range of 0 to 40°C
- Location where condensation occurs due to abrupt temperature changes
- Relative humidity less than 20%RM or greater than 85%RM
- Location exposed to corrosive gases or combustible gases
- Location exposed to significant amount of dust, salt or iron powder
- Location subject to direct vibration or impact
- Location exposed to direct sunlight
- Location where the product may come in contact with water, oil or chemical droplets

When using the product in any of the locations specified below, provide a sufficient shield.

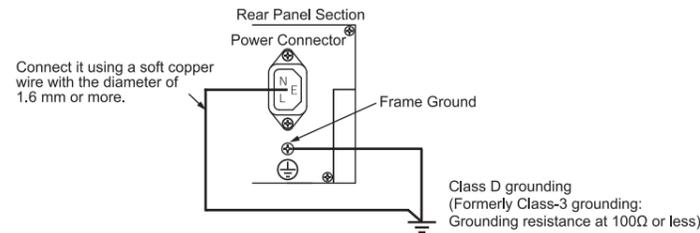
- Location subject to electrostatic noise
- Location where high electrical or magnetic field is present
- Location with the mains or power lines passing nearby

Installation and Noise Elimination

1. There is a cooling vent hole on the main body's rear panel section. Do not close the vent hole when the main body is installed.
2. When it required to fix the main body, fix it as follows using the optional mounting brackets (Model TT-FT: 4 sets with secure bolts and nuts) User supplied bolts to secure the brackets to the mounting surface or table.



3. Protective Ground



4. Noise Elimination Grounding (Frame Ground)

Connect it using a soft copper wire with the diameter of 1.6 mm or more to the frame ground on the main body (Refer to the above figure).

Do not share the ground wire with or connect to other equipment. Ground each controller. The same procedure is applied for the protective ground.



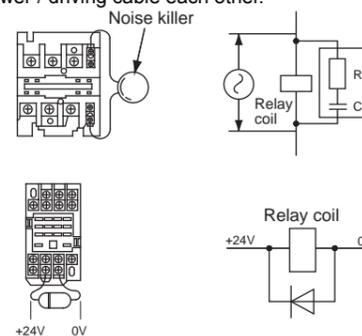
5. Precautions Regarding wiring Method

Separate the I/O cable, communication line and power / driving cable each other.

6. Noise Sources and Elimination

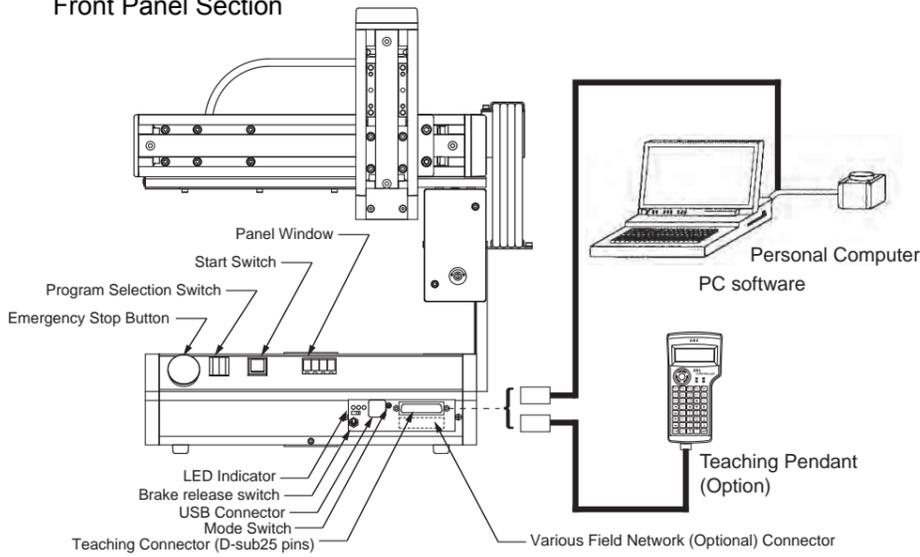
Carry out noise elimination measures for power devices on the same power path and in the same equipment. The following are examples of measures to eliminate noise sources:

- ① AC solenoid valves, magnet switches and relays  
[Measure] Install a Noise killer parallel with the coil.
- ② DC solenoid valves, magnet switches and relays  
[Measure] Install a diode parallel with the coil.  
Use a DC relay with a built-in diode.

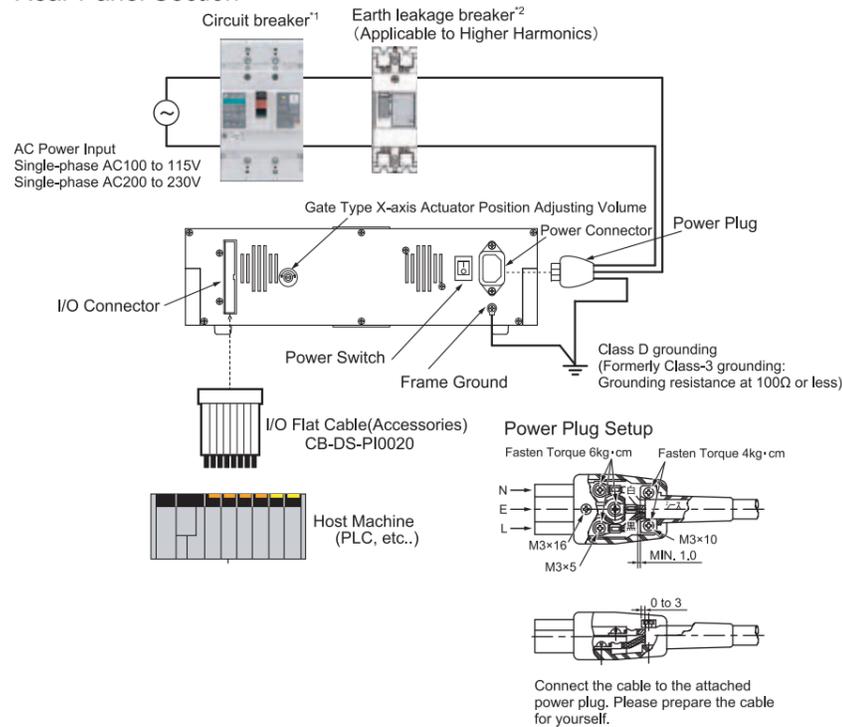


# Connection Diagram

## Front Panel Section



## Rear Panel Section



- \*1 For the selection of the circuit breaker, perform it according to the following items.  
 Breaker Teaching pendant Value > Power Capacity + AC Input Voltage  
 (Refer to the item for the controller specifications for the power capacity).  
 • The current reaches the maximum level when the servo-motor is turned on and the servo-motor exciting phase is detected. Select the circuit breaker rated current that does not trip the maximum current.  
 • Select the circuit breaker that does not trip with the rush current described in the controller specifications.  
 (Refer to the operating characteristic curve described in the manufacturer's catalog.)  
 • For the rated breaking current for the circuit breaker, select the breaking current value with which the current can be securely broken down even when short-circuit current passes.  
 Rated Breaking Current > Short-circuit Current = Primary Power Supply Capacity/Power Voltage.  
 • Select the breaking current value for the circuit breaker leaving some margin.

- \*2 When the leakage breaker is to be installed, it is required to select it with the purpose clarified such as protection from fire or human body protection.  
 Measure the leakage current at the location where the leakage breaker has been installed.  
 Use the "applicable to higher harmonics type" leakage breaker.

# I/O Signals

## Input

| Pin No. | Electric wire color | Port No. | Function in Standard Setting (in Delivery) | Remarks       |  |
|---------|---------------------|----------|--|---------------|--|
|         |                     |          |  | Parameter No. | Parameter Name                           |
| 1       | BR 1                | —        | I/O Power Source +24V                      | 30            | Input function select 000 <sup>1</sup> 2 |
| 2       | RD 1                | 016      | Universal Input                            | 31            | Input function select 001                |
| 3       | OR 1                | 017      | Universal Input                            | 32            | Input function select 002                |
| 4       | YW 1                | 018      | Universal Input                            | 33            | Input function select 003                |
| 5       | GN 1                | 019      | Universal Input                            | 34            | Input function select 004                |
| 6       | BL 1                | 020      | Universal Input                            | 35            | Input function select 005                |
| 7       | PL 1                | 021      | Universal Input                            | 36            | Input function select 006                |
| 8       | GY 1                | 022      | Universal Input                            | 37            | Input function select 007 <sup>3</sup>   |
| 9       | WT 1                | 023      | Universal Input                            | 38            | Input function select 008 <sup>3</sup>   |
| 10      | BK 1                | 024      | Universal Input                            | 39            | Input function select 009 <sup>3</sup>   |
| 11      | BR 2                | 025      | Universal Input                            | 40            | Input function select 010 <sup>3</sup>   |
| 12      | RD 2                | 026      | Universal Input                            | 41            | Input function select 011 <sup>3</sup>   |
| 13      | OR 2                | 027      | Universal Input                            | 42            | Input function select 012 <sup>3</sup>   |
| 14      | YW 2                | 028      | Universal Input                            | 43            | Input function select 013 <sup>3</sup>   |
| 15      | GN 2                | 029      | Universal Input                            | 44            | Input function select 014                |
| 16      | BL 2                | 030      | Universal Input                            | 45            | Input function select 015                |
| 17      | PL 2                | 031      | Universal Input                            |               |  |

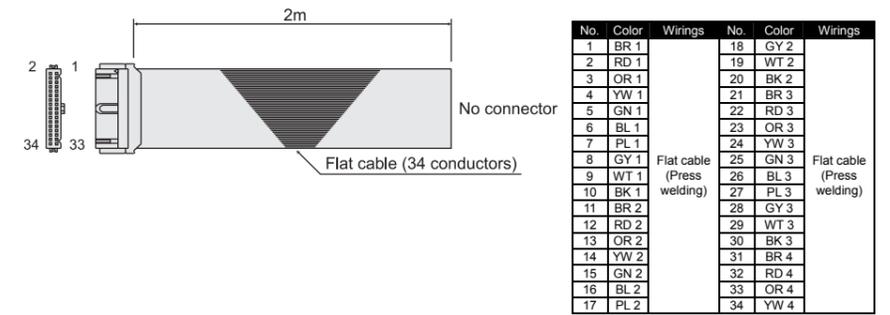
- \*1 Set the input functions using the I/O parameter Nos. 30 to 45 (Input Function Selection 000 to 015) and set the port Nos. that assign the each of the set functions, using the I/O parameter Nos. 282 to 298.  
 \*2 If the input function selection 000 (program start) is assigned to any port except for the Port No. 000, the start switch on the front panel is disabled.  
 \*3 When the input function selection 007 to 013 (program No. designating digital switch) are assigned to any port except for the port Nos. 007 to 013, the program change digital switch on the front panel is disabled.

## Output

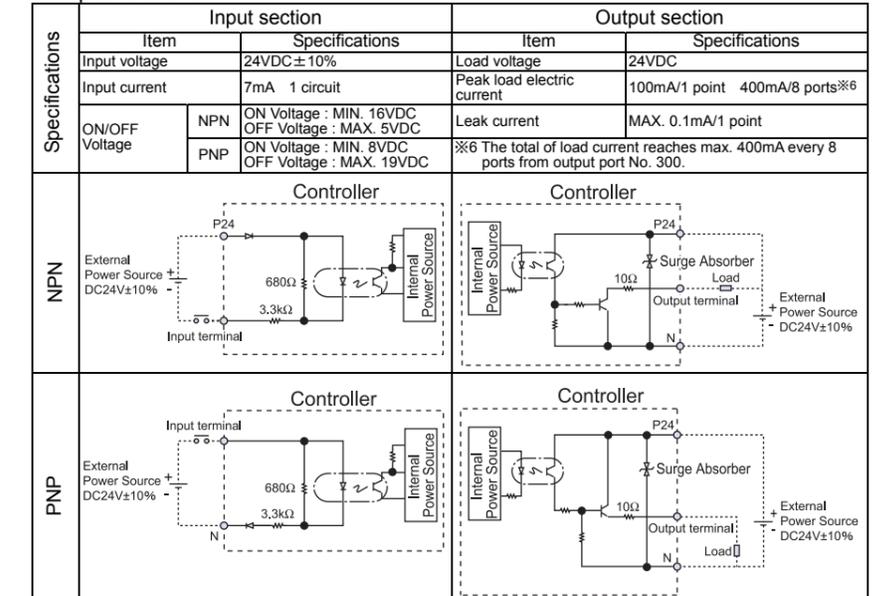
| Pin No. | Electric wire color | Port No. | Function in Standard Setting (in Delivery) | Remarks       |  |
|---------|---------------------|----------|--|---------------|--|
|         |                     |          |  | Parameter No. | Parameter Name   |
| 18      | GY 2                | 316      | Universal Output                           | 46            | Output function select 300 <sup>5</sup><br>Output function select 300 (Area 2) |
| 19      | WT 2                | 317      | Universal Output                           | 47            | Output function select 301 <sup>5</sup><br>Output function select 301 (Area 2) |
| 20      | BK 2                | 318      | Universal Output                           | 48            | Output function select 302 <sup>5</sup><br>Output function select 302 (Area 2) |
| 21      | BR 3                | 319      | Universal Output                           | 49            | Output function select 303 <sup>5</sup><br>Output function select 303 (Area 2) |
| 22      | RD 3                | 320      | Universal Output                           | 50            | Output function select 304 <sup>5</sup><br>Output function select 304 (Area 2) |
| 23      | OR 3                | 321      | Universal Output                           | 51            | Output function select 305 <sup>5</sup><br>Output function select 305 (Area 2) |
| 24      | YW 3                | 322      | Universal Output                           | 52            | Output function select 306 <sup>5</sup><br>Output function select 306 (Area 2) |
| 25      | GN 3                | 323      | Universal Output                           | 53            | Output function select 307 <sup>5</sup><br>Output function select 307 (Area 2) |
| 26      | BL 3                | 324      | Universal Output                           | 54            | Output function select 308 <sup>5</sup><br>Output function select 308 (Area 2) |
| 27      | PL 3                | 325      | Universal Output                           | 55            | Output function select 309 <sup>5</sup><br>Output function select 309 (Area 2) |
| 28      | GY 3                | 326      | Universal Output                           | 56            | Output function select 310 <sup>5</sup><br>Output function select 310 (Area 2) |
| 29      | WT 3                | 327      | Universal Output                           | 57            | Output function select 311 <sup>5</sup><br>Output function select 311 (Area 2) |
| 30      | BK 3                | 328      | Universal Output                           | 58            | Output function select 312 <sup>5</sup><br>Output function select 312 (Area 2) |
| 31      | BR 4                | 329      | Universal Output                           | 59            | Output function select 313 <sup>5</sup><br>Output function select 313 (Area 2) |
| 32      | RD 4                | 330      | Universal Output                           | 60            | Output function select 314 <sup>5</sup><br>Output function select 314 (Area 2) |
| 33      | OR 4                | 331      | Universal Output                           | 61            | Output function select 315 <sup>5</sup><br>Output function select 315 (Area 2) |
| 34      | YW 4                | —        | I/O Power Source 0V                        | 346           | Output function select 315 (Area 2)  |

- \*4 Set the output functions using the I/O parameter Nos. 46 to 61 (Output Function Selection 300 to 315) and set the port Nos. that assign the each of the set functions, using the I/O parameter Nos. 299 to 314. Also, setting the output functions using the I/O parameter Nos. 331 to 346 (Output Function Selection 300 Area 2 to 315 Area 2) and setting the Port Nos. that assign the each of the set functions, using the I/O parameter Nos. 315 to 330, are available.  
 When the system output signal is output to the LEDs on the panel window, use the Output Function Selection Area 2.  
 \*5 Because the output function selections 300 to 304 are allocated to the LEDs on the panel window, when the parameters of 46 to 50 are set to universal output, or the Port No. allocation is changed using the parameter 299 to 303 settings, the LEDs are disabled.

## I/O Flat Cable (Accessories) CB-DS-PIO020

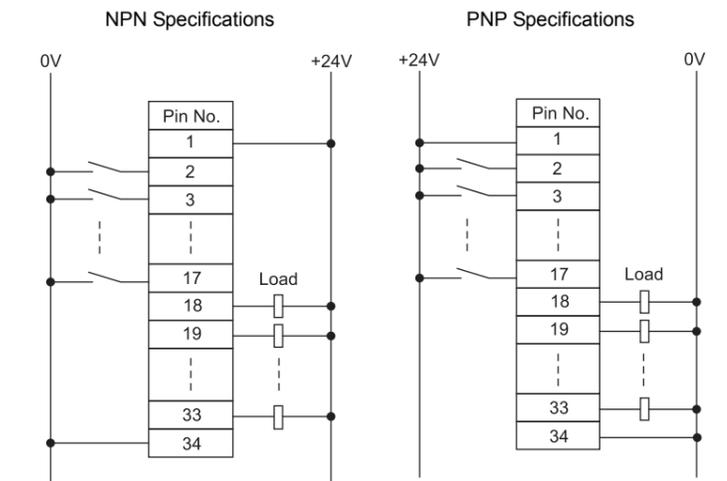


## I/O Specifications



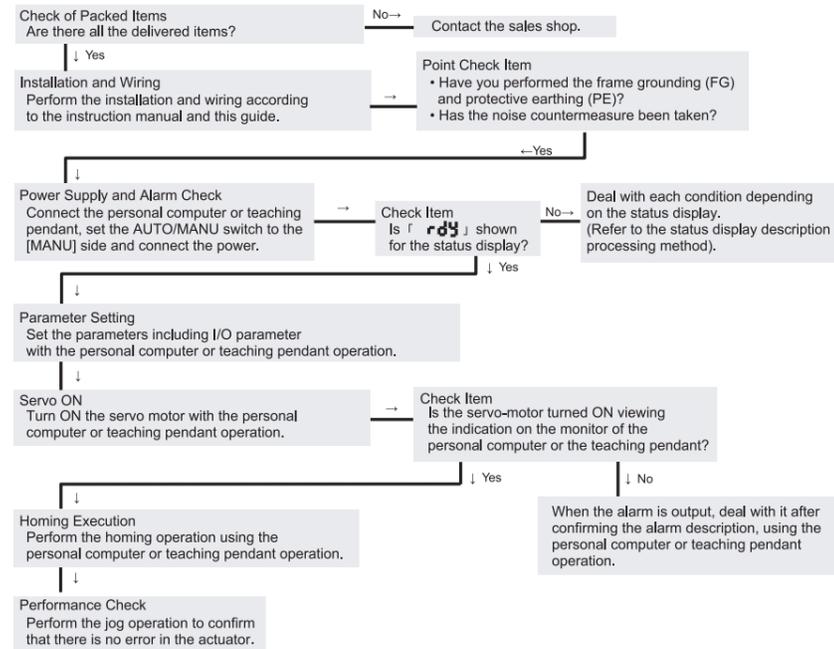
For the input and output, an equivalent circuit expressing the logic is used.

## I/O Circuit Diagrams



## Starting Procedures

When using this product for the first time, make sure to avoid mistakes and incorrect wiring by referring to the procedure below.



Set-up for operation is completed.

## Troubleshooting

The following alarm displays are frequently generated at the start-up operation. Deal with each of them referring to the following table.

| Status display | Status contents  | Cause and Remedy   |
|----------------|--|--|
| ErG            | During Emergency-stop  | It is not an alarm.<br><ul style="list-style-type: none"> <li>It is caused when the emergency stop button is not cleared on the front panel. Clear it.</li> <li>It is generated when the emergency stop switch in the teaching pendant or the personal computer application software is not cancelled. In such case, cancel it.</li> <li>It is generated when the personal computer cable is not connected to the emergency stop box.</li> </ul> |
| dSF            | Deadman switch OFF   | It is not an alarm.<br><ul style="list-style-type: none"> <li>It generated when the AUTO/MANU switch has been set to "MANU" and the personal computer or the teaching pendant is not connected. Connect the personal computer or the teaching pendant or set the AUTO/MANU switch to "AUTO".</li> <li>When the actuator is to be started up, hold the deadman switch on the teaching pendant to turn it on.</li> </ul>                           |
| RCF            | AC Power Interruption<br>Momentary Power Failure<br>Power Voltage Drop | It is generated when the power voltage is not supplied. Check the power supply.  |
| EE69           | 24V I/O Error  | It is generated when the +24V power for I/O is not supplied. Check the power supply.<br>(Procedure for starting up I/O 24V power unit without connection)<br>Set both the I/O parameter No. 10 and No. 12 to "0".<br>In this case, the I/O connection is invalid.  |
| EdS            | Field Bus Error  | It is generated when the field bus link connection is not established. Check the link cable connection, I/O parameter and PLC parameter settings.<br>(How to start up the controller without connecting the field bus)<br>Set both the I/O parameter No. 10 and No. 12 to "0".   |

**IAI**  
Quality and Innovation

## IAI Corporation

Head Office: 577-1 Obane Shimizu-KU Shizuoka City Shizuoka 424-0103, Japan  
 TEL +81-54-364-5105 FAX +81-54-364-2589  
 website: www.iai-robot.co.jp/

Technical Support available in USA, Europe and China

## IAI America, Inc.

Head Office: 2690 W, 237th Street Torrance, CA 90505  
 TEL (310) 891-6015 FAX (310) 891-0815  
 Chicago Office: 1261 Hamilton Parkway Itasca, IL 60143  
 TEL (630) 467-9900 FAX (630) 467-9912  
 Atlanta Office: 1220 Kennestone Circle Suite 108 Marietta, GA 30066  
 TEL (678) 354-9470 FAX (678) 354-9471  
 website: www.intelligentactuator.com

## IAI Industrieroboter GmbH

Ober der Röth 4, D-65824 Schwalbach am Taunus, Germany  
 TEL 06196-88950 FAX 06196-889524

## IAI (Shanghai) Co., Ltd.

SHANGHAI JIAHUA BUSINESS CENTER A8-303, 808, Hongqiao Rd. Shanghai 200030, China  
 TEL 021-6448-4753 FAX 021-6448-3992  
 website: www.iai-robot.com

# Tabletop Robot TT

---

Operation Manual 10th Edition



***IAI America, Inc.***





## Please Read Before Use

Thank you for purchasing our product.

This Operation Manual explains the handling methods, structure and maintenance of this product, among others, providing the information you need to know to use the product safely.

Before using the product, be sure to read this manual and fully understand the contents explained herein to ensure safe use of the product.

The CD or DVD that comes with the product contains operation manuals for IAI products.

When using the product, refer to the necessary portions of the applicable operation manual by printing them out or displaying them on a PC.

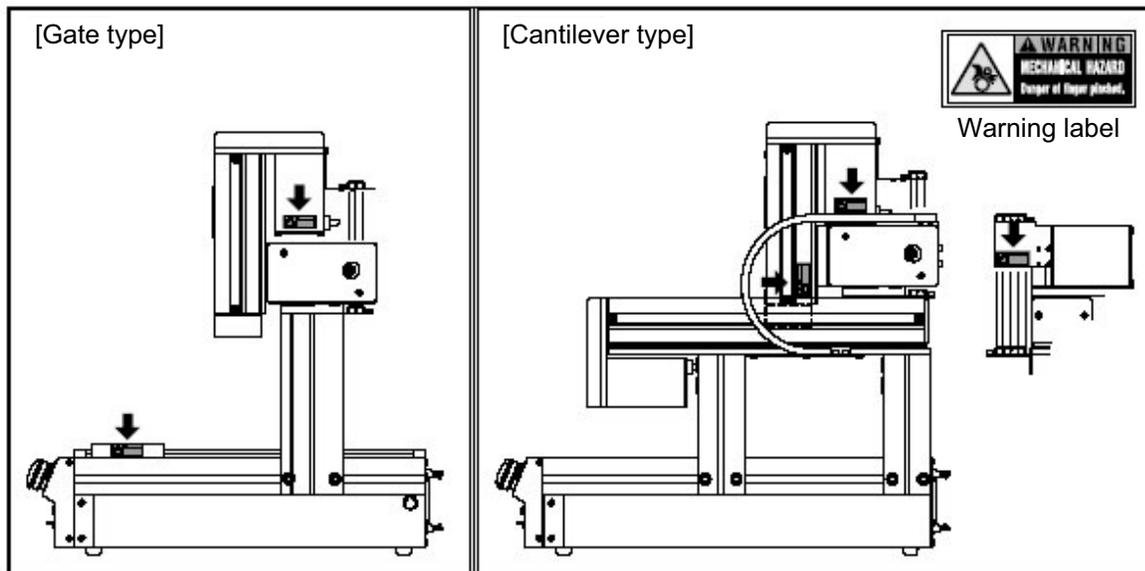
After reading the Operation Manual, keep it in a convenient place so that whoever is handling this product can reference it quickly when necessary.

### [Important]

- This Operation Manual is original.
- The product cannot be operated in any way unless expressly specified in this Operation Manual. IAI shall assume no responsibility for the outcome of any operation not specified herein.
- Information contained in this Operation Manual is subject to change without notice for the purpose of product improvement.
- If you have any question or comment regarding the content of this manual, please contact the IAI sales office near you.
- Using or copying all or part of this Operation Manual without permission is prohibited.
- The company names, names of products and trademarks of each company shown in the sentences are registered trademarks.

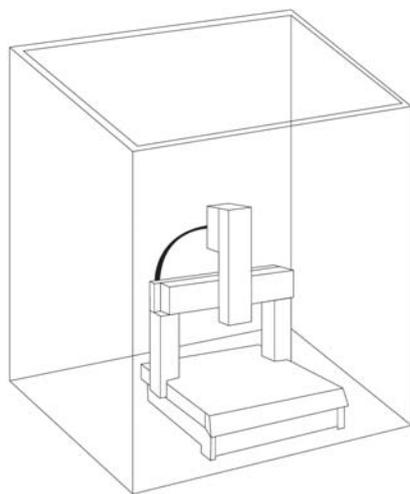
## 1. Notes on operation

To prevent pinching of fingers, do not bring your fingers near the following areas where a warning label is attached while the actuator is moving.



## 2. Installation of safety cage

It is strongly recommended that the robot be enclosed by a safety cage to ensure safety of the operator. When the robot is enclosed by a safety cage, the robot will satisfy the Machine Directives regardless of how it is used by the operator.



← Enclose by a safety cage.

3. The maximum sound pressure level of this robot is 76.4 dB.

4. After grease has been applied to the guide and ball screw during maintenance and inspection, be sure to install the covers.



INTELLIGENT ACTUATOR

---

## **CE Marking**

If a compliance with the CE Marking is required, please follow Overseas Standards Compliance Manual (ME0287) that is provided separately.



## 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.)

#### ■ Contact Us

This robot has been designed and manufactured with the utmost attention and care. Should you find any defect, however, or have any question regarding the handling of the robot, please contact IAI at the address and numbers specified at the end of this manual.



## Table of Contents

|   |    |
|---|----|
| Safety Guide .....  | 1  |
| Chapter 1 Installation.....   | 9  |
| 1.1 Introduction .....  | 9  |
| 1.2 Models .....  | 10 |
| 1.3 Safety Precautions.....   | 11 |
| 1.4 Warranty.....   | 12 |
| 1.4.1 Warranty Period.....  | 12 |
| 1.4.2 Scope of Warranty.....  | 12 |
| 1.4.3 Honoring the Warranty.....  | 12 |
| 1.4.4 Limited Liability.....  | 12 |
| 1.4.5 Conditions of Conformance with Applicable Standards/Regulations, Etc., and Applications.... | 13 |
| 1.4.6 Other Items Excluded from Warranty.....   | 13 |
| 2. Specifications .....   | 14 |
| 2.1 Basic Specifications .....  | 14 |
| 2.2 Name and Function of Each Part.....   | 15 |
| 2.2.1 Robot Body .....  | 15 |
| 2.2.2 Front Panel.....  | 18 |
| 2.2.3 Codes Displayed on the Panel Window.....  | 21 |
| 2.2.4 Rear Panel .....  | 23 |
| 2.2.5 I/O Connector Pin Assignments .....   | 25 |
| 2.3 Interfaces of the Tabletop Robot.....   | 26 |
| 2.3.1 Standard Interface (Main Application Version 0.18 or Earlier) .....                         | 26 |
| 2.3.2 Standard Interface (Main Application Version 0.19 or Later) .....                           | 28 |
| 2.4 External I/O Specifications.....  | 32 |
| 2.4.1 NPN Specification .....   | 32 |
| 2.4.2 PNP Specification.....  | 34 |
| 2.5 External Dimensions.....  | 36 |
| 3. Installation Environment, Noise Measures and Other.....  | 44 |
| 3.1 Installation Environment .....  | 44 |
| 3.2 Installation.....   | 45 |
| 3.2.1 Brackets (Optional) .....   | 45 |
| 3.2.2 Installing the Load, Etc.....   | 45 |
| 3.2.3 Using the T-grooves .....   | 46 |
| 3.3 Power Source .....  | 46 |
| 3.4 Noise Measures and Grounding .....  | 47 |
| 3.4.1 Grounding .....   | 47 |
| 3.4.2 Noise sources and noise elimination .....   | 47 |
| 4. System Setup .....   | 49 |
| 4.1 Connecting the Tabletop Robot with Peripheral Equipment.....                                  | 49 |
| 4.2 I/O Connection Diagram (External DIOs) .....  | 50 |
| 4.2.1 NPN specification.....  | 50 |
| 4.2.2 PNP specification .....   | 51 |



|           |  |     |
|-----------|--|-----|
| Chapter 2 | Operation .....  | 52  |
| 1.        | Operation.....   | 52  |
| 1.1       | How to Start a Program .....   | 52  |
| 1.2       | Starting a Program by Auto-Start via Parameter Setting.....                | 53  |
| 1.3       | Starting via the Digital Program Selector Switch and Function Switch ..... | 54  |
| 2.        | Controller Data .....  | 55  |
| 2.1       | Data Structure.....  | 55  |
| 2.2       | Saving Data .....  | 56  |
| Chapter 3 | X-SEL Language Data .....  | 58  |
| 1.        | Values and Symbols Used in SEL Language .....                              | 58  |
| 2.        | Position Part.....   | 71  |
| 3.        | Command Part .....   | 72  |
| Chapter 4 | Commands .....   | 74  |
| 1.        | List of SEL Language Command Codes by Function .....                       | 74  |
| 1.1       | List of Commands by Function .....   | 74  |
| 1.2       | List of Commands in Alphabetical Order .....                               | 79  |
| 2.        | Explanation of Commands .....  | 84  |
| 3.        | Key Characteristics of Actuator Control Commands and Points to Note .....  | 224 |
| 3.1       | Continuous Movement Commands .....   | 224 |
| 3.2       | PATH/PSPL Commands .....   | 226 |
| 3.3       | CIR/ARC Commands.....  | 226 |
| 3.4       | CIR2/ARC2/ARCD/ARCC Commands .....   | 226 |
| 4.        | Palletizing Function .....   | 227 |
| 4.1       | How to Use .....   | 227 |
| 4.2       | Palletizing Setting .....  | 227 |
| 4.3       | Palletizing Calculation.....   | 233 |
| 4.4       | Palletizing Movement.....  | 234 |
| 4.5       | Program Examples .....   | 236 |
| 5.        | Pseudo-Ladder Task.....  | 244 |
| 5.1       | Basic Frame.....   | 244 |
| 5.2       | Ladder Statement Field.....  | 245 |
| 5.3       | Points to Note .....   | 245 |
| 5.4       | Program Example .....  | 246 |



|   |     |
|---|-----|
| Chapter 5 Maintenance and Inspection .....  | 247 |
| 1. Inspection Items and Inspection Intervals.....                                   | 247 |
| 2. Visual Inspection of the Exterior .....  | 247 |
| 3. Visual Inspection and Cleaning .....   | 247 |
| 3.1 Cleaning.....   | 247 |
| 3.2 Interior Inspection .....   | 248 |
| 3.3 Internal Cleaning.....  | 248 |
| 4. Greasing the Guides.....   | 248 |
| 4.1 Applicable Grease .....   | 248 |
| 4.2 How to Apply Grease .....   | 248 |
| 5. Greasing the Ball Screw .....  | 249 |
| 5.1 Applicable Grease .....   | 249 |
| 5.2 How to Apply Grease .....   | 249 |
| 6. Timing Belt.....   | 249 |
| 6.1 Inspecting the Belt .....   | 249 |
| 6.2 Applicable Belt.....  | 249 |
| 6.3 Belt Replacement Procedure .....  | 250 |
| Appendix .....  | 251 |
| © How to Create a Program .....   | 251 |
| 1. Position Table .....   | 251 |
| 2. Program Format .....   | 252 |
| 3. Positioning to Five Positions.....   | 253 |
| 4. How to Use TAG and GOTO .....  | 254 |
| 5. Moving Back and Forth between Two Points.....                                    | 255 |
| 6. Path Operation .....   | 256 |
| 7. Output Control during Path Movement .....  | 257 |
| 8. Circular/Arc Operation .....   | 258 |
| 9. Home-return Completion Output .....  | 259 |
| 10. Moving an Axis Selectively based on Input and Outputting a Completion Signal .. | 260 |
| 11. Changing the Moving Speed .....   | 261 |
| 12. Changing the Speed during Movement .....  | 262 |
| 13. Local/Global Classification of Variables and Flags .....                        | 263 |
| 14. How to Use Subroutines.....   | 264 |
| 15. Pausing the Operation.....  | 265 |
| 16. Aborting the Operation 1 (CANC).....  | 266 |



|   |     |
|---|-----|
| 17. Aborting the Operation 2 (STOP) .....                           | 267 |
| 18. Moving to a Specified Position Number .....                     | 268 |
| 19. Conditional Jump .....  | 269 |
| 20. Waiting for Multiple Inputs .....                               | 270 |
| 21. How to Use Offset .....   | 271 |
| 22. Executing an Operation n Times .....                            | 272 |
| 23. Constant-pitch Feed Operation.....                              | 273 |
| 24. Jogging.....  | 274 |
| 25. Switching Programs.....   | 275 |
| 26. Aborting a Program .....  | 276 |
| <br>  |     |
| ⊙ How to Use Internal DIOs.....                                     | 277 |
| 1. Internal DIs and Dedicated Functions.....                        | 277 |
| 2. Showing User SEL Program Data on the 7-segment LED Display ..... | 278 |
| <br>  |     |
| ⊙ List of Parameters .....  | 281 |
| 1. I/O Parameters .....   | 282 |
| 2. Parameters Common to All Axes .....                              | 296 |
| 3. Axis-Specific Parameters.....                                    | 299 |
| 4. Driver Card Parameters.....                                      | 303 |
| 5. Encoder Parameters.....  | 307 |
| 6. I/O-Slot Card Parameters .....                                   | 308 |
| 7. Other Parameters .....   | 309 |
| 8. Manual Operation Types .....                                     | 314 |
| 9. Use Examples of Key Parameters.....                              | 315 |
| ⊙ Error Level Control.....  | 317 |
| ⊙ Error List (Main application).....                                | 319 |
| ⊙ Error List (Main core).....                                       | 343 |
| ⊙ Troubleshooting of X-SEL Controller .....                         | 348 |
| Trouble Report Sheet.....   | 351 |
| Change History .....  | 352 |



## Safety Guide

“Safety Guide” has been written to use the machine safely and so prevent personal injury or property damage beforehand. Make sure to read it before the operation of this product.

### Safety Precautions for Our Products

The common safety precautions for the use of any of our robots in each operation.

| No. | Operation Description | Description  |
|-----|-----------------------|--|
| 1   | Model Selection       | <ul style="list-style-type: none"><li>● This product has not been planned and designed for the application where high level of safety is required, so the guarantee of the protection of human life is impossible. Accordingly, do not use it in any of the following applications.<ol style="list-style-type: none"><li>1) Medical equipment used to maintain, control or otherwise affect human life or physical health.</li><li>2) Mechanisms and machinery designed for the purpose of moving or transporting people (For vehicle, railway facility or air navigation facility)</li><li>3) Important safety parts of machinery (Safety device, etc.)</li></ol></li><li>● Do not use the product outside the specifications. Failure to do so may considerably shorten the life of the product.</li><li>● Do not use it in any of the following environments.<ol style="list-style-type: none"><li>1) Location where there is any inflammable gas, inflammable object or explosive</li><li>2) Place with potential exposure to radiation</li><li>3) Location with the ambient temperature or relative humidity exceeding the specification range</li><li>4) Location where radiant heat is added from direct sunlight or other large heat source</li><li>5) Location where condensation occurs due to abrupt temperature changes</li><li>6) Location where there is any corrosive gas (sulfuric acid or hydrochloric acid)</li><li>7) Location exposed to significant amount of dust, salt or iron powder</li><li>8) Location subject to direct vibration or impact</li></ol></li><li>● For an actuator used in vertical orientation, select a model which is equipped with a brake. If selecting a model with no brake, the moving part may drop when the power is turned OFF and may cause an accident such as an injury or damage on the work piece.</li></ul> |



| No. | Operation Description    | Description  |
|-----|--------------------------|--|
| 2   | Transportation           | <ul style="list-style-type: none"><li>● When carrying a heavy object, do the work with two or more persons or utilize equipment such as crane.</li><li>● When the work is carried out with 2 or more persons, make it clear who is to be the leader and who to be the follower(s) and communicate well with each other to ensure the safety of the workers.</li><li>● When in transportation, consider well about the positions to hold, weight and weight balance and pay special attention to the carried object so it would not get hit or dropped.</li><li>● Transport it using an appropriate transportation measure.<br/>The actuators available for transportation with a crane have eyebolts attached or there are tapped holes to attach bolts. Follow the instructions in the operation manual for each model.</li><li>● Do not step or sit on the package.</li><li>● Do not put any heavy thing that can deform the package, on it.</li><li>● When using a crane capable of 1t or more of weight, have an operator who has qualifications for crane operation and sling work.</li><li>● When using a crane or equivalent equipments, make sure not to hang a load that weighs more than the equipment's capability limit.</li><li>● Use a hook that is suitable for the load. Consider the safety factor of the hook in such factors as shear strength.</li><li>● Do not get on the load that is hung on a crane.</li><li>● Do not leave a load hung up with a crane.</li><li>● Do not stand under the load that is hung up with a crane.</li></ul> |
| 3   | Storage and Preservation | <ul style="list-style-type: none"><li>● The storage and preservation environment conforms to the installation environment. However, especially give consideration to the prevention of condensation.</li><li>● Store the products with a consideration not to fall them over or drop due to an act of God such as earthquake.</li></ul>  |
| 4   | Installation and Start   | <p>(1) Installation of Robot Main Body and Controller, etc.</p> <ul style="list-style-type: none"><li>● Make sure to securely hold and fix the product (including the work part). A fall, drop or abnormal motion of the product may cause a damage or injury. Also, be equipped for a fall-over or drop due to an act of God such as earthquake.</li><li>● Do not get on or put anything on the product. Failure to do so may cause an accidental fall, injury or damage to the product due to a drop of anything, malfunction of the product, performance degradation, or shortening of its life.</li><li>● When using the product in any of the places specified below, provide a sufficient shield.<ol style="list-style-type: none"><li>1) Location where electric noise is generated</li><li>2) Location where high electrical or magnetic field is present</li><li>3) Location with the mains or power lines passing nearby</li><li>4) Location where the product may come in contact with water, oil or chemical droplets</li></ol></li></ul>  |



| No. | Operation Description  | Description  |
|-----|------------------------|--|
| 4   | Installation and Start | <p>(2) Cable Wiring</p> <ul style="list-style-type: none"><li>● Use our company's genuine cables for connecting between the actuator and controller, and for the teaching tool.</li><li>● Do not scratch on the cable. Do not bend it forcibly. Do not pull it. Do not coil it around. Do not insert it. Do not put any heavy thing on it. Failure to do so may cause a fire, electric shock or malfunction due to leakage or continuity error.</li><li>● Perform the wiring for the product, after turning OFF the power to the unit, so that there is no wiring error.</li><li>● When the direct current power (+24V) is connected, take the great care of the directions of positive and negative poles. If the connection direction is not correct, it might cause a fire, product breakdown or malfunction.</li><li>● Connect the cable connector securely so that there is no disconnection or looseness. Failure to do so may cause a fire, electric shock or malfunction of the product.</li><li>● Never cut and/or reconnect the cables supplied with the product for the purpose of extending or shortening the cable length. Failure to do so may cause the product to malfunction or cause fire.</li></ul> <p>(3) Grounding</p> <ul style="list-style-type: none"><li>● The grounding operation should be performed to prevent an electric shock or electrostatic charge, enhance the noise-resistance ability and control the unnecessary electromagnetic radiation.</li><li>● For the ground terminal on the AC power cable of the controller and the grounding plate in the control panel, make sure to use a twisted pair cable with wire thickness <math>0.5\text{mm}^2</math> (AWG20 or equivalent) or more for grounding work. For security grounding, it is necessary to select an appropriate wire thickness suitable for the load. Perform wiring that satisfies the specifications (electrical equipment technical standards).</li><li>● Perform Class D Grounding (former Class 3 Grounding with ground resistance <math>100\Omega</math> or below).</li></ul> |



| No. | Operation Description  | Description   |
|-----|------------------------|---|
| 4   | Installation and Start | <p>(4) Safety Measures</p> <ul style="list-style-type: none"><li>● When the work is carried out with 2 or more persons, make it clear who is to be the leader and who to be the follower(s) and communicate well with each other to ensure the safety of the workers.</li><li>● When the product is under operation or in the ready mode, take the safety measures (such as the installation of safety and protection fence) so that nobody can enter the area within the robot's movable range. When the robot under operation is touched, it may result in death or serious injury.</li><li>● Make sure to install the emergency stop circuit so that the unit can be stopped immediately in an emergency during the unit operation.</li><li>● Take the safety measure not to start up the unit only with the power turning ON. Failure to do so may start up the machine suddenly and cause an injury or damage to the product.</li><li>● Take the safety measure not to start up the machine only with the emergency stop cancellation or recovery after the power failure. Failure to do so may result in an electric shock or injury due to unexpected power input.</li><li>● When the installation or adjustment operation is to be performed, give clear warnings such as "Under Operation; Do not turn ON the power!" etc. Sudden power input may cause an electric shock or injury.</li><li>● Take the measure so that the work part is not dropped in power failure or emergency stop.</li><li>● Wear protection gloves, goggle or safety shoes, as necessary, to secure safety.</li><li>● Do not insert a finger or object in the openings in the product. Failure to do so may cause an injury, electric shock, damage to the product or fire.</li><li>● When releasing the brake on a vertically oriented actuator, exercise precaution not to pinch your hand or damage the work parts with the actuator dropped by gravity.</li></ul> |
| 5   | Teaching               | <ul style="list-style-type: none"><li>● When the work is carried out with 2 or more persons, make it clear who is to be the leader and who to be the follower(s) and communicate well with each other to ensure the safety of the workers.</li><li>● Perform the teaching operation from outside the safety protection fence, if possible. In the case that the operation is to be performed unavoidably inside the safety protection fence, prepare the "Stipulations for the Operation" and make sure that all the workers acknowledge and understand them well.</li><li>● When the operation is to be performed inside the safety protection fence, the worker should have an emergency stop switch at hand with him so that the unit can be stopped any time in an emergency.</li><li>● When the operation is to be performed inside the safety protection fence, in addition to the workers, arrange a watchman so that the machine can be stopped any time in an emergency. Also, keep watch on the operation so that any third person can not operate the switches carelessly.</li><li>● Place a sign "Under Operation" at the position easy to see.</li><li>● When releasing the brake on a vertically oriented actuator, exercise precaution not to pinch your hand or damage the work parts with the actuator dropped by gravity.</li></ul> <p>* Safety protection Fence : In the case that there is no safety protection fence, the movable range should be indicated.</p>   |



| No. | Operation Description | Description   |
|-----|-----------------------|---|
| 6   | Trial Operation       | <ul style="list-style-type: none"><li>• When the work is carried out with 2 or more persons, make it clear who is to be the leader and who to be the follower(s) and communicate well with each other to ensure the safety of the workers.</li><li>• After the teaching or programming operation, perform the check operation one step by one step and then shift to the automatic operation.</li><li>• When the check operation is to be performed inside the safety protection fence, perform the check operation using the previously specified work procedure like the teaching operation.</li><li>• Make sure to perform the programmed operation check at the safety speed. Failure to do so may result in an accident due to unexpected motion caused by a program error, etc.</li><li>• Do not touch the terminal block or any of the various setting switches in the power ON mode. Failure to do so may result in an electric shock or malfunction.</li></ul> |
| 7   | Automatic Operation   | <ul style="list-style-type: none"><li>• Check before starting the automatic operation or rebooting after operation stop that there is nobody in the safety protection fence.</li><li>• Before starting automatic operation, make sure that all peripheral equipment is in an automatic-operation-ready state and there is no alarm indication.</li><li>• Make sure to operate automatic operation start from outside of the safety protection fence.</li><li>• In the case that there is any abnormal heating, smoke, offensive smell, or abnormal noise in the product, immediately stop the machine and turn OFF the power switch. Failure to do so may result in a fire or damage to the product.</li><li>• When a power failure occurs, turn OFF the power switch. Failure to do so may cause an injury or damage to the product, due to a sudden motion of the product in the recovery operation from the power failure.</li></ul>                                 |



| No. | Operation Description      | Description   |
|-----|----------------------------|---|
| 8   | Maintenance and Inspection | <ul style="list-style-type: none"><li>● When the work is carried out with 2 or more persons, make it clear who is to be the leader and who to be the follower(s) and communicate well with each other to ensure the safety of the workers.</li><li>● Perform the work out of the safety protection fence, if possible. In the case that the operation is to be performed unavoidably inside the safety protection fence, prepare the “Stipulations for the Operation” and make sure that all the workers acknowledge and understand them well.</li><li>● When the work is to be performed inside the safety protection fence, basically turn OFF the power switch.</li><li>● When the operation is to be performed inside the safety protection fence, the worker should have an emergency stop switch at hand with him so that the unit can be stopped any time in an emergency.</li><li>● When the operation is to be performed inside the safety protection fence, in addition to the workers, arrange a watchman so that the machine can be stopped any time in an emergency. Also, keep watch on the operation so that any third person can not operate the switches carelessly.</li><li>● Place a sign “Under Operation” at the position easy to see.</li><li>● For the grease for the guide or ball screw, use appropriate grease according to the Operation Manual for each model.</li><li>● Do not perform the dielectric strength test. Failure to do so may result in a damage to the product.</li><li>● When releasing the brake on a vertically oriented actuator, exercise precaution not to pinch your hand or damage the work parts with the actuator dropped by gravity.</li><li>● The slider or rod may get misaligned OFF the stop position if the servo is turned OFF. Be careful not to get injured or damaged due to an unnecessary operation.</li><li>● Pay attention not to lose the cover or untightened screws, and make sure to put the product back to the original condition after maintenance and inspection works.<br/>Use in incomplete condition may cause damage to the product or an injury.</li></ul> <p>* Safety protection Fence : In the case that there is no safety protection fence, the movable range should be indicated.</p> |
| 9   | Modification and Dismantle | <ul style="list-style-type: none"><li>● Do not modify, disassemble, assemble or use of maintenance parts not specified based at your own discretion.</li></ul>  |
| 10  | Disposal                   | <ul style="list-style-type: none"><li>● When the product becomes no longer usable or necessary, dispose of it properly as an industrial waste.</li><li>● When removing the actuator for disposal, pay attention to drop of components when detaching screws.</li><li>● Do not put the product in a fire when disposing of it.<br/>The product may burst or generate toxic gases.</li></ul>  |
| 11  | Other                      | <ul style="list-style-type: none"><li>● Do not come close to the product or the harnesses if you are a person who requires a support of medical devices such as a pacemaker. Doing so may affect the performance of your medical device.</li><li>● See Overseas Specifications Compliance Manual to check whether complies if necessary.</li><li>● For the handling of actuators and controllers, follow the dedicated operation manual of each unit to ensure the safety.</li></ul>  |



## Alert Indication

The safety precautions are divided into “Danger”, “Warning”, “Caution” and “Notice” according to the warning level, as follows, and described in the Operation Manual for each model.

| Level   | Degree of Danger and Damage   | Symbol  |
|---------|---|---|
| Danger  | This indicates an imminently hazardous situation which, if the product is not handled correctly, will result in death or serious injury.        |  Danger  |
| Warning | This indicates a potentially hazardous situation which, if the product is not handled correctly, could result in death or serious injury.       |  Warning |
| Caution | This indicates a potentially hazardous situation which, if the product is not handled correctly, may result in minor injury or property damage. |  Caution |
| Notice  | This indicates lower possibility for the injury, but should be kept to use this product properly.   |  Notice  |



## Caution in Handling

---

1. Do not set speeds and accelerations/decelerations equal to or greater than the respective ratings.

If the actuator is operated at a speed or acceleration/deceleration exceeding the allowable value, abnormal noise or vibration, failure, or shorter life may result.

In the case of interpolated operation of combined axes, the speed and acceleration/deceleration settings should correspond to the minimum values among all combined axes.

2. Keep the load moment within the allowable value.

If the actuator is operated under a load equal to or greater than the allowable load moment, abnormal noise or vibration, failure, or shorter life may result. In an extreme case, flaking may occur.

3. Make sure to attach the actuator properly by following this Operation manual.

Using the product with the actuator not being certainly retained or affixed may cause abnormal noise, vibration, malfunction or shorten the product life.



## Chapter 1 Installation

### 1.1 Introduction

Thank you for purchasing the Tabletop Robot.

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 Tabletop Robot is an all-in-one actuator that can be used as an independent standalone robot. The robot can also be used to control various peripheral equipment by utilizing the robot's built-in controller and supplied input/output (general-purpose input/output) connector. In general, connecting additional equipment will make the system larger and more complex, which often increases the probability of accident due to malfunction, carelessness, etc. If you are configuring the Tabletop Robot to operate with other equipment, please take sufficient safety measures.

- Actuator duty

Based on the relationship of service life and precision, keep the duty to 50% or less for all actuators of IAI as a guideline.

The duty is calculated by the formula below:

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

- 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.

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.



## 1.2 Models

## Example of Model Code

TT — A3 — I — 2020 — 05B — DV  
 [1] [2] [3] [4] [5] [6]

Model table

| [1] Series | [2] Type   | [3] Encoder type   | [4] XY stroke (mm) | [5] Z stroke                      | [6] Options  |
|------------|--|--------------------|--------------------|-----------------------------------|--|
| TT         | A2<br>(Gate 2-axis type)<br>C2<br>(Cantilever 2-axis type) | I<br>(Incremental) | 2020<br>(200 mm)   | 05B<br>(50 mm)<br>10B<br>(100 mm) | DV<br>(DeviceNet)<br>CC<br>(CC-Link)<br>PR<br>(ProfiBus)<br>ET<br>(Ethernet)<br>FT<br>(Mounting bracket)<br>P<br>(External I/O: PNP specification) |
|            | A3<br>(Gate 3-axis type)<br>C3<br>(Cantilever 3-axis type) |                    | 4040<br>(400 mm)   |                                   |  |

**⚠ Caution**

### 1.3 Safety Precautions

This system product was developed as a drive unit for an automated machine, etc., and as such the maximum torque and speed are limited to levels acceptable for an automatically driven machine. However, strict observance of the following items is requested to prevent unforeseen danger.

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. 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.
3. 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.
4. 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.



## 1.4 Warranty

### 1.4.1 Warranty Period

One of the following periods, whichever is shorter:

- 18 months after shipment from our company
- 12 months after delivery to the specified location

### 1.4.2 Scope of Warranty

Our products are covered by warranty when all of the following conditions are met. Faulty products covered by warranty will be replaced or repaired free of charge:

- (1) The breakdown or problem in question pertains to our product as delivered by us or our authorized dealer.
- (2) The breakdown or problem in question occurred during the warranty period.
- (3) The breakdown or problem in question occurred while the product was in use for an appropriate purpose under the conditions and environment of use specified in the operation manual and catalog.
- (4) The breakdown of problem in question was caused by a specification defect or problem, or by a quality issue with our product.

Note that breakdowns due to any of the following reasons are excluded from the scope of warranty:

- [1] Anything other than our product
- [2] Modification or repair performed by a party other than us (unless we have approved such modification or repair)
- [3] Anything that could not be easily predicted with the level of science and technology available at the time of shipment from our company
- [4] A natural disaster, man-made disaster, incident or accident for which we are not liable
- [5] Natural fading of paint or other symptoms of aging
- [6] Wear, depletion or other expected result of use
- [7] Operation noise, vibration or other subjective sensation not affecting function or maintenance

Note that the warranty only covers our product as delivered and that any secondary loss arising from a breakdown of our product is excluded from the scope of warranty.

### 1.4.3 Honoring the Warranty

As a rule, the product must be brought to us for repair under warranty.

### 1.4.4 Limited Liability

- (1) We shall assume no liability for any special damage, consequential loss or passive loss such as a loss of expected profit arising from or in connection with our product.
- (2) We shall not be liable for any program or control method created by the customer to operate our product or for the result of such program or control method.



#### 1.4.5 Conditions of Conformance with Applicable Standards/Regulations, Etc., and Applications

- (1) If our product is combined with another product or any system, device, etc., used by the customer, the customer must first check the applicable standards, regulations and/or rules. The customer is also responsible for confirming that such combination with our product conforms to the applicable standards, etc. In such a case we will not be liable for the conformance of our product with the applicable standards, etc.
- (2) Our product is for general industrial use. It is not intended or designed for the applications specified below, which require a high level of safety. Accordingly, as a rule our product cannot be used in these applications. Contact us if you must use our product for any of these applications:
  - [1] Medical equipment pertaining to maintenance or management of human life or health
  - [2] A mechanism or mechanical equipment intended to move or transport people (such as a vehicle, railway facility or aviation facility)
  - [3] Important safety parts of mechanical equipment (such as safety devices)
  - [4] Equipment used to handle cultural assets, art or other irreplaceable items
- (3) Contact us at the earliest opportunity if our product is to be used in any condition or environment that differs from what is specified in the catalog or operation manual.

#### 1.4.6 Other Items Excluded from Warranty

The price of the product delivered to you does not include expenses associated with programming, the dispatch of engineers, etc. Accordingly, a separate fee will be charged in the following cases even during the warranty period:

- [1] Guidance for installation/adjustment and witnessing of test operation
- [2] Maintenance and inspection
- [3] Technical guidance and education on operating/wiring methods, etc.
- [4] Technical guidance and education on programming and other items related to programs



## 2. Specifications

### 2.1 Basic Specifications

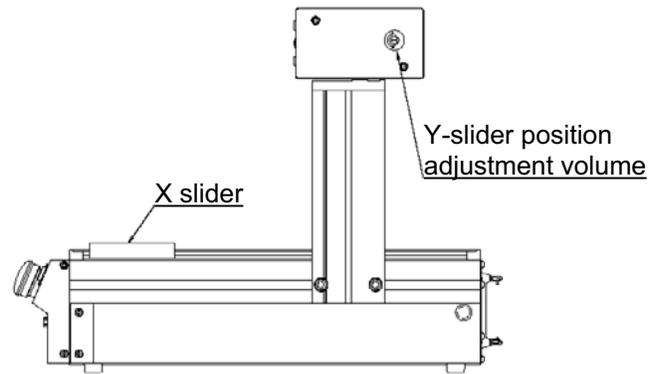
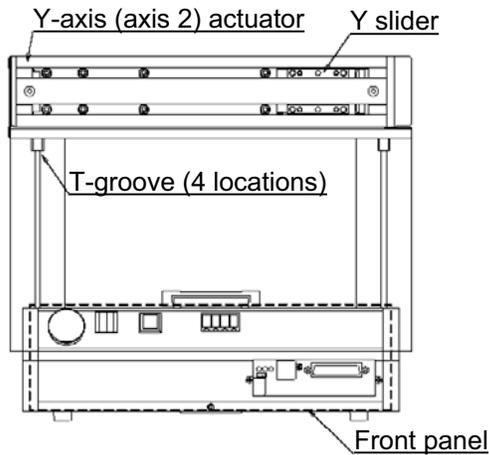
| Item                                  | Description  |
|---------------------------------------|--|
| Number of controlled axes             | Maximum 3 axes (Factory setting)   |
| Power-source voltage                  | Single-phase, 100 to 230 VAC $\pm$ 10%   |
| Power-source frequency                | 50 Hz/60 Hz  |
| Withstand voltage                     | 2000 V 1 minute  |
| Rush current                          | 15 A (100 VAC), 30 A (200 VAC)   |
| Leak current                          | 0.75 mA (60 Hz)  |
| Resistance to momentary power failure | Max 500 $\mu$ s  |
| Electric-shock protection mechanism   | Class 1 basic isolation  |
| Surrounding air temperature range     | 5°C to 40°C  |
| Surrounding humidity range            | 20% to 90% (Non-condensing)  |
| Storage temperature range             | -10°C to 65°C  |
| Maximum speed                         | 300 mm/sec   |
| Rated acceleration                    | Gate type: 0.3G, Cantilever type: 0.2G   |
| Programming language                  | Super SEL language   |
| Program steps                         | 6000 steps (total)   |
| Number of positions                   | 3000 positions   |
| Number of programs                    | 64 programs  |
| Multi-tasking                         | 16 programs  |
| Standard inputs                       | 16 points (General-purpose inputs, port Nos. 016 to 031)   |
| Standard outputs                      | 16 points (General-purpose outputs, port Nos. 316 to 331)  |
| Dedicated inputs                      | Digital switch for program number input<br>Function switch, etc.   |
| Dedicated outputs                     | Alarm status indicator LED<br>Ready status indicator LED<br>Emergency-stop status indicator LED<br>Home-return completion status indicator LED, etc. |
| Serial communication                  | For teaching pendant/PC connection   |
| Supported Fieldbus standards          | CC-LINK DeviceNet Profibus<br>ModBus/TCP Ethernet  |

Note: The parameters are normally set to the above general-purpose input and general-purpose output port numbers before shipment.

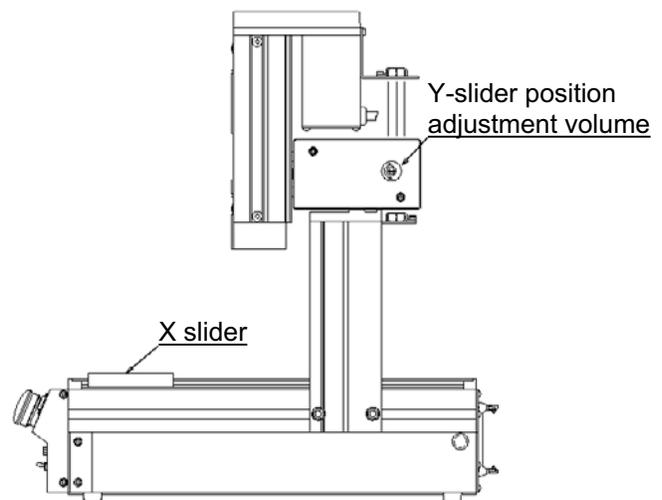
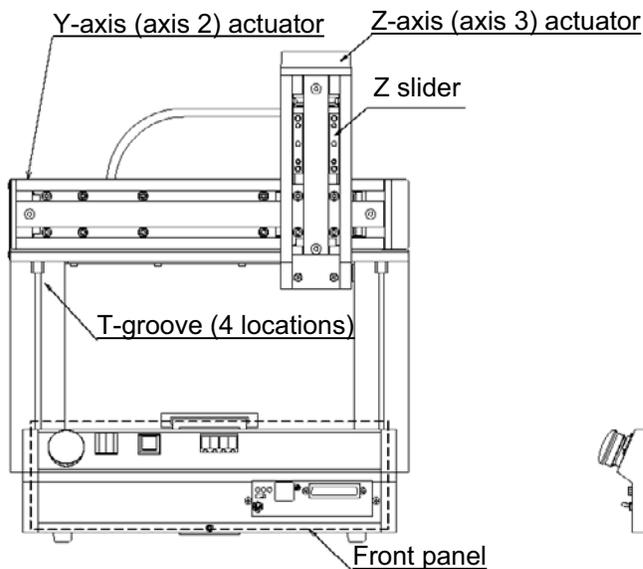
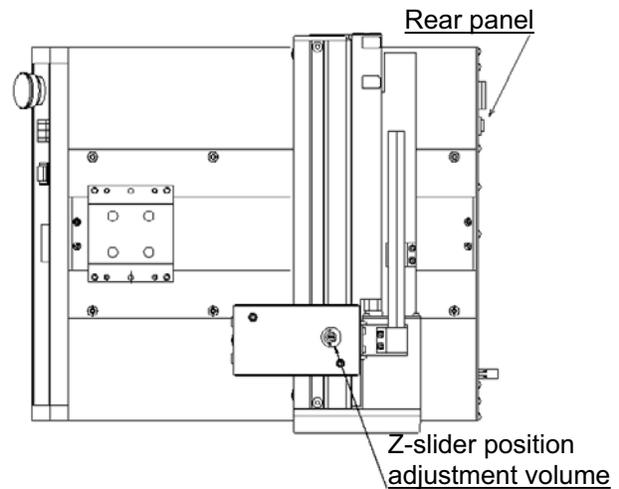
## 2.2 Name and Function of Each Part

### 2.2.1 Robot Body

#### Gate 2-axis type

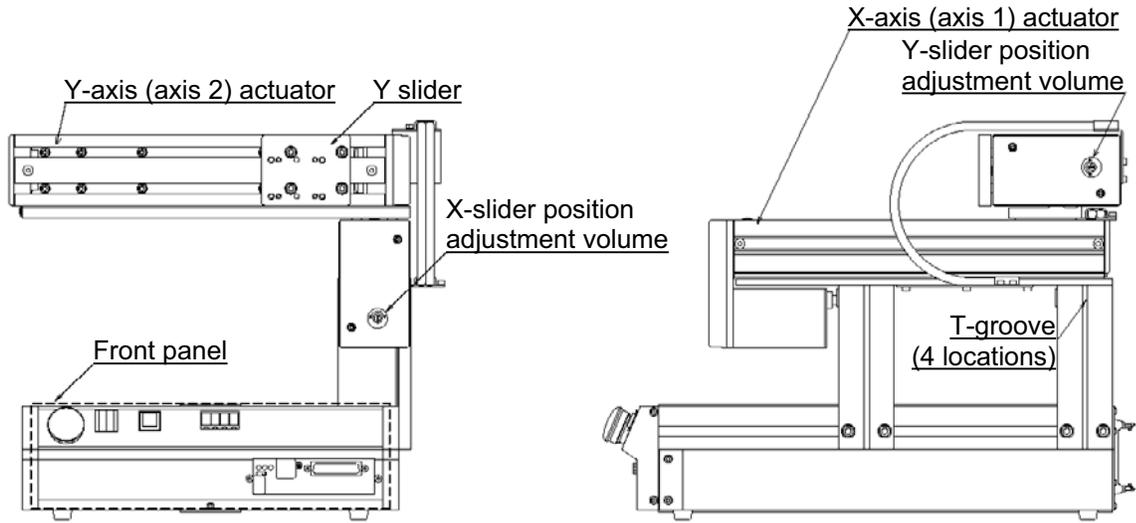


#### Gate 3-axis type

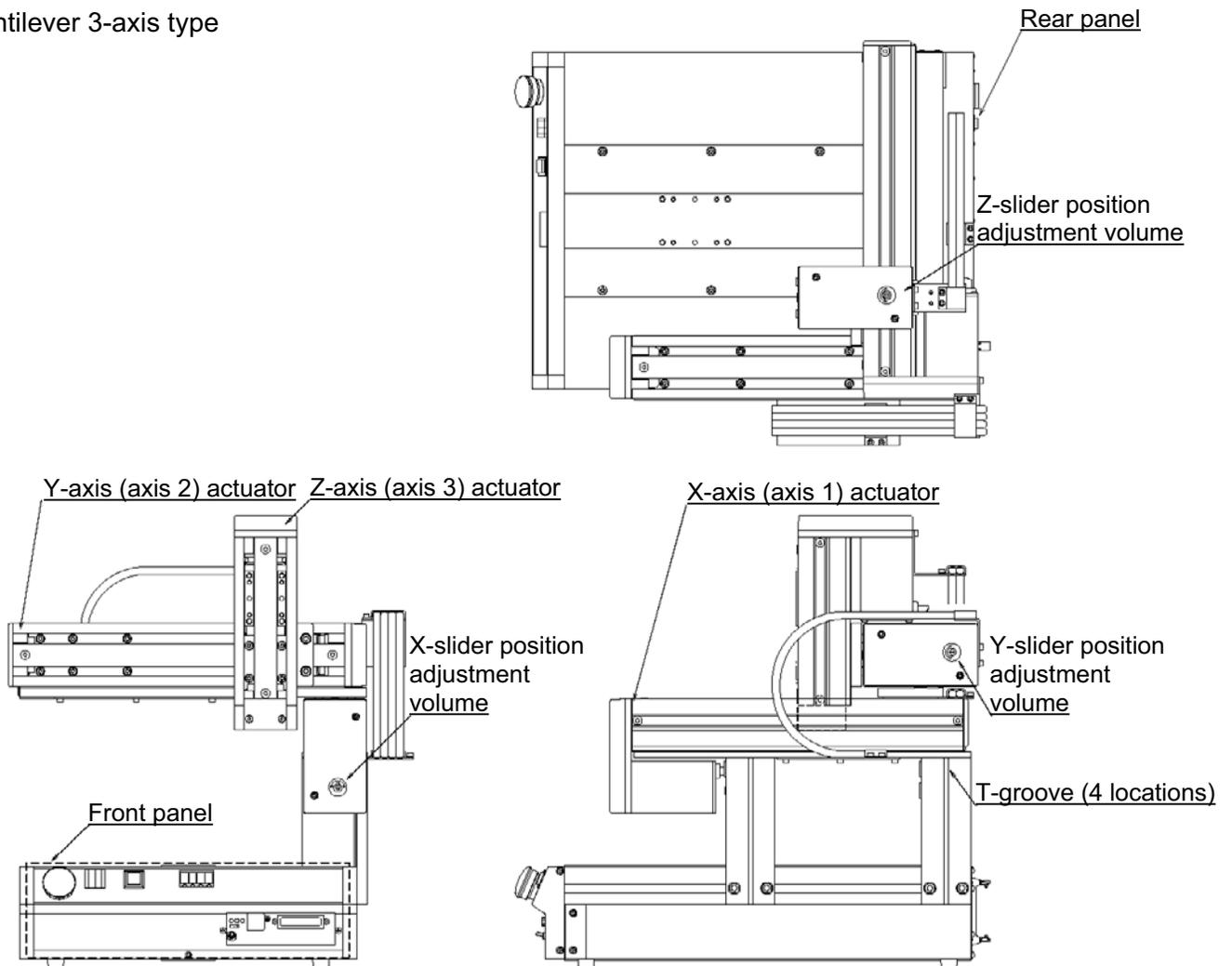




Cantilever 2-axis type



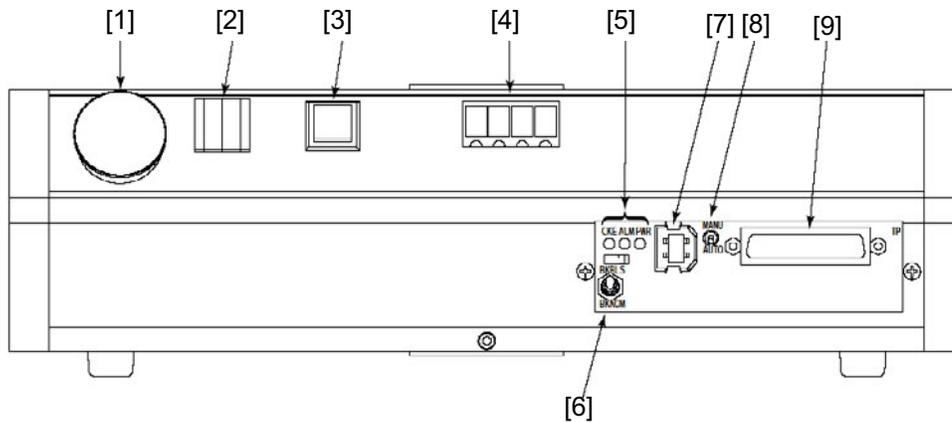
Cantilever 3-axis type





- X-axis actuator  
Various loads can be attached to the X-axis actuator of the gate-type robot.
  - Y-axis actuator  
Various tools can be attached to the Y-axis actuator of the 2-axis robot.
  - Z-axis actuator  
Various tools can be attached to the Z-axis actuator of the 3-axis robot.
  - T-groove  
Auxiliary tools can be attached by utilizing the T-grooves/T-slots and nuts.
  - Position adjustment volume  
You can fine-tune the slider position easily by turning this volume with a flathead screwdriver, etc. This function is useful when manually adjusting the slider position to read position data. This adjustment volume is provided on various actuators.
- (Note) Before fine-tuning the slider position using this volume, be sure to actuate an emergency stop.  
Do not insert an adjustment tool, finger or other object into the operating range of the robot while the servo is ON or slider is operating.

## 2.2.2 Front Panel



- [1] Emergency button (emergency-stop button)  
This switch is used to cut off the drive power when the robot must be stopped in case of emergency.
- [2] Digital program selector switch  
This switch provides a 2-digit decimal digital switch input for selecting the program you want to start from among the group of programs stored in the Tabletop Robot. Pressing the start switch [3] will start the selected program.
- [3] Start switch (function switch)  
This switch issues a trigger to start the program set by the digital program selector switch [2].  
(Factory setting)  
This switch is enabled in the AUTO mode.  
In the MANU mode, this switch is enabled after the teaching pendant or PC software has been connected online. (Once the teaching pendant or PC software is connected online, the switch will remain enabled until the robot is restarted (via software reset), even after the connection is switched offline.)  
(This switch turns ON/OFF input port No. 000. Since I/O parameter No. 30 is set to "1" at the factory, input port No. 000 is used as the program start signal (dedicated input). You can use input port No. 000 as a general-purpose input by setting I/O parameter No. 30 to "0.")

Note: The parameters are normally set to the above input port numbers before shipment.

**[4] Panel window**

The panel window consists of a 4-digit, 7-segment LED display and four LED lamps indicating the status of the robot.

The status indicated by each LED lamp when the lamp is lit is explained below:

RDY: The robot is ready to perform PIO program operation.  
(This lamp is connected to dedicated output No. 301.)

ALM: An error of operation-cancellation level or higher has occurred.  
(This lamp is connected to dedicated output No. 300.)

EMG: An emergency stop has been actuated.  
(This lamp is connected to dedicated output No. 302.)

HPS: All axes have completed their home return.  
(This lamp is connected to dedicated output No. 304.)

For the specific codes shown on the 4-digit, 7-segment LED display, refer to 2.2.3, "Codes Displayed on the Panel Window" or the "Error Code Table."

Note: The parameters are normally set to the above output port numbers before shipment.

**[5] LED indicator lamps**

The status indicated by each LED lamp when the lamp is lit is explained below:

CKE: System clock error

ALM: CPU alarm (system-down level error)

PWR: Power ON

**[6] Brake switch**

This switch is used to release the axis brake.

Tilt the switch upward (BKRLS side) to forcibly release the brake, or tilt it downward (BKNOM side) to allow the brake to be controlled automatically by the controller. Set this switch to the BKNOM side in normal conditions of use.

**[7] USB connector**

This connector is used for USB connection. Use it to connect the PC software to the controller via USB cable.

Applicable connector: USB connector B: XM7B-0442

Mating connector: USB cable

**Notes**

- If the USB port is used, all TT robots comprising the system must be connected one by one to install the USB driver included in the CD-ROM "X-SEL PC Software IA-101-TT-USB." For details on how to install the driver, refer to the operation manual for X-SEL PC software.
- If the USB port is used, a dummy plug must be connected to the teaching connector [9].  
Dummy plug model: DP-1

**[8] Mode switch**

This switch is used to specify the operation mode of the Tabletop Robot.

Tilt the switch upward to select the MANU mode (manual mode), or tilt it downward to select the AUTO mode (automatic mode).

Operations from the teaching pendant or PC software (such as teaching) must be performed in the MANU mode. (They cannot be performed in the AUTO mode.)

Auto program start is enabled in the AUTO mode. (The function cannot be used in the MANU mode.)



## [9] Teaching 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  |



2.2.3 Codes Displayed on the Panel Window

(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 E   | 3             | Emergency stop is being actuated (except during the update mode).                        |
| a P E   | 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 E 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 E | 7             | All servo axes are interlocked (except during the update mode).                          |
| E A X X | 8             | Message level error  |
| E A X X | 8             | Message level error  |
| r U d E | 9             | Core update mode   |
| U d E   | 9             | Core update is in progress.  |
| F U d E | 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 E   | 9             | Debug mode   |
| P 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.



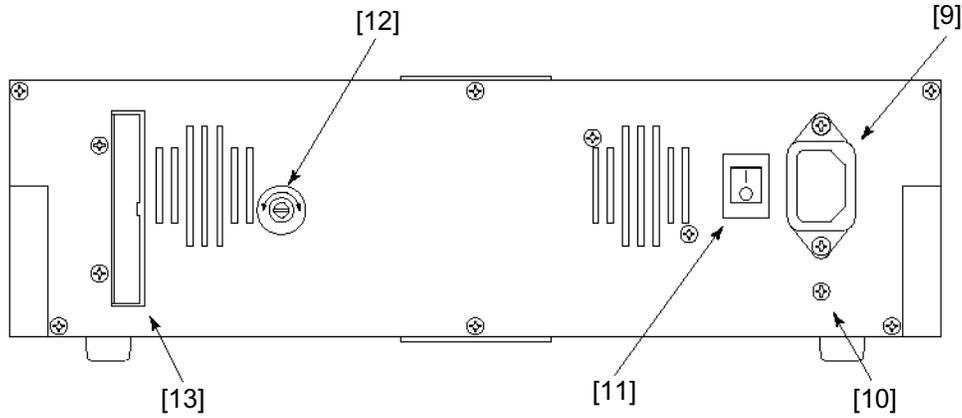
## (2) Core

| 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.   |
| E       | U | d | A | 2             | Application update has completed.  |
| E       | - | - | - | 2             | Hardware test mode process   |
|         | E | r | A | 2             | Clearing the application flash ROM.  |
| E       | E | r | A | 2             | Application flash ROM has been cleared.  |
|         | J | P | A | 2             | Jump to the application  |
| E       | H | F | E | 2             | Core flash-ROM check process   |
| E       | H | F | A | 2             | Application flash-ROM check process  |
| E       | H | S | d | 2             | SDRAM check process  |

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



### 2.2.4 Rear Panel



#### [9] Power connector

Connect the power cable to this connector.

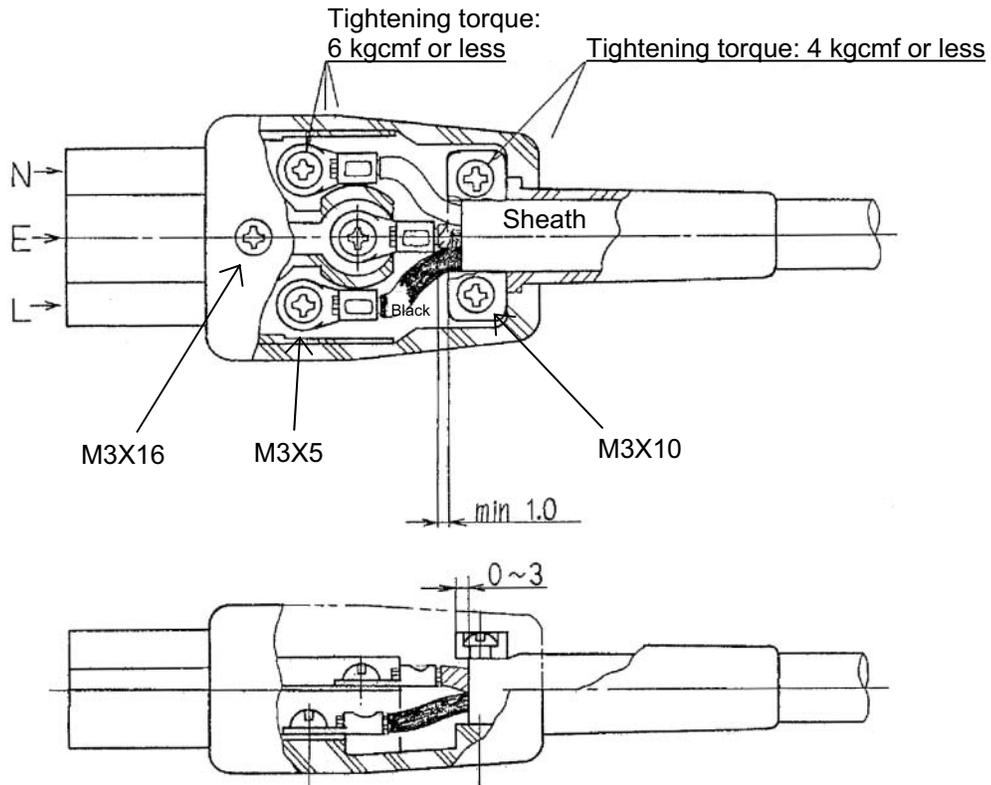
Use the supplied socket for cable connection with the power connector.

(Note) The allowable range of power-source voltage is 100 to 230 VAC ( $\pm 10\%$ ).

Providing a power cable and attaching it to the supplied socket is the user's responsibility.

Use a cable appropriate for the power-source voltage used.

#### How to attach a cable to the supplied socket



**[10] Ground terminal**

This terminal is used to connect FG of the enclosure to ground.

**[11] Power switch****[12] Gate X-axis actuator position adjustment volume (This volume is not available on the cantilever type.)**

You can fine-tune the X-axis slider position easily by turning this volume with a flathead screwdriver, etc. This function is useful when manually adjusting the slider position to read position data.

(Note) Before fine-tuning the slider position using this volume, be sure to actuate an emergency stop. Do not insert an adjustment tool, finger or other object into the operating range of the robot while the servo is ON or slider is operating.

**[13] I/O connector (general-purpose I/Os)**

This general-purpose I/O connector is used to connect peripheral equipment, etc.

It is a 34-pin flat connector that comprises 16 general-purpose input/16 general-purpose output DIOs.

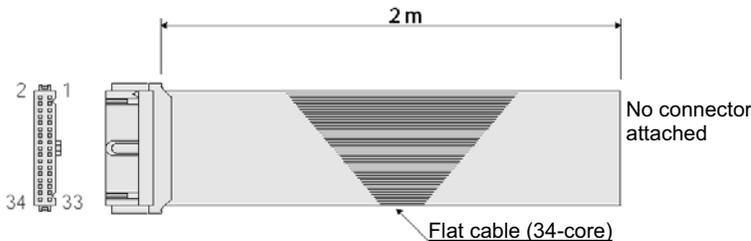


### 2.2.5 I/O Connector Pin Assignments

| Pin No. | Category | Port No. | Function                | Cable color            |
|---------|----------|----------|-------------------------|------------------------|
| 1       | +24 V    |          | I/O power supply + 24 V | Brown-1                |
| 2       | Input    | 016      | General-purpose input   | Red-1                  |
| 3       |          | 017      | General-purpose input   | Orange-1               |
| 4       |          | 018      | General-purpose input   | Yellow-1               |
| 5       |          | 019      | General-purpose input   | Green-1                |
| 6       |          | 020      | General-purpose input   | Blue-1                 |
| 7       |          | 021      | General-purpose input   | Purple-1               |
| 8       |          | 022      | General-purpose input   | Gray-1                 |
| 9       |          | 023      | General-purpose input   | White-1                |
| 10      |          | 024      | General-purpose input   | Black-1                |
| 11      |          | 025      | General-purpose input   | Brown-2                |
| 12      |          | 026      | General-purpose input   | Red-2                  |
| 13      |          | 027      | General-purpose input   | Orange-2               |
| 14      |          | 028      | General-purpose input   | Yellow-2               |
| 15      |          | 029      | General-purpose input   | Green-2                |
| 16      |          | 030      | General-purpose input   | Blue-2                 |
| 17      |          | 031      | General-purpose input   | Purple-2               |
| 18      |          | Output   | 316                     | General-purpose output |
| 19      | 317      |          | General-purpose output  | White-2                |
| 20      | 318      |          | General-purpose output  | Black-2                |
| 21      | 319      |          | General-purpose output  | Brown-3                |
| 22      | 320      |          | General-purpose output  | Red-3                  |
| 23      | 321      |          | General-purpose output  | Orange-3               |
| 24      | 322      |          | General-purpose output  | Yellow-3               |
| 25      | 323      |          | General-purpose output  | Green-3                |
| 26      | 324      |          | General-purpose output  | Blue-3                 |
| 27      | 325      |          | General-purpose output  | Purple-3               |
| 28      | 326      |          | General-purpose output  | Gray-3                 |
| 29      | 327      |          | General-purpose output  | White-3                |
| 30      | 328      |          | General-purpose output  | Black-3                |
| 31      | 329      |          | General-purpose output  | Brown-4                |
| 32      | 330      |          | General-purpose output  | Red-4                  |
| 33      | 331      |          | General-purpose output  | Orange-4               |
| 34      | 0 V      |          |                         | I/O power supply 0 V   |

Note: The parameters are normally set to the above port numbers before shipment.

#### I/O flat cable (supplied) Model: CB-DS-PIO020



| No. | Color   | Wiring                      | No. | Color   | Wiring                      |
|-----|---------|-----------------------------|-----|---------|-----------------------------|
| 1   | Brown1  | Flat cable, pressure welded | 18  | Gray2   | Flat cable, pressure welded |
| 2   | Red1    |                             | 19  | White2  |                             |
| 3   | Orange1 |                             | 20  | Black2  |                             |
| 4   | Yellow1 |                             | 21  | Brown-3 |                             |
| 5   | Green1  |                             | 22  | Red3    |                             |
| 6   | Blue1   |                             | 23  | Orange3 |                             |
| 7   | Purple1 |                             | 24  | Yellow3 |                             |
| 8   | Gray1   |                             | 25  | Green3  |                             |
| 9   | White1  |                             | 26  | Blue3   |                             |
| 10  | Black1  |                             | 27  | Purple3 |                             |
| 11  | Brown-2 |                             | 28  | Gray3   |                             |
| 12  | Red2    |                             | 29  | White3  |                             |
| 13  | Orange2 |                             | 30  | Black3  |                             |
| 14  | Yellow2 |                             | 31  | Brown-4 |                             |
| 15  | Green2  |                             | 32  | Red4    |                             |
| 16  | Blue2   |                             | 33  | Orange4 |                             |
| 17  | Purple2 |                             | 34  | Yellow4 |                             |



## 2.3 Interfaces of the Tabletop Robot

### 2.3.1 Standard Interface (Main Application Version 0.18 or Earlier)

The standard interface of the Tabletop Robot uses input port Nos. 000 to 047 and output port Nos. 300 to 347.

The standard interface is subject to limitations on use.

Only input port Nos. 016 to 031 and output port Nos. 316 to 331 can send/receive signals to/from peripheral equipment via the I/O connector on the rear panel as external DIOs.

Other ports are used as internal DIOs, dedicated ports for switches/LEDs on the front panel or ports used by SEL programs, or reserved for future expansion.

[Internal DI, Internal DO]

- Internal DI No. 000 is an input port connected from the start switch on the front panel.
- Internal DI Nos. 007 to 010 and Nos. 011 to 013 are input ports connected from the digital switch on the front panel.
- Although dedicated functions can be assigned to internal DI Nos. 001 to 006, 014 and 015, these ports cannot be controlled directly. To control internal DI Nos. 001 to 006, 014 and 015, turn ON/OFF internal DO Nos. 308 to 315 in a SEL program. For details, refer to © Appendix, "How to Use Internal DIOs."
- Internal DI No. 300 to 304 is an output port to the panel window LED and start switch LED from the front panel.
- Internal DO Nos. 332 to 346 are used to control the 7-segment LED display in the panel window on the front panel.

System information and user program data can be shown alternately on the 7-segment LED display by using a SEL program. For details, refer to © Appendix, "How to Use Internal DIOs."



Internal DIO Table

|             | Port No.   | Function   |                            | Port No.   | Function   |
|-------------|------------|--|----------------------------|------------|--|
| Internal DI | 000        | Start  | Internal DO                | 300        | ALM (LED on the front panel)                             |
|             | 001        | (Software reset)   |                            | 301        | RDY (LED on the front panel)                             |
|             | 002        | (Servo ON)   |                            | 302        | EMG (LED on the front panel)                             |
|             | 003        | (Auto program start)   |                            | 303        | Automatic operation mode (start switch LED)              |
|             | 004        | (Software interlock)   |                            | 304        | HPS (LED on the front panel)                             |
|             | 005        | (Pause reset)  |                            | 305        | For future expansion                                     |
|             | 006        | (Pause)  |                            | 306        | For future expansion                                     |
|             | 007        | Program number specification<br>Ones place of the digital switch |                            | 307        | For future expansion                                     |
|             | 008        |  |                            | 308        | Internal DI No. 001 ON/OFF                               |
|             | 009        |  |                            | 309        | Internal DI No. 002 ON/OFF                               |
|             | 010        | Program number specification<br>Tens place of the digital switch |                            | 310        | Internal DI No. 003 ON/OFF                               |
|             | 011        |  |                            | 311        | Internal DI No. 004 ON/OFF                               |
|             | 012        |  |                            | 312        | Internal DI No. 005 ON/OFF                               |
|             | 013        | (Drive-source cutoff input)                                      |                            | 313        | Internal DI No. 006 ON/OFF                               |
|             | 014        |  |                            | 314        | Internal DI No. 014 ON/OFF                               |
| 015         | 315        |  | Internal DI No. 015 ON/OFF |            |  |
| External DI | 016 to 031 | General-purpose input (I/O connector on the rear panel)          | External DO                | 316 to 331 | General-purpose output (I/O connector on the rear panel) |
| Internal DI | 032        | For future expansion   | Internal DO                | 332        | 7-segment user display digit specification               |
|             | 033        |  |                            | 333        | 7-segment user display digit specification               |
|             | 034        |  |                            | 334        | For future expansion                                     |
|             | 035        |  |                            | 335        | For future expansion                                     |
|             | 036        |  |                            | 336        | For future expansion                                     |
|             | 037        |  |                            | 337        | 7-segment display refresh                                |
|             | 038        |  |                            | 338        | 7-segment user/system alternate display                  |
|             | 039        |  |                            | 339        | 7-segment user display specification                     |
|             | 040        |  |                            | 340        | DT0 (7-segment user display bit)                         |
|             | 041        |  |                            | 341        | DT1 (7-segment user display bit)                         |
|             | 042        |  |                            | 342        | DT2 (7-segment user display bit)                         |
|             | 043        |  |                            | 343        | DT3 (7-segment user display bit)                         |
|             | 044        |  |                            | 344        | DT4 (7-segment user display bit)                         |
|             | 045        |  |                            | 345        | DT5 (7-segment user display bit)                         |
|             | 046        |  |                            | 346        | DT6 (7-segment user display bit)                         |
| 047         | 347        | For future expansion   |                            |            |  |
| External DI | 048 to 299 | Used for field network (Optional)                                | External DO                | 348 to 599 | Used for field network (Optional)                        |



### 2.3.2 Standard Interface (Main Application Version 0.19 or Later)

The input port to which to assign the input function selection from 000 to 015 currently set by “Input function selection \*\*\*” can be set (changed) using the I/O parameter “Physical input port number for input function selection \*\*\*.”

The output port to which to assign the output function selection from 300 to 315 currently set by “Output function selection \*\*\*” can be set (changed) using the I/O parameter “Physical output port number for output function selection \*\*\*.”

In addition to output function selections 300 to 315 described above, you can also use the I/O parameter “Physical output port number for output function selection \*\*\* (area 2)” to set (assign) an output port for the output function selection from 300 (area 2) to 315 (area 2) currently set by “Output function selection \*\*\* (area 2),” and output the applicable signal from the specified port.

Note: The above functions are supported by the X SEL PC software of version 7.0.2.0 or later.

#### (1) Assignment example of input function selection

The following is an example of assigning input function selection 000 (start), currently set by “Input function selection 000,” to a different input port.

Set the function of input function selection 000 (start) using I/O parameter No. 30, “Input function selection 000.” For details, refer to Appendix, “Parameter List.”

The physical input port number for input function selection 000 (start) is set by I/O parameter No. 283, “Physical input port number for input function selection 000.”

If “016” is set in this parameter, for example, the function of input function selection 000 (start) is assigned to “input port No. 016.”

Accordingly, input port 016 becomes the signal input port for input function selection 000 (start). After the assignment has been changed, “input port No. 000” returns to a general-purpose input port.

Note, however, that the above parameter will become invalid when “-1 (default value: normally the parameter is set to this value before shipment)” is set, in which case the function of input function selection 000 (start) will be assigned to “input port No. 000” as shown in the internal DIO table.

#### [Notes]

- If input function selection 000 (start) is assigned to a different input port, the start switch on the front panel will no longer function as the “program start signal.”
- If any of input function selections 007 to 013 is assigned to a different input port, the digital program selector switch on the front panel will no longer function as the “start program number.”
- If a network is available, input function selections 000 to 015 can also be assigned to port Nos. 048 to 299 assigned to the network.

Note: Although ports of desired output numbers can be set separately, error No. 685, “I/O function selection physical port number error” will generate if duplicate port numbers are set or the “start program number” is set to a non-continuous port.



(2) Assignment example of output function selection

The following is an example of assigning output function selection 300 (ALM), currently set by “Output function selection 300,” to a different output port.

Set the function of output function selection 300 using I/O parameter No. 46, “Output function selection 300.” For details, refer to Appendix, “Parameter List.”

The physical output port number for output function selection 300 (ALM) is set by I/O parameter No. 299, “Physical output port number for output function selection 300.”

If “316” is set in this parameter, for example, the function of output function selection 300 (ALM) is assigned to “output port No. 316.”

Accordingly, the signal of output function selection 300 (ALM) is output to output port 316.

Note, however, that the above parameter will become invalid when “0 (default value: normally the parameter is set to this value before shipment)” is set, in which case the function of output function selection 300 (ALM) will be assigned to “output port No. 300” as shown in the internal DIO table.

After the assignment has been changed, “output port No. 300” returns to a general-purpose output port.

- \* To output system outputs to an external device, it is recommended that the signals be output separately using “Output function selection \*\*\* (area 2)” and “Physical output port number for output function selection \*\*\* (area 2)” explained later.

[Notes]

- If output function selection 300 (ALM) is assigned to a different output port, the panel window LED “ALM” on the front panel will no longer function. As a result, this LED will not illuminate even when the ALM signal is output.
- If output function selection 301 (RDY) is assigned to a different output port, the panel window LED “RDY” on the front panel will no longer function. As a result, this LED will not illuminate even after the controller becomes ready (ready to perform PIO program operation).
- If output function selection 302 (EMG) is assigned to a different output port, the panel window LED “EMG” on the front panel will no longer function. As a result, this LED will not illuminate even when the EMG signal is output (emergency stop is actuated).
- If output function selection 303 (start switch) is assigned to a different output port, the start switch LED on the front panel will no longer function. As a result, this LED will not illuminate even during continuous operation.
- If output function selection 304 (HSP) is assigned to a different output port, the panel window LED “HSP” on the front panel will no longer function. As a result, this LED will not illuminate even when the HSP signal is output (all valid axes completed home return).
- Even when the input port number assigned to a given input function selection \*\*\* is changed by setting “Physical input port number for input function selection \*\*\*” accordingly, the functions where the ON/OFF statuses of output port Nos. 308 to 315 are reflected in input port Nos. 1 to 6, 14 and 15 will be maintained, as shown in the internal DIO table. For example, setting “Input function selection 001” and “Physical input port number for input function selection 001” to “1” (soft reset) and “16,” respectively, and then turning output port No. 308 ON will turn input port No. 1 ON, but soft reset will not be executed.

Note: Although ports of desired output numbers can be set separately, error No. 685, “I/O function selection physical port number error” will generate if duplicate port numbers are set.



(3) Assignment example of output function selection (area 2)

Output function selection 300 (area 2) (ALM), currently set by “Output function selection 300 (area 2),” can be assigned to the output port set by “Physical output port number for output function selection 300 (area 2)” to output the applicable signal from this port. An example is given below.

Set the function of output function selection 300 (area 2) using I/O parameter No. 331, “Output function selection 300 (area 2).” For details, refer to Appendix, “Parameter List.”

The physical output port number for output function selection 300 (area 2) (ALM) is set by I/O parameter No. 315, “Physical output port number for output function selection 300 (area 2).”

If “316” is set in this parameter, for example, the function of output function selection 300 (area 2) (ALM) is assigned to “output port No. 316.”

Accordingly, the signal of output function selection 300 (area 2) (ALM) is output to output port 316. Note, however, that the above parameter will become invalid when “0 (default value: normally the parameter is set to this value before shipment)” is set, in which case the signal will not be output.

Based on the above setting, the ALM signal can now be output to a different port (output port set for area 2) without disabling the ALM LED on the front panel (without changing the setting of “Physical input port number for output function selection 300” for output signal selection 300).

Note: Although ports of desired output numbers can be set separately, error No. 685, “I/O function selection physical port number error” will generate if duplicate port numbers are set.



## (4) Use example

The following is a setting example of assigning the system IOs to external DIOs as shown below when the external DIOs are assigned to input port Nos. 16 to 31 and output port Nos. 316 to 331 (default settings: the external DIOs are normally assigned to these ports before shipment). Settings that allow the panel window LEDs (RDY, ALM, EMG, HPS) to continue functioning normally are explained.

Input port No. 16 = Program start number (ON edge) (specified by BCD)

Input port No. 17 = Servo ON signal

Input port Nos. 18 to 23 = Start program number

Input port No. 24 = Error reset (ON edge)

Input port No. 25 = Home return of all valid axes (ON edge)

Output port No. 316 = Error of operation-cancellation level or higher (ON)

Output port No. 317 = READY output (PIO trigger program operation permitted)

Output port No. 318 = Emergency stop output (ON)

Output port No. 319 = Output during automatic operation

Output port No. 320 = Output if all valid axes completed home return (coordinates have been confirmed)

Output port Nos. 321 to 323 = Output when axis 1 to 3 servos are ON

## I/O Parameter Settings

| No.        | Parameter name   | Settings   | Remarks  |
|------------|--|------------|--|
| 30         | Input function selection 000   | 1          | 1 (default value) = Program start signal (ON edge) (specified by BCD)                |
| 32         | Input function selection 002   | 1          | 1 = Servo ON   |
| 37 to 42   | Input function selection 007 to 012  | 1          | 1 (default value) = Start program number   |
| 43         | Input function selection 013   | 2          | 2 = Error reset (ON edge)  |
| 45         | Input function selection 015   | 1          | 1 = Home return of all valid axes (ON edge)  |
| 283        | Physical input port number for input function selection 000  | 16         | Input port number = 16   |
| 285        | Physical input port number for input function selection 002  | 17         | Input port number = 17   |
| 290 to 295 | Physical input port number for input function selection 007 to 012   | 18 to 23   | Input port number = 18 to 23   |
| 296        | Physical input port number for input function selection 013  | 24         | Input port number = 24   |
| 298        | Physical input port number for input function selection 015  | 25         | Input port number = 25   |
| 315 to 330 | Physical input port number for output function selection 300 (area 2) to physical input port number for output function selection 307 (area 2) | 316 to 323 | Output port numbers = 316 to 323   |
| 331        | Output function selection 300 (area 2)   | 1          | 1 = Error of operation-cancellation level or higher (ON)                             |
| 332        | Output function selection 301 (area 2)   | 1          | 1 = READY output (PIO trigger program operation permitted)                           |
| 333        | Output function selection 302 (area 2)   | 1          | 1 = Emergency stop output (ON)   |
| 334        | Output function selection 303 (area 2)   | 2          | 2 = Output during automatic operation (Other parameter No. 12)                       |
| 335        | Output function selection 304 (area 2)   | 2          | 2 = Output if all valid axes completed home return (coordinates have been confirmed) |
| 336 to 338 | Output function selection 305 (area 2) to output function selection 307 (area 2)   | 2          | 2 = Output when axis 1 to 3 servos are ON (system monitor task output)               |

## [Notes]

- If input function selection 000 (start) is assigned to a different input port, the start switch on the front panel will no longer function as the “program start signal.”
- If any of input function selections 007 to 013 (digital switches) is assigned to a different input port, the digital program selector switch on the front panel will no longer function as the “start program number.”

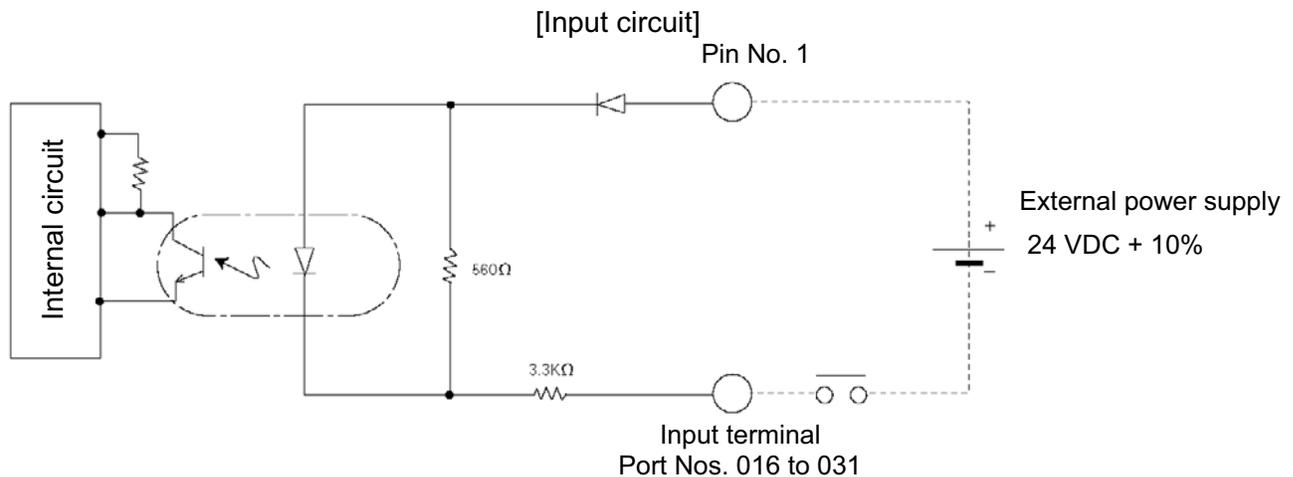
## 2.4 External I/O Specifications

### 2.4.1 NPN Specification

#### (1) Input part

#### External Input Specifications (NPN Specification)

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



Note: The parameters are normally set to the above port numbers before shipment.

#### ⚠ 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.

#### © Input signals to the Tabletop Robot



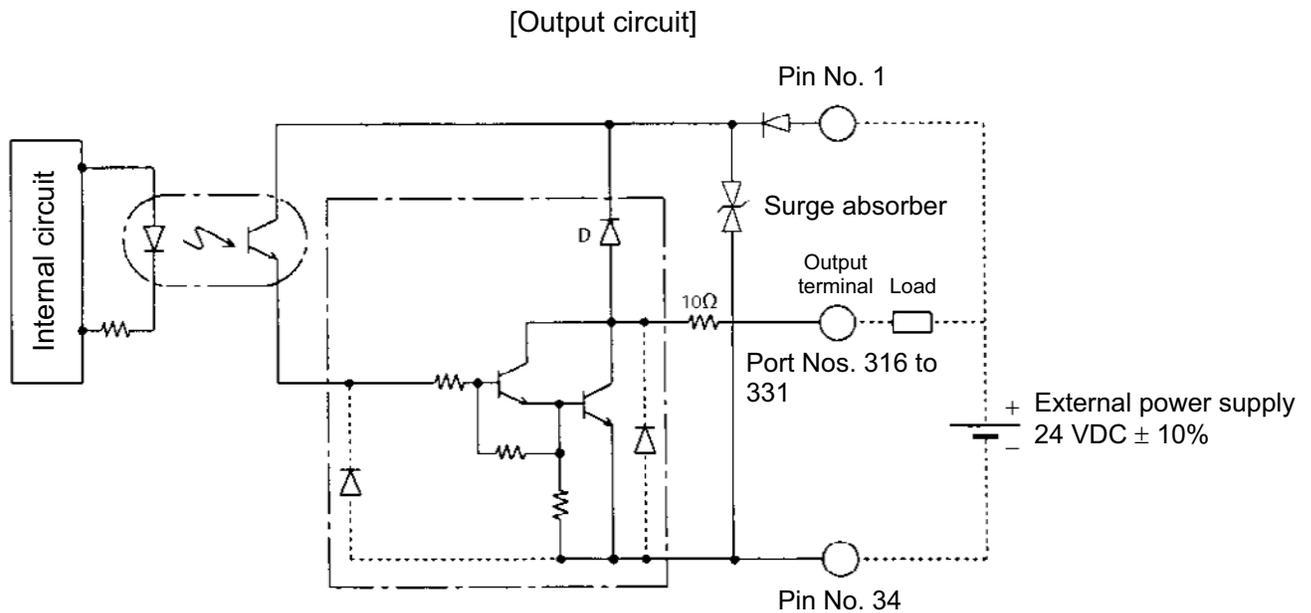
At the default settings, the system recognizes the ON/OFF durations of input signals if they are approx. 4 msec or longer. The ON/OFF duration settings can also be changed using I/O parameter No. 20 (input filtering frequency).

## (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                      |                         |
| Isolation method     | Photocoupler isolation                     |                         |
| 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. 316.



Note: The parameters are normally set to the above port numbers before shipment.

**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.

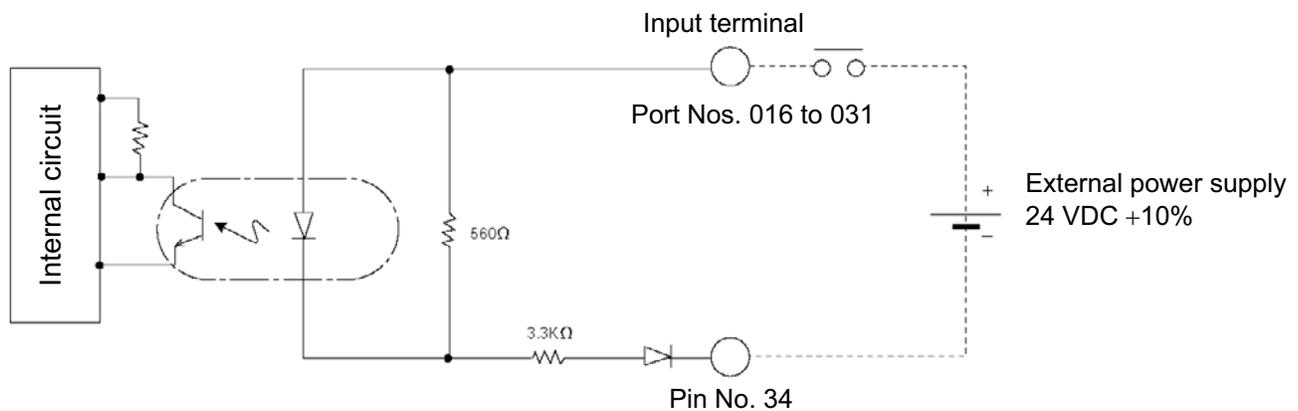
## 2.4.2 PNP Specification

## (1) Input part

External Input Specifications (PNP Specification)

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

## [Input circuit]



Note: The parameters are normally set to the above port numbers before shipment.

**⚠ 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.

## ◎ Input signals to the Tabletop Robot



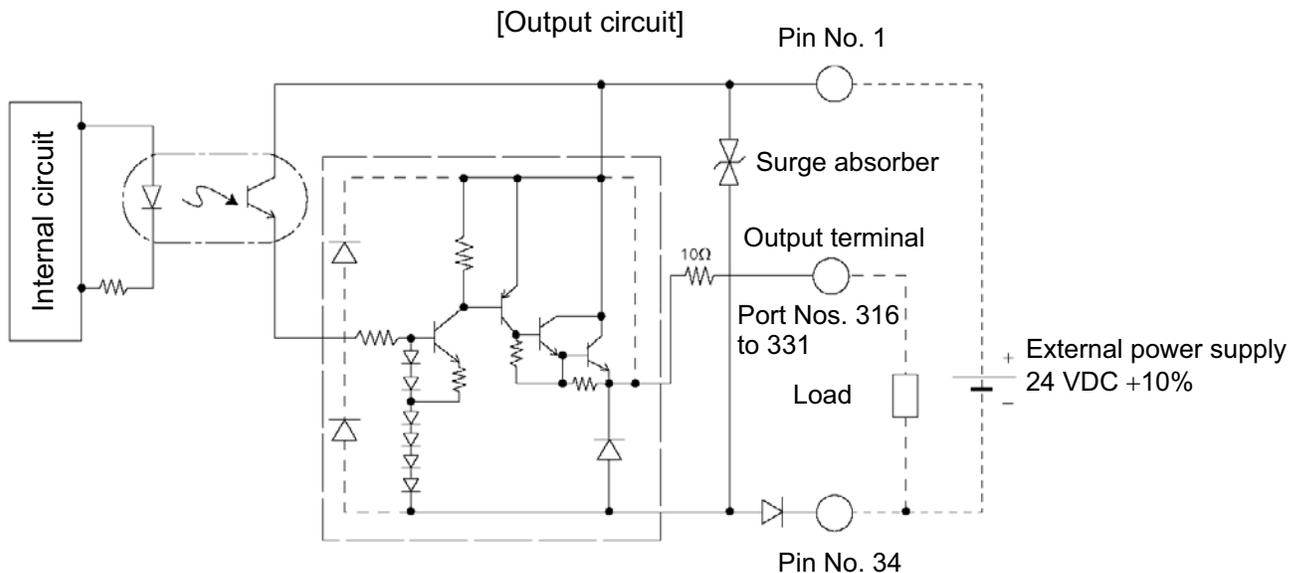
At the default settings, the system recognizes the ON/OFF durations of input signals if they are approx. 4 msec or longer. The ON/OFF duration settings can also be changed using I/O parameter No. 20 (input filtering frequency).

## (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                           |                         |
| Isolation method     | Photocoupler isolation                          |                         |
| External devices     | [1] Miniature relay<br>[2] Sequencer input unit |                         |

Note) 400 mA is the maximum total load current of every eight ports from output port No. 300.



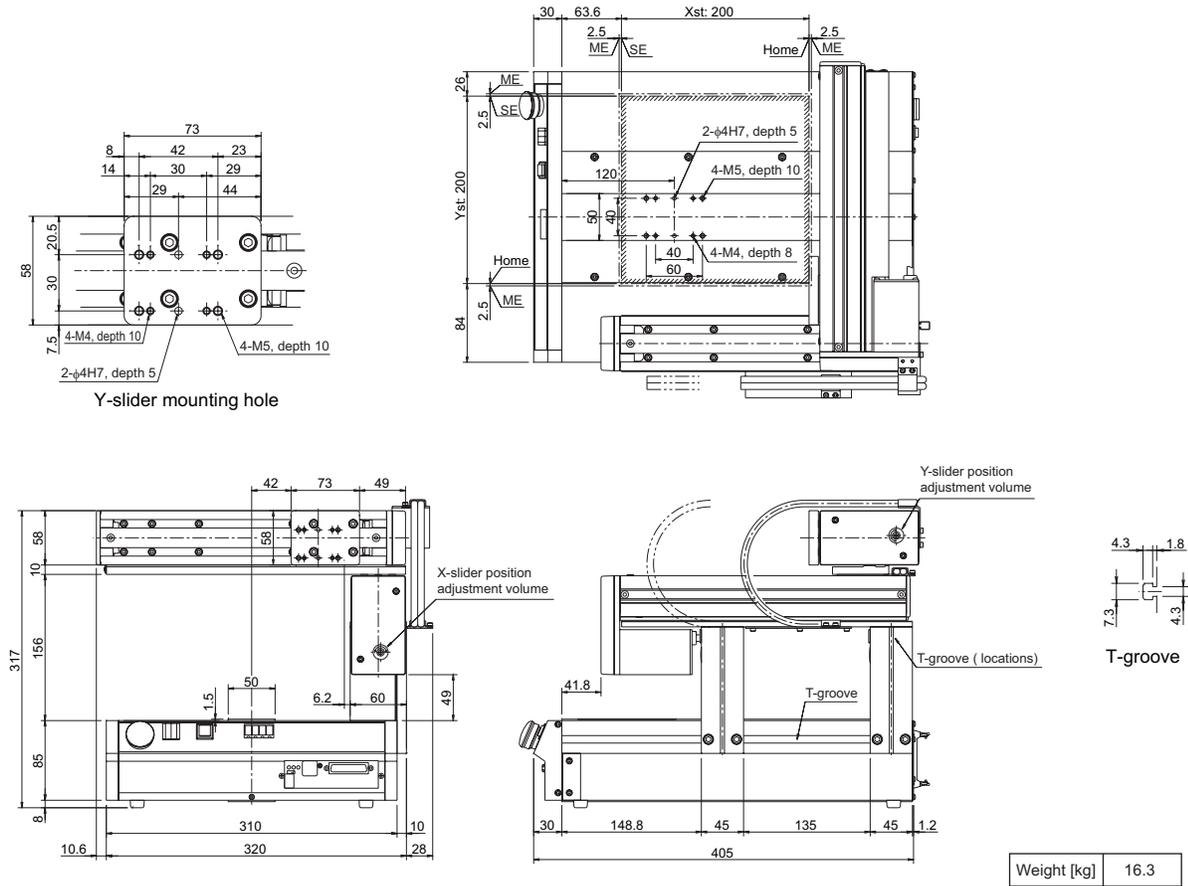
Note: The parameters are normally set to the above port numbers before shipment.

**⚠ 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.

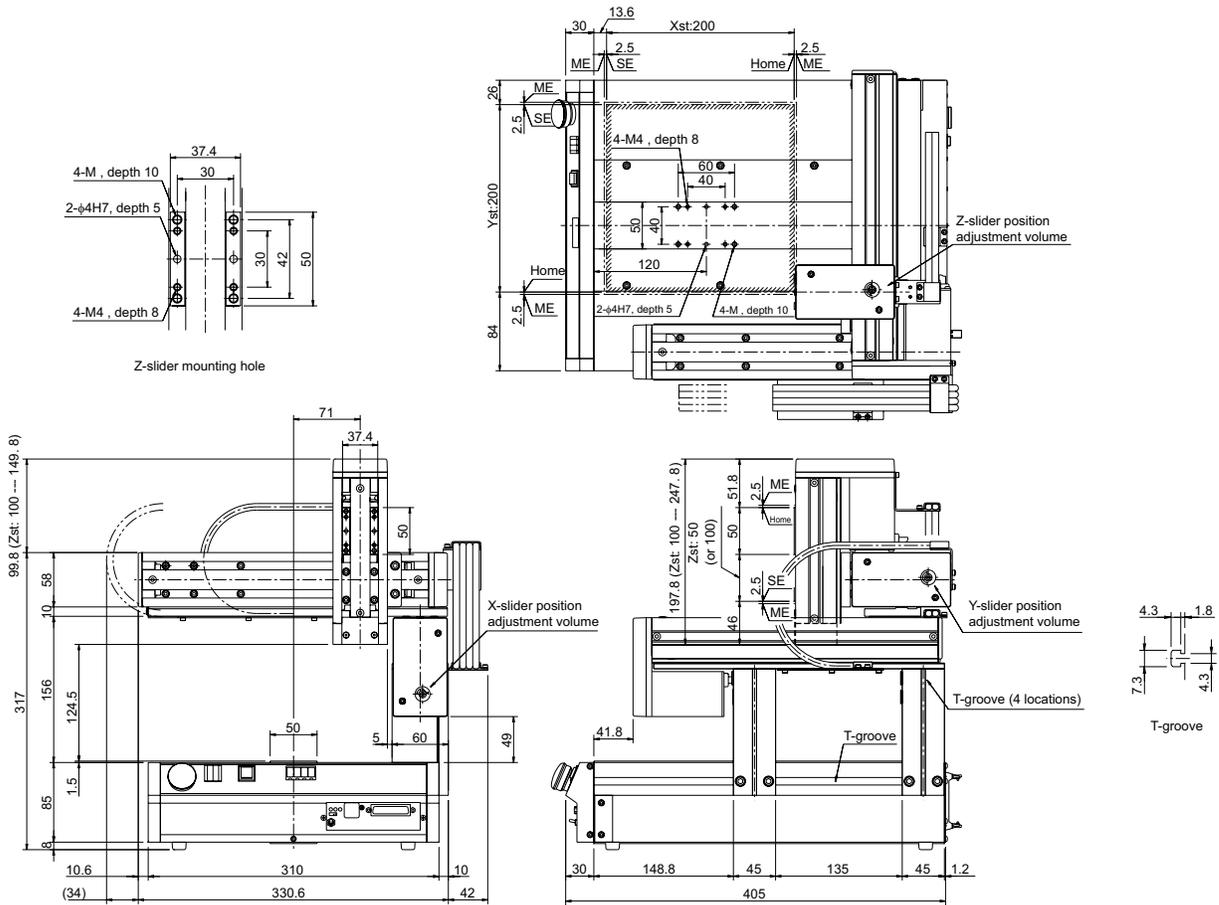
## 2.5 External Dimensions

### Compact, cantilever 2-axis type with 200-mm XY-axis stroke





Compact, cantilever 3-axis type with 200-mm XY-axis stroke



|             |    |
|-------------|----|
| Weight [kg] | 18 |
|-------------|----|

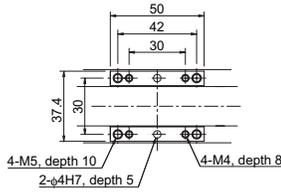




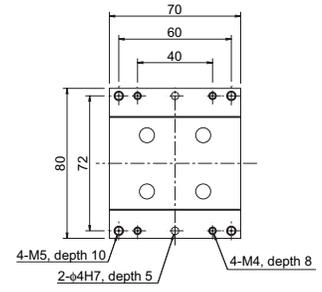
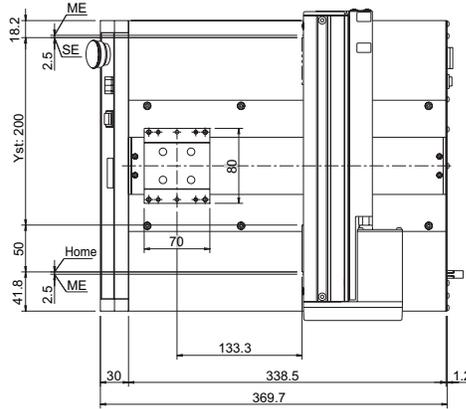


Compact, gate 2-axis type with 200-mm XY-axis stroke

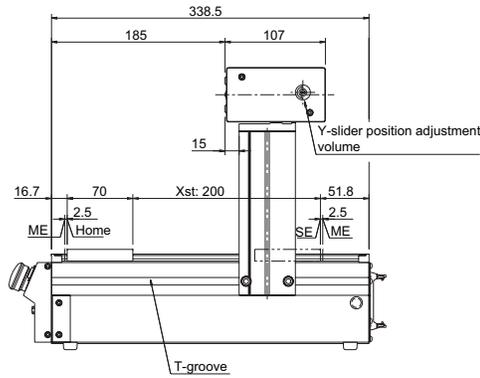
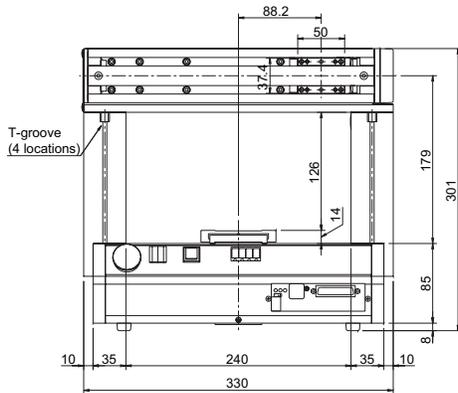
2. Specifications



Y-slider mounting hole



Z-slider mounting hole

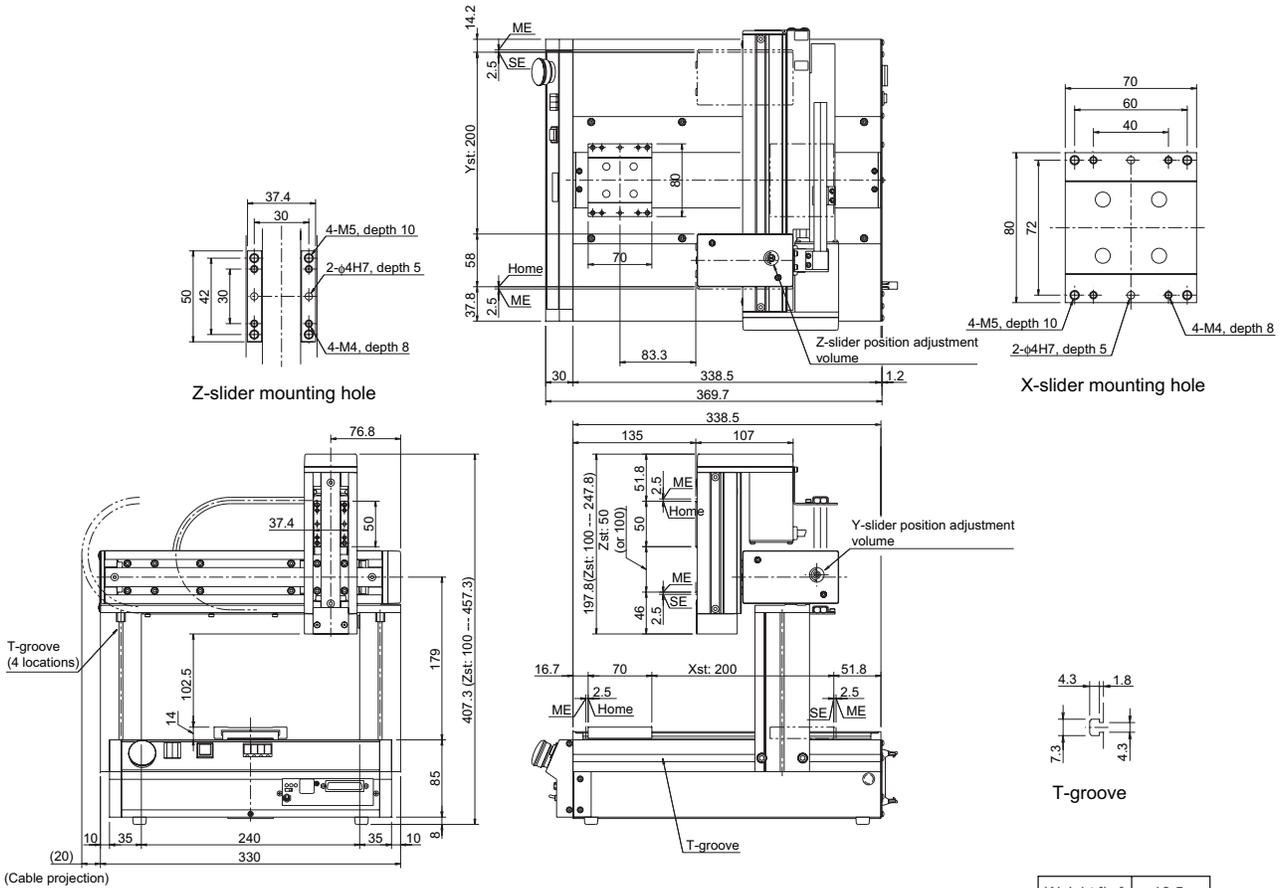


T-groove

|             |      |
|-------------|------|
| Weight [kg] | 14.8 |
|-------------|------|



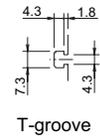
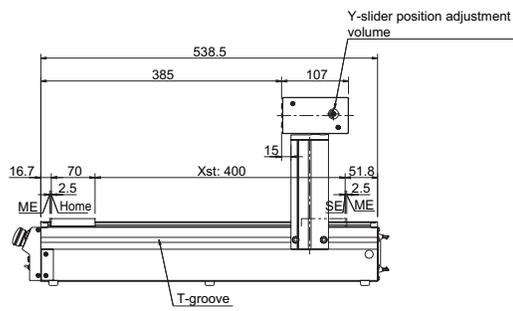
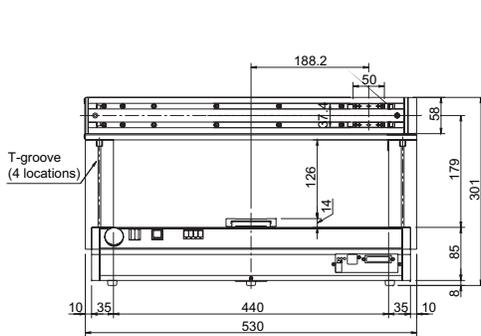
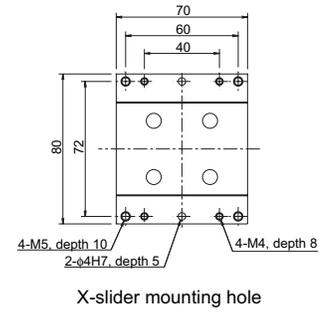
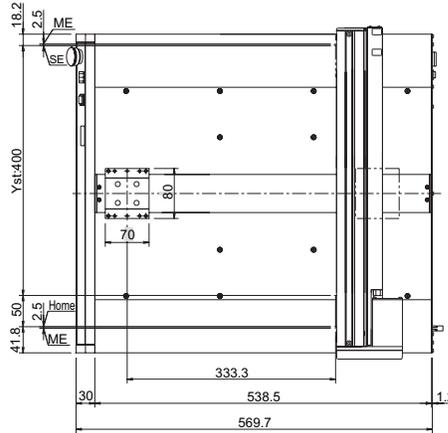
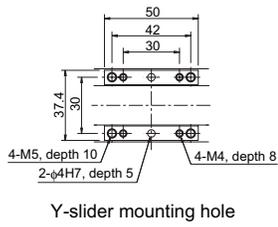
Compact, gate 3-axis type with 200-mm XY-axis stroke



|             |      |
|-------------|------|
| Weight [kg] | 16.5 |
|-------------|------|



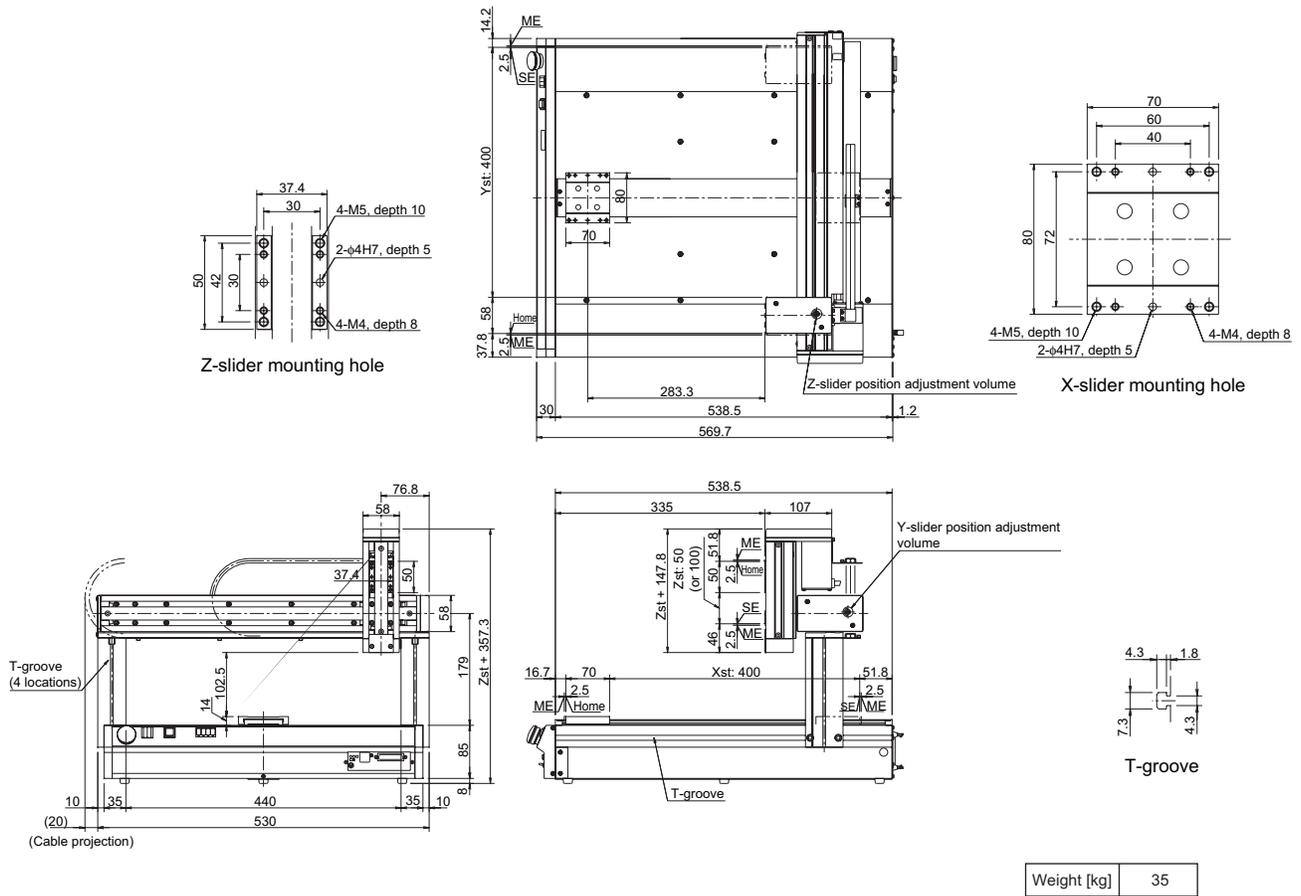
### Large, gate 2-axis type with 400-mm XY-axis stroke



|             |    |
|-------------|----|
| Weight [kg] | 33 |
|-------------|----|



### Large, gate 3-axis type with 400-mm XY-axis stroke





### 3. Installation Environment, Noise Measures and Other

As for the use environment, the products explained in this manual can be used in an environment of pollution degree 2\*1 or equivalent.

\*1 Pollution degree 2: An environment normally subject to non-conductive contaminants only, but conductive contaminants may generate temporarily due to bedewing.  
(EN60947-5-1)

#### 3.1 Installation Environment

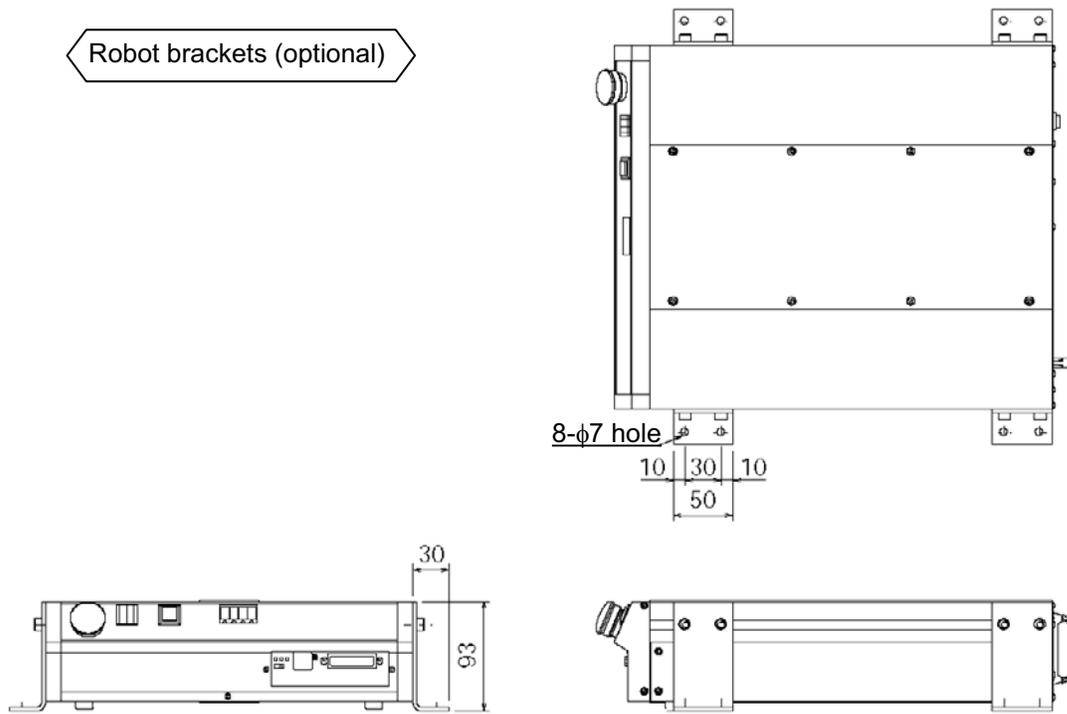
- (1) When installing and wiring the Tabletop Robot, do not block the ventilation holes provided for cooling. (Insufficient ventilation will not only prevent the robot from functioning fully, but it may also result in failure.)
- (2) Prevent foreign matter from entering the Tabletop Robot through the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



### 3.2 Installation

#### 3.2.1 Brackets (Optional)

The Tabletop Robot is fitted with rubber feet to prevent movement on the installed surface. However, the robot may still move depending on the conditions of use (load, acceleration/deceleration). Optional brackets securely affix the robot and prevent it from moving. Install the brackets to the robot using the dedicated T-slots and hexagon socket head bolts (M6x12).



#### 3.2.2 Installing the Load, Etc.

Tapped holes and positioning holes are provided on the X-axis slider (gate type only), Y-axis slider and Z-axis slider. Use these holes to install a load, tool, etc., to each slider. For details, refer to the external dimension view of the robot.



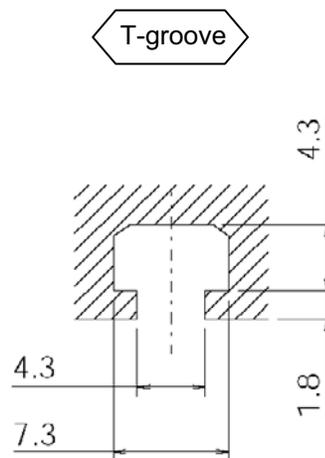
### 3.2.3 Using the T-grooves

T-grooves of M4 size are provided on the frame that supports the actuator.

Auxiliary tools and other items can be installed in these T-grooves using nuts.

Use of square nuts is recommended for affixing items using T-grooves, but general hexagonal nuts can also be used.

As for the bolts used for installation, pay attention to their length to ensure that the tip of the bolt will not contact the bottom of the T-groove.



### 3.3 Power Source

The Tabletop Robot takes power from a single-phase power source of 100 to 230 VAC  $\pm$  10%.

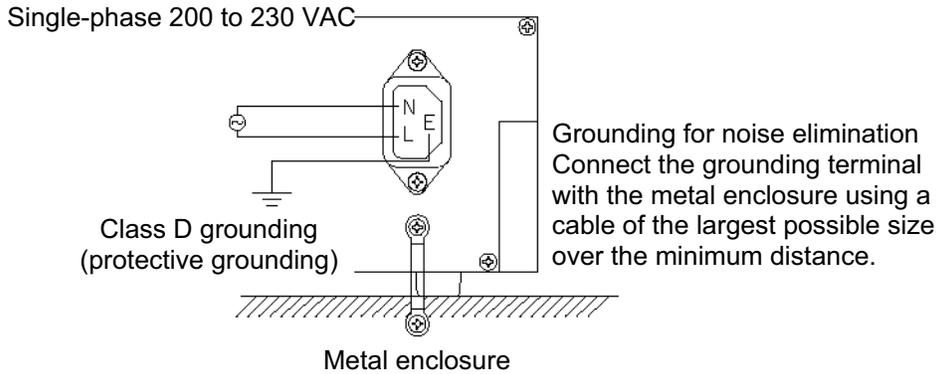
The voltage-source capacity varies depending on the power-source voltage and number of axes, as follows.

|                |        | Power-source voltage |        |
|----------------|--------|----------------------|--------|
|                |        | 100 V                | 200 V  |
| Number of axes | 2 axes | 150 VA               | 155 VA |
|                | 3 axes | 210 VA               | 215 VA |

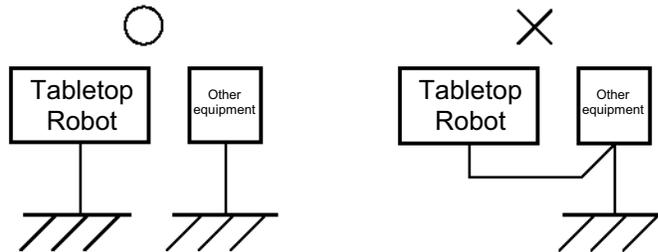
### 3.4 Noise Measures and Grounding

#### 3.4.1 Grounding

Power terminal E is used for protective grounding. Provide Class D grounding from this terminal.



Provide dedicated grounding from the Tabletop Robot.



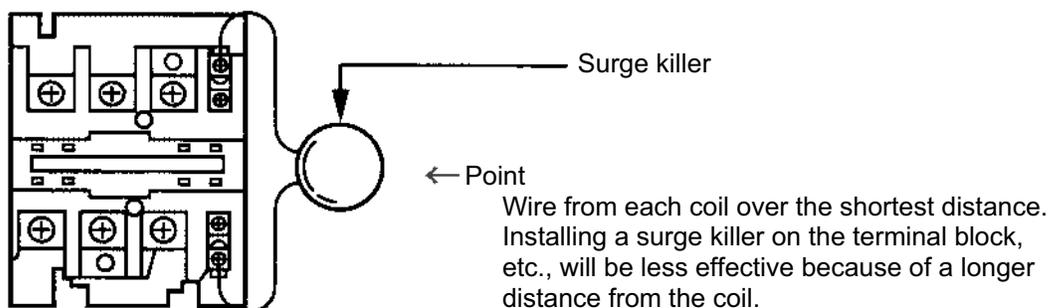
Do not ground the robot as shown above.

#### 3.4.2 Noise sources and noise elimination

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

[1] AC solenoid valve, magnet switch, relay

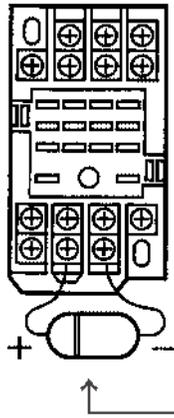
Measure --- Install a surge killer in parallel with the coil.





[2] DC solenoid valve, magnet switch, relay

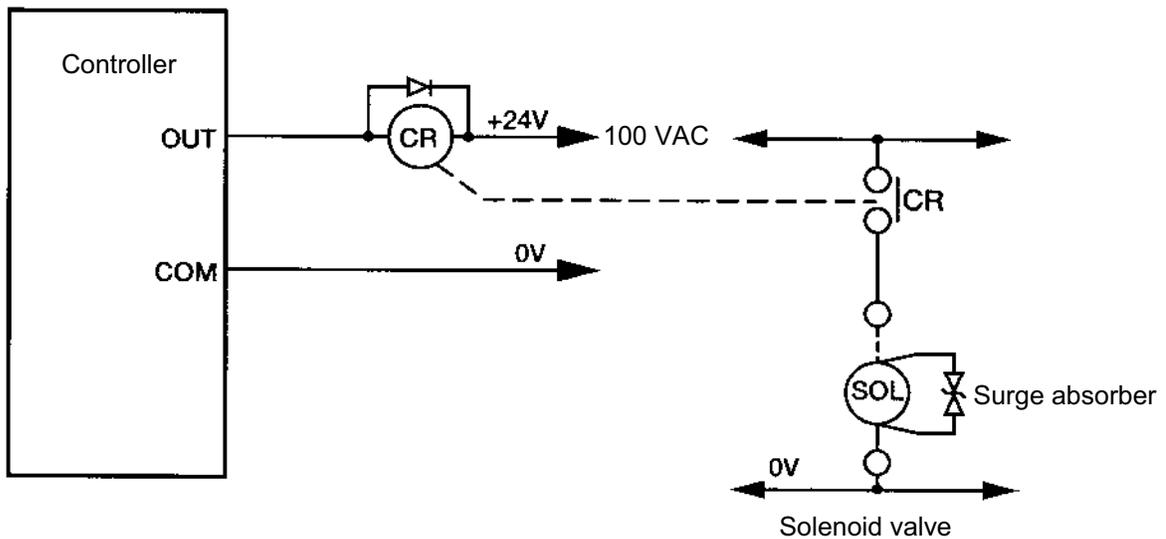
Measure --- Install a diode in parallel with the coil. Determine the diode capacity in accordance with the load capacity.



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

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.

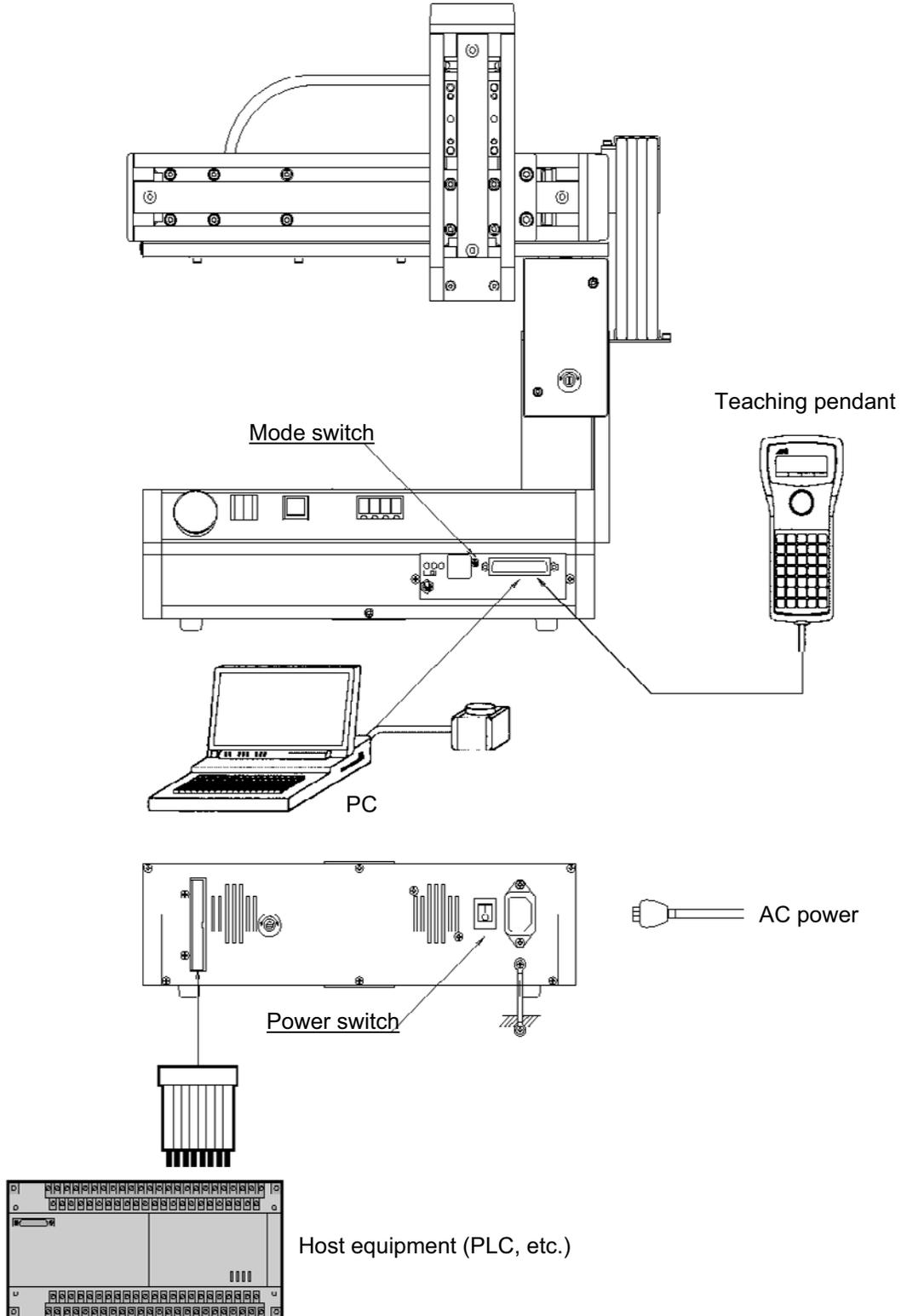
Reference Circuit Diagram





## 4. System Setup

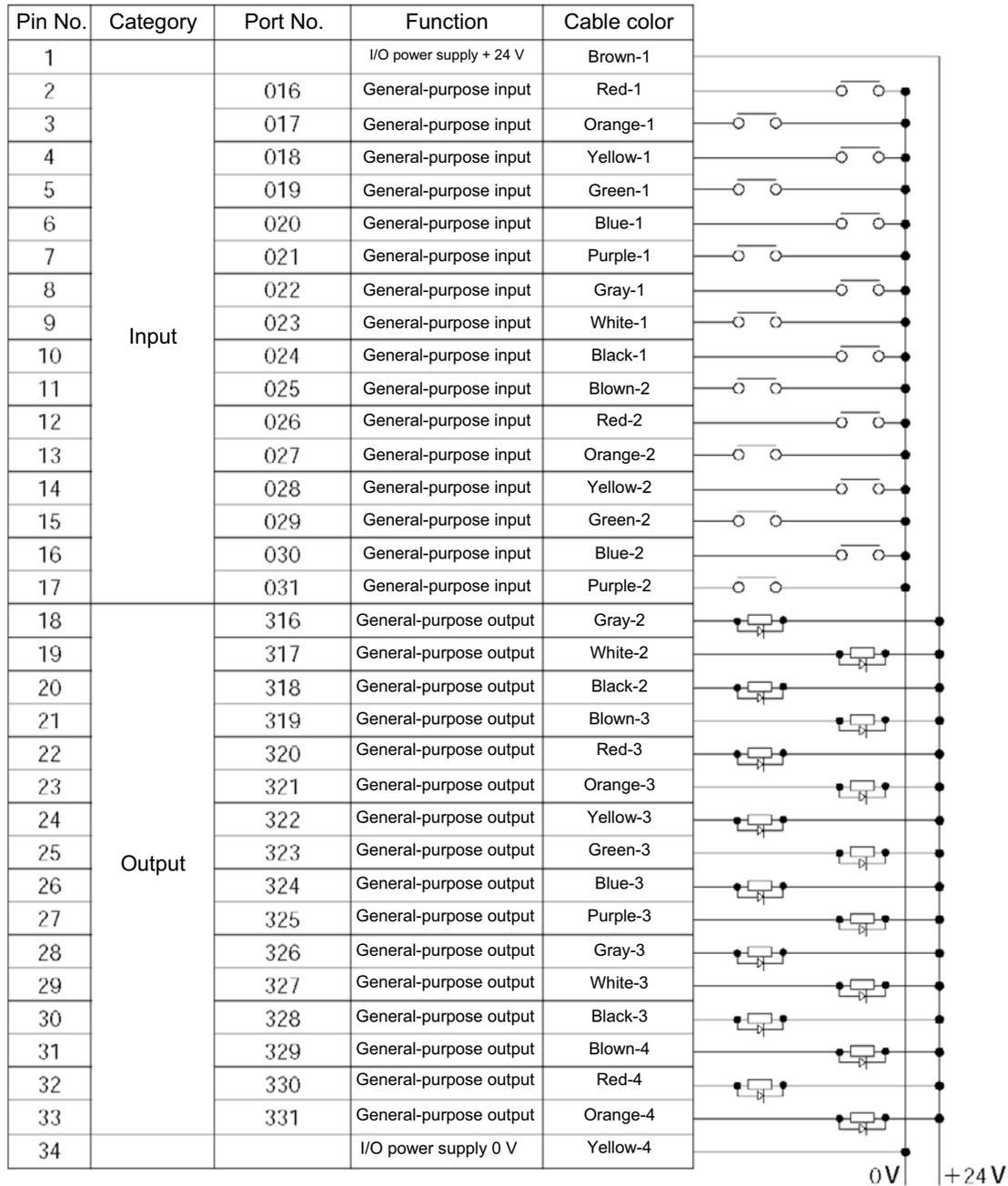
### 4.1 Connecting the Tabletop Robot with Peripheral Equipment





### 4.2 I/O Connection Diagram (External DIOs)

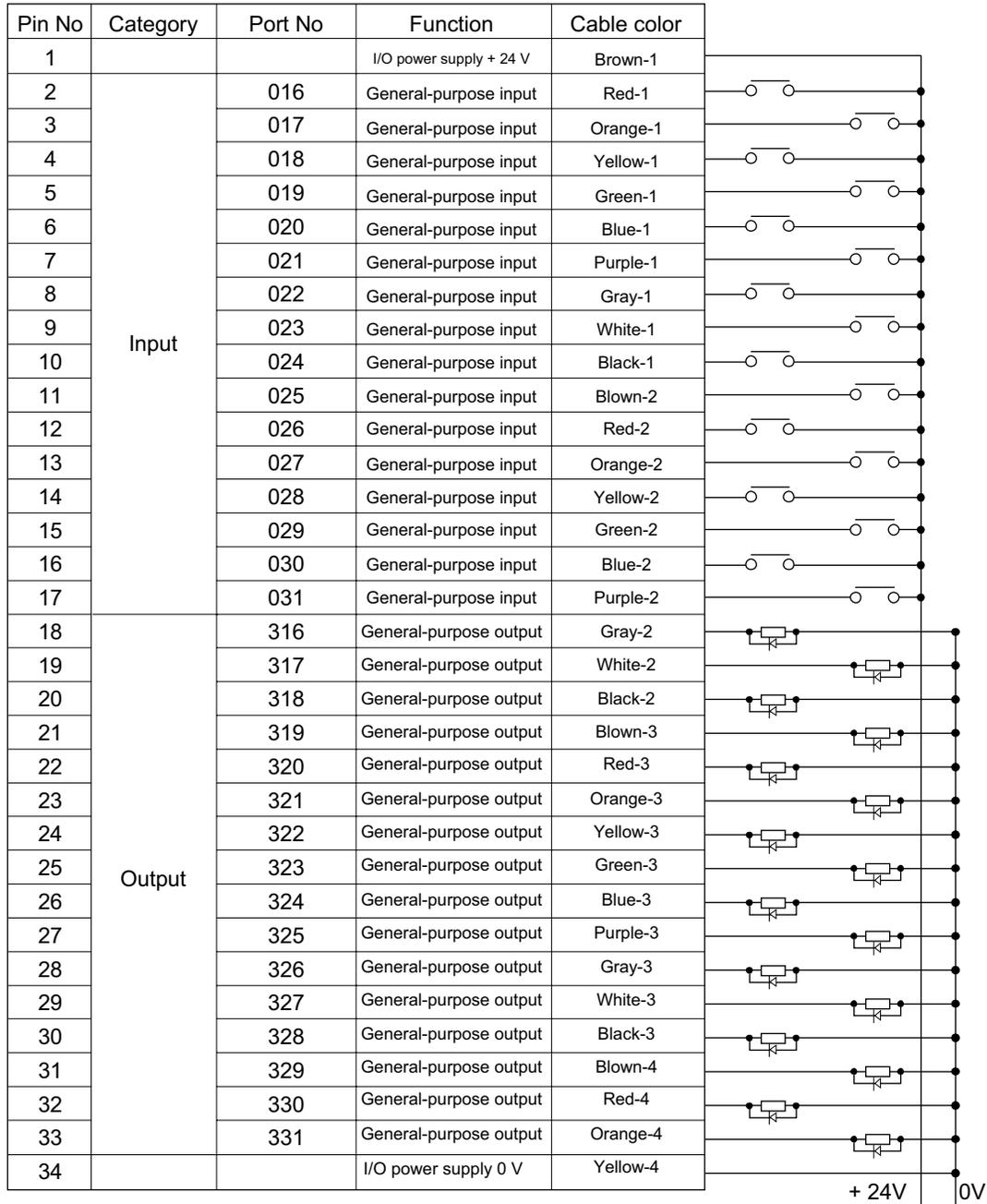
#### 4.2.1 NPN specification



Note: The parameters are normally set to the above port numbers before shipment.



4.2.2 PNP specification



Note: The parameters are normally set to the above port numbers before shipment.



## Chapter 2 Operation

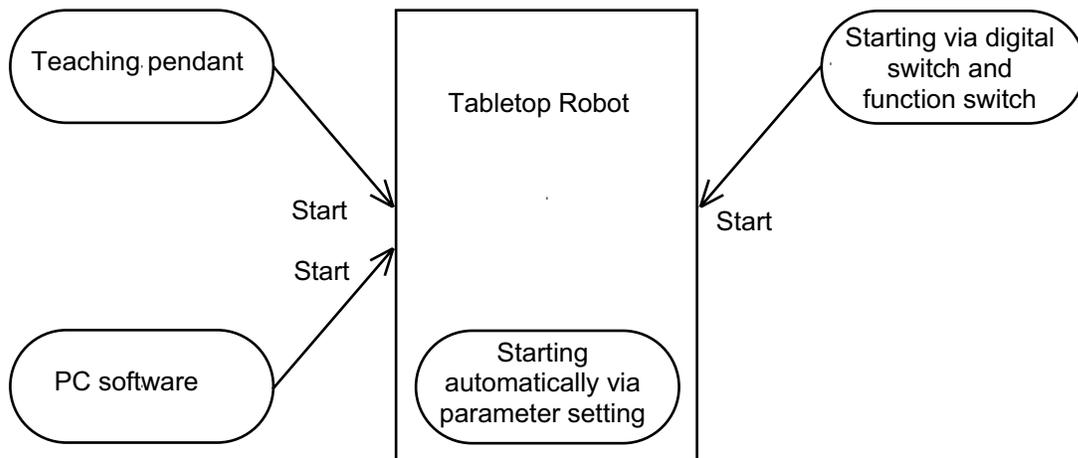
### 1. Operation

#### 1.1 How to Start a Program

With the Tabletop Robot, the stored programs can be started (run) using the four methods specified below:

- Starting from the PC software
- Starting from the teaching pendant
- Starting automatically by a parameter setting (auto start)
- Starting via the digital program selector switch and function switch

The starting methods using the PC software and teaching pendant are used for simple operation checks as part of debugging process. For the specific operating procedures to start Tabletop Robot programs from the PC software or teaching pendant, refer to the operation manual for the PC software or teaching pendant.

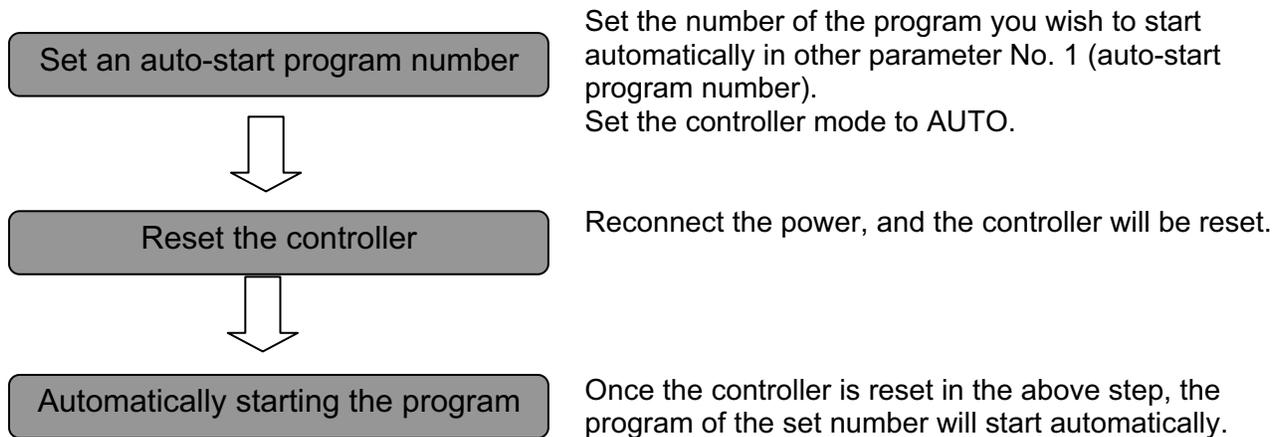




## 1.2 Starting a Program by Auto-Start via Parameter Setting

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

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



### **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.



### 1.3 Starting via the Digital Program Selector Switch and Function Switch

Set a desired program number using the digital program selector switch. Pressing the function switch will start the specified program.

To use this starting method, one of the following conditions must be satisfied:

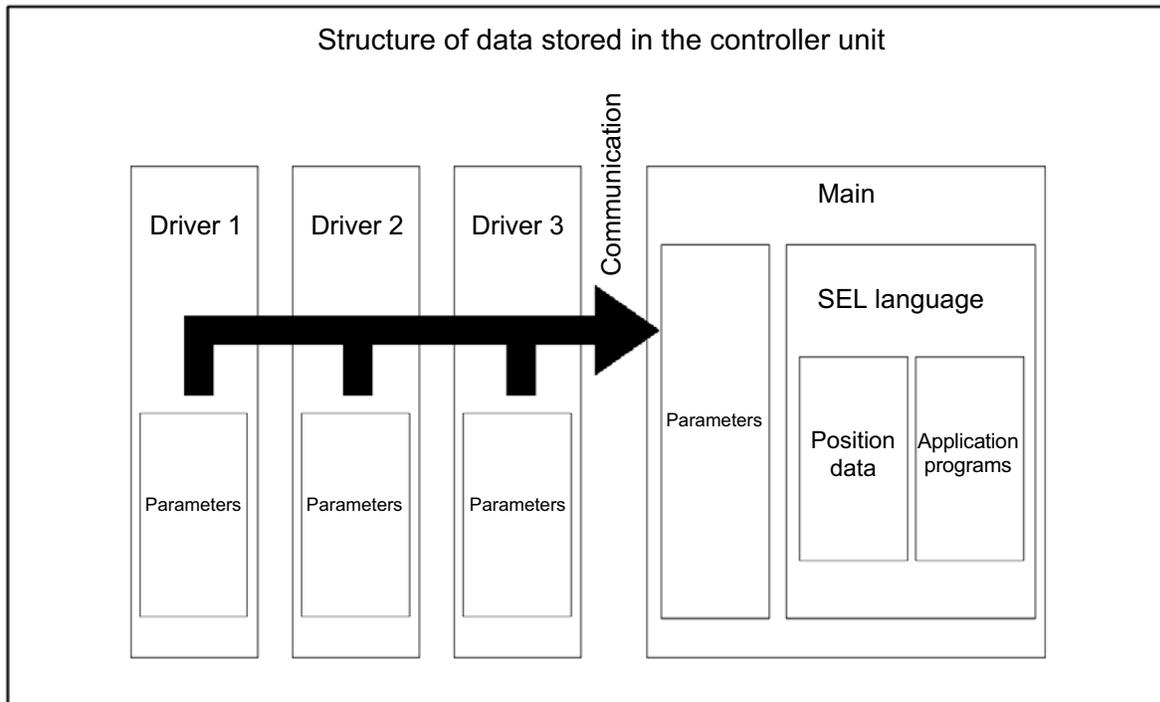
- The robot is in the AUTO mode.
- The robot is in the MANU mode, where the teaching pendant or PC software has been connected to the robot online and the robot is not yet restarted. (Once the teaching pendant or PC software is connected online and the digital switch is set, the specified program can be started using the function switch even after the connection is switched offline.)



## 2. Controller Data

### 2.1 Data Structure

The controller unit of the Tabletop Robot stores parameters as well as position data and application programs needed to execute SEL commands.

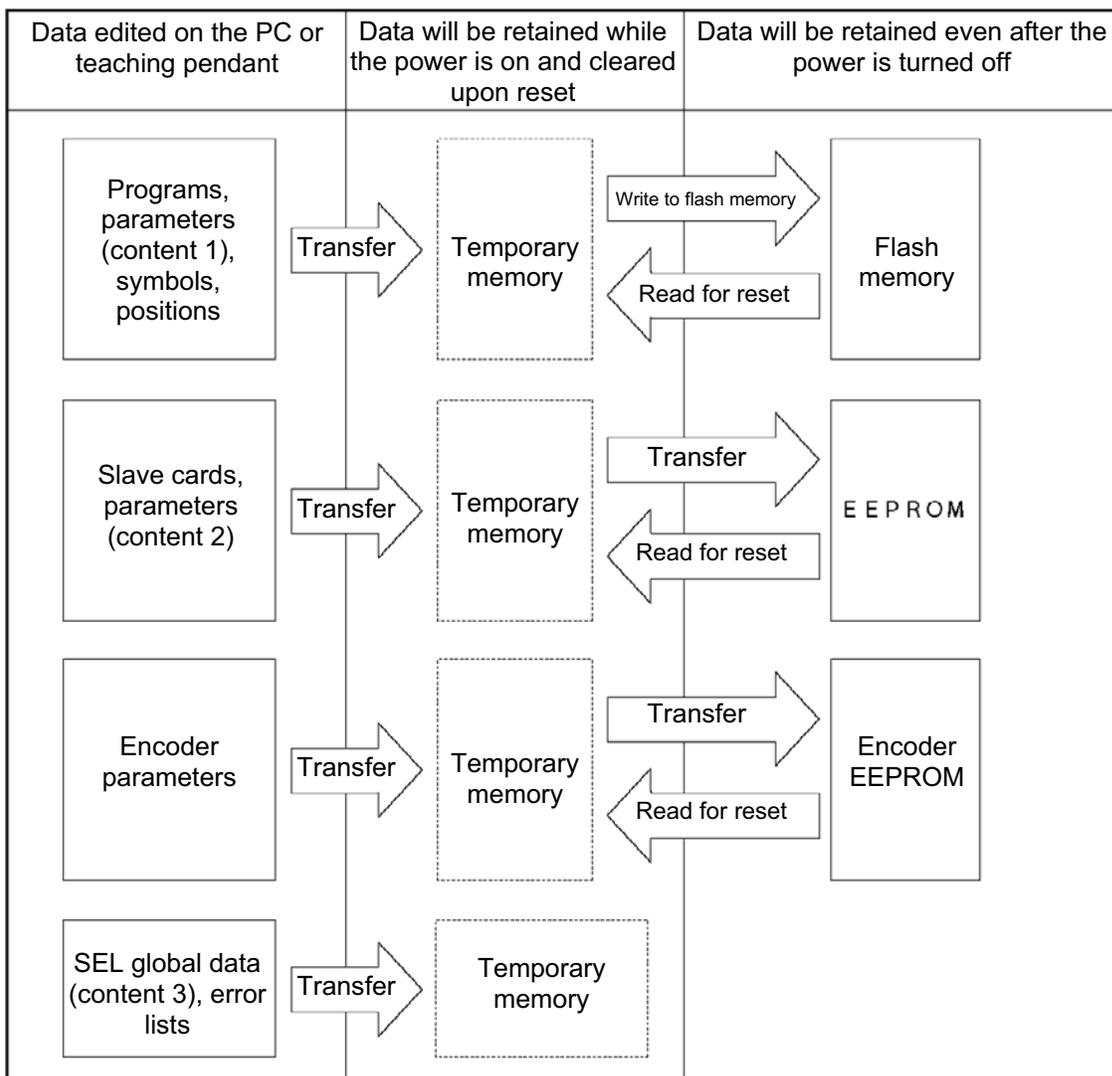


The user must create position data and application programs. The parameters are predefined, but their settings can be changed in accordance with the user's system. Refer to Appendix, "List of Parameters," for details on the parameters.

## 2.2 Saving Data

When data is created/edited using the PC software or teaching pendant is sent to the controller (or when the [WRT] key is pressed on the teaching pendant), the data is stored in the controller's temporary memory. The data stored in the controller's temporary memory will be erased once the controller is powered off or restarted (via software reset). For important data, always write to the flash memory so that they will not be lost.

Note: Global data (variables, flags and strings) and error lists will be erased once the controller is powered off or restarted (via software reset). (These data cannot be retained after the power is turned off.) The error list is retained after a software reset, but will be cleared once the power is turned off.



- 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

Since the programs, parameters, symbols and positions are read from the flash memory at restart, the data in the temporary memory will return to the original data before editing 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).



### 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.



## Chapter 3 X-SEL Language Data

### 1. Values and Symbols Used in SEL Language

#### 1.1 List of Values and Symbols Used

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

| Function  | Global range                           | Local range  | Remarks                                 |
|---|--|--|---|
| Input port  | 000 to 299 (300)                       |  | Varies depending on the function.       |
| Output port   | 300 to 599 (300)                       |  | Varies depending on the function.       |
| Flag  | 600 to 899 (300)                       | 900 to 999 (100)   |   |
| Variable (integer)                                  | 200 to 299 (100)<br>1200 to 1299 (100) | 1 to 99 (99)<br>1001 to 1099 (99)  | 99 is used for IN, INB, OUT, OUTB, etc. |
| Variable (real)                                     | 300 to 399 (100)<br>1300 to 1399 (100) | 100 to 199 (100)<br>1100 to 1199 (100)   | 199 is used for PPUT, PGET, PARG, etc.  |
| String  | 300 to 999 (700)                       | 1 to 299 (299)   |   |
| Tag number  |  | 1 to 99 (99)   |   |
| Subroutine number                                   |  | 1 to 99 (99)   |   |
| Zone number   | 1 to 4 (4)                             |  |   |
| Pallet number                                       |  | 1 to 10 (10)   |   |
| Axis number   | 1 to 3 (3)                             |  | Varies depending on the function.       |
| Axis pattern  | 0 to 111                               |  |   |
| Position number                                     | 1 to 3000 (3000)                       |  |   |
| Program number                                      | 1 to 64 (64)                           |  |   |
| Step number   | 1 to 6000 (6000)                       |  |   |
| Task level  | NORMAL/HIGH (2)                        |  |   |
| SIO channel number                                  | 1 to 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 to 7299 (300)                     |  |   |
| Virtual output port (SEL user program → SEL system) | 7300 to 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.<br>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 are retained until the controller is powered off.
- The variables and flags in the local range are cleared when the program is started (the data are also cleared when the controller is powered off).
- 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 Tabletop Robot 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 Tabletop Robot.)

If position data are used in internal operations as numeric data (repeated multiplications and divisions), the precision 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 (External DIOs)

(1) Input ports

Used as input ports for limit switches, sensor switches, etc.

|                         |
|-------------------------|
| Input number assignment |
| 016 to 031 (standard)   |

(2) Output ports

Used as various output ports.

|                          |
|--------------------------|
| Output number assignment |
| 316 to 331 (standard)    |

Note: The parameters are normally assigned to the above input port and output port numbers before shipment.



## 1-3 Virtual I/O Ports

## (1) Virtual input ports

| Port No.     | Function   |
|--------------|--|
| 7000         | Always OFF   |
| 7001         | Always ON  |
| 7002         | Voltage low warning for system-memory backup battery   |
| 7003         | Abnormal voltage of system-memory backup battery   |
| 7004         | For future expansion = Use prohibited  |
| 7005         | For future expansion = Use prohibited  |
| 7006         | Top-level system error = Message level error is present  |
| 7007         | Top-level system error = Operation-cancellation level error is present   |
| 7008         | Top-level system error = Cold-start level error is present   |
| 7009         | For future expansion = Use prohibited  |
| 7010         | Drive-source cutoff factor is present (including when waiting for cutoff reset input)  |
| 7011         | Latch signal indicating that all-operation-cancellation factor is present (latch signal for recognizing 1-shot cancellation factor; latch is cancelled by 7300-ON) |
| 7012         | All-operation-pause factor is present (including when waiting for restart switch signal) (Valid only during automatic operation recognition)                       |
| 7013         | All-servo-axis-interlock factor is present (all-operation-pause factor + interlock input-port factor)  |
| 7014         | For future expansion = Use prohibited  |
| 7015         | For future expansion = Use prohibited  |
| 7016         | For future expansion = Use prohibited  |
| 7017         | For future expansion = Use prohibited  |
| 7018         | For future expansion = Use prohibited  |
| 7019         | For future expansion = Use prohibited  |
| 7020         | For future expansion = Use prohibited  |
| 7021         | For future expansion = Use prohibited  |
| 7022         | For future expansion = Use prohibited  |
| 7023 to 7030 | For future expansion = Use prohibited  |
| 7031         | For future expansion = Use prohibited  |
| 7032         | For future expansion = Use prohibited  |
| 7033         | For future expansion = Use prohibited  |
| 7034         | For future expansion = Use prohibited  |
| 7035         | For future expansion = Use prohibited  |
| 7036         | For future expansion = Use prohibited  |
| 7037         | For future expansion = Use prohibited  |
| 7038 to 7040 | For future expansion = Use prohibited  |
| 7041 to 7070 | For future expansion = Use prohibited  |
| 7071         | In AUTO mode   |
| 7072         | During automatic operation   |
| 7073 to 7100 | For future expansion = Use prohibited  |
| 7101         | Running program No. 01 (including during pause)  |
| ~            | ~  |
| 7164         | Running program No. 64 (including during pause)  |
| 7165 to 7299 | For future expansion = Use 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 to 7380 | For future expansion = Use prohibited  |
| 7381 to 7399 | For future expansion = Use prohibited  |
| 7400 to 7599 | For future expansion = Use prohibited  |



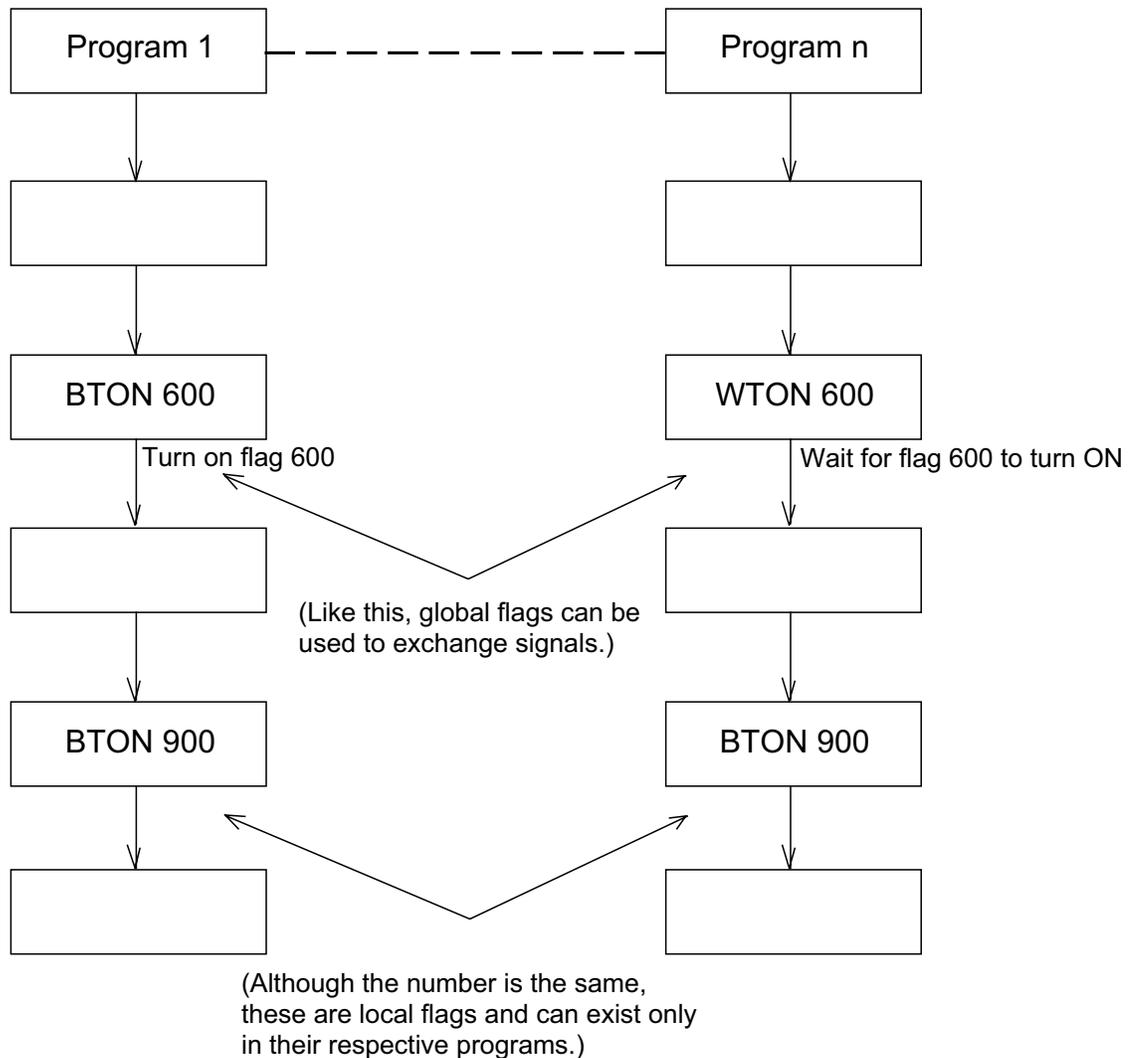
### 1-4 Flags

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.

General-purpose flags (global flags) are retained until the controller is powered off. Dedicated flags (local flags) are cleared when the program is started.

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

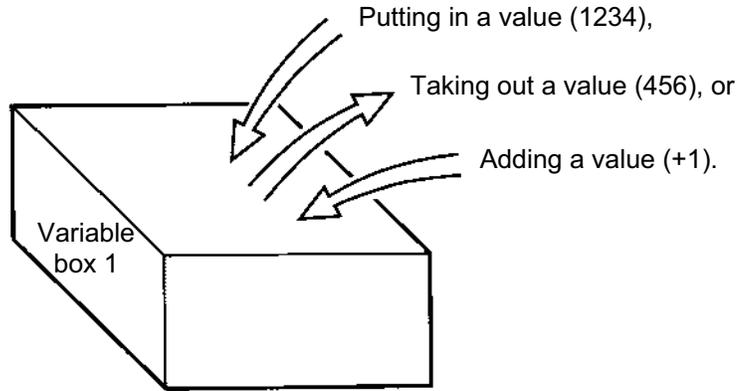


1-5 Variables

(1) Meaning of variable

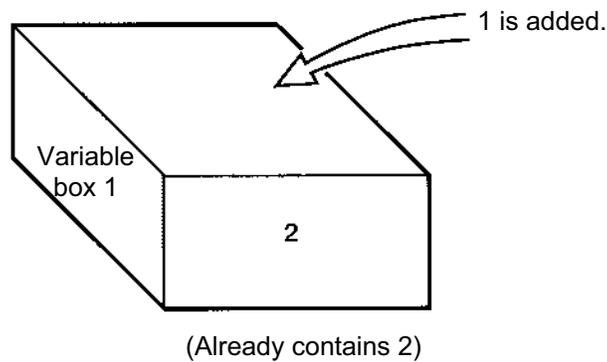
“Variable” is a technical term used in software programming. Simply put, it means “a box in which a value is put.” Variables can be used in many ways, such as putting in or taking out a value and performing addition or subtraction.

A variable can be used in many ways, such as:



| Command | Operand 1 | Operand 2 |
|---------|-----------|-----------|
| ADD     | 1         | 1         |

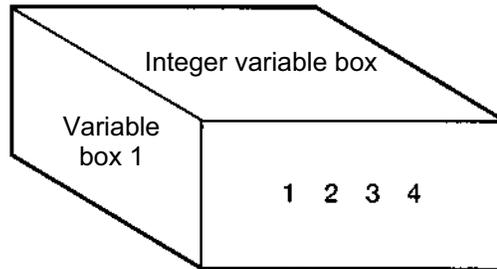
If this command is applied to variable box 1, which already contains 2, then 1 will be added to the current value and 3 will result.





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

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

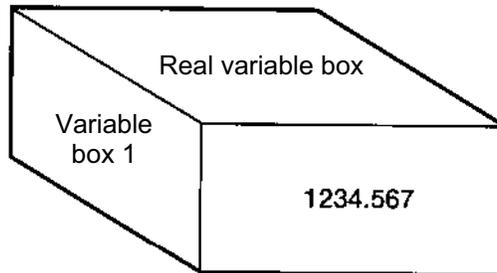


|                         |                            |                             |                            |
|-------------------------|----------------------------|-----------------------------|----------------------------|
| Integer variable number | 200 to 299<br>1200 to 1299 | Can be used in all programs | “Global integer variables” |
| Integer variable number | 1 to 99<br>1001 to 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 to 399<br>1300 to 1399 | Can be used in all programs | “Global real variables” |
| Real variable number | 100 to 199<br>1100 to 1199 | Used only in each program   | “Local real variables”  |

**Caution**

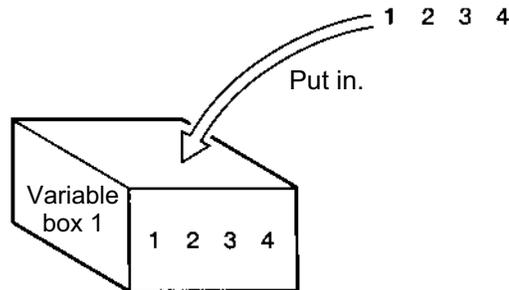
Real number 199 is a special register this system uses in real-number operations. Any value in the range from -99,999.9 to 999,999.9 (eight digits including a sign) can be input in programs.

**[3] Variables with "\*" (asterisk) (indirect specification)**

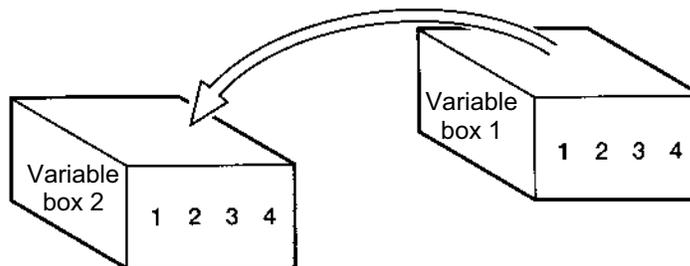
An "\*" (asterisk) is used to specify a variable.

In the following example, the content of variable box 1 will be put in variable box 2. If variable box 1 contains "1234," then "1234" will be put in variable box 2.

| Command | Operand 1 | Operand 2 |
|---------|-----------|-----------|
| LET     | 1         | 1234      |



| Command | Operand 1 | Operand 2 |
|---------|-----------|-----------|
| LET     | 2         | *1        |



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

An "\*" is also used when indirectly specifying a symbol variable (refer to 1-8, "Symbols").

| Command | Operand 1 | Operand 2 |
|---------|-----------|-----------|
| LET     | ABC       | 1         |
| LET     | BCD       | 2         |
| ADD     | ABC       | *BCD      |

Put 1 in variable ABC.

Put 2 in variable BCD.

Add the content of variable BCD, or 2, to variable ABC.  
(The content of variable ABC becomes 3.)

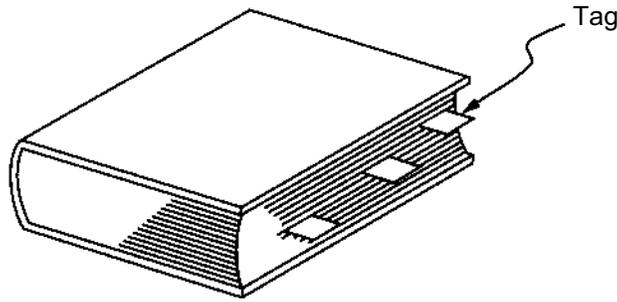


### 1-6 Tags

The term “tag” means “heading.”

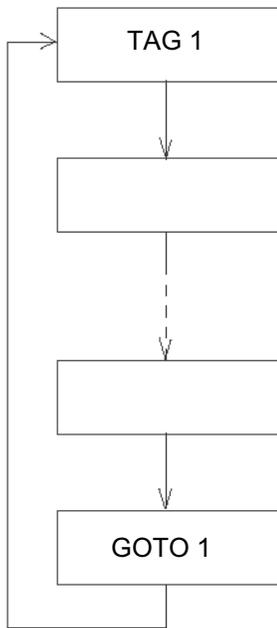
Tags are used in the same way you attach labels to the pages in a book you want to reference frequently.

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



| Command | Operand 1                             |
|---------|---------------------------------------|
| TAG     | Tag number (Integer between 1 and 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) |

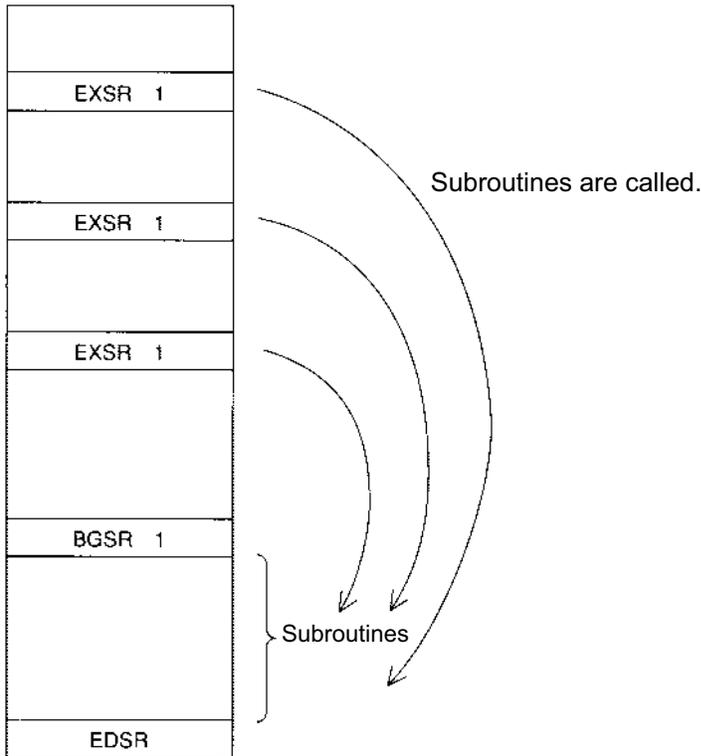
Subroutine execution command

| 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 Tabletop Robot, values such as variable numbers and flag numbers can be handled as symbols. For the method to edit symbols, refer to “Editing Symbols” in the operation manual for X-SEL teaching pendant or “Symbol Edit Window” in the operation manual for X-SEL PC software.

### (1) Supported symbols

The following items can be expressed using symbols:

Variable number, flag number, tag number, subroutine number, program number, position number, input port number, output port number, axis number, constant

### (2) Description rules of symbols

[1] A maximum of nine single-byte alphanumeric characters or underscore starting with an alphabet

(Note: The length of a character-string literal must not exceed eight single-byte characters.)

\* If the PC software version is 1.1.0.5 or later or the teaching pendant version is 1.04 or later, an underscore can be used as the first character in a symbol.

\* If the PC software version is 1.1.0.5 or later, single-byte ASCII code characters from 21h to 7Eh (limited to those that can be input via keyboard) can be used as the second and subsequent 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 symbols can be used in all SEL programs: 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 number | How axis is stated |
|-------------|--------------------|
| 1           | Axis 1             |
| 2           | Axis 2             |
| 3           | Axis 3             |



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

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

- Commands that use axis specification based on axis number  
BASE, PPUT, PGET, 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 number | Axis 3  | Axis 2 | Axis 1  |
| Used        | 1       | 1      | 1       |
| Not used    | 0       | 0      | 0       |

[Example] When axes 1 and 2 are used

Axis 2



011 --- (The 0 in front is not necessary. With the 0 removed, the expression reads "11.")



Axis 1

[Example] When axes 1 and 3 are used

Axis 3



101 --- (In this case, the 0 is needed to indicate the position of axis 3.)



Axis 1

Indirect specification of axis pattern in a variable

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

[Example] To perform home return for axis 3 only, you can specify as follows based on axis pattern:

```
HOME 100
```

In indirect specification, 100 (binary) is expressed as 4 (decimal), so the same operation can be specified as follows:

```
LET 6 4
HOME *6
```

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

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



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, speeds, accelerations and decelerations are set and stored.

\* Maximum  
Gate type: 0.3 G  
Cantilever type: 0.2 G

1 to 300/mm sec

\* Maximum  
Gate type: 0.3 G  
Cantilever type: 0.2 G

| Position No. | Axis 1 | Axis 2 | Axis 3 | Speed | Acceleration | Deceleration |
|--------------|--------|--------|--------|-------|--------------|--------------|
| 1            |        |        |        |       |              |              |
| 2            |        |        |        |       |              |              |
| 3            |        |        |        |       |              |              |
| ⋮            |        |        |        |       |              |              |
| 2998         |        |        |        |       |              |              |
| 2999         |        |        |        |       |              |              |
| 3000         |        |        |        |       |              |              |

- \* 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.

The effective speed and acceleration are determined based on the following priorities.

| Priority | Speed                                | Acceleration (deceleration)  |
|----------|--------------------------------------|--|
| 1        | Position data value set in operand 1 | Position data value set in operand 1   |
| 2        | Value set by a VEL command           | Value set by an ACC (DCL) command  |
| 3        |                                      | All-axis parameter No. 11, "Default acceleration"<br>(All-axis parameter No. 12, "Default deceleration") |

### 3. Command Part

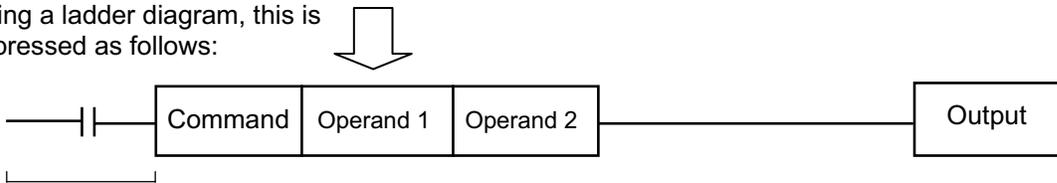
The primary feature of SEL language is its very simple command structure. Since the structure is simple, there is no need for a compiler (to translate into computer language) and high-speed operation is possible via an interpreter (the program runs as commands are translated).

#### 3.1 SEL language Structure

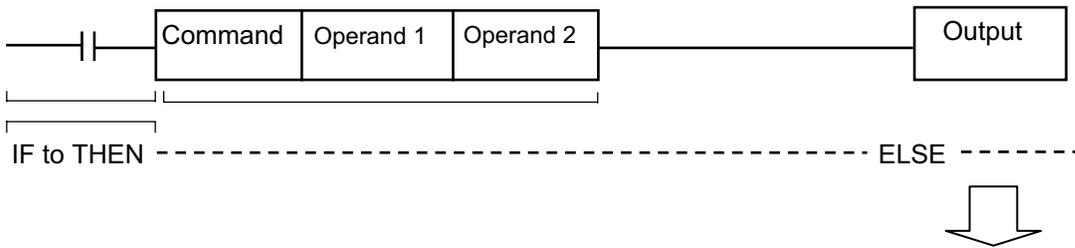
The table below shows the structure of one command step.

| Extension condition<br>(AND, OR) | Input condition<br>(I/O, flag) | Command, declaration    |           |           | Output<br>(Output port, flag) |
|----------------------------------|--------------------------------|-------------------------|-----------|-----------|-------------------------------|
|                                  |                                | Command,<br>declaration | Operand 1 | Operand 2 |                               |
|                                  |                                |                         |           |           |                               |

Using a ladder diagram, this is expressed as follows:

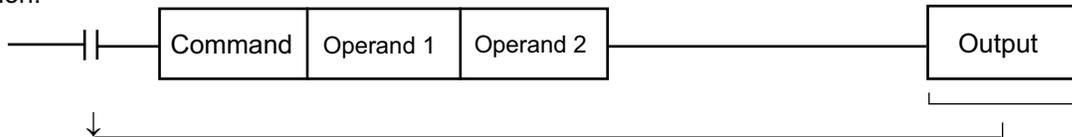


(1) The condition before the command is equivalent to "IF ~ THEN..." in BASIC.



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

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



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

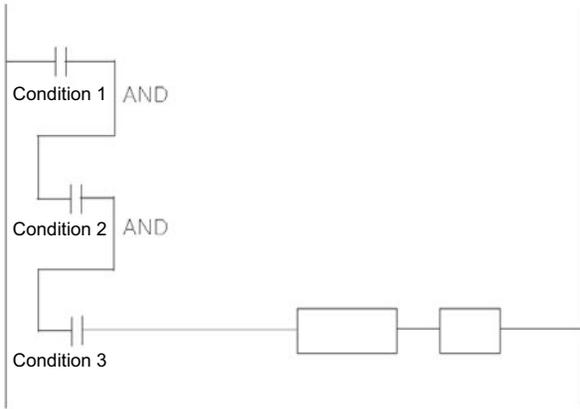


### 3.2 Extension Condition

Conditions can be combined in a complex manner.

AND extension

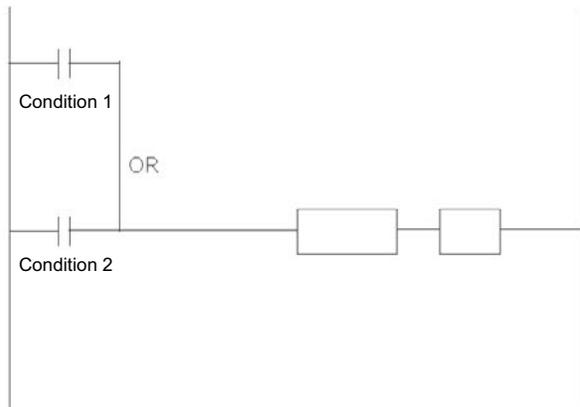
(Ladder diagram)



(SEL language)

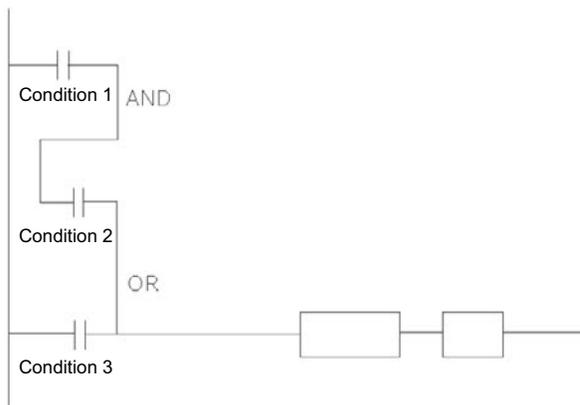
| Extension condition | Input condition | Command |           |           | Output |
|---------------------|-----------------|---------|-----------|-----------|--------|
|                     |                 | Command | Operand 1 | Operand 2 |        |
|                     | Condition 1     |    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

OR extension



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

AND extension and OR extension



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



## Chapter 4 Commands

### 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.

Once an “actuator control declaration” command is executed in a program, the command will remain valid as long as the program is running. To change the values (in operand 1, operand 2, etc.) already set by the “actuator control declaration” command, the necessary parts of the program must be set again. In other words, the values set by the last executed command will prevail.

The output field will be turned OFF when the command is executed. Once the execution is completed, the output field may be turned ON depending on the operation type condition in the output field. (The output field will remain OFF if the condition is not satisfied.)

Note: The output field of a comparison command CPXX (CPEQ, CPNE, CPGT, CPGE, CPLT or CPLE) will not be turned OFF when the command is executed.

#### 1.1 List of Commands by Function

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                            | 84   |
|                      | Optional  | TRAN    | Copy-destination variable           | Copy-source variable  | ZR  | Copy                              | 84   |
|                      | Optional  | CLR     | Start-of-clear variable             | End-of-clear variable | ZR  | Clear variable                    | 85   |
| Arithmetic operation | Optional  | ADD     | Augend variable                     | Addend                | ZR  | Add                               | 86   |
|                      | Optional  | SUB     | Minuend variable                    | Subtrahend            | ZR  | Subtract                          | 86   |
|                      | Optional  | MULT    | Multiplicand variable               | Multiplier            | ZR  | Multiply                          | 87   |
|                      | Optional  | DIV     | Dividend variable                   | Divisor               | ZR  | Divide                            | 87   |
|                      | Optional  | MOD     | Remainder assignment variable       | Divisor               | ZR  | Calculate remainder               | 88   |
| Function operation   | Optional  | SIN     | Sine assignment variable            | Operand [radian]      | ZR  | Sine                              | 89   |
|                      | Optional  | COS     | Cosine assignment variable          | Operand [radian]      | ZR  | Cosine                            | 89   |
|                      | Optional  | TAN     | Tangent assignment variable         | Operand [radian]      | ZR  | Tangent                           | 90   |
|                      | Optional  | ATN     | Inverse-tangent assignment variable | Operand               | ZR  | Inverse tangent                   | 90   |
|                      | Optional  | SQR     | Root assignment variable            | Operand               | ZR  | Root                              | 91   |
| Logical operation    | Optional  | AND     | AND operand variable                | Operand               | ZR  | Logical AND                       | 92   |
|                      | Optional  | OR      | OR operand variable                 | Operand               | ZR  | Logical OR                        | 93   |
|                      | Optional  | EOR     | Exclusive-OR operand variable       | Operand               | ZR  | Logical exclusive-OR              | 94   |
| Comparison           | Optional  | CPXX    | Comparison variable                 | Comparison value      | <u>EQ, NE,</u><br><u>GT, GE,</u><br><u>LT, LE</u> | Compare                           | 95   |
| Timer                | Optional  | TIMW    | Wait time (sec)                     | Prohibited            | TU  | Wait                              | 96   |
|                      | Optional  | TIMC    | Program number                      | Prohibited            | CP  | Cancel waiting                    | 97   |
|                      | Optional  | GTTM    | Time assignment variable            | Prohibited            | CP  | Get time                          | 98   |
| I/O, flag operation  | Optional  | BTXX    | Start output, flag                  | (End output, flag)    | CP  | Output, flag [ON, OF, NT]         | 99   |
|                      | Optional  | BTPN    | Output port, flag                   | Timer setting         | CP  | Output ON pulse                   | 100  |
|                      | Optional  | BTPF    | Output port, flag                   | Timer setting         | CP  | Output OFF pulse                  | 101  |
|                      | Optional  | WTXX    | I/O, flag                           | (Wait time)           | TU  | Wait for I/O, flag [ON, OF]       | 102  |
|                      | Optional  | IN      | Head I/O, flag                      | End I/O, flag         | CC  | Input binary (32 bits max.)       | 103  |
|                      | Optional  | INB     | Head I/O, flag                      | Conversion digits     | CC  | Input BCD (8 digits max.)         | 104  |
|                      | Optional  | OUT     | Head output, flag                   | End I/O, flag         | CC  | Output binary (32 bits max.)      | 105  |
|                      | Optional  | OUTB    | Head output, flag                   | Conversion digits     | CC  | Output BCD (8 digits max.)        | 106  |
|                      | Optional  | FMIO    | Format type                         | Prohibited            | CP  | Set IN (B)/OUT (B) command format | 107  |



Operation type in the output field

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

| Category                     | Condition  | Command | Operand 1                               | Operand 2                              | Output                | Function                                   | Page |
|------------------------------|------------|---------|---|--|-----------------------|--|------|
| Program control              | Optional   | GOTO    | Jump-destination tag number             | Prohibited                             | CP                    | Jump                                       | 110  |
|                              | Prohibited | TAG     | Declaration tag number                  | Prohibited                             | CP                    | Declare jump destination                   | 110  |
|                              | Optional   | EXSR    | Execution subroutine number             | Prohibited                             | CP                    | Execute subroutine                         | 111  |
|                              | Prohibited | BGSR    | Declaration subroutine number           | Prohibited                             | CP                    | Start subroutine                           | 111  |
|                              | Prohibited | EDSR    | Prohibited                              | Prohibited                             | CP                    | End subroutine                             | 112  |
| Task management              | Optional   | EXIT    | Prohibited                              | Prohibited                             | CP                    | End program                                | 113  |
|                              | Optional   | EXPG    | Execution program number                | (Execution program number)             | CC                    | Start program                              | 114  |
|                              | Optional   | ABPG    | Stop program number                     | (Stop program number)                  | CC                    | Stop other program                         | 115  |
|                              | Optional   | SSPG    | Pause program number                    | (Pause program number)                 | CC                    | Pause program                              | 116  |
|                              | Optional   | RSPG    | Resumption program number               | (Resumption program number)            | CC                    | Resume program                             | 117  |
| Position operation           | Optional   | PGET    | Axis number                             | Position number                        | CC                    | Assign position to variable 199            | 118  |
|                              | Optional   | PPUT    | Axis number                             | Position number                        | CP                    | Assign value of variable 199               | 119  |
|                              | Optional   | PCLR    | Start position number                   | End position number                    | CP                    | Clear position data                        | 120  |
|                              | Optional   | PCPY    | Copy-destination position number        | Copy-source position number            | CP                    | Copy position data                         | 121  |
|                              | Optional   | PRED    | Read axis pattern                       | Save-destination position number       | CP                    | Read current axis position                 | 122  |
|                              | Optional   | PRDQ    | Axis number                             | Variable number                        | CP                    | Read current axis position (1 axis direct) | 123  |
|                              | Optional   | PTST    | Confirmation axis pattern               | Confirmation position number           | CC                    | Confirm position data                      | 124  |
|                              | Optional   | PVEL    | Speed [mm/sec]                          | Assignment-destination position number | CP                    | Assign position speed                      | 125  |
|                              | Optional   | PACC    | Acceleration [G]                        | Assignment-destination position number | CP                    | Assign position acceleration               | 126  |
|                              | Optional   | PDCL    | Deceleration [G]                        | Assignment-destination position number | CP                    | Assign position deceleration               | 127  |
|                              | Optional   | PAXS    | Axis-pattern assignment variable number | Position number                        | CP                    | Read axis pattern                          | 128  |
|                              | Optional   | PSIZ    | Size assignment variable number         |  | CP                    | Confirm position size                      | 129  |
|                              | Optional   | GVEL    | Variable number                         | Position number                        | CP                    | Get speed data                             | 130  |
|                              | Optional   | GACC    | Variable number                         | Position number                        | CP                    | Get acceleration data                      | 131  |
|                              | Optional   | GDCL    | Variable number                         | Position number                        | CP                    | Get deceleration data                      | 132  |
| Actuator control declaration | Optional   | VEL     | Speed [mm/sec]                          | Prohibited                             | CP                    | Set speed                                  | 133  |
|                              | Optional   | OVRD    | Speed ratio [%]                         | Prohibited                             | CP                    | Set speed coefficient                      | 134  |
|                              | Optional   | ACC     | Acceleration [G]                        | Prohibited                             | CP                    | Set acceleration                           | 135  |
|                              | Optional   | DCL     | Deceleration [G]                        | Prohibited                             | CP                    | Set deceleration                           | 136  |
|                              | Optional   | SCRV    | Ratio [%]                               | Prohibited                             | CP                    | Set sigmoid motion ratio                   | 137  |
|                              | Optional   | OFST    | Setting axis pattern                    | Offset value [mm]                      | CP                    | Set offset                                 | 138  |
|                              | Optional   | DEG     | Division angle [deg]                    | Prohibited                             | CP                    | Set division angle                         | 139  |
|                              | Optional   | BASE    | Reference axis number                   | Prohibited                             | CP                    | Set reference axis                         | 140  |
|                              | Optional   | GRP     | Valid axis pattern                      | Prohibited                             | CP                    | Set group axes                             | 141  |
|                              | Optional   | HOLD    | (Input port to pause)                   | (HOLD type)                            | CP                    | Declare port to pause                      | 142  |
|                              | Optional   | CANC    | (Input port to abort)                   | (CANC type)                            | CP                    | Declare port to abort                      | 143  |
|                              | Optional   | VLMX    | Prohibited                              | Prohibited                             | CP                    | Specify VLMX speed                         | 144  |
|                              | Optional   | DIS     | Distance                                | Prohibited                             | CP                    | Set spline division distance               | 145  |
|                              | Optional   | POTP    | 0 or 1                                  | Prohibited                             | CP                    | Set PATH output type                       | 146  |
|                              | Optional   | PAPR    | Distance                                | Speed                                  | CP                    | Set PUSH command distance, speed           | 147  |
| Optional                     | QRTN       | 0 or 1  | Prohibited                              | CP                                     | Set quick-return mode | 148  |      |



Operation type in the output field

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

| Category                 | Condition  | Command           | Operand 1                   | Operand 2                        | Output                        | Function   | Page |
|--------------------------|--|-------------------|-----------------------------|----------------------------------|-------------------------------|--|------|
| Actuator control command | Optional   | SVXX              | Operation axis pattern      | Prohibited                       | PE                            | Servo [ON, OF]   | 149  |
|                          | Optional   | HOME              | Home-return axis pattern    | Prohibited                       | PE                            | Return to home   | 150  |
|                          | Optional   | MOVP              | Destination position number | Prohibited                       | PE                            | Move to specified position   | 151  |
|                          | Optional   | MOVL              | Destination position number | Prohibited                       | PE                            | Move to specified position via interpolation                             | 152  |
|                          | Optional   | MVPI              | Travel position number      | Prohibited                       | PE                            | Move to relative position  | 153  |
|                          | Optional   | MVLI              | Travel position number      | Prohibited                       | PE                            | Move to relative position via interpolation                              | 154  |
|                          | Optional   | PATH              | Start position number       | End position number              | PE                            | Move along path  | 155  |
|                          | Optional   | JXWX              | Axis operation pattern      | Start I/O, flag                  | PE                            | Jog [FN, FF, BN, BF]   | 156  |
|                          | Optional   | STOP              | Axis stop pattern           | Prohibited                       | CP                            | Decelerate and stop axis   | 157  |
|                          | Optional   | PSPL              | Start position number       | End position number              | PE                            | Move along spline  | 158  |
|                          | Optional   | PUSH              | Target position number      | Prohibited                       | PE                            | Move by push motion  | 159  |
|                          | Optional   | CIR2              | Passing position 1 number   | Passing position 2 number        | PE                            | Move along circle 2 (arc interpolation)                                  | 161  |
|                          | Optional   | ARC2              | Passing position number     | End position number              | PE                            | Move along arc 2 (arc interpolation)                                     | 162  |
|                          | Optional   | CIRS              | Passing position 1 number   | Passing position 2 number        | PE                            | Move three-dimensionally along circle                                    | 163  |
|                          | Optional   | ARCS              | Passing position number     | Passing position number          | PE                            | Move three-dimensionally along arc                                       | 164  |
|                          | Optional   | CHVL              | Axis pattern                | Speed                            | CP                            | Change speed   | 165  |
|                          | Optional   | ARCD              | End position number         | Center angle [deg]               | PE                            | Move along arc via specification of end position and center angl         | 166  |
|                          | Optional   | ARCC              | Center position number      | Center angle [deg]               | PE                            | Move along arc via specification of center position and center angle     | 167  |
|                          | Optional   | PBND              | Axis pattern                | Distance                         | CP                            | Set positioning band   | 168  |
|                          | Optional   | CIR               | Passing position 1 number   | Passing position 2 number        | PE                            | Move along circle (CIR2 is recommended)                                  | 169  |
|                          | Optional   | ARC               | Passing position number     | End position number              | PE                            | Move along arc (ARC2 is recommended)                                     | 170  |
|                          | Refer to the page on palletizing for commands relating to arch motion. |                   |                             |                                  |                               |  |      |
| Optional                 | ARCH   | Position number   | Position number             | PE                               | Arch motion                   | 220  |      |
| Optional                 | ACHZ   | Axis number       | Prohibited                  | CP                               | Declare arch-motion Z-axis    | 210  |      |
| Optional                 | ATRG   | Position number   | Position number             | CP                               | Set arch trigger              | 211  |      |
| Optional                 | AEXT   | (Position number) | Prohibited                  | CP                               | Set arch-motion composition   | 212  |      |
| Optional                 | OFAZ   | Offset value      | Prohibited                  | CP                               | Set arch-motion Z-axis offset | 212  |      |
| Structural IF            | Optional   | IFXX              | Comparison variable         | Comparison value                 | CP                            | Compare [EQ, NE, GT, GE, LT, LE]   | 171  |
|                          | Optional   | ISXX              | Column number               | Column number, character literal | CP                            | Compare strings  | 172  |
|                          | Prohibited   | ELSE              | Prohibited                  | Prohibited                       | CP                            | Declare execution destination when IF command condition is not satisfied | 173  |
|                          | Prohibited   | EDIF              | Prohibited                  | Prohibited                       | CP                            | Declare end of IF  | 173  |
| Structural DO            | Optional   | DWXX              | Comparison variable         | Comparison value                 | CP                            | Loop [EQ, NE, GT, GE, LT, LE]  | 174  |
|                          | Optional   | LEAV              | Prohibited                  | Prohibited                       | CP                            | Pull out from DO   | 174  |
|                          | Optional   | ITER              | Prohibited                  | Prohibited                       | CP                            | Repeat DO  | 175  |
|                          | Prohibited   | EDDO              | Prohibited                  | Prohibited                       | CP                            | Declare end of DO  | 175  |
| Multi-branching          | Optional   | SLCT              | Prohibited                  | Prohibited                       | CP                            | Declare start of multi-branching   | 176  |
|                          | Prohibited   | WHXX              | Comparison variable         | Comparison value                 | CP                            | Branch value [EQ, NE, GT, GE, LT, LE]                                    | 177  |
|                          | Prohibited   | WSXX              | Column number               | Column number, character literal | CP                            | Branch character string [EQ, NE]   | 178  |
|                          | Prohibited   | OTHE              | Prohibited                  | Prohibited                       | CP                            | Declare branching destination when condition is not satisfied            | 179  |
|                          | Prohibited   | EDSL              | Prohibited                  | Prohibited                       | CP                            | Declare end of SLCT  | 179  |



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                            | 180  |
|                                | Optional  | PGST                    | Variable number | Program number                   | CP         | Get program status                         | 181  |
|                                | Optional  | SYST                    | Variable number | Prohibited                       | CP         | Get system status                          | 182  |
| Zone                           | Optional  | WZNA                    | Zone number     | Axis pattern                     | CP         | Wait for zone ON, with AND                 | 183  |
|                                | Optional  | WZNO                    | Zone number     | Axis pattern                     | CP         | Wait for zone ON, with OR                  | 184  |
|                                | Optional  | WZFA                    | Zone number     | Axis pattern                     | CP         | Wait for zone OFF, with AND                | 185  |
|                                | Optional  | WZFO                    | Zone number     | Axis pattern                     | CP         | Wait for zone OFF, with OR                 | 186  |
| Communication                  | Optional  | OPEN                    | Channel number  | Prohibited                       | CP         | Open channel                               | 187  |
|                                | Optional  | CLOS                    | Channel number  | Prohibited                       | CP         | Close channel                              | 187  |
|                                | Optional  | READ                    | Channel number  | Column number                    | CC         | Read from channel                          | 188  |
|                                | Optional  | TMRD                    | Timer setting   | Prohibited                       | CP         | Set READ timeout value                     | 189  |
|                                | Optional  | WRIT                    | Channel number  | Column number                    | CP         | Output to channel                          | 190  |
|                                | Optional  | SCHA                    | Character code  | Prohibited                       | CP         | Set end character                          | 191  |
| String operation               | Optional  | SCPY                    | Column number   | Column number, character literal | CC         | Copy character string                      | 192  |
|                                | Optional  | SCMP                    | Column number   | Column number, character literal | EQ         | Compare character strings                  | 193  |
|                                | Optional  | SGET                    | Variable number | Column number, character literal | CP         | Get character                              | 194  |
|                                | Optional  | SPUT                    | Column number   | Data                             | CP         | Set character                              | 195  |
|                                | Optional  | STR                     | Column number   | Data                             | CC         | Convert character string; decimal          | 196  |
|                                | Optional  | STRH                    | Column number   | Data                             | CC         | Convert character string; hexadecimal      | 197  |
|                                | Optional  | VAL                     | Variable number | Column number, character literal | CC         | Convert character string data; decimal     | 198  |
|                                | Optional  | VALH                    | Variable number | Column number, character literal | CC         | Convert character string data; hexadecimal | 199  |
| Optional                       | SLEN      | Character string length | Prohibited      | CP                               | Set length | 200  |      |



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                                     | 201     |
|                                | Prohibited   | EDPA               | Prohibited          | Prohibited      | CP   | Declare end of palletizing setting                                       | 201     |
|                                | Optional   | PAPI               | Count               | Count           | CP   | Set palletizing counts   | 202     |
|                                | Optional   | PAPN               | Pattern number      | Prohibited      | CP   | Set palletizing pattern  | 202     |
|                                | Optional   | PASE               | Axis number         | Axis number     | CP   | Set palletizing axes   | 203     |
|                                | Optional   | PAPT               | Pitch               | Pitch           | CP   | Set palletizing pitches  | 203     |
|                                | Optional   | PAST               | (Position number)   | Prohibited      | CP   | Set palletizing reference point  | 204     |
|                                | Optional   | PAPS               | Position number     | Prohibited      | CP   | Set 3 palletizing points for teaching                                    | 205     |
|                                | Optional   | PSLI               | Offset amount       | (Count)         | CP   | Set zigzag   | 206     |
|                                | Optional   | PCHZ               | (Axis number)       | Prohibited      | CP   | Set palletizing Z-axis   | 207     |
|                                | Optional   | PTRG               | Position number     | Position number | CP   | Set palletizing arch triggers  | 208     |
|                                | Optional   | PEXT               | (Position number)   | Prohibited      | CP   | Set palletizing composition  | 209     |
|                                | Optional   | OFPZ               | Offset amount       | Prohibited      | CP   | Set palletizing Z-axis offset  | 209     |
|                                | Optional   | ACHZ               | Axis number         | Prohibited      | CP   | Declare arch-motion Z-axis   | 210     |
|                                | Optional   | ATRG               | Position number     | Position number | CP   | Set arch triggers  | 211     |
|                                | Optional   | AEXT               | (Position number)   | Prohibited      | CP   | Set arch-motion composition  | 212     |
|                                | Optional   | OFAZ               | Offset amount       | Prohibited      | CP   | Set arch-motion Z-axis offset  | 212     |
|                                | Optional   | PTNG               | Palletizing number  | Variable number | CP   | Get palletizing position number  | 213     |
|                                | Optional   | PINC               | Palletizing number  | Prohibited      | CC   | Increment palletizing position number by 1                               | 213     |
|                                | Optional   | PDEC               | Palletizing number  | Prohibited      | CC   | Decrement palletizing position number by 1                               | 214     |
|                                | Optional   | PSET               | Palletizing number  | Data            | CC   | Set palletizing position number directly                                 | 214     |
|                                | Optional   | PARG               | Palletizing number  | Axis number     | CP   | Get palletizing angle  | 215     |
|                                | Optional   | PAPG               | Palletizing number  | Position number | CP   | Get palletizing calculation data   | 215     |
| Optional                       | PMVP   | Palletizing number | (Position number)   | PE              | Move to palletizing points via PTP           | 216  |         |
| Optional                       | PMVL   | Palletizing number | (Position number)   | PE              | Move to palletizing points via interpolation | 217  |         |
| Optional                       | PACH   | Palletizing number | Position number     | PE              | Palletizing-point arch motion                | 218  |         |
| Optional                       | ARCH   | Position number    | Position number     | PE              | Arch motion                                  | 220  |         |
| Building of pseudo-ladder task | Extension conditions LD (LOAD), A (AND), O (OR), AB (AND BLOCK) and OB (OR BLOCK) are supported. |                    |                     |                 |  |  |         |
|                                | Optional   | CHPR               | 0 or 1              | Prohibited      | CP   | Change task level  | 222     |
|                                | Prohibited   | TPCD               | 0 or 1              | Prohibited      | CP   | Specify processing to be performed when input condition is not specified | 222     |
|                                | Prohibited   | TSLP               | Time                | Prohibited      | CP   | Task sleep   | 223     |
|                                | Optional   | OUTR               | Output, flag number | Prohibited      | CP   | Output relay for ladder  | See 245 |
|                                | Optional   | TIMR               | Local flag number   | Timer setting   | CP   | Timer relay for ladder   | See 245 |



## 1.2 List of Commands in Alphabetical Order

Operation type in the output field

CC: Command was executed successfully

ZR: Operation result is zero

PE: Operation is complete

CP: Command part has passed

TU: Time up

EQ: Operand 1 = Operand 2

NE: Operand 1 ≠ Operand 2

GT: Operand 1 > Operand 2

GE: Operand 1 ≥ Operand 2

LT: Operand 1 < Operand 2

LE: Operand 1 ≤ Operand 2

| Command  | Page | Condition  | Operation1                          | Operation2                | Output | Function   |
|----------|------|------------|-------------------------------------|---------------------------|--------|--|
| <b>A</b> |      |            |                                     |                           |        |  |
| ABPG     | 115  | Optional   | Stop program number                 | (Stop program number)     | CC     | Stop other program   |
| ACC      | 135  | Optional   | Acceleration                        | Prohibited                | CP     | Set acceleration   |
| ACHZ     | 210  | Optional   | Axis number                         | Prohibited                | CP     | Declare arch-motion Z-axis   |
| ADD      | 86   | Optional   | Augend variable                     | Addend                    | ZR     | Add  |
| AEXT     | 212  | Optional   | (Position number)                   | Prohibited                | CP     | Set arch-motion composition  |
| AND      | 92   | Optional   | AND operand variable                | Operand                   | ZR     | Logical AND  |
| ARC      | 170  | Optional   | Passing position number             | End position number       | PE     | Move along arc   |
| ARC2     | 162  | Optional   | Passing position number             | End position number       | PE     | Move along arc 2   |
| ARCC     | 167  | Optional   | Center position number              | Center angle              | PE     | Move along arc via specification of center position and center angle |
| ARCD     | 166  | Optional   | End position number                 | Center angle              | PE     | Move along arc via specification of end position and center angle    |
| ARCH     | 220  | Optional   | Position number                     | Position number           | PE     | Arch motion  |
| ARCS     | 164  | Optional   | Passing position number             | Passing position number   | PE     | Move three-dimensionally along arc                                   |
| ATN      | 90   | Optional   | Inverse-tangent assignment variable | Operand                   | ZR     | Inverse tangent  |
| ATRG     | 211  | Optional   | Position number                     | Position number           | CP     | Set arch trigger   |
| AXST     | 180  | Optional   | Variable number                     | Axis number               | CP     | Get axis status  |
| <b>B</b> |      |            |                                     |                           |        |  |
| BASE     | 140  | Optional   | Reference axis number               | Prohibited                | CP     | Set reference axis   |
| BGPA     | 201  | Optional   | Palletizing number                  | Prohibited                | CP     | Declare start of palletizing setting                                 |
| BGSR     | 111  | Prohibited | Declaration subroutine number       | Prohibited                | CP     | Start subroutine   |
| BTPF     | 101  | Optional   | Output port, flag                   | Timer setting             | CP     | Output OFF pulse   |
| BTPN     | 100  | Optional   | Output port, flag                   | Timer setting             | CP     | Output ON pulse  |
| BTXX     | 99   | Optional   | Start output, flag                  | (End output, flag)        | CP     | Output, flag [ON, OF, NT]  |
| <b>C</b> |      |            |                                     |                           |        |  |
| CANC     | 143  | Optional   | (Input port to abort)               | (CANC type)               | CP     | Declare port to abort  |
| CHPR     | 222  | Optional   | 0 or 1                              | Prohibited                | CP     | Change task level  |
| CHVL     | 165  | Optional   | Axis pattern                        | Speed                     | CP     | Change speed   |
| CIR      | 169  | Optional   | Passing position 1 number           | Passing position 2 number | PE     | Move along circle  |
| CIR2     | 161  | Optional   | Passing position 1 number           | Passing position 2 number | PE     | Move along circle 2  |
| CIRS     | 163  | Optional   | Passing position 1 number           | Passing position 2 number | PE     | Move three-dimensionally along circle                                |
| CLOS     | 187  | Optional   | Channel number                      | Prohibited                | CP     | Close channel  |
| CLR      | 85   | Optional   | Start-of-clear variable             | End-of-clear variable     | ZR     | Clear variable   |
| COS      | 89   | Optional   | Cosine assignment variable          | Operand                   | ZR     | Cosine   |
| CPXX     | 95   | Optional   | Comparison variable                 | Comparison value          |        | Compare  |
| <b>D</b> |      |            |                                     |                           |        |  |
| DCL      | 136  | Optional   | Deceleration                        | Prohibited                | CP     | Set deceleration   |
| DEG      | 139  | Optional   | Division angle                      | Prohibited                | CP     | Set division angle   |
| DIS      | 145  | Optional   | Distance                            | Prohibited                | CP     | Set spline division distance   |
| DIV      | 87   | Optional   | Dividend variable                   | Divisor                   | ZR     | Divide   |
| DWXX     | 174  | Optional   | Comparison variable                 | Comparison value          | CP     | Loop [EQ, NE, GT, GE, LT, LE]  |



Operation type in the output field

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

| Command  | Page | Condition  | Operation1                    | Operation2                       | Output | Function   |
|----------|------|------------|-------------------------------|----------------------------------|--------|--|
| <b>E</b> |      |            |                               |                                  |        |  |
| EDDO     | 175  | Prohibited | Prohibited                    | Prohibited                       | CP     | Declare end of DO  |
| EDIF     | 173  | Prohibited | Prohibited                    | Prohibited                       | CP     | Declare end of IF  |
| EDPA     | 201  | Prohibited | Prohibited                    | Prohibited                       | CP     | Declare end of palletizing setting                                       |
| EDSL     | 179  | Prohibited | Prohibited                    | Prohibited                       | CP     | Declare end of SLCT  |
| EDSR     | 112  | Prohibited | Prohibited                    | Prohibited                       | CP     | End subroutine   |
| ELSE     | 173  | Prohibited | Prohibited                    | Prohibited                       | CP     | Declare execution destination when IF command condition is not satisfied |
| EOR      | 94   | Optional   | Exclusive-OR operand variable | Operand                          | ZR     | Logical exclusive-OR   |
| EXIT     | 113  | Optional   | Prohibited                    | Prohibited                       | CP     | End program  |
| EXPG     | 114  | Optional   | Execution program number      | (Execution program number)       | CC     | Start program  |
| EXSR     | 111  | Optional   | Execution subroutine number   | Prohibited                       | CP     | Execute subroutine   |
| <b>F</b> |      |            |                               |                                  |        |  |
| FMIO     | 107  | Optional   | Format type                   | Prohibited                       | CP     | Set IN (B)/OUT (B) command format  |
| <b>G</b> |      |            |                               |                                  |        |  |
| GACC     | 131  | Optional   | Variable number               | Position number                  | CP     | Get acceleration data  |
| GDCL     | 132  | Optional   | Variable number               | Position number                  | CP     | Get deceleration data  |
| GOTO     | 110  | Optional   | Jump-destination tag number   | Prohibited                       | CP     | Jump   |
| GRP      | 141  | Optional   | Valid axis pattern            | Prohibited                       | CP     | Set group axes   |
| GTTM     | 98   | Optional   | Time assignment variable      | Prohibited                       | CP     | Get time   |
| GVEL     | 130  | Optional   | Variable number               | Position number                  | CP     | Get speed data   |
| <b>H</b> |      |            |                               |                                  |        |  |
| HOLD     | 142  | Optional   | (Input port to pause)         | (HOLD type)                      | CP     | Declare port to pause  |
| HOME     | 150  | Optional   | Home-return axis pattern      | Prohibited                       | PE     | Return to home   |
| <b>I</b> |      |            |                               |                                  |        |  |
| IFXX     | 171  | Optional   | Comparison variable           | Comparison value                 | CP     | Compare [EQ, NE, GT, GE, LT, LE]   |
| INB      | 104  | Optional   | Head I/O, flag                | Conversion digits                | CC     | Input BCD (8 digits max.)  |
| IN       | 103  | Optional   | Head I/O, flag                | End I/O, flag                    | CC     | Input binary (32 bits max.)  |
| ISXX     | 172  | Optional   | Column number                 | Column number, character literal | CP     | Compare strings  |
| ITER     | 175  | Optional   | Prohibited                    | Prohibited                       | CP     | Repeat DO  |
| <b>J</b> |      |            |                               |                                  |        |  |
| JXWX     | 156  | Optional   | Axis operation pattern        | Start I/O, flag                  | PE     | Jog [FN, FF, BN, BF]   |
| <b>L</b> |      |            |                               |                                  |        |  |
| LEAV     | 174  | Optional   | Prohibited                    | Prohibited                       | CP     | Pull out from DO   |
| LET      | 84   | Optional   | Assignment variable           | Assigned value                   | ZR     | Assign   |
| <b>M</b> |      |            |                               |                                  |        |  |
| MOD      | 88   | Optional   | Remainder assignment variable | Divisor                          | ZR     | Calculate remainder  |
| MOVL     | 152  | Optional   | Destination position number   | Prohibited                       | PE     | Move to specified position via interpolation                             |
| MOVVP    | 151  | Optional   | Destination position number   | Prohibited                       | PE     | Move to specified position   |
| MULT     | 87   | Optional   | Multiplicand variable         | Multiplier                       | ZR     | Multiply   |
| MVLI     | 154  | Optional   | Travel position number        | Prohibited                       | PE     | Move to relative position via interpolation                              |
| MVPI     | 153  | Optional   | Travel position number        | Prohibited                       | PE     | Move to relative position  |



Operation type in the output field

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

| Command  | Page | Condition  | Operation1                              | Operation2                             | Output | Function  |
|----------|------|------------|---|--|--------|---|
| <b>O</b> |      |            |   |  |        |   |
| OFAZ     | 212  | Optional   | Offset amount                           | Prohibited                             | CP     | Set arch-motion Z-axis offset                                 |
| OFMZ     | 209  | Optional   | Offset amount                           | Prohibited                             | CP     | Set palletizing Z-axis offset                                 |
| OFST     | 138  | Optional   | Setting axis pattern                    | Offset value                           | CP     | Set offset  |
| OPEN     | 187  | Optional   | Channel number                          | Prohibited                             | CP     | Open channel  |
| OR       | 93   | Optional   | OR operand variable                     | Operand                                | ZR     | Logical OR  |
| OTHE     | 179  | Prohibited | Prohibited                              | Prohibited                             | CP     | Declare branching destination when condition is not satisfied |
| OUT      | 105  | Optional   | Head output, flag                       | End I/O, flag                          | CC     | Output binary (32 bits max.)                                  |
| OUTB     | 106  | Optional   | Head output, flag                       | Conversion digits                      | CC     | Output BCD (8 digits max.)                                    |
| OUTR     | 245  | Optional   | Output, flag number                     | Prohibited                             | CP     | Output relay for ladder                                       |
| OVRD     | 134  | Optional   | Speed ratio                             | Prohibited                             | CP     | Set speed ratio   |
| <b>P</b> |      |            |   |  |        |   |
| PACC     | 126  | Optional   | Acceleration                            | Assignment-destination position number | CP     | Assign position acceleration                                  |
| PACH     | 218  | Optional   | Palletizing number                      | Position number                        | PE     | Palletizing-point arch motion                                 |
| PAPG     | 215  | Optional   | Palletizing number                      | Position number                        | CP     | Get palletizing calculation data                              |
| PAPI     | 202  | Optional   | Count                                   | Count                                  | CP     | Set palletizing counts  |
| PAPN     | 202  | Optional   | Pattern number                          | Prohibited                             | CP     | Set palletizing pattern                                       |
| PAPR     | 147  | Optional   | Distance                                | Prohibited                             | CP     | Set PUSH command distance, speed                              |
| PAPS     | 205  | Optional   | Position number                         | Prohibited                             | CP     | Set 3 palletizing points for teaching                         |
| PAPT     | 203  | Optional   | Pitch                                   | Pitch                                  | CP     | Set palletizing pitches                                       |
| PARG     | 215  | Optional   | Palletizing number                      | Axis number                            | CP     | Get palletizing angle   |
| PASE     | 203  | Optional   | Axis number                             | Axis number                            | CP     | Set palletizing axes  |
| PAST     | 204  | Optional   | (Position number)                       | Prohibited                             | CP     | Set palletizing reference point                               |
| PATH     | 155  | Optional   | Start position number                   | End position number                    | PE     | Move along path   |
| PAXS     | 128  | Optional   | Axis-pattern assignment variable number | Position number                        | CP     | Read axis pattern   |
| PBND     | 168  | Optional   | Axis pattern                            | Distance                               | CP     | Set positioning band  |
| PCHZ     | 207  | Optional   | (Axis number)                           | Prohibited                             | CP     | Set palletizing Z-axis  |
| PCLR     | 120  | Optional   | Start position number                   | End position number                    | CP     | Clear position data   |
| PCPY     | 121  | Optional   | Copy-destination position number        | Copy-source position number            | CP     | Copy position data  |
| PDCL     | 127  | Optional   | Deceleration                            | Assignment-destination position number | CP     | Assign position deceleration                                  |
| PDEC     | 214  | Optional   | Palletizing number                      | Prohibited                             | CC     | Decrement palletizing position number by 1                    |
| PEXT     | 209  | Optional   | (Position number)                       | Prohibited                             | CP     | Set palletizing composition                                   |
| PGET     | 118  | Optional   | Axis number                             | Position number                        | CC     | Assign position to variable 199                               |
| PGST     | 181  | Optional   | Variable number                         | Program number                         | CP     | Get program status  |
| PINC     | 213  | Optional   | Palletizing number                      | Prohibited                             | CC     | Increment palletizing position number by 1                    |
| PMVL     | 217  | Optional   | Palletizing number                      | (Position number)                      | PE     | Move to palletizing points via interpolation                  |
| PMVP     | 216  | Optional   | Palletizing number                      | (Position number)                      | PE     | Move to palletizing points via PTP                            |
| POTP     | 146  | Optional   | 0 or 1                                  | Prohibited                             | CP     | Set PATH output type  |
| PPUT     | 119  | Optional   | Axis number                             | Position number                        | CP     | Assign value of variable 199                                  |
| PRDQ     | 123  | Optional   | Axis number                             | Variable number                        | CP     | Read current axis position (1 axis direct)                    |
| PRED     | 122  | Optional   | Read axis pattern                       | Save-destination position number       | CP     | Read current axis position                                    |
| PSET     | 214  | Optional   | Palletizing number                      | Data                                   | CC     | Set palletizing position number directly                      |
| PSIZ     | 129  | Optional   | Size assignment variable number         |  | CP     | Confirm position size   |
| PSLI     | 206  | Optional   | Offset amount                           | (Count)                                | CP     | Set zigzag  |
| PSPL     | 158  | Optional   | Start position number                   | End position number                    | PE     | Move along spline   |
| PTNG     | 213  | Optional   | Palletizing number                      | Variable number                        | CP     | Get palletizing position number                               |



Operation type in the output field

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

| Command  | Page | Condition | Operation1                | Operation2                             | Output | Function                              |
|----------|------|-----------|---------------------------|--|--------|---------------------------------------|
| <b>P</b> |      |           |                           |  |        |                                       |
| PTRG     | 208  | Optional  | Position number           | Position number                        | CP     | Set palletizing arch triggers         |
| PTST     | 124  | Optional  | Confirmation axis pattern | Confirmation position number           | CP     | Confirm position data                 |
| PUSH     | 159  | Optional  | Target position number    | Prohibited                             | PE     | Move by push motion                   |
| PVEL     | 125  | Optional  | Speed                     | Assignment-destination position number | CP     | Assign position speed                 |
| <b>Q</b> |      |           |                           |  |        |                                       |
| QRTN     | 148  | Optional  | 0 or 1                    | Prohibited                             | CP     | Set quick-return mode                 |
| <b>R</b> |      |           |                           |  |        |                                       |
| READ     | 188  | Optional  | Channel number            | Column number                          | CC     | Read from channel                     |
| RSPG     | 117  | Optional  | Resumption program number | (Resumption program number)            | CC     | Resume program                        |
| <b>S</b> |      |           |                           |  |        |                                       |
| SCHA     | 191  | Optional  | Character code            | Prohibited                             | CP     | Set end character                     |
| SCMP     | 193  | Optional  | Column number             | Column number, character literal       | EQ     | Compare character strings             |
| SCPY     | 192  | Optional  | Column number             | Column number, character literal       | CC     | Copy character string                 |
| SCRV     | 137  | Optional  | Ratio                     | Prohibited                             | CP     | Set sigmoid motion ratio              |
| SGET     | 194  | Optional  | Variable number           | Column number, character literal       | CP     | Get character                         |
| SIN      | 89   | Optional  | Sine assignment variable  | Operand                                | ZR     | Sine                                  |
| SLCT     | 176  | Optional  | Prohibited                | Prohibited                             | CP     | Declare start of multi-branching      |
| SLEN     | 200  | Optional  | Character string length   | Prohibited                             | CP     | Set length                            |
| SPUT     | 195  | Optional  | Column number             | Data                                   | CP     | Set character                         |
| SQR      | 91   | Optional  | Root assignment variable  | Operand                                | ZR     | Root                                  |
| SSPG     | 116  | Optional  | Pause program number      | (Pause program number)                 | CC     | Pause program                         |
| STOP     | 157  | Optional  | Axis stop pattern         | Prohibited                             | CP     | Decelerate and stop axis              |
| STR      | 196  | Optional  | Column number             | Data                                   | CC     | Convert character string; decimal     |
| STRH     | 197  | Optional  | Column number             | Data                                   | CC     | Convert character string; hexadecimal |
| SUB      | 86   | Optional  | Minuend variable          | Subtrahend                             | ZR     | Subtract                              |
| SVXX     | 149  | Optional  | Operation axis pattern    | Prohibited                             | PE     | Servo [ON, OF]                        |
| SYST     | 182  | Optional  | Variable number           | Prohibited                             | CP     | Get system status                     |



Operation type in the output field

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

| Command  | Page | Condition  | Operation1                  | Operation2                       | Output | Function   |
|----------|------|------------|-----------------------------|----------------------------------|--------|--|
| <b>T</b> |      |            |                             |                                  |        |  |
| TAG      | 110  | Prohibited | Declaration tag number      | Prohibited                       | CP     | Declare jump destination   |
| TAN      | 90   | Optional   | Tangent assignment variable | Operand                          | ZR     | Tangent  |
| TIMC     | 97   | Optional   | Program number              | Prohibited                       | CP     | Cancel waiting   |
| TIMR     | 245  | Optional   | Local flag number           | Timer setting                    | CP     | Timer relay for ladder   |
| TIMW     | 96   | Optional   | Wait time                   | Prohibited                       | TU     | Wait   |
| TMRD     | 189  | Optional   | Timer setting               | Prohibited                       | CP     | Set READ timeout value   |
| TPCD     | 222  | Prohibited | 0 or 1                      | Prohibited                       | CP     | Specify processing to be performed when input condition is not specified |
| TRAN     | 84   | Optional   | Copy-destination variable   | Copy-source variable             | ZR     | Copy   |
| TSLP     | 223  | Prohibited | Time                        | Prohibited                       | CP     | Task sleep   |
| <b>V</b> |      |            |                             |                                  |        |  |
| VAL      | 198  | Optional   | Variable number             | Column number, character literal | CC     | Convert character string data; decimal                                   |
| VALH     | 199  | Optional   | Variable number             | Column number, character literal | CC     | Convert character string data; hexadecimal                               |
| VEL      | 133  | Optional   | Speed                       | Prohibited                       | CP     | Set speed  |
| VLMX     | 144  | Optional   | Prohibited                  | Prohibited                       | CP     | Specify VLMX speed   |
| <b>W</b> |      |            |                             |                                  |        |  |
| WHXX     | 177  | Prohibited | Comparison variable         | Comparison value                 | CP     | Branch value [EQ, NE, GT, GE, LT, LE]                                    |
| WRIT     | 190  | Optional   | Channel number              | Column number                    | CC     | Output to channel  |
| WSXX     | 178  | Prohibited | Column number               | Column number, character literal | CP     | Branch character string [EQ, NE]   |
| WTXX     | 102  | Optional   | I/O, flag                   | (Wait time)                      | TU     | Wait for I/O, flag [ON, OF]  |
| WZFA     | 185  | Optional   | Zone number                 | Axis pattern                     | CP     | Wait for zone OFF, with AND  |
| WZFO     | 186  | Optional   | Zone number                 | Axis pattern                     | CP     | Wait for zone OFF, with OR   |
| WZNA     | 183  | Optional   | Zone number                 | Axis pattern                     | CP     | Wait for zone ON, with AND   |
| WZNO     | 184  | Optional   | Zone number                 | Axis pattern                     | CP     | Wait for zone ON, with OR  |





- CLR (Clear variable)

| F 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).







- MOD (Remainder of division)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                    |           | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------------|-----------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1          | Operand 2 |                          |
| Optional                                  | Optional                       | MOD                     | Variable<br>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.<br>1 ( $7 \div 3 = 2$ 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).<br>1 ( $7 \div 3 = 2$ with a remainder of 1) will be assigned to variable 2. |



## 1-3 Function Operation

## ● SIN (Sine operation)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                    |           | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------------|-----------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1          | Operand 2 |                          |
| Optional                                  | Optional                       | SIN                     | Variable<br>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  $\times \pi \div 180$

[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  $30 \times \pi \div 180$  (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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                    |           | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------------|-----------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1          | Operand 2 |                          |
| Optional                                  | Optional                       | COS                     | Variable<br>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  $\times \pi \div 180$

[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  $60 \times \pi \div 180$  (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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                    |           | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------------|-----------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1          | Operand 2 |                          |
| Optional                                  | Optional                       | TAN                     | Variable<br>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  $\times \pi \div 180$

[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.<br>45 $\times \pi \div 180$ (radian)<br>(45° will be converted to radian and assigned to variable 101.) |
| LET  | 101 | 45       |   |
| 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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                    |           | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------------|-----------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1          | Operand 2 |                          |
| Optional                                  | Optional                       | ATN                     | Variable<br>number | Data      | ZR                       |

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

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

[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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                    |           | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------------|-----------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1          | Operand 2 |                          |
| Optional                                  | Optional                       | SQR                     | Variable<br>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).











## 1-6 Timer

## ● TIMW (Timer)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |           |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-----------|------------|--------------------------|
|   |                                | Command,<br>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).





## ● GTTM (Get time)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                    |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------------|------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1          | Operand 2  |                          |
| Optional                                  | Optional                       | GTTM                    | Variable<br>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). |

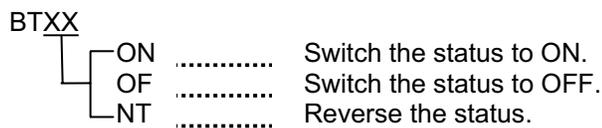


## 1-7 I/O, Flag Operation

## ● BTXX (Output port, flag operation)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |              |                | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------|----------------|--------------------------|
|   |                                | Command,<br>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.



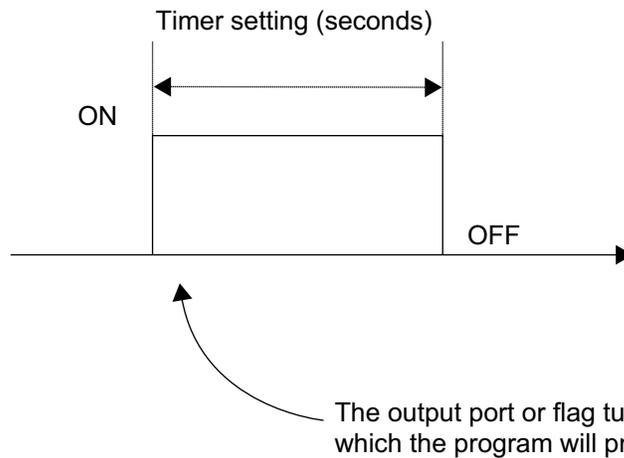
|             |      |     |     |   |
|-------------|------|-----|-----|---|
| [Example 1] | BTON | 316 |     | Turn ON output port 316.  |
| [Example 2] | BTOF | 316 | 323 | Turn OFF output ports 316 through 323.  |
| [Example 3] | LET  | 1   | 600 | Assign 600 to variable 1.   |
|             | BTNT | *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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                      |               | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|----------------------|---------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1            | Operand 2     |                          |
| Optional                                  | Optional                       | BTPN                    | Output port,<br>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).



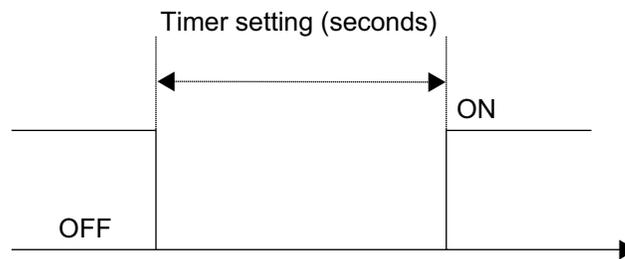
- (Note 1) If this command is executed with respect to an output port or flag already ON, the output port or flag will be turned OFF upon elapse of the timer setting.
- (Note 2) If the program ends after the command has been executed but before the timer is up, the output port or flag will not be turned 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    316    1    Turn ON output port 316 for 1 second.  
                          BTPN    600    10    Turn ON flag 600 for 10 seconds.

## ● BTPF (Output OFF pulse)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                      |               | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|----------------------|---------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1            | Operand 2     |                          |
| Optional                                  | Optional                       | BTPF                    | Output port,<br>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).



The output port or flag turns OFF here, after which the program will proceed to the next step.

- (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 | 316 | 1  | Turn OFF output port 316 for 1 second. |
| BTPF | 600 | 10 | Turn OFF flag 600 for 10 seconds.      |

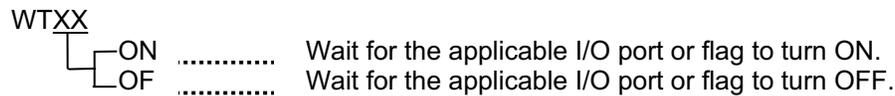


- WTXX (Wait for I/O port, flag)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |           |           | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-----------|-----------|--------------------------|
|   |                                | Command,<br>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).

Note) A local flag cannot be entered in operand 1.



|             |      |     |     |   |
|-------------|------|-----|-----|---|
| [Example 1] | WTON | 16  |     | Wait for input port 16 to turn ON.  |
| [Example 2] | WTOF | 324 | 10  | Wait for 10 seconds for output port 324 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   | 18  | Assign 18 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 18) to turn OFF. |

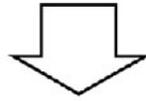


- IN (Read I/O, flag as binary)

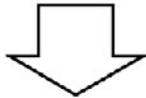
| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |           |           | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-----------|-----------|--------------------------|
|   |                                | Command,<br>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            |
| 23    | 22    | 21    | 20    | 19    | 18    | 17    | 16    | ... | 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 | + | $2^2$ | +      | 0 | + | $2^0$ |   |     |
| 128   | + | 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.

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

[Example 1]      IN      16      23      Read input ports 16 through 23, to variable 99 as a binary.

[Example 2]      LET      1      16      Assign 16 to variable 1.  
                   LET      2      23      Assign 23 to variable 2.  
                   IN      \*1      \*2      Read the input ports from the content of variable 1 (input port 16) through the content of variable 2 (input port 23), to variable 99 as a binary.

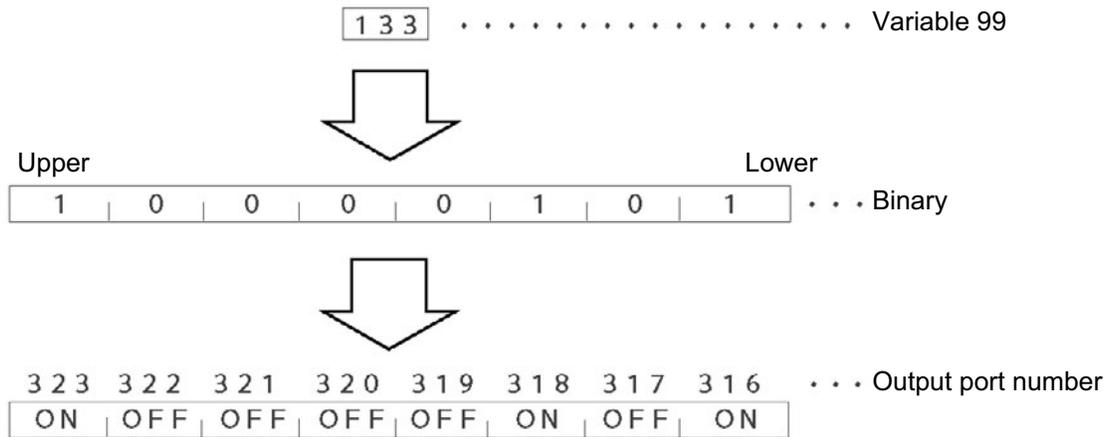




● OUT (Write output, flag as binary)

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

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



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

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

[Example 1]        OUT     316     323     Write the value in variable 99 to output ports 316 through 323 as a binary.

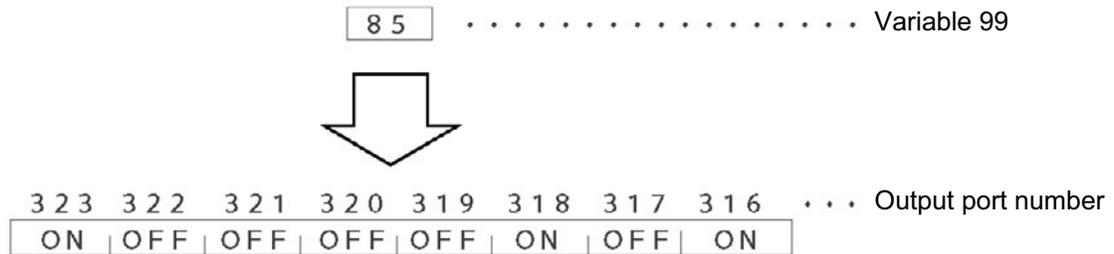
[Example 2]        LET     1       316     Assign 316 to variable 1.  
                       LET     2       323     Assign 323 to variable 2.  
                       OUT     \*1      \*2     Write the value in variable 99 to the output ports from the content of variable 1 (output port 316) through the content of variable 2 (output port 323) as a binary.



● OUTB (Write output, flag as BCD)

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

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



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

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

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

[Example 1]            OUTB    316        2            Write the value in variable 99 to the output ports from 316 for two digits (until output port 327) as a BCD.

[Example 2]            LET        1            316        Assign 316 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 316) for the content of variable 2 (two digits) (until output port 323) as a BCD.

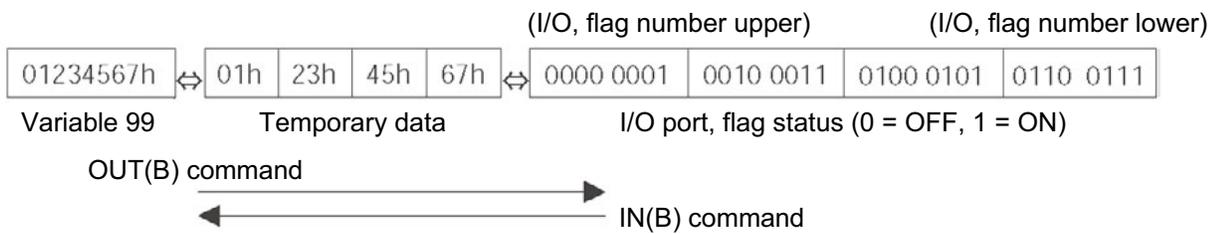


● FMIO (Set IN, INB, OUT, OUTB command format)

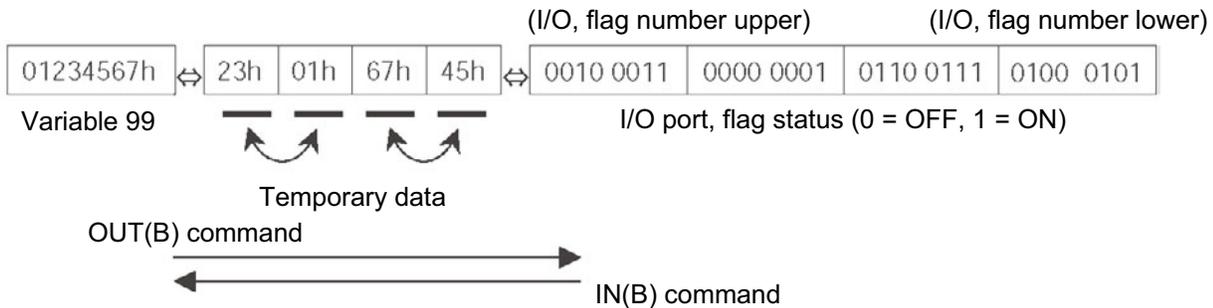
| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |             |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-------------|------------|--------------------------|
|   |                 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

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

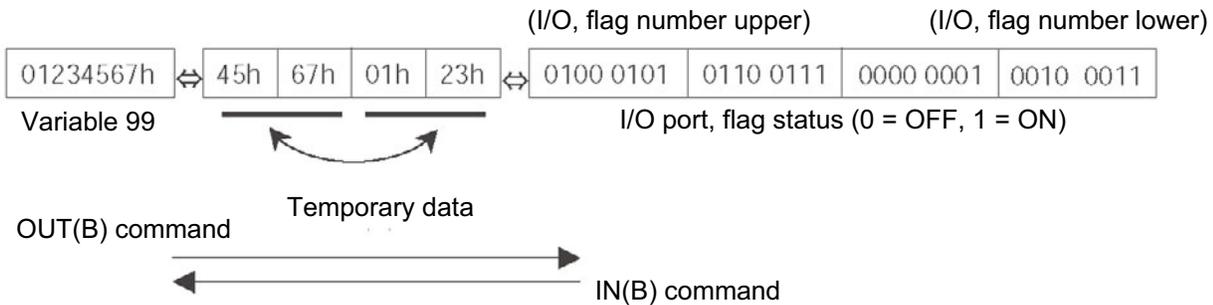
- [1] Operand 1 = 0 (Default status when a FMIO command has not been executed)  
Data is read or written without being reversed.



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

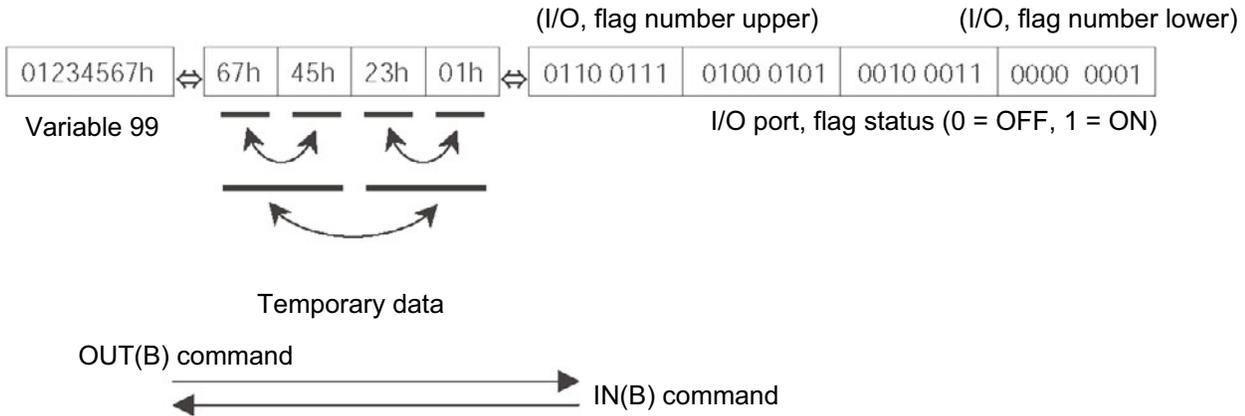


- [3] Operand 1 = 2  
Data is read or written after its upper 16 bits and lower 16 bits are reversed every 32 bits.



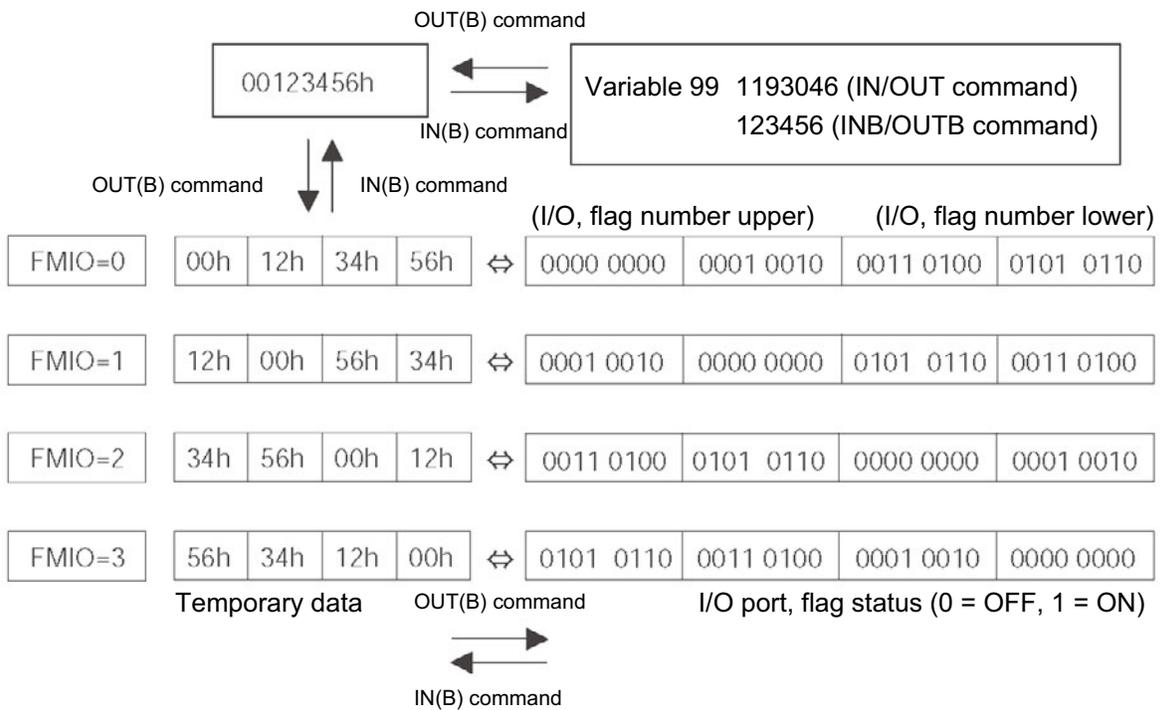


- [4] Operand 1 = 3  
Data is read or written after its upper 16 bits and lower 16 bits are reversed every 32 bits and its upper eight bits and lower eight bits are reversed every 16 bits.



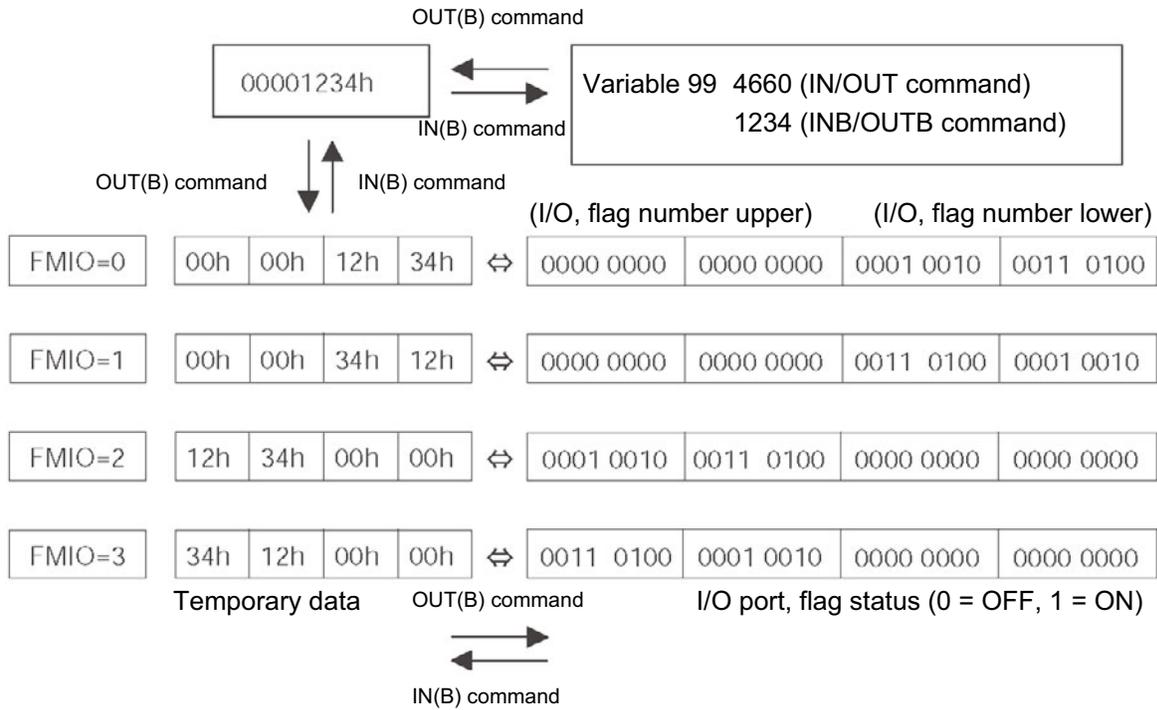
(Note) FMIO command is supported in main application version 0.56 or later, PC software version 2.0.45 or later and teaching pendant version 1.13 or later.

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

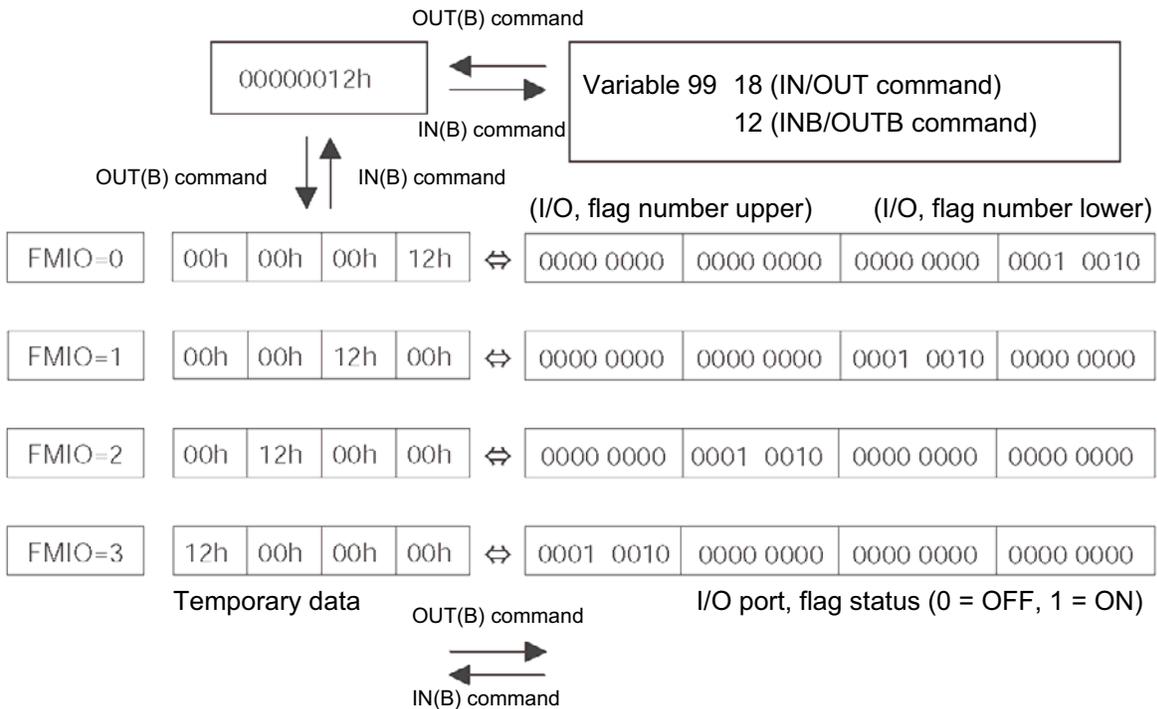




[Example 2] Variable 99 = 00001234h (Decimal: 4660, BCD: 1234)



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





## 1-8 Program Control

## ● GOTO (Jump)

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

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

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

[Example 1]

|      |   |                |
|------|---|----------------|
| TAG  | 1 | Set a tag.     |
| ⋮    |   |                |
| ⋮    |   |                |
| ⋮    |   |                |
| GOTO | 1 | Jump to tag 1. |

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

Since the maximum number of nests is defined for each conditional branching command or subroutine call, a nest will be infinitely repeated if an 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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |            |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|------------|------------|--------------------------|
|   |                                | Command,<br>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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                      |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|----------------------|------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1            | Operand 2  |                          |
| Optional                                  | Optional                       | EXSR                    | Subroutine<br>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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                      |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|----------------------|------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1            | Operand 2  |                          |
| Prohibited                                | Prohibited                     | BGSR                    | Subroutine<br>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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |            |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|------------|------------|--------------------------|
|   |                                | Command,<br>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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |            |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|------------|------------|--------------------------|
|   |                                | Command,<br>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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                   |                      | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-------------------|----------------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1         | Operand 2            |                          |
| Optional                                  | Optional                       | EXPG                    | Program<br>number | (Program<br>number ) | CC                       |

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

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

Error-generation/output-operation conditions

When one EXPG program is specified (only operand 1 is specified)

| Status of the<br>specified program | No program number error *1               |                        |  | Program number<br>error *1       |
|------------------------------------|--|------------------------|--|----------------------------------|
|                                    | Program already registered               |                        | Program not yet<br>registered                          |                                  |
|                                    | Program running                          | Program not<br>running |  |                                  |
| Error                              | A57<br>"Multiple program<br>start error" | None                   | C03<br>"Non-registered program<br>specification error" | C2C<br>"Program number<br>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<br>specified program | No program number error *2                                 |  |  | Program<br>number error *1       |
|------------------------------------|--|--|--|----------------------------------|
|                                    | Registered program exists inside the<br>specified range *3 |  | None of programs inside<br>the specified range are<br>registered |                                  |
|                                    | Running program<br>exists inside the<br>specified range    | None of programs<br>inside the<br>specified range<br>are running |  |                                  |
| Error                              | A57<br>"Multiple program<br>start error"                   | None   | C03<br>"Non-registered program<br>specification error"           | C2C<br>"Program<br>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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                   |                  | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-------------------|------------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1         | Operand 2        |                          |
| Optional                                  | Optional                       | ABPG                    | Program<br>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<br>specified program | No program number error *1 |                        |                               | Program number error<br>*1    |
|------------------------------------|----------------------------|------------------------|-------------------------------|-------------------------------|
|                                    | Program already registered |                        | Program not yet<br>registered |                               |
|                                    | Program running            | Program not<br>running |                               |                               |
| Error                              | None                       | None                   | None                          | C2C<br>"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<br>specified program | No program number error *3                                 |  |  | Program<br>number error *1       |
|------------------------------------|--|--|--|----------------------------------|
|                                    | Registered program exists inside the<br>specified range *4 |  | None of programs inside<br>the specified range are<br>registered |                                  |
|                                    | Running program<br>exists inside the<br>specified range    | None of programs<br>inside the<br>specified range<br>are running |  |                                  |
| Error                              | None   | None   | None   | C2C<br>"Program<br>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)

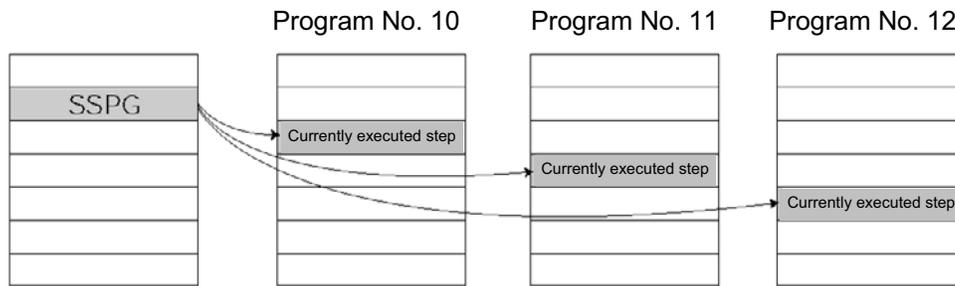
| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                   |                  | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-------------------|------------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1         | Operand 2        |                          |
| Optional                                  | Optional                       | SSPG                    | Program<br>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<br>"Non-registered program specification error" | C2C<br>"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<br>"Non-registered program specification error"        | C2C<br>"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.



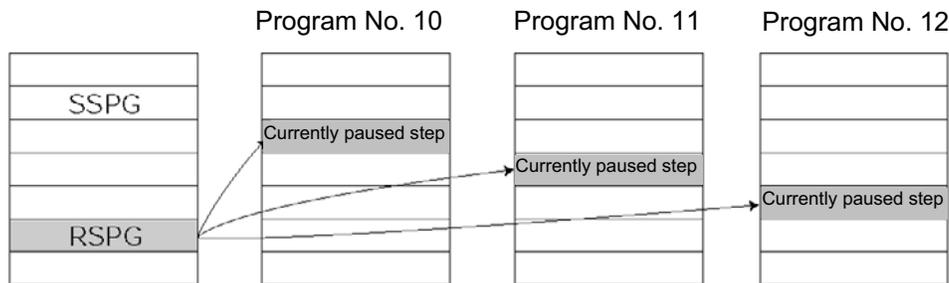
● RSPG (Resume program)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                   |                     | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-------------------|---------------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1         | Operand 2           |                          |
| Optional                                  | Optional                       | RSPG                    | Program<br>number | (Program<br>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<br>"Non-registered program specification error" | C2C<br>"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<br>"Non-registered program specification error"        | C2C<br>"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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |             |                    | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-------------|--------------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1   | Operand 2          |                          |
| Optional                                  | Optional                       | PGET                    | Axis number | Position<br>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.

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









- PRED (Read current position)

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

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

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

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

11 (binary) → 3 (decimal)

LET     1     3      Assign 3 to variable 1.

PRED   \*1    \*10

[Example 3]      LET     1     10      Assign 10 to variable 1.  
PRED   11    \*1      Read the current positions of axes 1 and 2 to the content of variable 1 (position 10).



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

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

This command is available with the PC software of version 1.1.0.5 or later and teaching pendant of version 1.05 or later.

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

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



- PTST (Check position data)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |              |                    | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------|--------------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1    | Operand 2          |                          |
| Optional                                  | Optional                       | PTST                    | Axis pattern | Position<br>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 axes 1 and 2 at position 10.  
Output 300 will turn OFF if the position data is given as follows:

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

11 (binary) → 3 (decimal)

LET    1        3                    Assign 3 to variable 1.  
PTST   \*1       10        300

[Example 3]    LET    1        11                    Assign 11 to variable 1.  
PTST   11       \*1        600    Turn ON flag 600 if there are no valid values in the data of axes 1 and 2 at the content of variable 1 (position 11).  
Flag 600 will turn ON if the position data is given as follows:

| No. | Axis 1   | Axis 2   | Axis 3   | Speed | Acceleration | Deceleration |
|-----|----------|----------|----------|-------|--------------|--------------|
| 10  | 100.000  | 50.000   | XXXX.XXX | XXX   | XXXX         | XXXX         |
| 11  | XXXX.XXX | XXXX.XXX | 200.000  | XXX   | XXXX         | XXXX         |



- PVEL (Assign speed data)

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

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

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

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



- PACC (Assign acceleration data)

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

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

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

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



- PDCL (Assign deceleration data)

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

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

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



- PAXS (Read axis pattern)

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

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

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

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

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

|     | Axis 1  | Axis 2  |       |                 |
|-----|---------|---------|-------|-----------------|
| 98  | XX.XXX  | XX.XXX  | ..... | 0 0 = 0 + 0 = 0 |
| 99  | 100.XXX | XX.XXX  | ..... | 0 1 = 0 + 1 = 1 |
| 100 | XX.XXX  | 150.000 | ..... | 1 0 = 2 + 0 = 2 |
| 101 | 100.000 | 50.000  | ..... | 1 1 = 2 + 1 = 3 |



- PSIZ (Check position data size)

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

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

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

| Position No. | Axis 1  | Axis 2  | Axis 3  | Vel | Acc  | Dcl  |
|--------------|---------|---------|---------|-----|------|------|
| 1            | XXX.XXX | XXX.XXX | XXX.XXX | XXX | X.XX | X.XX |
| 2            | XXX.XXX | XXX.XXX | XXX.XXX | XXX | X.XX | X.XX |
| •            |         |         |         |     |      |      |
| •            |         |         |         |     |      |      |
| •            |         |         |         |     |      |      |
| •            |         |         |         |     |      |      |
| 10           | 50.000  | 100.000 | 150.000 | 200 | 0.30 | 0.30 |
| •            |         |         |         |     |      |      |
| •            |         |         |         |     |      |      |

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



- GACC (Get acceleration data)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                    |                    | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------------|--------------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1          | Operand 2          |                          |
| Optional                                  | Optional                       | GACC                    | Variable<br>number | Position<br>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.

| Position No. | Axis 1  | Axis 2  | Axis 3  | Vel | Acc  | Dcl  |
|--------------|---------|---------|---------|-----|------|------|
| 1            | XXX.XXX | XXX.XXX | XXX.XXX | XXX | X.XX | X.XX |
| 2            | XXX.XXX | XXX.XXX | XXX.XXX | XXX | X.XX | X.XX |
| •            |         |         |         |     |      |      |
| •            |         |         |         |     |      |      |
| •            |         |         |         |     |      |      |
| •            |         |         |         |     |      |      |
| 10           | 50.000  | 100.000 | 150.000 | 200 | 0.30 | 0.30 |
| •            |         |         |         |     |      |      |
| •            |         |         |         |     |      |      |

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



- GDCL (Get deceleration data)

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

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

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

| Position No. | Axis 1  | Axis 2  | Axis 3  | Vel | Acc  | Dcl  |
|--------------|---------|---------|---------|-----|------|------|
| 1            | XXX.XXX | XXX.XXX | XXX.XXX | XXX | X.XX | X.XX |
| 2            | XXX.XXX | XXX.XXX | XXX.XXX | XXX | X.XX | X.XX |
| •            |         |         |         |     |      |      |
| •            |         |         |         |     |      |      |
| •            |         |         |         |     |      |      |
| •            |         |         |         |     |      |      |
| 10           | 50.000  | 100.000 | 150.000 | 200 | 0.30 | 0.30 |
| •            |         |         |         |     |      |      |
| •            |         |         |         |     |      |      |

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



## 1-11 Actuator Control Declaration

## ● VEL (Set speed)

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

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

(Note 1) Decimal places cannot be used. An error will generate

(Note 2) The minimum speed is 1 mm/s.

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

[Example 2]      VEL      200              Set the speed to 200 mm/s.  
                      MOV P    2              Move to point 2 at 200 mm/s.

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





- ACC (Set acceleration)

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

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

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

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

(Note) Setting an acceleration exceeding the specified range for the actuator may generate an error. It may also result in a failure or shorter product life.  
 Maximum acceleration: 0.3 G for the gate type and 0.2 G for the cantilever type



- DCL (Set deceleration)

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

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

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

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

(Note) Setting a deceleration exceeding the specified range for the actuator may generate an error. It may also result in a failure or shorter product life.  
Maximum acceleration: 0.3 G for the gate type and 0.2 G for the cantilever type

● SCR V (Set sigmoid motion ratio)

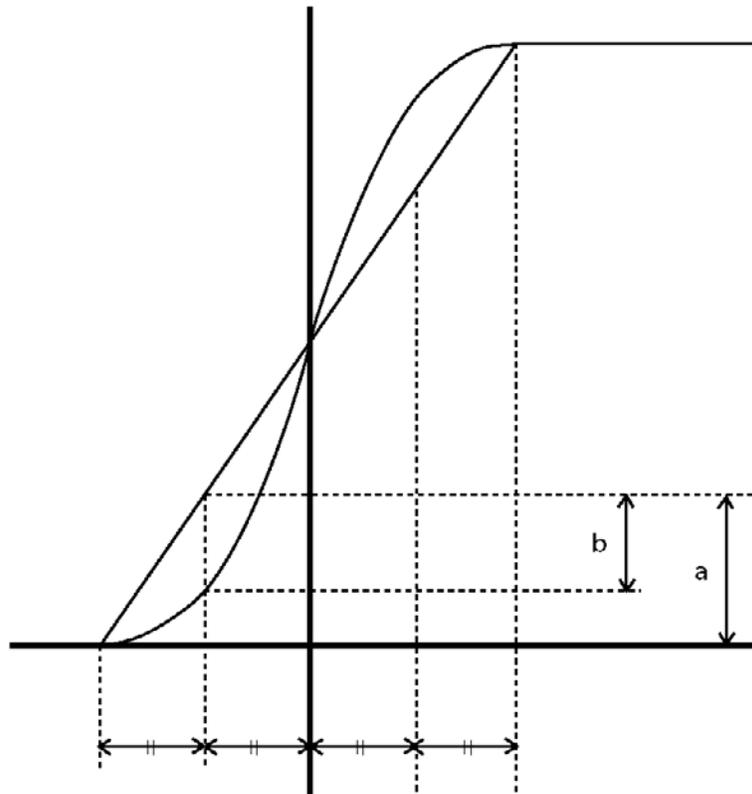
| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |           |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-----------|------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1 | Operand 2  |                          |
| Optional                                  | Optional                       | SCR V                   | 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 (%).

$$\frac{b}{a} \times 100 (\%)$$

If the ratio is not set using this command or 0% is set, a trapezoid motion will be implemented.

A SCR V command can be used with the following commands: MOVP, MOVL, MVPI, MVLI, JBWF, JBWN, JFWF, JFWN



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



- OFST (Set offset)

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

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

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

[Example 1]      OFST    100    50    Add 50 mm to the specified positions of axes 3.  
                      :  
                      OFST    100    0    Return the offsets of axes 3 to 0.

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

100 (binary) → 4 (decimal)  
LET     1     4     Assign 4 to variable 1.  
OFST    \*1    50  
          :  
OFST    \*1    0

[Example 3]      LET     1     100    Assign 100 to variable 1.  
                      OFST    101    \*1    Add the content of variable 1 (100 mm) to the specified positions of axes 1 and 3.



- DEG (Set arc angle)

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

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

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

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



- BASE (Specify axis base)

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

Function] Count the axes sequentially based on the axis number specified in operand 1 being the first axis.

A BASE command can be used with PRED, PRDQ, AXST, actuator-control, ARCH, PACH, PMVP, PMVL and zone commands. Note that each zone range is assigned to the actuator via parameter.

[Example 1]      HOME    1            Axis 1 returns to the home.  
                   BASE    2            Axis 2 is considered the first axis.  
                   HOME    1            Axis 2 returns to the home.

Hereafter, axes 2 to 4 will operate based on the specifications for axes 1 to 3 (axis number, axis pattern, position data, etc.).

[Example 2]      LET      1      3        Assign 3 to variable 1.  
                   BASE    \*1            The content of variable 1 (axis 3) will be considered as the first axis.

Hereafter, axes 3 and 4 will operate based on the specifications for axes 1 and 2.



- GRP (Set group axes)

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

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

[Example 1]

|      |     |   |   |
|------|-----|---|---|
| GRP  | 110 |   | Data of axes 2 and 3 become valid.  |
| CIR2 | 1   | 2 | Axis-pattern error will not generate even if data is set for axis 1 to 3. |

[Example 2] The axis pattern can be specified indirectly using a variable. When the command in [Example 1] is rephrased based on indirect specification using a variable:  
 110 (binary) → 6 (decimal)  
 LET 1 6 Assign 6 to variable 1.  
 GRP \*1  
 CIR2 1 2



● HOLD (Hold: Declare axis port to pause)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                              |             | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|------------------------------|-------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1                    | Operand 2   |                          |
| Optional                                  | Optional                       | HOLD                    | (Input port,<br>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.

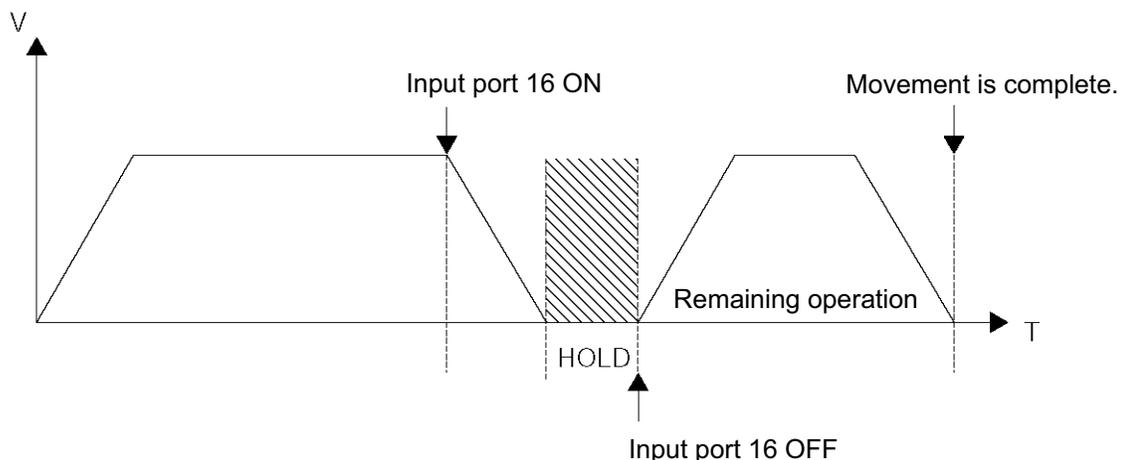
If any axis currently stopped via a HOLD servo OFF is moved by external force, etc., from the stopped position, and when the servo of that axis was ON prior to the HOLD stop, the axis will move to the original stopped position when the HOLD is cancelled before resuming operation.

(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 JXWX and PATH operations.)

(Note 3) Following a pause of home return, the operation will resume from the beginning of the home-return sequence.

[Example]            HOLD    16    0    The axes will decelerate to a stop when input port 16 turns ON.



● CANC (Cancel: Declare axis port to abort)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                              |             | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|------------------------------|-------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1                    | Operand 2   |                          |
| Optional                                  | Optional                       | CANC                    | (Input port,<br>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)

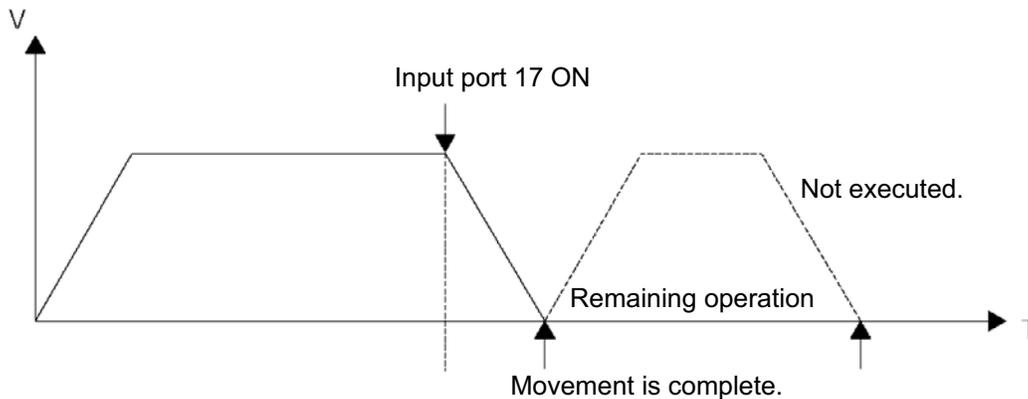
The CANC type is set to "0" (contact a) when the program is started.

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 JXWX and PATH operations.)

[Example] CANC 17 0 The axes will decelerate to a stop when input port 17 turns ON.





- VLMX (Specify VLMX speed)

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

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

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

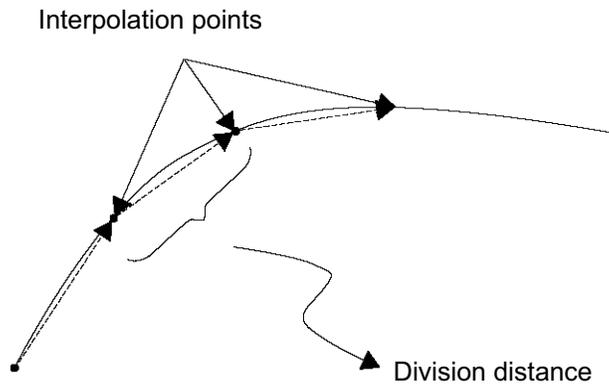
[Example]

|      |     |   |  |
|------|-----|---|--|
| VEL  | 200 | ] | The speed becomes 200 mm/sec in this section.  |
| MOVP | 1   |   |  |
| MOVP | 2   | ] | The speed becomes VLMX mm/sec in this section. |
| VLMX |     |   |  |
| MOVP | 3   | ] |  |
| MOVP | 4   |   |  |

● DIS (Set division distance at spline movement)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |           |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-----------|------------|--------------------------|
|   |                                | Command,<br>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.

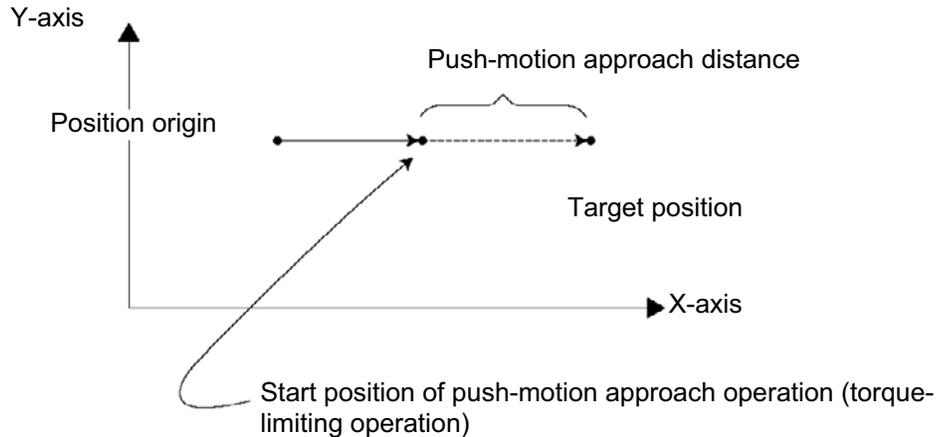
[Example]            DIS            10            Set the division distance to 10 mm.



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

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |           |           | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-----------|-----------|--------------------------|
|   |                                | Command,<br>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.

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

## ● QRTN (Set quick-return mode)

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

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

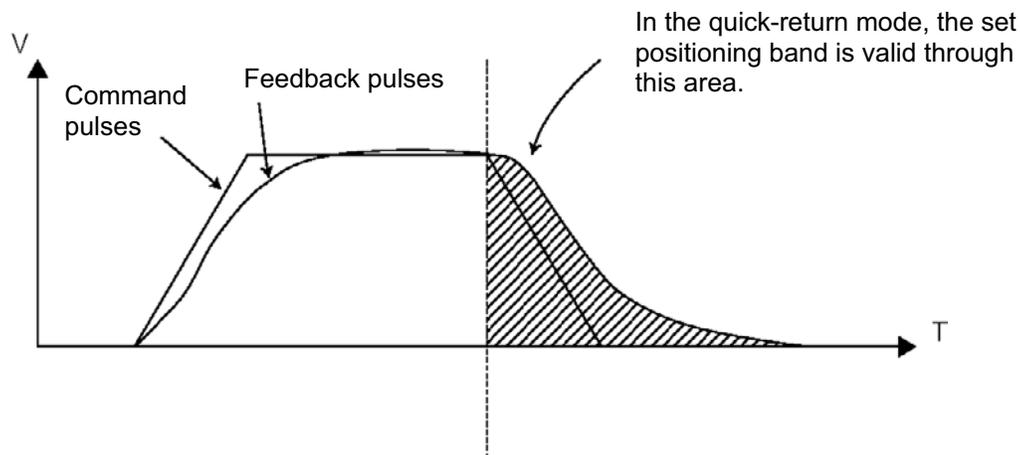
[1] QRTN [Operand 1] = 0 (Normal mode)

Positioning is deemed complete when all command pulses have been output and the current position is inside the positioning band.

\* If a deceleration command is currently executed in the quick-return mode, the system will wait for all command pulses to be output.

[2] QRTN [Operand 1] = 1 (Quick-return mode)

Positioning is deemed complete when “a normal deceleration command is currently executed (excluding deceleration due to a stop command, etc.) or all command pulses have been output” AND “the current position is inside the positioning band.” This setting is used to perform other processing during deceleration, in conjunction with a PBNB command.



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

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

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

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

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

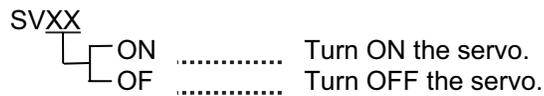


## 1-12 Actuator Control Command

## ● SVXX (Turn ON/OFF servo)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |              |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------|------------|--------------------------|
|   |                                | Command,<br>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.



[Example 1] SVON 110 Turn ON the servos of axes 2 and 3. Nothing will occur if the axis servos are already ON.

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

```

110 (binary) → 6 (decimal)
LET 1 6 Assign 6 to variable 1.
SVON *1

```



Warning : Turning the servo ON near the mechanical end may disturb the magnetic pole phase detection, and may cause the magnetic pole unconfirmed error or the magnetic pole detection error.  
Put the slider or rod away from the mechanical end when turning the servo ON.



- HOME (Return to home)

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

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

(Note) Following a pause of home return, the operation will resume from the beginning of the home-return sequence.

[Example 1] HOME 100 Axis 3 return to the home.

[Example 2] The axis pattern can be specified indirectly using a variable.  
When the command in [Example 1] is rephrased based on indirect specification using a variable:  
100 (binary) → 4 (decimal)  
LET 1 4 Assign 4 to variable 1.  
HOME \*1



Warning : Turning the servo ON near the mechanical end may disturb the magnetic pole phase detection, and may cause the magnetic pole unconfirmed error or the magnetic pole detection error.  
Put the slider or rod away from the mechanical end when turning the servo ON.



● **MOVP (Move PTP by specifying position data)**

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

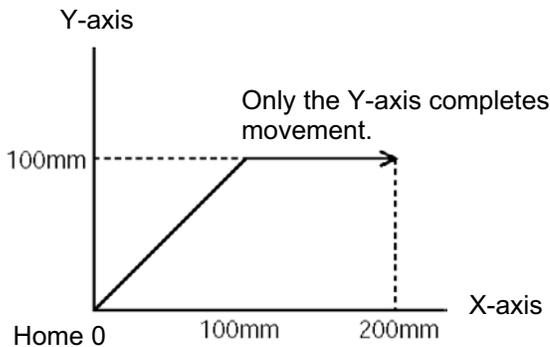
[Function] Move the actuator to the position corresponding to the position number specified in operand 1, without interpolation (PTP stands for “Point-to-Point”).  
The output will turn OFF at the start of axis movement, and turn ON when the movement is complete.

[Example 1]      VEL      100      Set the speed to 100 mm/sec.  
                  MOVP    1      Move the axes to the position corresponding to position No. 1 (200, 100).

[Example 2]      VEL      100      Set the speed to 100 mm/sec.  
                  LET      1      2      Assign 2 to variable 1.  
                  MOVP    \*1      Move the axes to the position corresponding to the content of variable 1 (position No. 2, or (100, 100)).

| No. | X-axis  | Y-axis  | Speed | Acceleration | Deceleration |
|-----|---------|---------|-------|--------------|--------------|
| 1   | 200.000 | 100.000 | XXX   | XXXX         | XXXX         |
| 2   | 100.000 | 100.000 | XXX   | XXXX         | XXXX         |

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



Each axis moves at the specified speed.



- **MOVL** (Move by specifying position data)

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

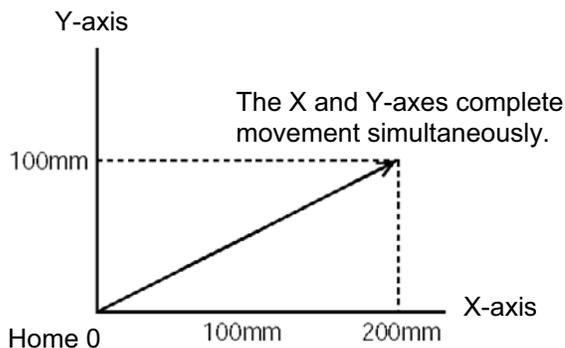
[Function] Move the actuator to the position corresponding to the position number specified in operand 1, with interpolation.  
The output will turn OFF at the start of axis movement, and turn ON when the movement is complete.

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

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

| No. | X-axis  | Y-axis  | Speed | Acceleration | Deceleration |
|-----|---------|---------|-------|--------------|--------------|
| 1   | 200.000 | 100.000 | XXX   | XXXX         | XXXX         |
| 2   | 100.000 | 100.000 | XXX   | XXXX         | XXXX         |

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



The X and Y-axes complete movement simultaneously.

The tip of each axis moves at the specified speed.





- MVLI (Move via incremental interpolation)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                    |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------------|------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1          | Operand 2  |                          |
| Optional                                  | Optional                       | MVLI                    | Position<br>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.

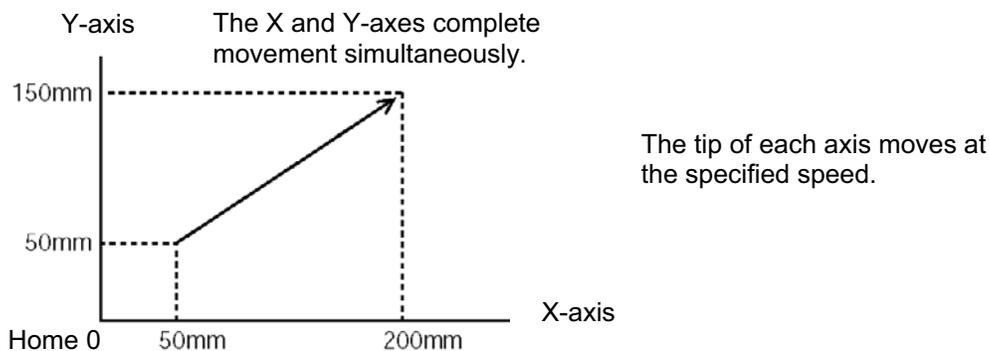
Movement may not occur if the specified travel distance is below the resolution (1 pulse):  
1 pulse: Lead [mm] / 16384 --- Standard product with a gear ratio of 1:1

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

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

| No. | X-axis  | Y-axis  | Speed | Acceleration | Deceleration |
|-----|---------|---------|-------|--------------|--------------|
| 1   | 200.000 | 100.000 | XXX   | XXXX         | XXXX         |
| 2   | 100.000 | 100.000 | XXX   | XXXX         | XXXX         |

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

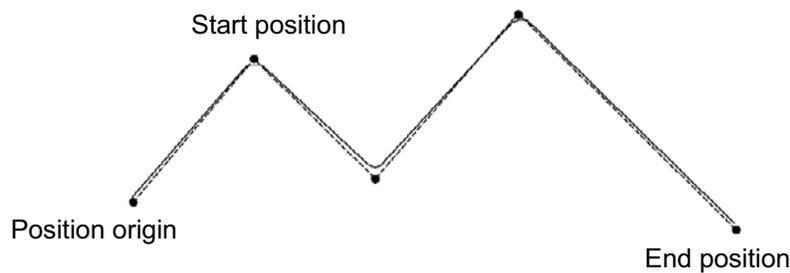




- PATH (Move along path)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                          |                        | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------------------|------------------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1                | Operand 2              |                          |
| Optional                                  | Optional                       | PATH                    | Start position<br>number | End position<br>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]      VEL      100            Set the speed to 100 mm/sec.  
                  PATH     100    120        Move continuously from position Nos. 100 to 120.

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





- STOP (Stop movement)

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

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

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

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

[Example 1]        STOP    100            Decelerate and stop axis 3.

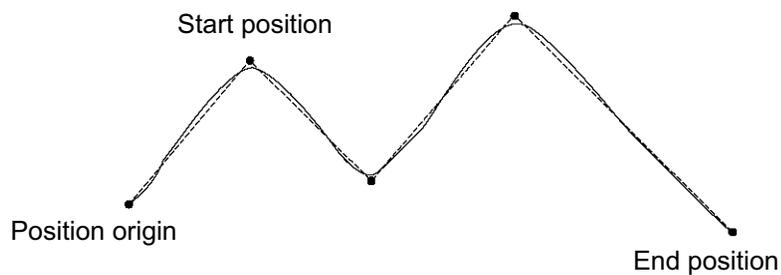
[Example 2]        The axis pattern can be specified indirectly using a variable.  
When the command in [Example 1] is rephrased based on indirect specification using a variable:  
100 (binary) → 4 (decimal)  
LET    1     12    Assign 4 to variable 1.  
STOP   \*1



- PSPL (Move along spline)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                          |                        | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------------------|------------------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1                | Operand 2              |                          |
| Optional                                  | Optional                       | PSPL                    | Start position<br>number | End position<br>number | PE                       |

[Function] Continuously move from the specified start position to end position via interpolation along a spline-interpolation curve.  
The output type in the output field can be set using an actuator-declaration command POTP.  
If invalid data is set for any position number between the start and end position numbers, that position number will be skipped during continuous movement.



(The above diagram is only an example.)

(Note) If the acceleration and deceleration are different between points, the speeds will not be connected smoothly.

In this case, input in operand 1 the point number of the next target, instead of the predicted current position upon execution of the applicable command.

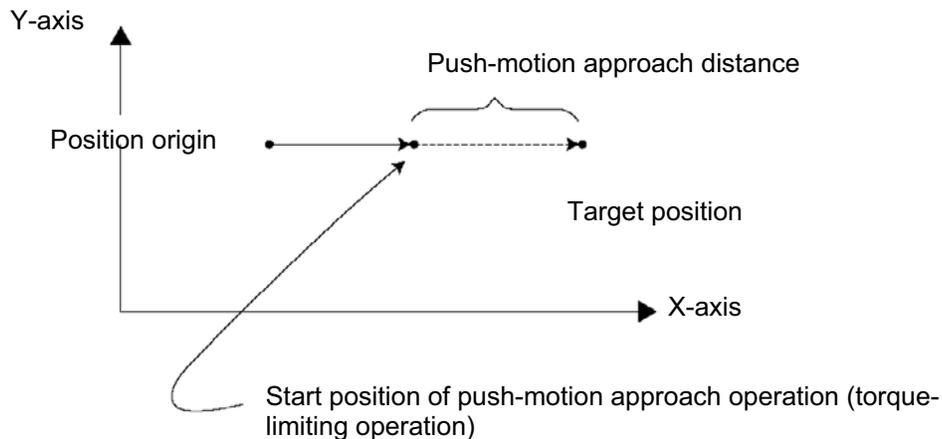
(Inputting a point number corresponding to the predicted current position will trigger movement to the same point during continuous movement, thereby causing the speed to drop.)

[Example]      VEL      100      Set the speed to 100 mm/sec.  
                  PSPL    100    120    Continuously move from position Nos. 100 to 120 along a spline-interpolation curve.

## ● PUSH (Move by push motion)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                              |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|------------------------------|------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1                    | Operand 2  |                          |
| Optional                                  | Optional                       | PUSH                    | Target<br>position<br>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 (torque-limiting operation) will be performed. The speed of push-motion approach operation (torque-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 a single axis. If multiple axes are specified, an “Error No. C91, Multiple push-axes specification error” will generate.
- (Note 2) A push-motion approach speed exceeding the maximum speed permitted by the system will be clamped at the maximum speed. (The maximum system speed is not the maximum practical speed. Determine a practical speed by considering the impact upon contact, etc.)



[Example]      PAPER                    100                    20  
                   MOV                        2  
                   PUSH                    10

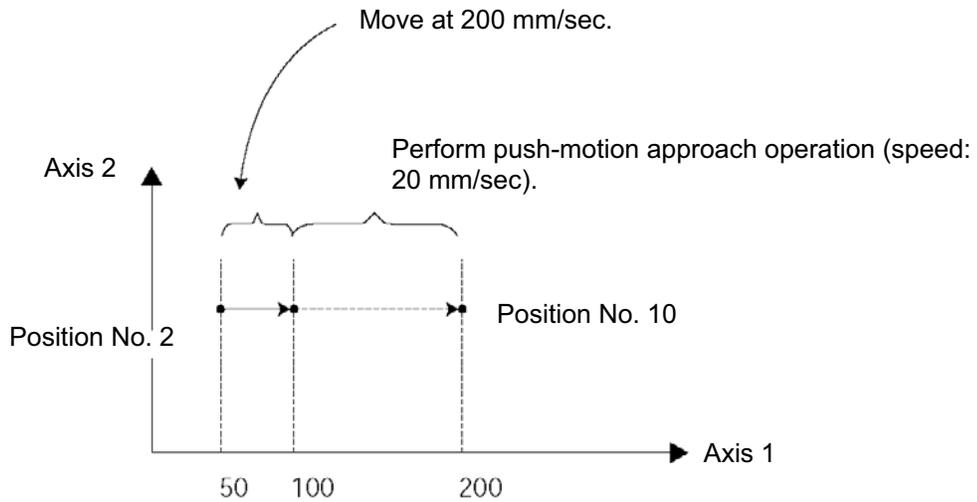
Set the push-motion approach distance to 100 mm and push-motion approach speed to 20 mm/sec.

Move from the current position to position No. 2.

Perform push-motion movement from position Nos. 2 to 10.

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

| Position No. | Axis 1  | Axis 2  | Axis 3  | Vel | Acc  | Dcl  |
|--------------|---------|---------|---------|-----|------|------|
| 1            | XXX.XXX | XXX.XXX | XXX.XXX | XXX | X.XX | X.XX |
| 2            | 50.000  | 100.000 | XXX.XXX | XXX | X.XX | X.XX |
| •            |         |         |         |     |      |      |
| •            |         |         |         |     |      |      |
| •            |         |         |         |     |      |      |
| •            |         |         |         |     |      |      |
| 10           | 200.000 |         |         | 200 | 0.30 | 0.30 |
| •            |         |         |         |     |      |      |
| •            |         |         |         |     |      |      |







● CIRS (Move three-dimensionally along circle)

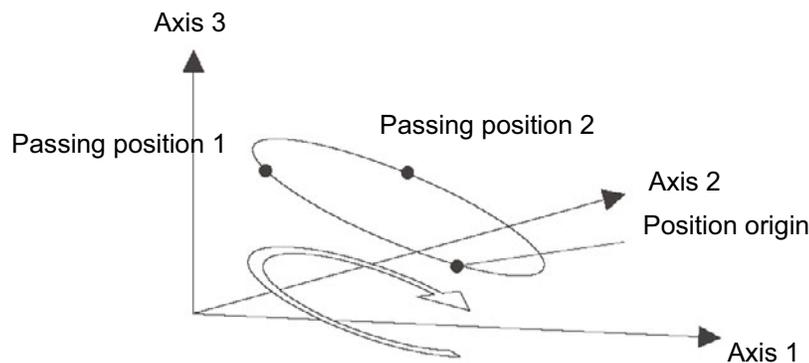
| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                                 |                                 | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|---------------------------------|---------------------------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1                       | Operand 2                       |                          |
| Optional                                  | Optional                       | CIRS                    | Passing<br>position 1<br>number | Passing<br>position 2<br>number | PE                       |

This command is available with the PC software of version 1.1.0.5 or later and teaching pendant of version 1.05 or later.

[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 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.

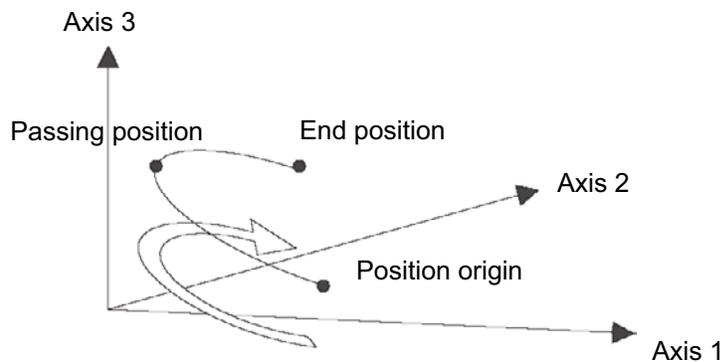
(Note 3) If the circle diameter is small with respect to the set speed, the speed may be limited. (Increasing the acceleration/deceleration will reduce the speed limitation, but they must not exceed the range permitted by the actuator.)

● ARCS (Move three-dimensionally along arc)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                               |                        | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-------------------------------|------------------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1                     | Operand 2              |                          |
| Optional                                  | Optional                       | ARCS                    | Passing<br>position<br>number | End position<br>number | PE                       |

This command is available with the PC software of version 1.1.0.5 or later and teaching pendant of version 1.05 or later.

[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 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. (Increasing the acceleration/deceleration will reduce the speed limitation, but they must not exceed the range permitted by the actuator.)



● CHVL (Change speed)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |              |           | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------|-----------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1    | Operand 2 |                          |
| Optional                                  | Optional                       | CHVL                    | Axis pattern | Speed     | CP                       |

[Function] Change the speed of the axes operating in other task.  
When a CHVL command is executed, the speed of the axes specified in operand 1 will change to the value specified in operand 2.

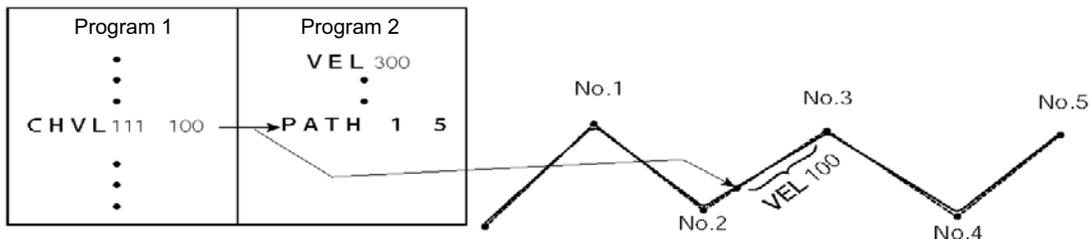
- (Note 1) This command is not valid on an axis operated by a CIR, ARC, PSPL, PUSH, ARCH, PACH, CIRS or ARCS command.
- (Note 2) Executing a CHVL command for an axis operating in sigmoid motion (SCRV command) will generate an "Error No. CC1, Speed-change condition error."
- (Note 3) This is a temporary speed-change command issued from other task to the active packet (point). It is not affected by the data declared by VEL.

| Program 1    | Program 2 |
|--------------|-----------|
| CHVL 111 100 | VEL 300   |
|              | .         |
|              | MOVP 1    |
|              | MOVP 2    |
|              | MOVP 3    |
|              | .         |

If CHVL is executed in program 1 while MOVP 2 is executed in program 2, the travel speed of MOVP 2 will become 100 mm/sec. The speeds of other move commands will remain 300 mm/sec.

The axis pattern can be specified indirectly using a variable.  
When program 1 is rephrased based on indirect specification using a variable:  
111 (binary) → 7 (decimal)  
LET 1 7 Assign 7 to variable 1.  
CHVL \*1 100

- (Note 4) Since this command is valid only for the packet that is active at the time of execution of the command for an axis subject to continuous motion in a PATH command, etc., caution must be exercised against the timing shift. The packet handling will be put on hold during speed-change processing, so caution must also be exercised against the locus shift.



If CHVL is executed in program 1 while PATH is executed in program 2, or specifically during the PATH movement from point No. 2 to point No. 3, the speed specified by CHVL (100 mm/sec in the above example) will become valid only during the PATH movement to point No. 3. Other travel speeds will remain at the speed specified by VEL (300 mm/sec in the above example).

- (Note 5) Override of the CHVL call task will be applied, so caution must be exercised.
- (Note 6) The maximum speed of the specified axis completing home return will be clamped by the minimum value set in "Axis-specific parameter No. 28, Maximum operating speed of each axis" or "Axis-specific parameter No. 27, Maximum speed limited by maximum motor speed" with respect to the specified axis and related interpolation axes currently operating. To prevent the maximum speed from being limited due to the effect of other axis whose maximum speed is lower than the speed specified in the CHVL command, issue a CHVL command in multiple steps corresponding to the respective axes having different maximum speeds. In particular, specification of a CHVL command in a separate step is recommended for a rotating axis.

[Example] CHVL 111 500 ⇒ CHVL 111 500  
CHVL 1000 500

● ARCD (Move along arc via specification of end position and center angle (arc interpolation))

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

[Function] Move along an arc originating from the current position and terminating at the end position, via arc interpolation.

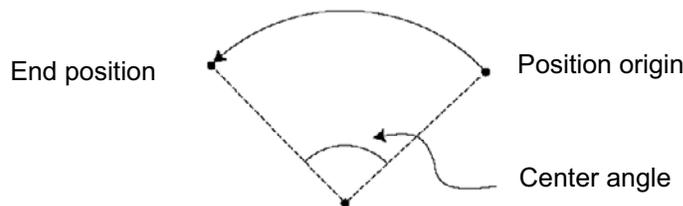
Specify the end position of movement in operand 1, and the center angle formed by the position origin and end position in operand 2. The center angle is set in a range from  $-359.999$  to  $-0.001$  or from  $0.001$  to  $359.999$ . A positive value indicates CCW (counterclockwise) movement, while a 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 acceleration in all-axis parameter No. 11 (Default deceleration in all-axis parameter No. 12) |

If speed is not set, a "C88 speed specification error" will generate.

If acceleration/deceleration is not valid, a "C89 acceleration/deceleration specification error" will generate.



(Note 1) This command is valid on arbitrary orthogonal planes. (Axis 2 may be selected automatically prior to axis 1 in accordance with the position data.)

[Example]           VEL       100           Set the speed to 100 mm/sec.  
                       ARCD    100    120       Move along an arc from the position origin to position No. 100 for a center angle of 120 degrees (CCW direction).

(Note 2) The rotating direction of the actual operation locus varies depending on whether the system is of gate type or cantilever type. Be sure to perform test operation to check the rotating direction.

● ARCC (Move along arc via specification of center position and center angle (arc interpolation))

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                              |              | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|------------------------------|--------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1                    | Operand 2    |                          |
| Optional                                  | Optional                       | ARCC                    | Center<br>position<br>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.

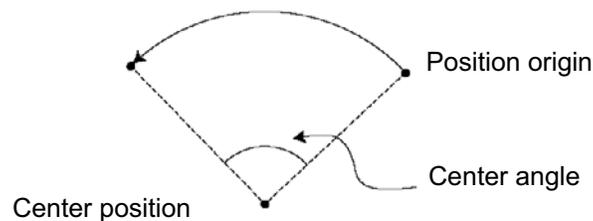
Specify the center position in operand 1, and the center angle formed by the position origin and end position in operand 2. The center angle is set in a range from  $-3600$  to  $3600$  degrees ( $\pm 10$  revolutions). A positive value indicates CCW (counterclockwise-direction) movement, while a negative 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 acceleration in all-axis parameter No. 11 (Default deceleration in all-axis parameter No. 12) |

If speed is not set, a "C88 speed specification error" will generate.

If acceleration/deceleration is not valid, a "C89 acceleration/deceleration specification error" will generate.



(Note 1) This command is valid on arbitrary orthogonal planes. (Axis 2 may be selected automatically prior to axis 1 in accordance with the position data.)

[Example]           VEL     100           Set the speed to 100 mm/sec.  
                  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).

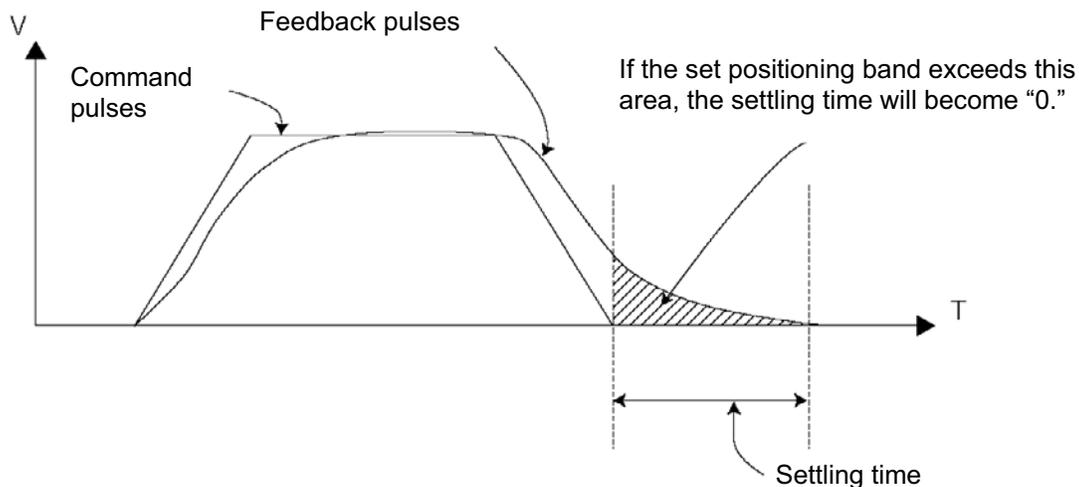
(Note 2) The rotating direction of the actual operation locus varies depending on whether the system is of gate type or cantilever type. Be sure to perform test operation to check the rotating direction.



● PBNB (Set positioning band)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |              |           | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------|-----------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1    | Operand 2 |                          |
| Optional                                  | Optional                       | PBNB                    | Axis pattern | Distance  | CP                       |

[Function] Set the positioning completion band for the axes in the axis pattern specified in operand 1. The distance in operand 2 is set in mm. As a rule, positioning is deemed complete when all command pulses have been output and the current position is inside the positioning band. Therefore, this command is effective if you wish to reduce the tact time by shortening the approximate positioning settling time. (Normally a setting of approx. 3 to 5 mm will have effect, but the effect must be confirmed on the actual machine.)  
(This command can be combined with a QRTN command for special purposes. Refer to the section on QRTN command for details.)



- (Note 1) If positioning band is not set with a PBNB command, the value set in “Axis-specific parameter No. 58, Positioning band” will be used.
- (Note 2) If the positioning band is changed, the new setting will remain valid even after the program ends. Therefore, to build a system using PBNB commands, a positioning band must be expressly specified with a PBNB command before operation of each program. An assumption that the positioning band will be reset to the original value when the operation ends in other program may lead to an unexpected problem, because the positioning band will become different from what is anticipated in case the applicable program is aborted due to error, etc.
- (Note 3) The value set in “Axis-specific parameter No. 58, Positioning band” will not be written by a PBNB command.

[Example 1] PBNB 11 5 Set the positioning band for axes 1 and 2 to 5 mm after this command.

[Example 2] The axis pattern can be specified indirectly using a variable.  
When the command in [Example 1] is rephrased based on indirect specification using a variable:  
11 (binary) → 3 (decimal)  
LET 1 3 Assign 3 to variable 1.  
PBNB \*1 5







1-13 Structural IF

● IFXX (Structural IF)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                    |           | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------------|-----------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1          | Operand 2 |                          |
| Optional                                  | Optional                       | IFXX                    | Variable<br>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 |   |    |                             |
| 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]

```

600  VEL      100      Set the speed to 100 mm/sec.
      IFEQ     1        1  Select an axis.
      IFGE     2        0  Select a moving direction.
      JFWN     01       5  Move axis 1 forward.
      ELSE
      JBWN     01       5  Move axis 1 backward.
      EDIF
      ELSE
      IFLT     2        0  Select a moving direction.
      JBWN     10       5  Move axis 2 backward.
      ELSE
      JFWN     10       5  Move axis 2 forward.
      EDIF
      EDIF
    
```

Jog by selecting axis 1/axis 2 by variable 1 and forward/backward (+/-) by variable 2.  
 Nothing will happen if flag 600 is OFF, in which case the program will proceed to the step next to the last EDIF.

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



● ISXX (Compare strings)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                  |   | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|------------------|---|--------------------------|
|   |                                | Command,<br>declaration | Operand 1        | Operand 2                                 |                          |
| Optional                                  | Optional                       | ISXX                    | Column<br>number | Column<br>number,<br>character<br>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]

```

VEL    100          Set the speed to 100 mm/sec.
SCPY   10          'GOFD' (Move
                  forward)
SCPY   14          'GOBK' (Move
                  backward)
LET    1           5
LET    2           14
SLEN   4           Set the number of comparing characters to 4.
ISEQ   1           '1AXS' (Axis 1) Select an axis.
ISEQ   5           10          Select a moving direction.
JFWN   01          5           Move axis 1 forward.
ELSE
JBWN   01          5           Move axis 1 backward.
EDIF
ELSE
ISNE   *1          *2          Select a moving direction.
JFWN   10          5           Move axis 2 backward.
ELSE
JBWN   10          5           Move axis 2 forward.
EDIF
EDIF

```

Jog by selecting axis 1/axis 2 by columns 1 to 4 and forward/backward by columns 5 to 8.  
 Nothing will happen if flag 600 is OFF, in which case the program will proceed to the step next to the last EDIF.  
 If columns 1 to 8 contain the following data, axis 1 will be moved forward.

|    |    |    |    |   |   |   |   |
|----|----|----|----|---|---|---|---|
| 1  | 2  | 3  | 4  | 5 | 6 | 7 | 8 |
| 1A | XS | GO | FD |   |   |   |   |

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



- ELSE (Else)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |            |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|------------|------------|--------------------------|
|   |                                | Command,<br>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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |            |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|------------|------------|--------------------------|
|   |                                | Command,<br>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.





- ITER (Repeat)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |            |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|------------|------------|--------------------------|
|   |                                | Command,<br>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
      :
      :
600  ITER
      :
      EDDO
  
```

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

Forcibly switch the control to an EDDO command and perform end judgment, if flag 600 is ON.

- EDDO (End DO WHILE)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |            |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|------------|------------|--------------------------|
|   |                                | Command,<br>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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |            |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|------------|------------|--------------------------|
|   |                                | Command,<br>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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                    |           | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------------|-----------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1          | Operand 2 |                          |
| Prohibited                                | Prohibited                     | WHXX                    | Variable<br>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
WHEQ    1      10      [1] will be executed if the content of variable 1 is 10.
:
[1]
:
WHGT    1      *2      This command will be executed if the content of variable
:                      1 is greater than the content of variable 2.
:                      Since variable 1 (= 20) > variable 2 (=10), [2] will be
:                      executed.

[2]
:
OTHE
:
[3]
:
EDSL
:
(4)
:

```

This command will be executed if none of the conditions are satisfied. In this example, since [2] was executed, [3] will not be executed.

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.

\* 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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                  |   | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|------------------|---|--------------------------|
|   |                                | Command,<br>declaration | Operand 1        | Operand 2                                 |                          |
| Prohibited                                | Prohibited                     | WSXX                    | Column<br>number | Column<br>number,<br>character<br>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
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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |            |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|------------|------------|--------------------------|
|   |                                | Command,<br>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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |            |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|------------|------------|--------------------------|
|   |                                | Command,<br>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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                    |             | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------------|-------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1          | Operand 2   |                          |
| Optional                                  | Optional                       | AXST                    | Variable<br>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 hexadecimal, 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:

$$3188 \div 16 = 199 \text{ ,,},4$$

$$199 \div 16 = 12 (= C) \text{ ,,},7$$

$$3188 = 12 (= C) \times 16^2 + 7 \times 16^1 + 4$$

$$= C74 (\text{HEX}) (\text{Hexadecimal number})$$

Therefore, an "Error No. C74, Actual-position soft limit over error" is present.



- PGST (Get program status)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                    |                   | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------------|-------------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1          | Operand 2         |                          |
| Optional                                  | Optional                       | PGST                    | Variable<br>number | Program<br>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 hexadecimal, the status to be stored (program error number) is a decimal. Therefore, the decimal program error numbers must be converted to hexadecimal.

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



- SYST (Get system status)

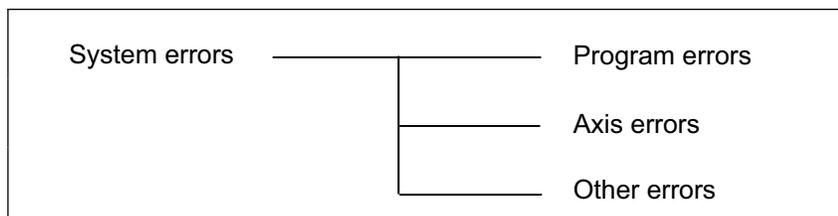
| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                    |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------------|------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1          | Operand 2  |                          |
| Optional                                  | Optional                       | SYST                    | Variable<br>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 hexadecimal, they must be converted to decimals.

(Note 3) Relationship of error statuses



\* An axis error that generates during operation with a program command will be registered both as a program error and an axis error.

[Example] SYST 1 Read the system error number to variable 1.



1-17 Zone

● WZNA (Wait for zone ON, with AND)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |             |              | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-------------|--------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1   | Operand 2    |                          |
| Optional                                  | Optional                       | WZNA                    | Zone number | Axis pattern | CP                       |

[Function] Wait for the zone status of all axes (AND) specified by the axis pattern in operand 2 to become ON (inside zone) with respect to the zone specified in operand 1.

(Note 1) The zone status of axes not yet completing home return will remain OFF (outside zone).

(Note 2) A maximum of four areas can be set as zones for each axis ("Axis-specific parameter Nos. 86 to 97").

(Note 3) Zone output can be specified using "Axis-specific parameter Nos. 88, 91, 94 and 97" irrespective of this command.

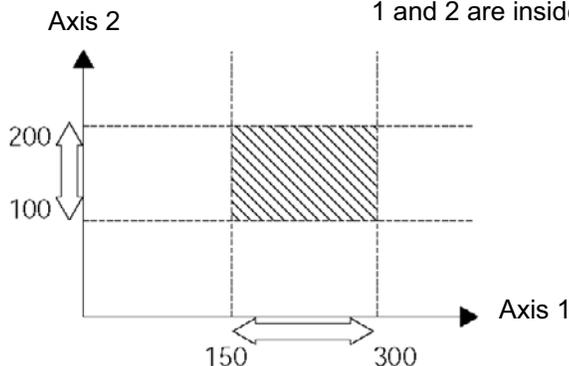
[Example 1]      WZNA      1              11              If the parameters are set as follows, the program will wait until the zone status of axes 1 and 2 becomes ON (inside the shaded area shown in the diagram below).

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

11 (binary) → 3 (decimal)  
 LET      5              3              Assign 3 to variable 5.  
 WZNA      1              \*5

|   |  |        |   |
|---|--|--------|---|
|   | Axis 1   | Axis 2 |   |
| { | "Axis-specific parameter No. 86, Zone 1 max." 300000 | 200000 | } |
|   | (Value is set in units of 0.001 mm)                  |        |   |
| { | "Axis-specific parameter No. 87, Zone 1 min." 150000 | 100000 | } |
|   | (Value is set in units of 0.001 mm)                  |        |   |

The program will proceed to the next step if both axes 1 and 2 are inside the shaded area.





● WZNO (Wait for zone ON, with OR)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |             |              | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-------------|--------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1   | Operand 2    |                          |
| Optional                                  | Optional                       | WZNO                    | Zone number | Axis pattern | CP                       |

[Function] Wait for the zone status of any of the axes (OR) specified by the axis pattern in operand 2 to become ON (inside zone) with respect to the zone specified in operand 1.

(Note 1) The zone status of axes not yet completing home return will remain OFF (outside zone).

(Note 2) A maximum of four areas can be set as zones for each axis ("Axis-specific parameter Nos. 86 to 97").

(Note 3) Zone output can be specified using "Axis-specific parameter Nos. 88, 91, 94 and 97" irrespective of this command.

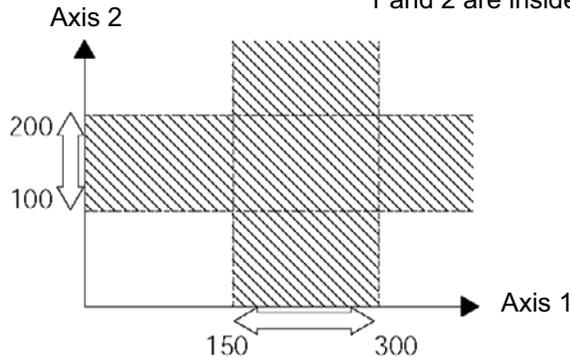
[Example 1]      WZNO    1            11            If the parameters are set as follows, the program will wait until the zone status of axes 1 or 2 becomes ON (inside the shaded area shown in the diagram below).

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

11 (binary) → 3 (decimal)  
 LET        5            3            Assign 3 to variable 5.  
 WZNO      1            \*5

|   |  |        |   |
|---|--|--------|---|
|   | Axis 1   | Axis 2 |   |
| { | "Axis-specific parameter No. 86, Zone 1 max." 300000 | 200000 | } |
|   | (Value is set in units of 0.001 mm)                  |        |   |
| { | "Axis-specific parameter No. 87, Zone 1 min." 150000 | 100000 | } |
|   | (Value is set in units of 0.001 mm)                  |        |   |

The program will proceed to the next step if both axes 1 and 2 are inside the shaded area.





● WZFA (Wait for zone OFF, with AND)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |             |              | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-------------|--------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1   | Operand 2    |                          |
| Optional                                  | Optional                       | WZFA                    | Zone number | Axis pattern | CP                       |

[Function] Wait for the zone status of all axes (AND) specified by the axis pattern in operand 2 to become OFF (outside zone) with respect to the zone specified in operand 1.

- (Note 1) The zone status of axes not yet completing home return will remain OFF (outside zone).
- (Note 2) A maximum of four areas can be set as zones for each axis ("Axis-specific parameter Nos. 86 to 97").
- (Note 3) Zone output can be specified using "Axis-specific parameter Nos. 88, 91, 94 and 97" irrespective of this command.

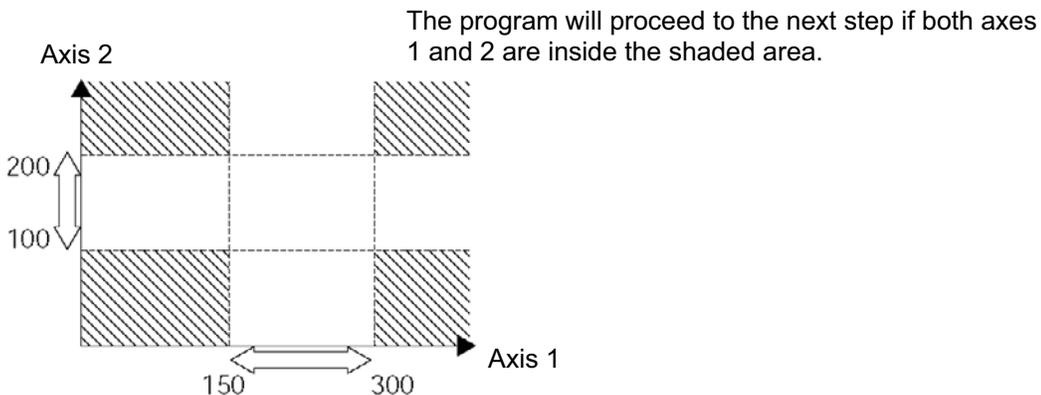
[Example]            WZFA    1            11            If the parameters are set as follows, the program will wait until the zone status of axes 1 and 2 becomes OFF (inside the shaded area shown in the diagram below)

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

```

11 (binary) → 3 (decimal)
LET    5            3            Assign 3 to variable 5.
WZFA   1            *5
    
```

|  |                  |                  |  |
|--|------------------|------------------|--|
| "Axis-specific parameter No. 86, Zone 1 max."<br>(Value is set in units of 0.001 mm) | Axis 1    300000 | Axis 2    200000 |  |
| "Axis-specific parameter No. 87, Zone 1 min."<br>(Value is set in units of 0.001 mm) | 150000           | 100000           |  |





● WZFO (Wait for zone OFF, with OR)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |             |              | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-------------|--------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1   | Operand 2    |                          |
| Optional                                  | Optional                       | WZFO                    | Zone number | Axis pattern | CP                       |

[Function] Wait for the zone status of any of the axes (OR) specified by the axis pattern in operand 2 to become OFF (outside zone) with respect to the zone specified in operand 1.

- (Note 1) The zone status of axes not yet completing home return will remain OFF (outside zone).
- (Note 2) A maximum of four areas can be set as zones for each axis ("Axis-specific parameter Nos. 86 to 97").
- (Note 3) Zone output can be specified using "Axis-specific parameter Nos. 88, 91, 94 and 97" irrespective of this command.

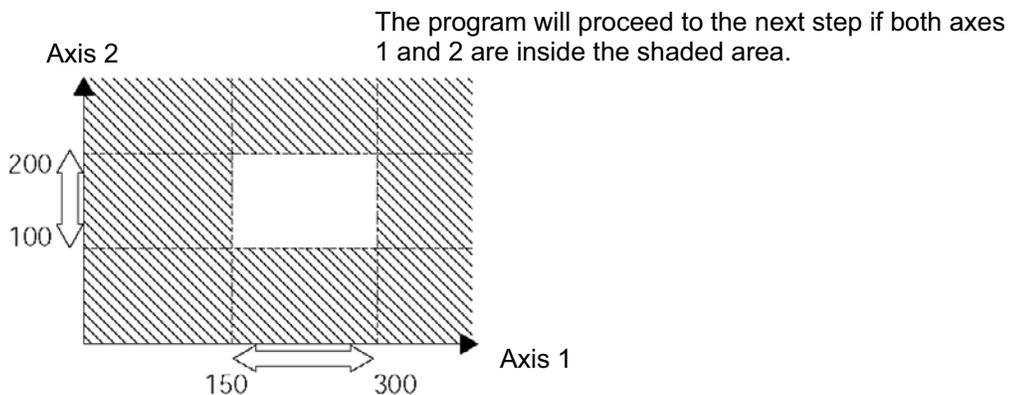
[Example 1]      WZFO      1              11              If the parameters are set as follows, the program will wait until the zone status of axes 1 or 2 becomes OFF (inside the shaded area shown in the diagram below).

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

```

11 (binary) → 3 (decimal)
LET     5             3             Assign 3 to variable 5.
WZFO    1             *5
    
```

|   |  |        |        |   |
|---|--|--------|--------|---|
|   |  | Axis 1 | Axis 2 |   |
| { | "Axis-specific parameter No. 86, Zone 1 max."<br>(Value is set in units of 0.001 mm) | 300000 | 200000 | } |
|   | "Axis-specific parameter No. 87, Zone 1 min."<br>(Value is set in units of 0.001 mm) | 150000 | 100000 |   |





## 1-18 Communication

## ● OPEN (Open channel)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                   |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-------------------|------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1         | Operand 2  |                          |
| Optional                                  | Optional                       | OPEN                    | Channel<br>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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                   |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-------------------|------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1         | Operand 2  |                          |
| Optional                                  | Optional                       | CLOS                    | Channel<br>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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                   |                  | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-------------------|------------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1         | Operand 2        |                          |
| Optional                                  | Optional                       | READ                    | Channel<br>number | Column<br>number | CC                       |

[Function] Read a character string from the channel specified in operand 1 to the column specified in operand 2. Read will end when the character specified by a SCHA command is received. Either a local or global column may be specified. A return code will be stored in a local variable (factory setting: variable 99) immediately after this command is executed. Whether or not the command has been executed successfully can be checked by checking the return code. If necessary, specify the processing to be performed when the command has been aborted due to an error. Specifying "0" in operand 2 will execute a dummy read (clear the receive buffer and disable receive) (return code: successful completion). The tool versions that support "0" input in operand 2 are shown below. Even if "0" cannot be input from a tool, indirect specification is still available.  
(Note) Dummy read (operand 2 = 0) is not supported by channel Nos. 31 to 34 (Ethernet option).

[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.
TRAN      1      99     Assign the return code (variable 99) to variable 1.
CLOS      1      Close the channel.
SLCT
The program branches to the processing corresponding to each return code.
(Note) Using a GOTO command to branch out of or into a SLCT-EDSL syntax
is prohibited.

WHEQ      1      0      If the content of variable 1 is "0" (successful completion), [1] will be executed.
:
[1]
:
WHEQ      1      1      If the content of variable 1 is "1" (timeout), [2] will be executed. If necessary,
:
[2]
:
WHEQ      11     2      If the content of variable 1 is "2" (timer cancelled), [3] will be executed. If
:
[3]
:
OTHE
:
[4]
:
EDSL
If any of the conditions is satisfied and the corresponding command is executed,
the processing flow will move here.
:
(The remainder is omitted.)
:

```

- Return code of the READ command

The return code is stored in a local variable. The variable number can be set by "Other parameter No. 24." The default variable number is "99."

- 0: READ completed successfully (Receive complete)
- 1: READ timeout (the timeout value is set by a TMRD command) (Continue to receive)
- 2: READ 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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |               |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|---------------|------------|--------------------------|
|   |                                | Command,<br>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 (factory setting: variable 99) immediately after the READ command is executed.  
 If necessary, specify the processing to be performed upon timeout.)  
 Setting the timer to "0" will allow the READ command to wait infinitely, without timeout, until the end character is read.  
 The timer setting is input in seconds (setting range: 0 to 99.00 seconds) including up to two decimal places.

(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.  |
| SLCT |    |    | The program branches to the processing corresponding to each return code.<br>(Note) Using a GOTO command to branch out of or into a SLCT-EDSL syntax is prohibited. |
| WHEQ | 1  | 0  | If the content of variable 1 is "0" (successful completion), [1] will be executed. Specify in [1] the processing to be performed upon successful completion.        |
| :    |    |    | [1]   |
| :    |    |    | :   |
| WHEQ | 1  | 1  | If the content of variable 1 is "1" (timeout), [2] will be executed. If necessary, specify the applicable processing in [2].  |
| :    |    |    | [2]   |
| :    |    |    | :   |
| WHEQ | 11 | 2  | If the content of variable 1 is "2" (timer cancelled), [3] will be executed. If necessary, specify the applicable processing in [3].                                |
| :    |    |    | [3]   |
| :    |    |    | :   |
| OTHE |    |    | If the content of variable 1 is not "0," "1" or "2," [4] will be executed. If necessary, specify the applicable error handling in [4].                              |
| :    |    |    | [4]   |
| :    |    |    | :   |
| EDSL |    |    | If any of the conditions is satisfied and the corresponding command is executed, the processing flow will move here.  |

Read completes successfully within 30 seconds → Variable No. 1 = 0

Timeout occurs → Variable No. 1 = 1

\* The return code of READ command may not be limited to 0 or 1. The variable to store the return code can be set in "Other parameter No. 24". Refer to the explanation of READ command for details.



- WRIT (Write)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                   |                  | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-------------------|------------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1         | Operand 2        |                          |
| Optional                                  | Optional                       | WRIT                    | Channel<br>number | Column<br>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.   |



- SCHA (Set end character)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                   |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-------------------|------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1         | Operand 2  |                          |
| Optional                                  | Optional                       | SCHA                    | Character<br>code | Prohibited | CP                       |

[Function] Set the end character to be used by a READ or WRIT command.  
Any character from 0 to 255 (character code used in BASIC, etc.) can be specified.

[Example] Refer to the sections on READ and WRIT commands.



## 1-19 String Operation

## ● SCPY (Copy character string)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                  |                                     | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|------------------|-------------------------------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1        | Operand 2                           |                          |
| Optional                                  | Optional                       | SCPY                    | Column<br>number | Column number,<br>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.





- SGET (Get character)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                    |                                     | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------------|-------------------------------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1          | Operand 2                           |                          |
| Optional                                  | Optional                       | SGET                    | Variable<br>number | Column number,<br>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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                  |           | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|------------------|-----------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1        | Operand 2 |                          |
| Optional                                  | Optional                       | SPUT                    | Column<br>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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                  |           | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|------------------|-----------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1        | Operand 2 |                          |
| Optional                                  | Optional                       | STR                     | Column<br>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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                  |           | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|------------------|-----------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1        | Operand 2 |                          |
| Optional                                  | Optional                       | STRH                    | Column<br>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 convert 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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                    |                                     | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------------|-------------------------------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1          | Operand 2                           |                          |
| Optional                                  | Optional                       | VAL                     | Variable<br>number | Column number,<br>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). |





- SLEN (Set length)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                            |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|----------------------------|------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1                  | Operand 2  |                          |
| Optional                                  | Optional                       | SLEN                    | Character<br>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,<br>VALH | ..... | Decimal part is invalid. |
| STR          | ..... | Decimal part is valid.   |

[Example] Refer to the examples of the above commands:



## 1-20 Palletizing-Related

## ● BGPA (Declare start of palletizing setting)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                       |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-----------------------|------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1             | Operand 2  |                          |
| Optional                                  | Optional                       | BGPA                    | Palletizing<br>number | Prohibited | CP                       |

Declare the start of a palletizing setting.

Once this command is executed, palletizing setting for the palletizing number specified in operand 1 will be enabled.

(In the case of an ACHZ, AEXT, OFAZ or ATRG command, setting is enabled without declaring BGPA.)

The input range of palletizing number is from 1 to 10.

When the palletizing setting is complete, execute EDPA.

Nested BGPAs are not supported. To declare start of another palletizing setting, execute an EDPA command and then execute a BGPA command again.

If the output field is specified, the output will turn ON after this command is executed.

Palletizing numbers are in the local range. Therefore, a given palletizing setting is valid only within the program in which it is set.

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

## ● EDPA (Declare end of palletizing setting)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |            |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|------------|------------|--------------------------|
|   |                                | Command,<br>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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |           |           | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-----------|-----------|--------------------------|
|   |                                | Command,<br>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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                   |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-------------------|------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1         | Operand 2  |                          |
| Optional                                  | Optional                       | PAPN                    | Pattern<br>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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |             |             | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-------------|-------------|--------------------------|
|   |                                | Command,<br>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 precision.

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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |           |           | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-----------|-----------|--------------------------|
|   |                                | Command,<br>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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                      |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|----------------------|------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1            | Operand 2  |                          |
| Optional                                  | Optional                       | PAST                    | (Position<br>number) | Prohibited | CP                       |

Set the reference point used in palletizing.

If a value is set in operand 1, that position number specified in operand 1 will be used to store the reference point data.

If no value is set in operand 1, the position-number setting for storing reference point data will become invalid.

This command is used in conjunction with PASE and PAPT.

If this command is not set, coordinates (0, 0) are used as the reference point. If this command is set, the set coordinates are used as the reference point in calculating the position coordinates of palletizing points.

Coordinates in both the PX and PY-axis directions must always be set as the reference-point coordinates. If a palletizing movement command such as PMVP or PMVL is executed, however, specification of palletizing Z-axis (PZ-axis) coordinate is optional. If a Z-axis coordinate is specified, movement in the PZ-axis direction will become enabled. Even if PZ-axis coordinate is not specified, operation will still be performed—just that the position will not move in the PZ-axis direction. Note, however, that an error will generate in the following cases: If this command and PZ-axis are set but the PX, PY and PZ-axes are not set as valid axes in the reference point data, an error will generate when position coordinates are calculated. If the palletizing Z-axis is not set and the PX and PY-axes are not set as valid axes in the reference point data, an error will also generate when position coordinates are calculated. “When position coordinates are calculated” means when PAPG (get palletizing calculation data) or any palletizing movement command such as PMVP, PMVL or PACH is executed. If this command is executed before BGPA is declared (= while palletizing setting is not enabled), an error will generate.

If the output field is specified, the output will turn ON after this command is executed.



- PAPS (Set palletizing points) For 3-point teaching

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                    |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------------|------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1          | Operand 2  |                          |
| Optional                                  | Optional                       | PAPS                    | Position<br>number | Prohibited | CP                       |

Specify the first position number among the three position numbers containing point data, 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.

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 a palletizing Z-axis (PZ-axis) is already declared, there must be two valid axes excluding the PZ-axis. If a PZ-axis is not declared yet, there must be two or three valid axes. 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.



- PSLI (Set zigzag)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                  |           | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|------------------|-----------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1        | Operand 2 |                          |
| Optional                                  | Optional                       | PSLI                    | Offset<br>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 (3) "Palletizing Setting" – (d) "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) Only when there are at least three axes.

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |               |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|---------------|------------|--------------------------|
|   |                                | Command,<br>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.

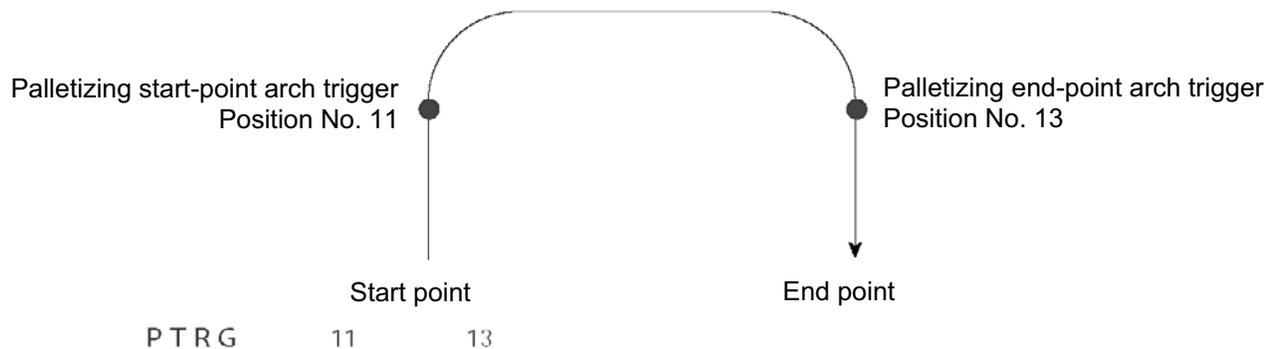
- PTRG (Set palletizing arch triggers)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                    |                    | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------------|--------------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1          | Operand 2          |                          |
| Optional                                  | Optional                       | PTRG                    | Position<br>number | Position<br>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 palletizing Z-axis (PZ-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.



(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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                      |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|----------------------|------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1            | Operand 2  |                          |
| Optional                                  | Optional                       | PEXT                    | (Position<br>number) | Prohibited | CP                       |

Set palletizing composition.

The position number specified in operand 1 will be set for use in composition.

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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |              |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------|------------|--------------------------|
|   |                                | Command,<br>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 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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |             |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-------------|------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1   | Operand 2  |                          |
| Optional                                  | Optional                       | ACHZ                    | Axis number | Prohibited | CP                       |

Specify the axis number representing the arch-motion Z direction.

The axis number specified in operand 1 will be set as the axis number representing the arch-motion Z direction.  
If the output field is specified, the output will turn ON after this command is executed.

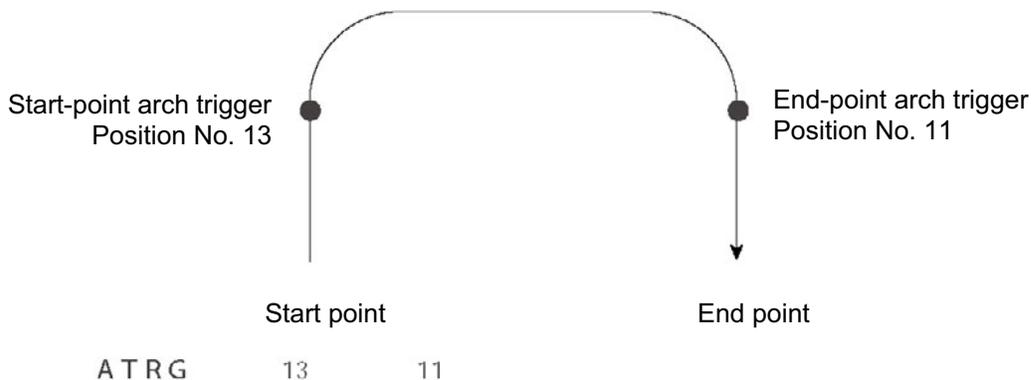
- ATRG (Set arch triggers)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                    |                    | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------------|--------------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1          | Operand 2          |                          |
| Optional                                  | Optional                       | ATRG                    | Position<br>number | Position<br>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.



(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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                      |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|----------------------|------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1            | Operand 2  |                          |
| Optional                                  | Optional                       | AEXT                    | (Position<br>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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |              |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------|------------|--------------------------|
|   |                                | Command,<br>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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                       |                    | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-----------------------|--------------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1             | Operand 2          |                          |
| Optional                                  | Optional                       | PTNG                    | Palletizing<br>number | Variable<br>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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                       |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-----------------------|------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1             | Operand 2  |                          |
| Optional                                  | Optional                       | PINC                    | Palletizing<br>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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                       |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-----------------------|------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1             | Operand 2  |                          |
| Optional                                  | Optional                       | PDEC                    | Palletizing<br>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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                       |           | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-----------------------|-----------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1             | Operand 2 |                          |
| Optional                                  | Optional                       | PSET                    | Palletizing<br>number | Data      | CC                       |

Set the value specified in operand 2 as the palletizing position number for the palletizing number specified in operand 1.

If the specified value is considered normal as a palletizing position calculated under the current palletizing setting, the value will be set. If not, the value will not be set.

If the output field is specified, the output will turn ON when the palletizing position number was successfully updated, and turn OFF if the update failed.



- PARG (Get palletizing angle)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                       |             | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-----------------------|-------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1             | Operand 2   |                          |
| Optional                                  | Optional                       | PARG                    | Palletizing<br>number | Axis number | CP                       |

Obtain the palletizing angle.

Calculate the palletizing angle (degrees) from the physical axis specified in operand 2 for the palletizing number specified in operand 1, and store the result in variable 199.

This command need not be executed, if not necessary.

If this command is executed after PAPS (set 3 palletizing points for teaching) is executed, the angle formed by the preferential axis and the specified physical axis will be calculated automatically. If this command is executed before PAPS is executed, or after both PAPS and PASE are executed in this order, an error will generate.

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 palletizing Z-axis (PZ-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.

The actual operating direction may have been reversed depending on the mechanism of the rotating axis and the setting of axis-specific parameter No. 6, "Operating-direction reversing selection." To use the value obtained by this command, be sure to confirm the actual operating direction.

If the output field is specified, the output will turn ON after this command is executed.

- PAPG (Get palletizing calculation data)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                       |                    | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-----------------------|--------------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1             | Operand 2          |                          |
| Optional                                  | Optional                       | PAPG                    | Palletizing<br>number | Position<br>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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                       |                      | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-----------------------|----------------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1             | Operand 2            |                          |
| Optional                                  | Optional                       | PMVP                    | Palletizing<br>number | (Position<br>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 palletizing Z-axis (PZ-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 PZ-direction position will move to the height of the specified position number by ignoring the palletizing calculation (only when three or more axes are available). Any data other than PZ-axis data contained in the position number specified in operand 2 will be ignored.

Absence of Z-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.



- PMVL (Move to palletizing points via interpolation)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                       |                      | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-----------------------|----------------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1             | Operand 2            |                          |
| Optional                                  | Optional                       | PMVL                    | Palletizing<br>number | (Position<br>number) | PE                       |

Move to the calculated palletizing points via interpolation.

The axes will move to the palletizing points specified in operand 1, via interpolation.

If the palletizing points are valid only for the PX/PY-axes (when palletizing Z-axis (PZ-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 PZ-direction position will move to the height of the specified position number by ignoring the palletizing calculation (only when three or more axes are available). Any data other than PZ-axis data contained in the position number specified in operand 2 will be ignored.

Absence of Z-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.

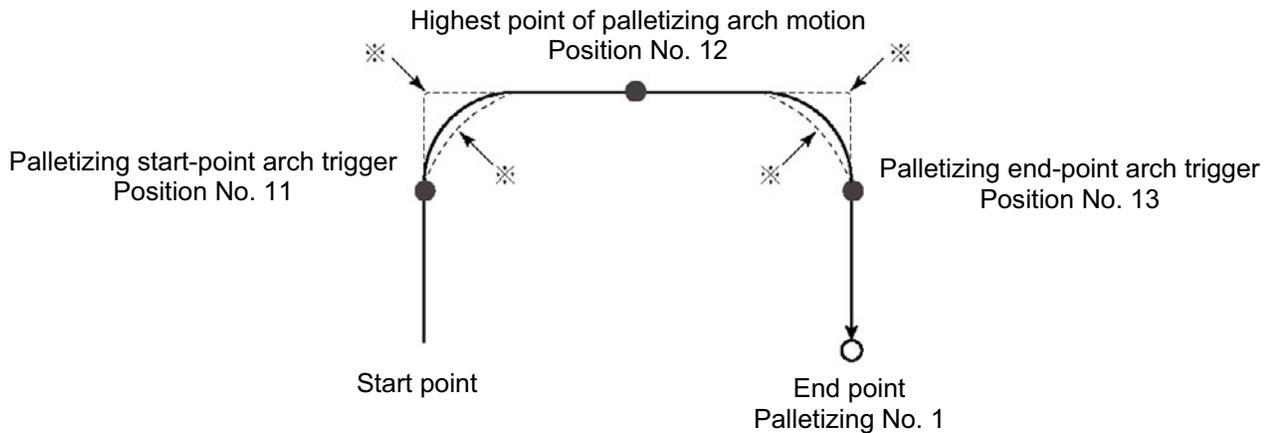
If palletizing Z-axis has not been declared, an error will generate.

## ● PACH (Palletizing-point arch motion)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                       |                    | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-----------------------|--------------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1             | Operand 2          |                          |
| Optional                                  | Optional                       | PACH                    | Palletizing<br>number | Position<br>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 pass near 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:

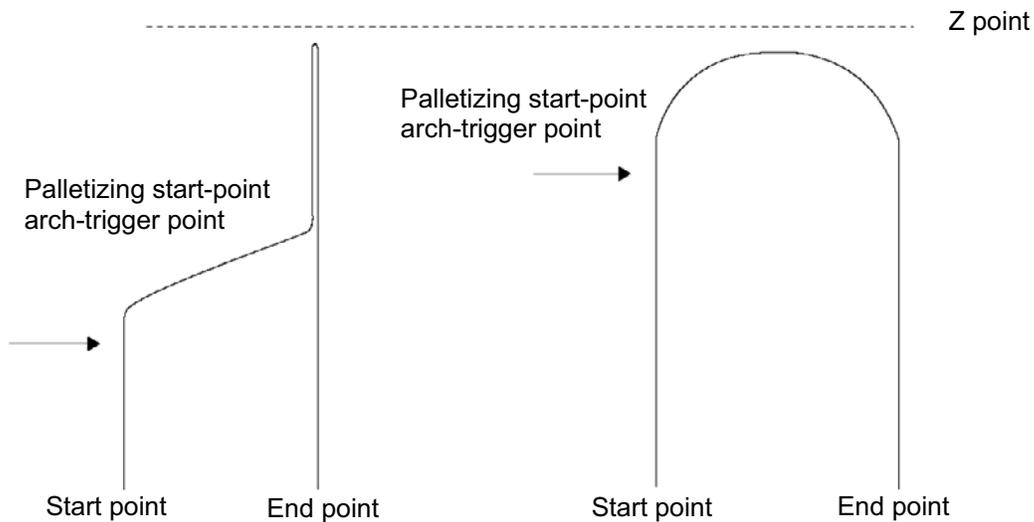


Fig. 5

In this case, change the palletizing arch triggers and Z point to increase the operation efficiency.

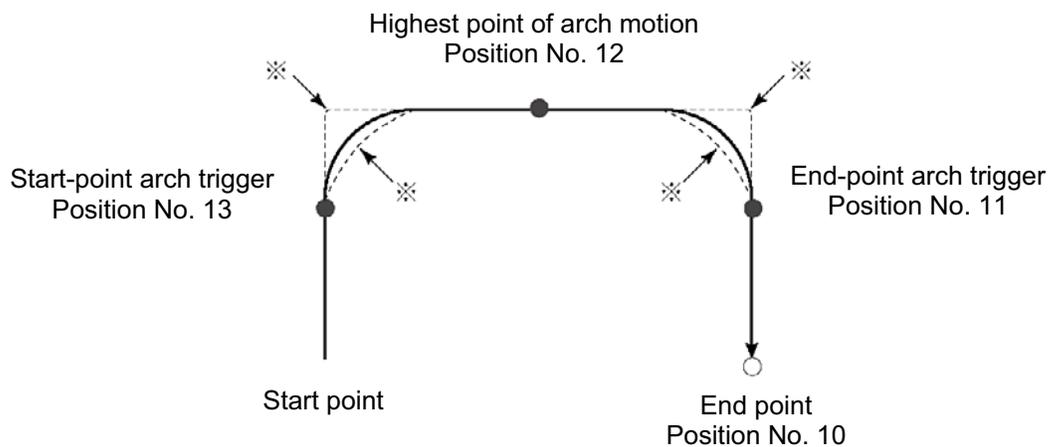
- If palletizing composition is set, axes other than the PX, PY and PZ-axes will also be operated if data is available for such axes. However, the composite axis will start/end operation at positions above the arch triggers.
- Executing this command will not increment the palletizing position number by 1.

## ● ARCH (Arch motion)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |                    |                    | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|--------------------|--------------------|--------------------------|
|   |                                | Command,<br>declaration | Operand 1          | Operand 2          |                          |
| Optional                                  | Optional                       | ARCH                    | Position<br>number | Position<br>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 pass near 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 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.



## 1-23 Building of Pseudo-Ladder Task

## ● CHPR (Change task level)

| Extension condition<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |           |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-----------|------------|--------------------------|
|   |                                | Command,<br>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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |           |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-----------|------------|--------------------------|
|   |                                | Command,<br>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<br>(LD, A, O, AB, OB) | Input condition<br>(I/O, flag) | Command, declaration    |           |            | Output<br>(Output, flag) |
|---|--------------------------------|-------------------------|-----------|------------|--------------------------|
|   |                                | Command,<br>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.



### 3. Key Characteristics of Actuator Control Commands and Points to Note

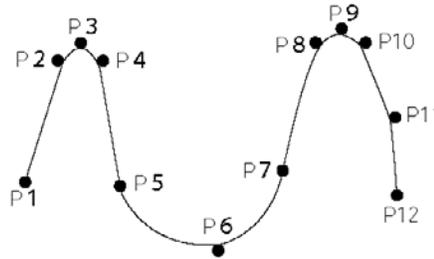
#### 3.1 Continuous Movement Commands

[PATH, CIR, ARC, PSPL, CIR2, ARC2, ARCD, ARCC, CIRS, ARCS]

[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.

```

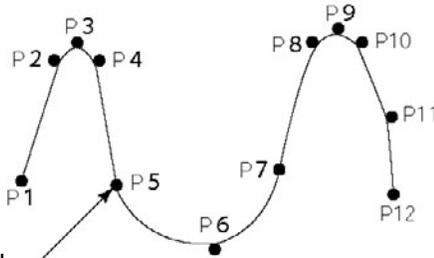
VEL      100
PATH    1      5
ARC2    6      7
PATH    8     12
    
```



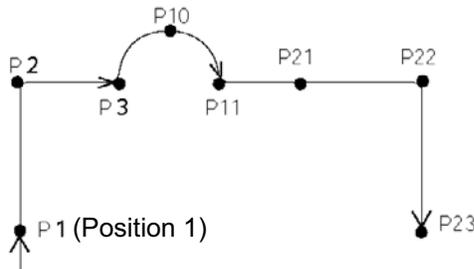
[2] Continuous movement will not be achieved if an input condition is specified for any continuous movement command.

```

VEL      100
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
VEL     100
...
PATH    1      3      316
ARC2   10     11     319
PATH   21     23     320
...
    
```

| Output field | Timing                                  |
|--------------|---|
| 316          | Turn ON as P1 approaches.               |
| 317          | Turn ON as P2 approaches.               |
| 318          | Turn ON as P3 approaches.               |
| 319          | Turn ON as P11 approaches.              |
| 320          | Turn ON as P21 approaches.              |
| 321          | Turn ON as P22 approaches.              |
| 322          | Turn ON when P23 operation is complete. |



[Example 2] (POTP = 0)

|      |     |    |     |
|------|-----|----|-----|
| VEL  | 100 |    |     |
| PATH | 1   | 3  | 316 |
| ARC2 | 10  | 11 | 319 |
| PATH | 21  | 23 | 320 |

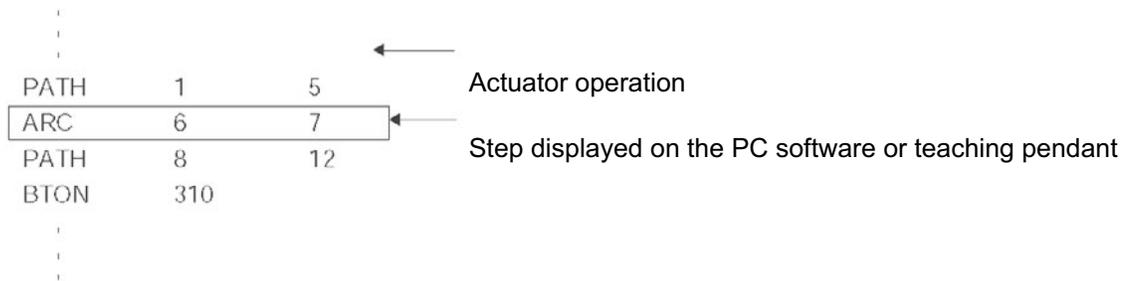
| Output field | Timing                                  |
|--------------|---|
| 316          | Turn ON as P3 approaches.               |
| 319          | Turn ON as P11 approaches.              |
| 320          | Turn ON when P23 operation is complete. |

[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.

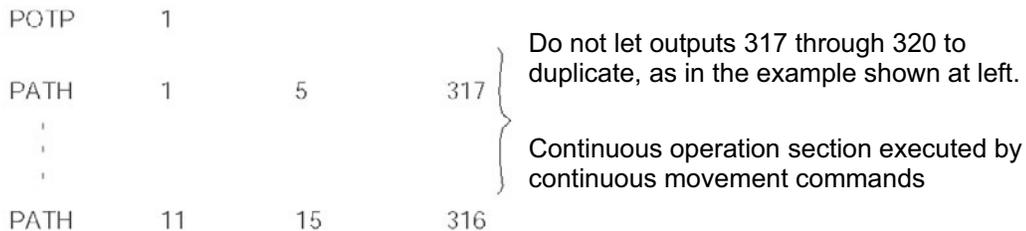
|    |      |    |    |     |
|----|------|----|----|-----|
|    | POTP | 1  |    |     |
|    | ⋮    |    |    |     |
|    | ⋮    |    |    |     |
|    | PATH | 1  | 3  | 316 |
| 20 | ARC2 | 10 | 11 | 319 |
|    | PATH | 21 | 23 | 320 |

| Output field | Timing                                  |
|--------------|---|
| 316          | Turn ON as P1 approaches.               |
| 317          | Turn ON as P2 approaches.               |
| 318          | Turn ON when P3 operation is complete.  |
| 319          | Turn ON as P11 approaches.              |
| 320          | Turn ON as P21 approaches.              |
| 321          | Turn ON as P22 approaches.              |
| 322          | 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.



[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.



The final output status of duplicate 317 through 320 is indeterminable, because it is affected by the positioning calculation time and the relationship of durations of actual operations.

### 3.2 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.

### 3.3 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.



### 3.4 CIR2/ARC2/ARCD/ARCC Commands

With a CIR2, ARC2, ARCD or ARCC command, the speed can be changed (only in the arc interpolation section) by inputting a speed for the point specified in operand 1. These commands are effective when you must lower the speed partially because the radius is small and the arc locus cannot be maintained inside the allowable range.

The speed and acceleration will take valid values based on the following priorities:

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



## 4. 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.

### 4.1 How to Use

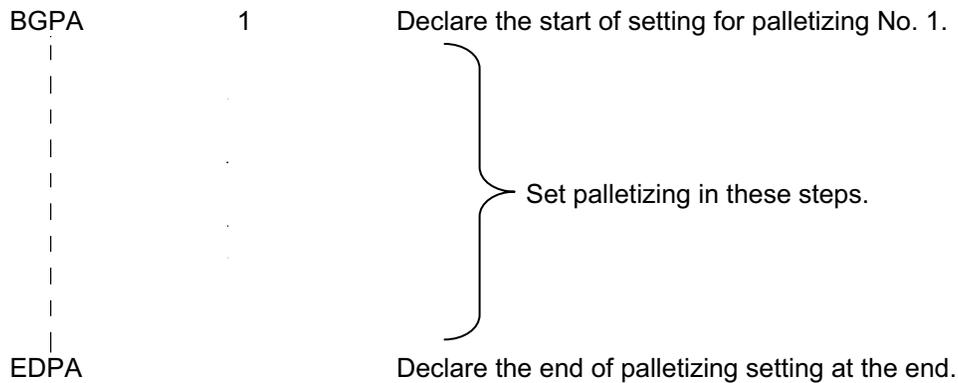
Use palletizing commands in the following steps:

- (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.

### 4.2 Palletizing Setting

Use the palletizing setting commands to set items necessary for palletizing operation. The setting items include the following:

- (1) Palletizing number setting --- Command: BGPA  
At the beginning of a palletizing setting, determine a palletizing number using a BGPA command to declare the start of palletizing setting.  
At the end, declare the end of palletizing setting using an EDPA command.



A maximum of 10 sets (palletizing Nos. 1 to 10) of palletizing setting can be specified for each program.

- (2) Palletizing pattern --- Command: PAPN  
 Select a pattern indicating the palletizing order.  
 The two patterns illustrated below are available.  
 The encircled numbers indicate the order of palletizing and are called "palletizing position numbers."

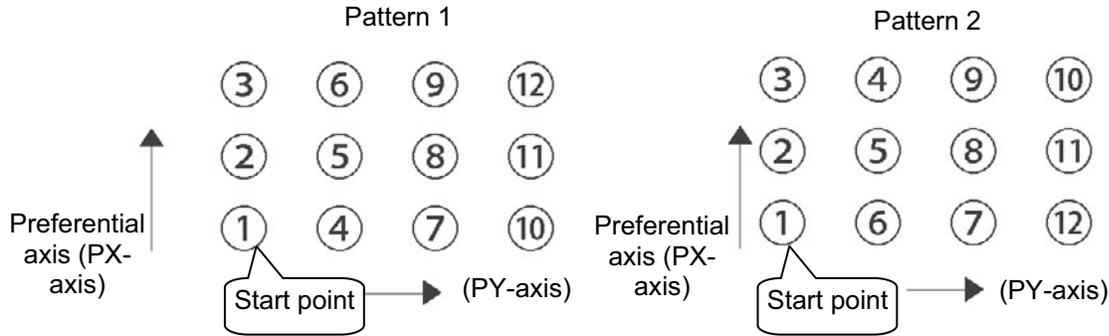


Fig. 1

PAPN      2      When pattern 2 is selected  
 (Setting is not necessary if pattern 1 is selected.)

The row from 1 to 3 to be placed first is called the "preferential axis (PX-axis)," while the other direction comprising the palletizing plane is called the "PY-axis."

- (3) Palletizing counts --- Command: PAPI  
 Set the palletizing counts.

PAPI      3      4      Count for preferential axis (PX-axis): 3, Count for PY-axis: 4

- (4) Palletizing position setting  
 Palletizing position setting is performed mainly by method A or B, as explained below. Set the palletizing positions for each palletizing setting based on method A or B.

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

### A. 3-point teaching method

To set the palletizing positions by 3-point teaching, store desired positions in position data fields as three continuous position data and then specify the first position number using a PAPS command.

This method allows you to set the PX-axis and PY-axis as three-dimensional axes not parallel with the actuators and not crossing with each other.

In the example shown below, position data [1], [3] and [10] are stored in three continuous position data fields.

When three points are taught from position No. 11

Position No. 11 [1]: Start point (First palletizing position)

Position No. 12 [3]: Palletizing position corresponding to the end point in the PX-axis direction

Position No. 13 [10]: Palletizing position corresponding to the end point in the PY-axis direction

The encircled numbers indicate palletizing position numbers (palletizing order).

Use a PAPS command to specify the position number corresponding to the start point.

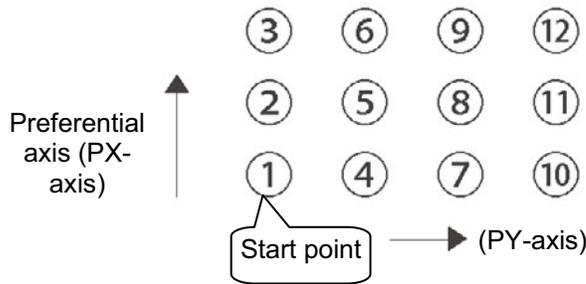


Fig. 1

PAPS 11

The pitches are calculated automatically from the count set for each axis.

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.

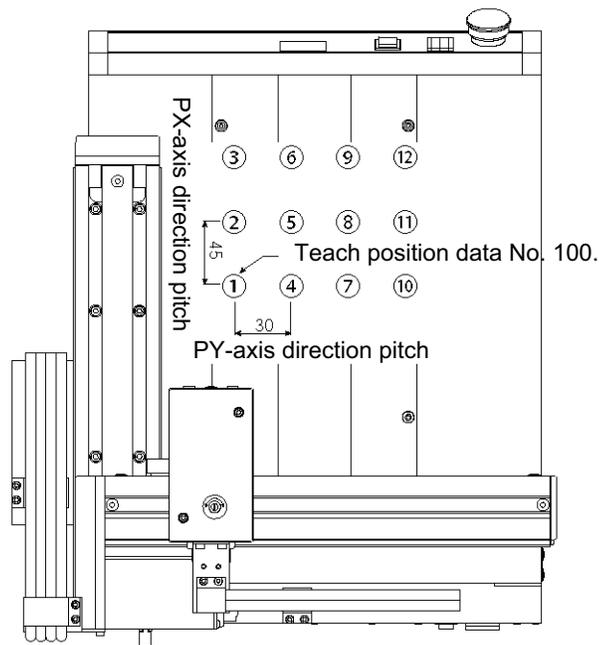
## B. Method to set palletizing positions in parallel with the actuators

**Palletizing reference point:** Store the position data of the start point (palletizing position No. 1) in a position data field and specify the applicable position number using a PAST command, as shown below.

**Palletizing pitches:** Use a PAPT command to specify the pitches in the PX-axis and PY-axis directions.

**Palletizing axes:** Use a PASE command to specify the two axes, one representing the PX-axis direction and the other representing the PY-axis direction, to be used in palletizing.

(An actuator axis number parallel with the preferential axis (PX-axis) and another perpendicular to the preferential axis)



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

(Note) When the above palletizing axes, palletizing pitches and palletizing reference point are used, the PX-axis and PY-axis must be parallel with the actuators and crossing with each other.

Select either method A or B for each palletizing setting.



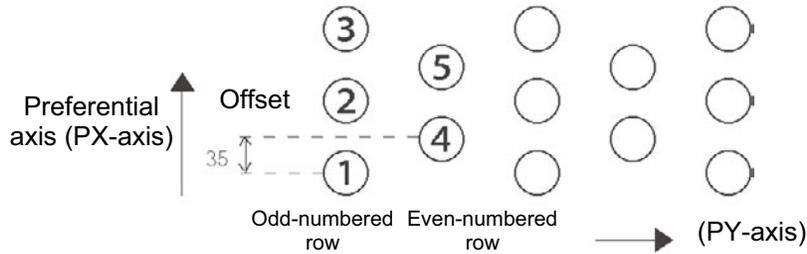
## (5) Zigzag setting --- Command: PSLI

Use a PSLI command to set a zigzag layout as shown below.

Zigzag offset: Offset amount in the preferential-axis direction, which will be applied when even-numbered rows are placed.

"Even-numbered rows" refer to the rows occurring at the even numbers based on the row placed first representing the first row.

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



PSLI 35 2

## (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

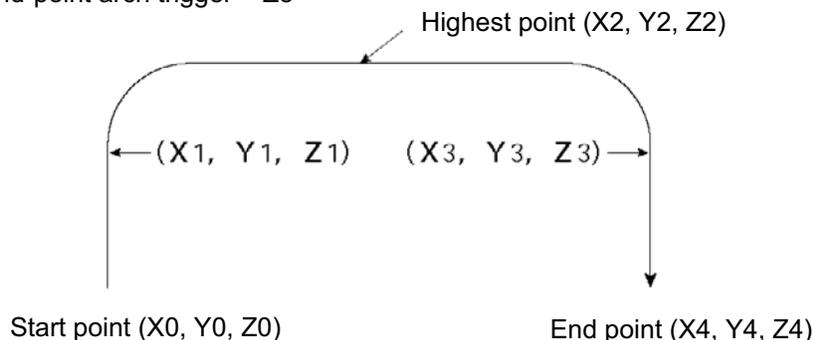
Specify when to start moving in other axis direction after the start of arch motion from the start point, as an arch-motion Z-direction coordinate position reached.

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. Examples include rotation angle. 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

### 4.3 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) ( $\theta$  in the figure below).  $\theta$  indicates an angle calculated by ignoring the coordinate in the palletizing Z-axis direction.  
In the figure below,  $\theta$  will become a negative value if axis 1 is used as the reference for angle calculation.

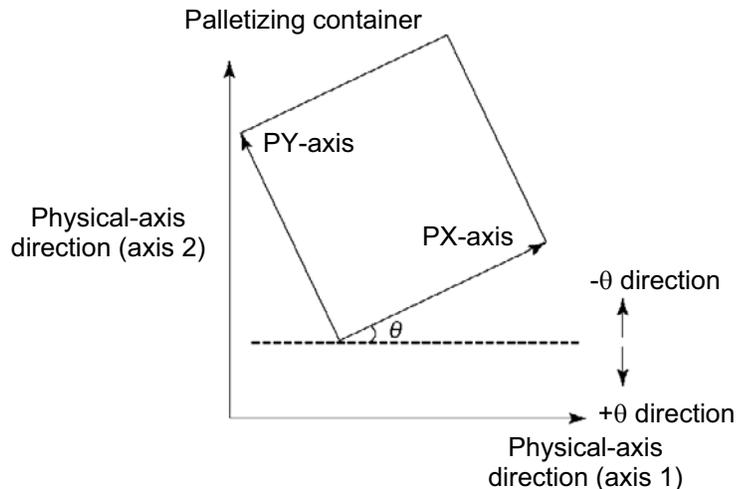


Fig. 4

Executing a "get palletizing angle" command (PARG) 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.

- (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.

### 4.4 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.)

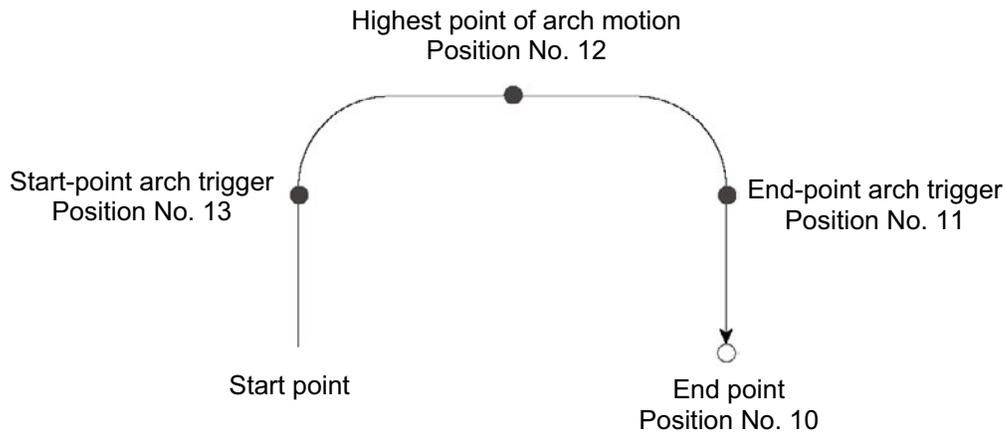
Two actuator axes will be required to comprise a two-dimensional plane. If a vertical axis (PZ-axis) is required, another axis must be set.

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.

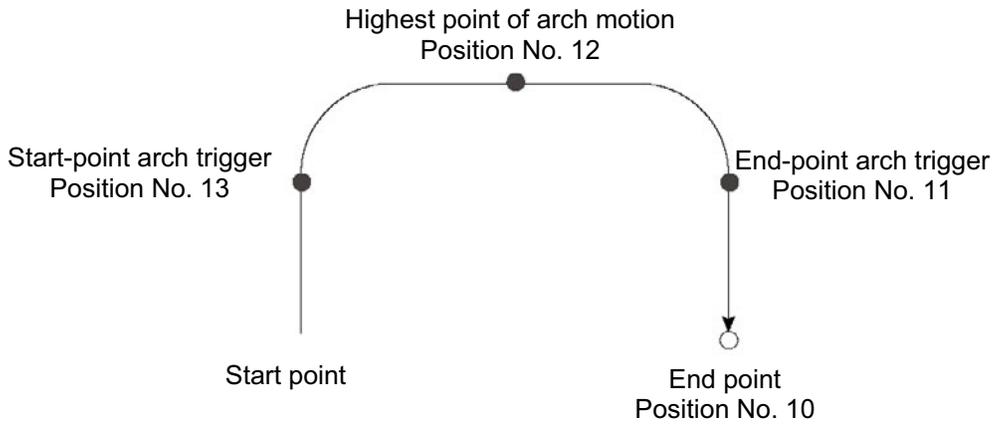
Palletizing arch motion must be set in a palletizing setting.



|      |    |    |
|------|----|----|
| 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 |



### 4.5 Program Examples

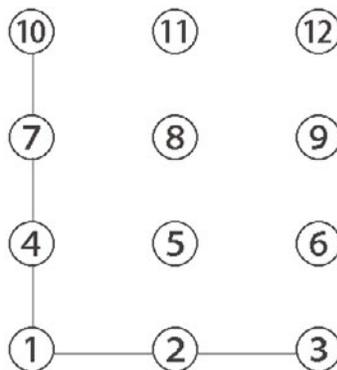
- (1) Simple program example (two-axis specification) using PAPS (set by 3-point teaching)  
 The example below specifies movement only and does not cover picking operation.

4. Palletizing Function

| Step | E | N | Cnd | Cmd  | Operand 1 | Operand 2 | Pst | Comment   |
|------|---|---|-----|------|-----------|-----------|-----|---|
| 1    |   |   |     | BGPA | 1         |           |     | Start setting palletizing No. 1.                |
| 2    |   |   |     | PAPI | 3         | 4         |     | Palletizing counts: 3 x 4                       |
| 3    |   |   |     | PAPS | 2         |           |     | Set by 3-point teaching.                        |
| 4    |   |   |     | EDPA |           |           |     | End setting palletizing No. 1.                  |
| 5    |   |   |     |      |           |           |     |   |
| 6    |   |   |     | VEL  | 200       |           |     | Speed: 200 mm/sec                               |
| 7    |   |   |     | MOVL | 1         |           |     | Move to picking position.                       |
| 8    |   |   |     | PSET | 1         |           |     | Set palletizing position number to 1.           |
| 9    |   |   |     | TAG  | 1         | 1         |     |   |
| 10   |   |   |     | PMVL | 1         |           |     | Move to palletizing position via interpolation. |
| 11   |   |   |     | MOVL | 1         |           |     | Move to picking position via interpolation.     |
| 12   |   |   |     | PINC | 1         |           | 600 | Increment palletizing position number by 1.     |
| 13   |   |   | 600 | GOTO | 1         |           |     | Beginning of loop if PINC is successful.        |
| 14   |   |   |     | EXIT |           |           |     | End   |

| No. | Axis 1  | Axis 2  | Vel | Acc | Dcl | Remarks                         |
|-----|---------|---------|-----|-----|-----|---------------------------------|
| 1   | 10.000  | 10.000  |     |     |     | Picking position                |
| 2   | 70.000  | 70.000  |     |     |     | Reference-point position data   |
| 3   | 148.000 | 71.000  |     |     |     | PX-axis end-point position data |
| 4   | 69.000  | 143.000 |     |     |     | PY-axis end-point position data |

PY-axis end-point coordinates  
 Position No. 4  
 (69, 143)



Reference point  
 Position No. 2  
 (70, 70)

PX-axis end-point coordinates  
 Position No. 3  
 (148, 71)

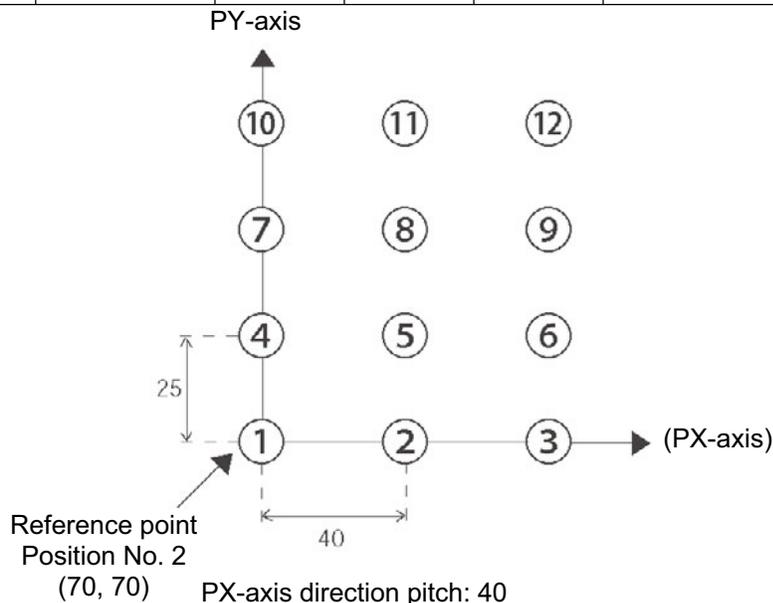
- Picking position  
 Position No. 1



(2) Simple program example (two-axis specification) using PAPS, PAPT and PAST  
 The example below specifies movement only and does not cover picking operation.

| Step | E | N | Cnd | Cmd  | Operand 1 | Operand 2 | Pst | Comment   |
|------|---|---|-----|------|-----------|-----------|-----|---|
| 1    |   |   |     | BGPA | 1         |           |     | Start setting palletizing No. 1.                |
| 2    |   |   |     | PAPI | 3         | 4         |     | Palletizing counts: 3 x 4                       |
| 3    |   |   |     | PASE | 1         | 2         |     | PX-axis = Axis 1, PY-axis = Axis 2              |
| 4    |   |   |     | PAPT | 40        | 25        |     | Pitch: X = 40, Y = 25                           |
| 5    |   |   |     | PAST | 2         |           |     | Position No. 2 as reference point               |
| 6    |   |   |     | EDPA |           |           |     | End setting palletizing No. 1.                  |
| 7    |   |   |     |      |           |           |     |   |
| 8    |   |   |     | VEL  | 200       |           |     | Speed: 200 mm/sec                               |
| 9    |   |   |     | MOVL | 1         |           |     | Move to picking position.                       |
| 10   |   |   |     | PSET | 1         | 1         |     | Set palletizing position number to 1.           |
| 11   |   |   |     | TAG  | 1         |           |     |   |
| 12   |   |   |     | PMVL | 1         |           |     | Move to palletizing position via interpolation. |
| 13   |   |   |     | MOVL | 1         |           |     | Move to picking position via interpolation.     |
| 14   |   |   |     | PINC | 1         |           | 600 | Increment palletizing position number by 1.     |
| 15   |   |   | 600 | GOTO | 1         |           |     | Beginning of loop if PINC is successful.        |
| 16   |   |   |     | EXIT |           |           |     | End   |

| No. | Axis 1 | Axis 2 | Vel | Acc | Dcl | Remarks                       |
|-----|--------|--------|-----|-----|-----|-------------------------------|
| 1   | 10.000 | 10.000 |     |     |     | Picking position              |
| 2   | 70.000 | 70.000 |     |     |     | Reference-point position data |



- Picking position Position No. 1

PX-axis direction pitch: 40  
 PY-axis direction pitch: 25  
 The PX-axis and PY-axis are parallel with axis 1 and axis 2, respectively.



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

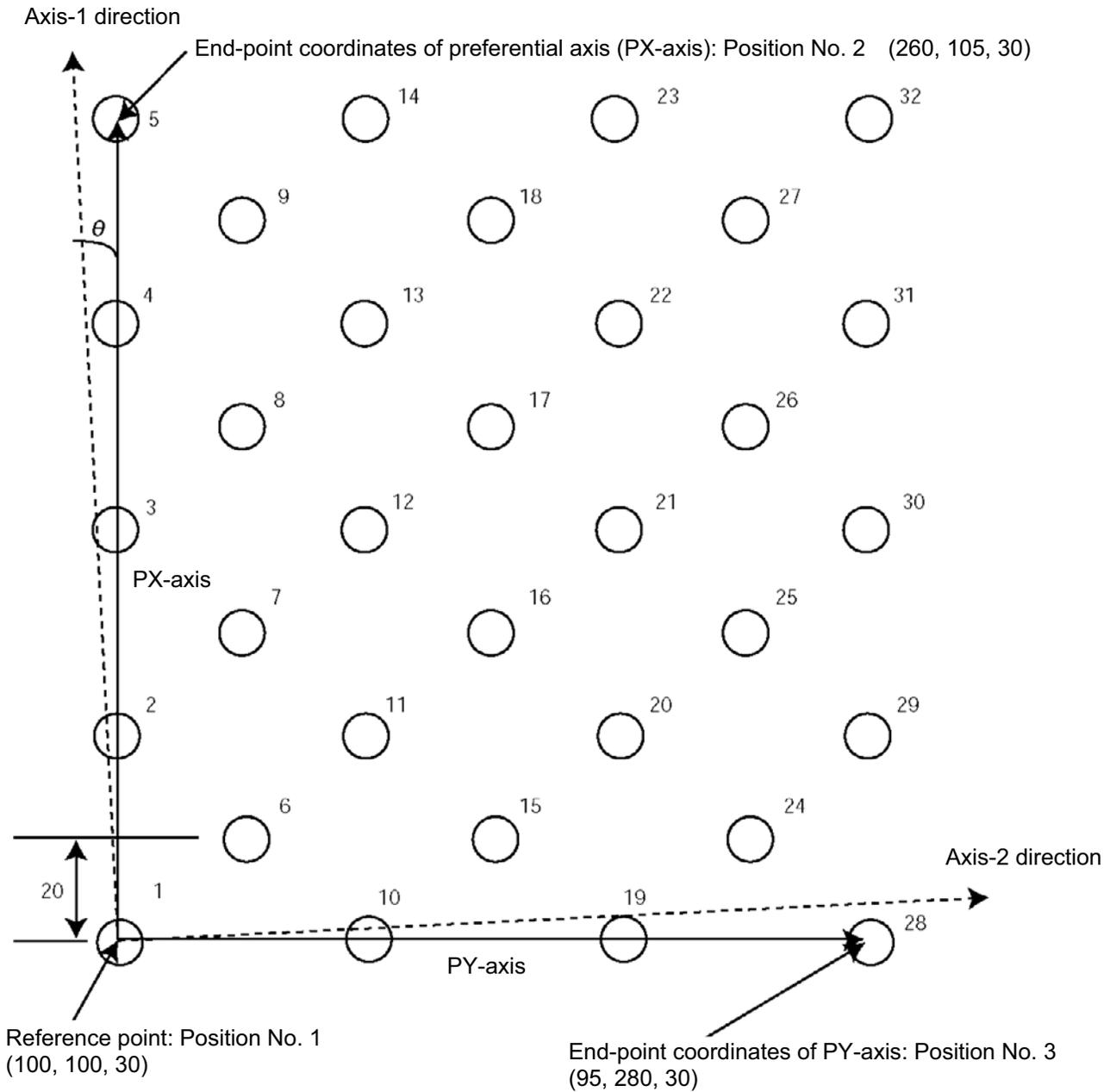
| Step | E | N | Cnd | Cmd                                      | Operand 1 | Operand 2 | Pst | Comment                               |  |
|------|---|---|-----|--|-----------|-----------|-----|---------------------------------------|--|
| 1    |   |   |     | BGPA                                     | 1         |           |     | Start setting palletizing No. 1.      |  |
| 2    |   |   |     |  |           |           |     |                                       |  |
| 3    |   |   |     | PAPI                                     | 5         | 7         |     | Palletizing counts: 5 x 7             |  |
| 4    |   |   |     | PAPN                                     | 1         |           |     | Palletizing pattern 1                 |  |
| 5    |   |   |     | PAPS                                     | 1         |           |     | Set by 3-point teaching.              |  |
| 6    |   |   |     |  |           |           |     | Use position No. 1 data.              |  |
| 7    |   |   |     | PSLI                                     | 20        | 4         |     | Zigzag offset = 20 mm                 |  |
| 8    |   |   |     | PCHZ                                     | 3         |           |     | Palletizing Z-axis = Axis 3           |  |
| 9    |   |   |     | PTRG                                     | 4         | 4         |     | Set palletizing arch triggers.        |  |
| 10   |   |   |     |  |           |           |     | Use position No. 4 data.              |  |
| 11   |   |   |     | OFPZ                                     | 10        |           |     | PZ-axis offset = 10 mm                |  |
| 12   |   |   |     |  |           |           |     |                                       |  |
| 13   |   |   |     |  |           |           |     | Use position No. 6 data.              |  |
| 14   |   |   |     | EDPA                                     |           |           |     |                                       |  |
| 15   |   |   |     |  |           |           |     |                                       |  |
| 16   |   |   |     |  |           |           |     |                                       |  |
| 17   |   |   |     |  |           |           |     |                                       |  |
| 18   |   |   |     |  |           |           |     |                                       |  |
| 19   |   |   |     |  |           |           |     |                                       |  |
| 20   |   |   |     | * ////////////////////////////////////// |           |           |     |                                       |  |
| 21   |   |   |     |  |           |           |     |                                       |  |
| 22   |   |   |     | ATRG                                     | 4         | 4         |     | Set arch triggers.                    |  |
| 23   |   |   |     |  |           |           |     | Use position No. 4 data.              |  |
| 24   |   |   |     | ACHZ                                     | 3         |           |     | Set arch-motion Z-axis.               |  |
| 25   |   |   |     |  |           |           |     |                                       |  |
| 26   |   |   |     | ACC                                      | 0.3       |           |     | Acceleration                          |  |
| 27   |   |   |     | DCL                                      | 0.3       |           |     | Deceleration                          |  |
| 28   |   |   |     | VLMX                                     |           |           |     |                                       |  |
| 29   |   |   |     |  |           |           |     |                                       |  |
| 30   |   |   |     | PSET                                     | 1         | 1         |     | Set palletizing position number to 1. |  |



| Step | E | N | Cnd | Cmd  | Operand 1 | Operand 2 | Pst | Comment  |
|------|---|---|-----|------|-----------|-----------|-----|--|
| 31   |   |   |     | MOVP | 8         |           |     | Move to picking position.                      |
| 32   |   |   |     |      |           |           |     |  |
| 33   |   |   |     | TAG  | 1         |           |     | Beginning of loop processing                   |
| 34   |   |   |     | PACH | 1         | 9         |     | Palletizing arch motion                        |
| 35   |   |   |     |      |           |           |     | Z point specified by Position No. 9            |
| 36   |   |   |     | ARCH | 8         | 9         |     | Arch motion                                    |
| 37   |   |   |     |      |           |           |     | Z point specified by Position No. 9            |
| 38   |   |   |     | PINC | 1         |           | 600 | Increment palletizing position number by 1.    |
| 39   |   |   | 600 | GOTO | 1         |           |     | Go to beginning of loop if PINC is successful. |
| 40   |   |   |     |      |           |           |     |  |
| 41   |   |   |     | EXIT |           |           |     | End of task                                    |
| 42   |   |   |     |      |           |           |     |  |
| 43   |   |   |     |      |           |           |     |  |
| 44   |   |   |     |      |           |           |     |  |
| 45   |   |   |     |      |           |           |     |  |

| No. | Axis 1  | Axis 2  | Axis 3 | Remarks                     |
|-----|---------|---------|--------|-----------------------------|
| 1   | 100.000 | 100.000 | 30.000 | Reference point data        |
| 2   | 260.000 | 105.000 | 30.000 | PX-axis end-point data      |
| 3   | 95.000  | 280.000 | 30.000 | PY-axis end-point data      |
| 4   | *.***   | *.***   | 10.000 | Arch-trigger point data     |
| 5   | *.***   | *.***   | *.***  | (Not used)                  |
| 6   | *.***   | *.***   | *.***  |                             |
| 7   | *.***   | *.***   | *.***  | (Not used)                  |
| 8   | 0.000   | 0.000   | 30.000 | Picking-position point data |
| 9   | *.***   | *.***   | 0.000  | Z point data                |
| 10  |         |         |        |                             |

Schematic diagram of placement-point positions based on the above program



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



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

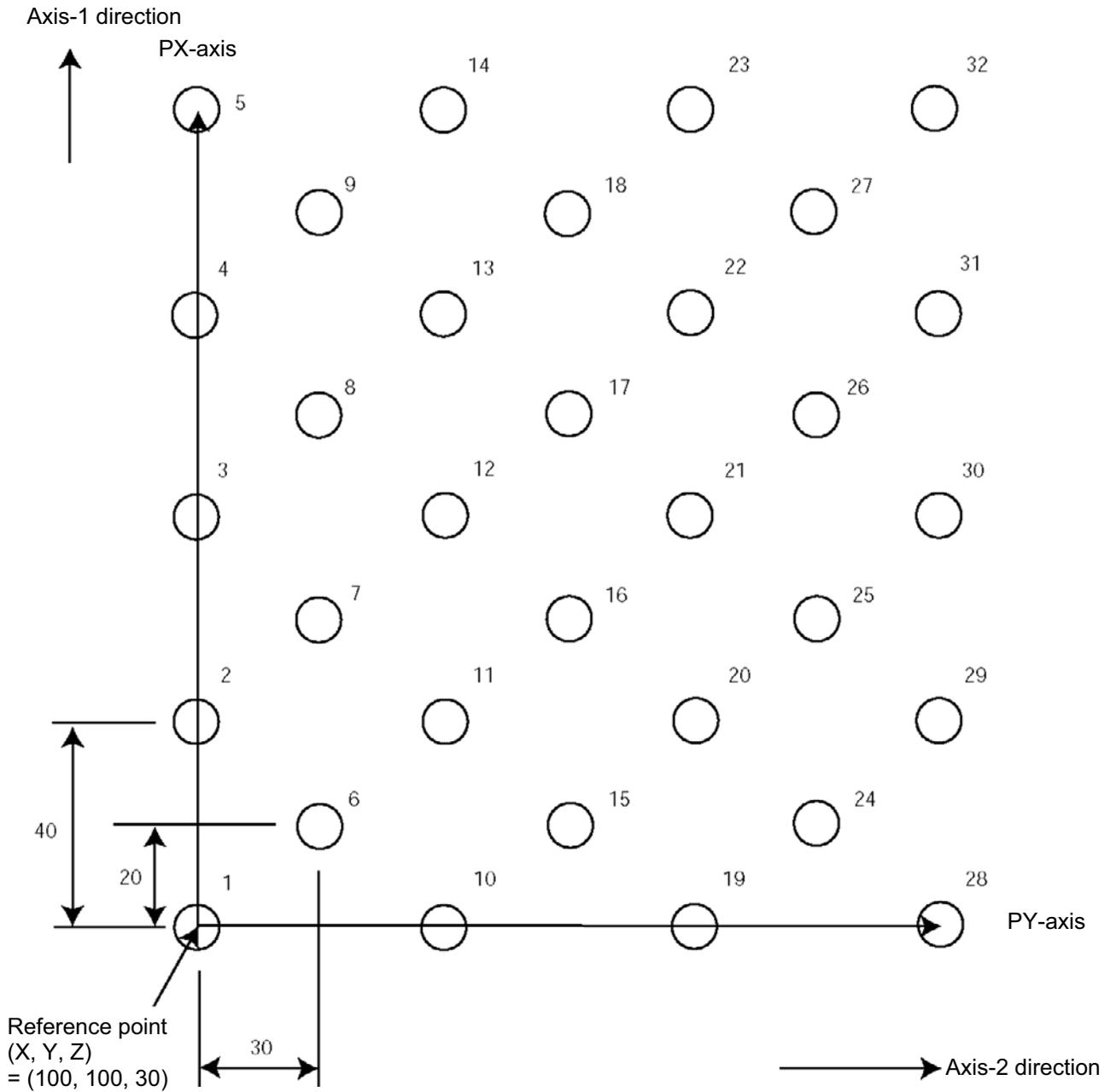
| Step | E  | N | Cnd | Cmd  | Operand 1 | Operand 2 | Pst | Comment                            |
|------|--|---|-----|------|-----------|-----------|-----|------------------------------------|
| 1    |  |   |     | BGPA | 1         |           |     | Start setting palletizing No. 1.   |
| 2    |  |   |     |      |           |           |     |                                    |
| 3    |  |   |     | PAPI | 5         | 7         |     | Palletizing counts: 5 x 7          |
| 4    |  |   |     | PAPN | 1         |           |     | Palletizing pattern 1              |
| 5    |  |   |     | PASE | 1         | 2         |     | PX-axis = Axis 1, PY-axis = Axis 2 |
| 6    |  |   |     | PAPT | 40        | 30        |     | Pitch (X = 40 mm, Y = 30 mm)       |
| 7    |  |   |     | PAST | 1         |           |     | Set reference point data.          |
| 8    |  |   |     |      |           |           |     | Use position No. 1 data.           |
| 9    |  |   |     | PSLI | 20        | 4         |     | Zigzag offset = 20 mm              |
| 10   |  |   |     |      |           |           |     | Zigzag count = 4                   |
| 11   |  |   |     | PCHZ | 3         |           |     | Palletizing Z-axis = Axis 3        |
| 12   |  |   |     | PTRG | 4         | 4         |     | Set palletizing arch triggers.     |
| 13   |  |   |     |      |           |           |     | Use position No. 4 data.           |
| 14   |  |   |     | OPFZ | 10        |           |     | PZ-axis offset = 10 mm             |
| 15   |  |   |     |      |           |           |     |                                    |
| 16   |  |   |     | EDPA |           |           |     |                                    |
| 17   |  |   |     |      |           |           |     |                                    |
| 18   | * ////////////////////////////////////// |   |     |      |           |           |     |                                    |
| 19   |  |   |     | ATRG | 4         | 4         |     | Set arch triggers.                 |
| 20   |  |   |     |      |           |           |     | Use position No. 4 data.           |
| 21   |  |   |     | ACHZ | 3         |           |     | Set arch-motion Z-axis.            |
| 22   |  |   |     |      |           |           |     |                                    |
| 23   |  |   |     | ACC  | 0.3       |           |     | Acceleration                       |
| 24   |  |   |     | DCL  | 0.3       |           |     | Deceleration                       |
| 25   |  |   |     | VLMX |           |           |     |                                    |
| 26   |  |   |     |      |           |           |     |                                    |
| 27   |  |   |     | PSET | 1         | 1         |     | Set palletizing position number.   |
| 28   |  |   |     | MOVP | 8         |           |     | Move to picking position.          |
| 29   | * ////////////////////////////////////// |   |     |      |           |           |     |                                    |
| 30   |  |   |     |      |           |           |     |                                    |



| Step | E | N | Cnd | Cmnd | Operand 1 | Operand 2 | Pst | Comment  |
|------|---|---|-----|------|-----------|-----------|-----|--|
| 31   |   |   |     | TAG  | 1         |           |     | Beginning of loop processing                   |
| 32   |   |   |     | PACH | 1         | 9         |     | Palletizing arch motion                        |
| 33   |   |   |     |      |           |           |     | Z point specified by Position No. 9            |
| 34   |   |   |     | ARCH | 8         | 9         |     | Arch motion                                    |
| 35   |   |   |     |      |           |           |     | Z point specified by Position No. 9            |
| 36   |   |   |     | PINC | 1         |           | 600 | Increment palletizing position number by 1.    |
| 37   |   |   | 600 | GOTO | 1         |           |     | Go to beginning of loop if PINC is successful. |
| 38   |   |   |     |      |           |           |     |  |
| 39   |   |   |     | EXIT |           |           |     | End of task                                    |
| 40   |   |   |     |      |           |           |     |  |

| No. | Axis 1  | Axis 2  | Axis 3 | Remarks                     |
|-----|---------|---------|--------|-----------------------------|
| 1   | 100.000 | 100.000 | 30.000 | Reference point data        |
| 2   | *.***   | *.***   | *.***  | (Not used)                  |
| 3   | *.***   | *.***   | *.***  | (Not used)                  |
| 4   | *.***   | *.***   | 10.000 | Arch-trigger point data     |
| 5   | *.***   | *.***   | *.***  | (Not used)                  |
| 6   | *.***   | *.***   | *.***  | (Not used)                  |
| 7   | *.***   | *.***   | *.***  | (Not used)                  |
| 8   | 0.000   | 0.000   | 30.000 | Picking-position point data |
| 9   | *.***   | *.***   | 0.000  | Z point data                |
| 10  |         |         |        |                             |

Schematic diagram of placement-point positions based on the above program



- The number shown at the top right of each circle indicates a 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



## 5. Pseudo-Ladder Task

With the Tabletop Robot, a pseudo-ladder task function can be used depending on the command and extension condition.

The input format is shown below. Note that this function must be used by expert engineers with a full knowledge of PLC software design.

### 5.1 Basic Frame

| Extension condition<br>E | N | Input condition | Command | Operand 1 | Operand2 | Output |                              |
|--------------------------|---|-----------------|---------|-----------|----------|--------|------------------------------|
|                          |   | Cnd             | Cmnd    |           |          | Pst    |                              |
| LD                       |   | 7001            | CHPR    | 1         |          |        | Ladder<br>statement<br>field |
|                          |   |                 | TPCD    | 1         |          |        |                              |
|                          |   |                 | TAG     | 1         |          |        |                              |
| .                        |   | .               | .       | .         |          |        |                              |
| .                        |   | .               | .       | .         |          |        |                              |
| .                        |   | .               | .       | .         |          |        |                              |
| LD                       |   | 7001            | TSLP    | 1 to 100  |          |        | Ladder<br>statement<br>field |
| .                        |   | .               | .       | .         |          |        |                              |
| .                        |   | .               | .       | .         |          |        |                              |
| .                        |   | .               | .       | .         |          |        |                              |
| .                        |   | .               | .       | .         |          |        |                              |
| .                        |   | .               | .       | .         |          |        |                              |
| LD                       |   | 7001            | TSLP    | 1 to 100  |          |        |                              |
| LD                       |   | 7001            | GOTO    | 1         |          |        |                              |
| LD                       |   | 7001            | EXIT    |           |          |        |                              |

\* Virtual input 7001: "Normally ON" contact

## 5.2 Ladder Statement Field

### [1] Extension conditions

|    |       |           |
|----|-------|-----------|
| LD | ..... | LOAD      |
| A  | ..... | AND       |
| O  | ..... | OR        |
| AB | ..... | AND BLOCK |
| OB | ..... | OR BLOCK  |

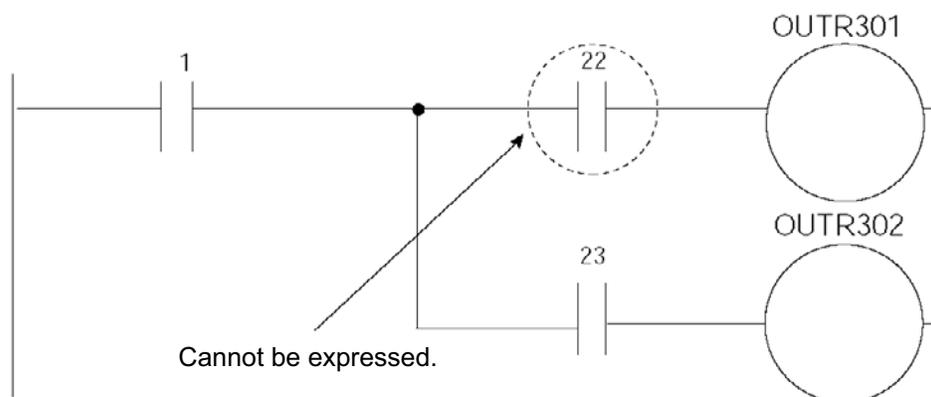
All of the above extension conditions can be used in non-ladder tasks.

### [2] Ladder commands

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

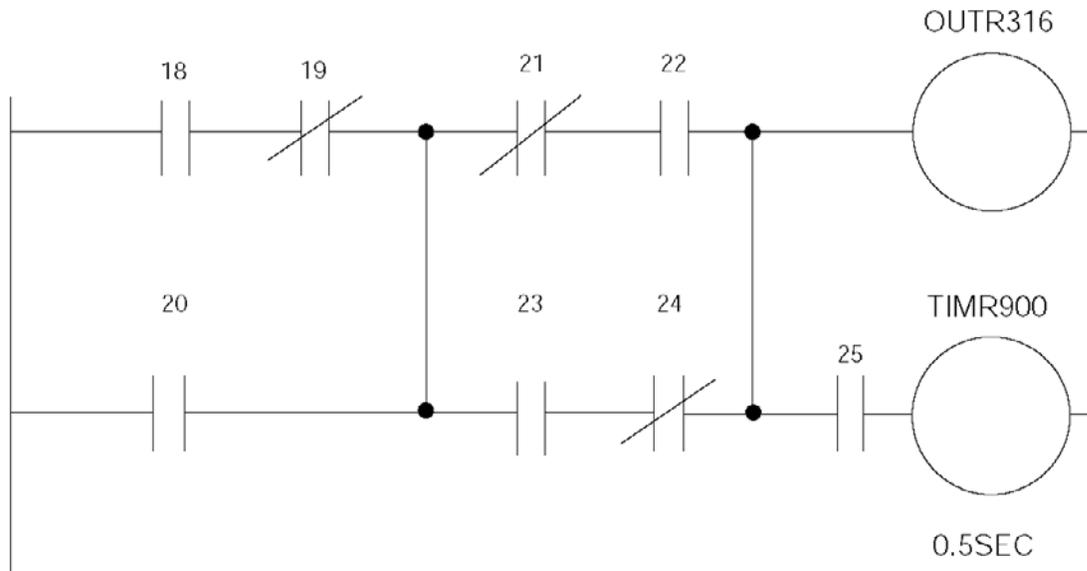
## 5.3 Points to Note

- 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.





### 5.4 Program Example



| Extension condition | N | Input condition | Command | Operand1  | Operand2  | Output |
|---------------------|---|-----------------|---------|-----------|-----------|--------|
| E                   | N | Cnd             | Cmnd    | Operand 1 | Operand 2 | Pst    |
| LD                  |   | 7001            | CHPR    | 1         |           |        |
|                     |   |                 | TPCD    | 1         |           |        |
|                     |   |                 | TAG     | 1         |           |        |
| LD                  |   | 18              |         |           |           |        |
| A                   | N | 19              |         |           |           |        |
| O                   |   | 20              |         |           |           |        |
| LD                  | N | 21              |         |           |           |        |
| A                   |   | 22              |         |           |           |        |
| LD                  |   | 23              |         |           |           |        |
| A                   | N | 24              |         |           |           |        |
| OB                  |   |                 |         |           |           |        |
| AB                  |   |                 | OUTR    | 316       |           |        |
| A                   |   | 25              | TIMR    | 900       | 0.5       |        |
| LD                  |   | 7001            | TSLP    | 3         |           |        |
| LD                  |   | 7001            | GOTO    | 1         |           |        |
| LD                  |   | 7001            | EXIT    |           |           |        |



## Chapter 5 Maintenance and Inspection

### 1. Inspection Items and Inspection Intervals

Perform the specified maintenance and inspection at the intervals listed below.

The schedule given below assumes that the robot is operated eight hours a day. If the robot is used continuously night and day or operated at higher utilization rates, shorten the inspection intervals accordingly.

|                           | Visual inspection of the exterior | Interior inspection | Greasing |
|---------------------------|-----------------------------------|---------------------|----------|
| Startup inspection        | ○                                 |                     |          |
| 1 month after operation   | ○                                 |                     |          |
| 6 months after operation  | ○                                 | ○                   |          |
| 1 year after operation    | ○                                 | ○                   | ○        |
| Every 6 months thereafter | ○                                 |                     |          |
| Every year                | ○                                 | ○                   | ○        |

### 2. Visual Inspection of the Exterior

Visually examine the exterior of the robot to check the following items.

|          |  |
|----------|--|
| Actuator | Looseness of actuator mounting bolts, etc. |
| Cables   | Damage, loose connector connection         |
| Overall  | Noise, vibration                           |

### 3. Visual Inspection and Cleaning

#### 3.1 Cleaning

- Clean the exterior as needed.
- Wipe off dirt with a soft cloth.
- Do not use strong compressed air on the actuator as this may force dust into the crevices.
- Do not use petroleum-based solvent on plastic parts or painted surfaces.
- If the robot is badly soiled, apply a neutral detergent or alcohol to a soft cloth, and wipe gently.



### 3.2 Interior Inspection

Turn off the power, remove the screw cover, and visually check the interior. Check the following items.

|            |   |
|------------|---|
| Actuator   | Looseness of robot mounting bolts, etc. |
| Guides     | Lubrication condition, soiling          |
| Ball screw | Lubrication condition, soiling          |

Visually inspect the interior to see if there is any dust or foreign matter in the robot. Also check the lubrication. Even if the grease you see around the parts is brown, the lubrication is fine as long as the traveling surface appears shiny. If the grease becomes dirty and dull or if the grease has worn away due to long hours of use, lubricate the parts after cleaning them.

### 3.3 Internal Cleaning

- Wipe off dirt with a soft cloth.
- Do not use strong compressed air on the actuator as this may force dust into the crevices.
- Do not use petroleum-based solvent, neutral detergent or alcohol.

## 4. Greasing the Guides

### 4.1 Applicable Grease

The Tabletop Robot is designed to use lithium grease for lubrication. The following grease is applied before the robot is shipped.

|                |                           |
|----------------|---------------------------|
| Idemitsu Kosan | Daphne Eponex Grease No.2 |
|----------------|---------------------------|

### 4.2 How to Apply Grease

Remove the screw cover and apply an appropriate amount of grease on the right and left rails.



**Caution:** In case the grease got into your eye, immediately go to see the doctor to get an appropriate care. After finishing the grease supply work, wash your hands carefully with water and soap to rinse the grease OFF.



## 5. Greasing the Ball Screw

### 5.1 Applicable Grease

The Tabletop Robot is designed to use lithium grease for lubrication. The following grease is applied before the robot is shipped.

|             |                   |
|-------------|-------------------|
| Kyodo Yushi | Multemp LRL No. 3 |
|-------------|-------------------|

### 5.2 How to Apply Grease

Remove the screw cover and apply an appropriate amount of grease on the right and left rails.



Caution: In case the grease got into your eye, immediately go to see the doctor to get an appropriate care. After finishing the grease supply work, wash your hands carefully with water and soap to rinse the grease OFF.

## 6. Timing Belt

### 6.1 Inspecting the Belt

Remove the pulley cover and visually inspect the belt.

Durability of the timing belt is affected significantly by the operating condition, and there is no standard guideline as to when the belt should be replaced. Generally, the belt is designed to withstand several millions of flexing loads. As a practical guideline, replace the timing belt when any of the conditions listed below is observed:

(If the belt needs to be replaced, please contact IAI's Engineering Service Section or Sales Section.)

- The teeth and end faces of the belt have worn significantly.
- The belt has swollen due to deposits of oil, etc.
- Cracks and other damages are found on the teeth or back of the belt.
- The belt has broken.

(If the belt needs to be replaced, please contact IAI's Engineering Service Section or Sales Section.)

### 6.2 Applicable Belt

The Tabletop Robot uses the following timing belt for its actuators. Should you require replacement of any belt used in your robot, please contact IAI's Engineering Service Section or Sales Section.

Timing belt: S3M; 6-mm wide, 190-mm long (Manufacturer: Bando Chemical Industries)



### 6.3 Belt Replacement Procedure

[1] Remove the pulley cover.

(With the gate X-axis actuator, remove the rear panel to access the pulley cover.)

[2] Hook a wire around the motor shaft.

(With the gate X-axis actuator, guide a wire through the belt replacement hole in the side face of the actuator and then hook the wire around the motor shaft.)

[3] Pull the end of the wire with a force of 2.4 to 2.6 kgf.

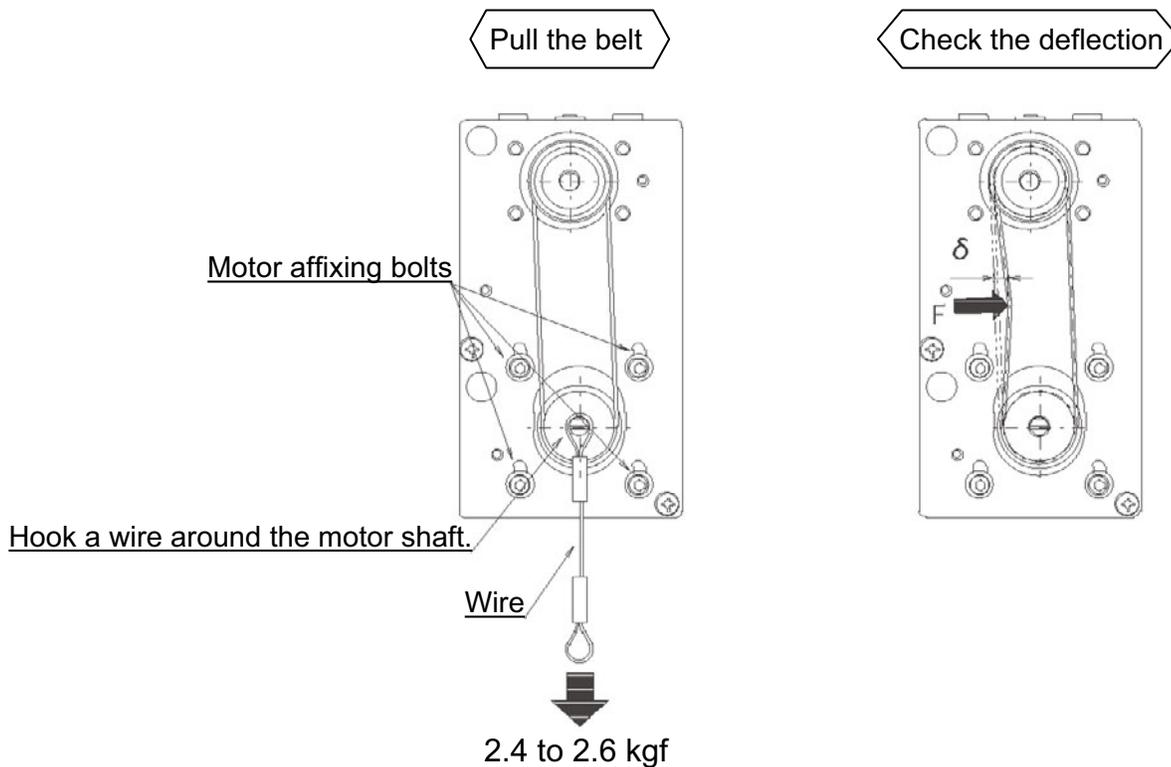
[4] Affix the motor.

[5] Check the deflection.

Tension load:  $F = 0.12$  to  $0.17$  kgf

Deflection:  $\delta = 1.04$  mm

(Note) When pulling the belt, hold the actuator with hands or otherwise prevent the actuator from moving.





## Appendix

### ◎ How to Create a Program

#### 1. Position Table

##### Position table

The Tabletop Robot can store 3,000 positions.

Positions are registered using the PC software or teaching pendant.

(Example with a 3-axis system)

| No.  | Axis1   | Axis2   | Axis3 | Vel | Acc | Dcl |
|------|---------|---------|-------|-----|-----|-----|
| 1    | 50.000  | 50.000  | 0.000 |     |     |     |
| 2    | 100.000 | 30.000  |       |     |     |     |
| 3    | 125.000 | 96.000  |       |     |     |     |
| 4    | 75.000  | 102.000 |       |     |     |     |
| 5    | 200.000 | 110.000 |       |     |     |     |
| 6    | 150.500 | 116.000 |       |     |     |     |
|      | ⋮       | ⋮       | ⋮     | ⋮   | ⋮   | ⋮   |
|      | ⋮       | ⋮       | ⋮     | ⋮   | ⋮   | ⋮   |
| 2994 |         |         |       |     |     |     |
| 2995 |         |         |       |     |     |     |
| 2996 |         |         |       |     |     |     |
| 2997 |         |         |       |     |     |     |
| 2998 |         |         |       |     |     |     |
| 2999 |         |         |       |     |     |     |
| 3000 |         |         |       |     |     |     |

- No.:** Specify a desired number in each program, and the actuator will move to the corresponding position registered under the number.
- Axis 1 to 3:** Enter a desired position for each axis under each position number.
- Vel:** Set a speed. The speed set in this field takes precedence over the speed specified directly in a program. In other words, specifying a position number will move the actuator to the applicable position at the speed specified in the Vel field under that position number.
- Acc:** Set an acceleration. The acceleration set in this field takes precedence over the acceleration specified directly in a program or set by a parameter.
- Dcl:** Set a deceleration. The deceleration set in this field takes precedence over the deceleration specified directly in a program or set by a parameter.



## 2. Program Format

### Program edit screen (PC software)

The Tabletop Robot supports a program consisting of up to 6,000 steps. Programs are edited using the PC software or teaching pendant.

| No. | B | E | N | Cnd | Cmnd | Operand 1 | Operand 2 | Pst | Comment |
|-----|---|---|---|-----|------|-----------|-----------|-----|---------|
| 3   |   |   |   |     | VEL  | 100       |           |     |         |
| 4   |   |   |   |     | ACC  | 0.3       |           |     |         |
| 5   |   |   |   |     | TAG  | 1         |           |     |         |
| 6   |   |   |   |     | EXSR | 5         |           |     |         |
| 7   |   |   |   |     | MOVP | 610       |           |     |         |
| 8   |   |   |   |     | MOVP | 599       |           |     |         |
| 9   |   |   |   |     | TINW | 0.3       |           |     |         |
| 10  |   |   |   |     | EXSR | 5         |           |     |         |
| 11  |   |   |   |     | MOVP | 601       |           |     |         |
| 12  |   |   |   |     | EXSR | 6         |           |     |         |
| 13  |   |   |   |     | TINW | 0.2       |           |     |         |
| 14  |   |   |   |     | MOVP | 610       |           |     |         |
| 15  |   |   |   |     | VEL  | 300       |           |     |         |
| 16  |   |   |   |     | EXSR | 1         |           |     |         |
| 17  |   |   |   |     | MOVP | 599       |           |     |         |

No.: Step number

B: Set a breakpoint. (This field can be accessed during online editing.)  
Using the mouse, click the "B" field in the line you want to set a breakpoint for. Once a breakpoint is set, "B" will be shown in the applicable line.  
\* Breakpoint --- Set a breakpoint in a step at which you want to pause the program run from the PC software.

E: Enter an extended condition (A, O, LD, AB, OB).

N: Specify "N" to indicate negation of the input condition.

Cnd: Enter an input condition.

Cmnd: Enter a SEL command.

Operand 1: Enter operand 1.

Operand 2: Enter operand 2.

Pst: Enter an output (operand 3).

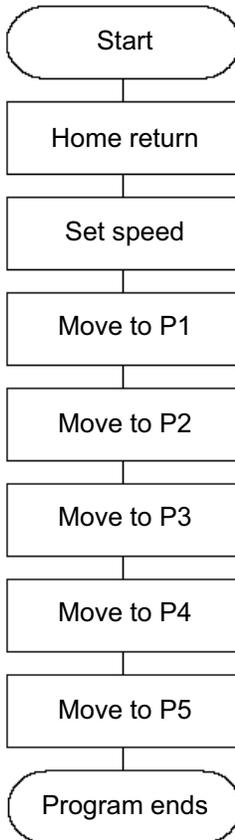
Comment: Enter a comment if necessary (using a maximum of 18 single-byte characters).

### 3. Positioning to Five Positions

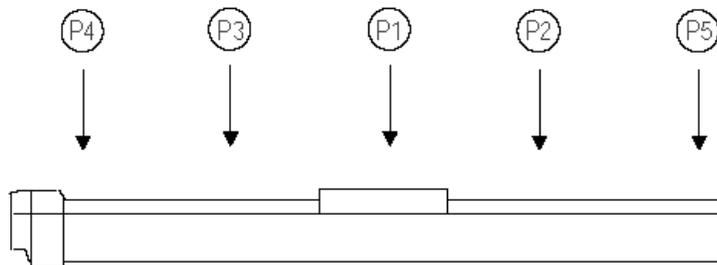
#### Description

Causes the actuator to move to positions 1 to 5 at a speed of 100 mm/sec following a home return. Only axis 1 is used.

#### Flowchart



- For the actuator to operate, a home return must have been completed and a speed must be set.
- The actuator moves to the position data coordinates specified by the movement commands.



#### Application program

| No. | B | E | N | Cnd | Cmd  | Operand 1 | Operand 2 | Pst | Comment            |
|-----|---|---|---|-----|------|-----------|-----------|-----|--------------------|
| 1   |   |   |   |     | HOME | 1         |           |     | Axis 1 Home Return |
| 2   |   |   |   |     | VEL  | 100       |           |     | Set Vel to 100mm/s |
| 3   |   |   |   |     | MOVL | 1         |           |     | Move to Point 1    |
| 4   |   |   |   |     | MOVL | 2         |           |     | Move to Point 2    |
| 5   |   |   |   |     | MOVL | 3         |           |     | Move to Point 3    |
| 6   |   |   |   |     | MOVL | 4         |           |     | Move to Point 4    |
| 7   |   |   |   |     | MOVL | 5         |           |     | Move to Point 5    |
| 8   |   |   |   |     | EXIT |           |           |     | End Program        |
| 9   |   |   |   |     |      |           |           |     |                    |

#### Position data

| No. | Axis1   |
|-----|---------|
| 1   | 100.000 |
| 2   | 150.000 |
| 3   | 50.000  |
| 4   | 0.000   |
| 5   | 200.000 |
| 6   |         |
| 7   |         |
| 8   |         |
| 9   |         |



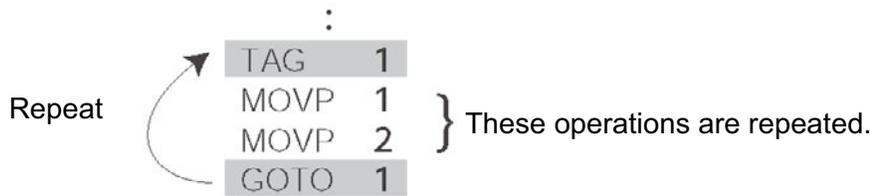
## 4. How to Use TAG and GOTO

### Description

Use GOTO and TAG commands if you want to repeat the same operation in the program or jump to desired steps based on certain conditions. A TAG can be defined in a step before or after a GOTO command.

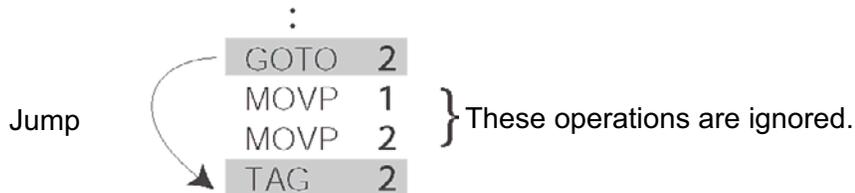
### Example of Use 1

Repeats the same operation.



### Example of Use 2

Jump to a specified step.



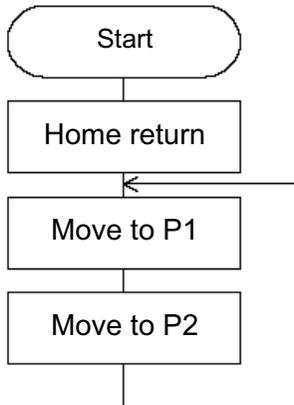


## 5. Moving Back and Forth between Two Points

### Description

Causes the actuator to move back and forth between two points repeatedly.

### Flowchart



- The actuator moves back and forth between P1 and P2 indefinitely.
- Axis 1 is used.
- Enter “TAG” in the first step of the repeated operation, and enter “GOTO” in the last step.

### Application program

| No. | B | E | N | Cnd | Cmd  | Operand 1 | Operand 2 | Pst | Comment |
|-----|---|---|---|-----|------|-----------|-----------|-----|---------|
| 1   |   |   |   |     | HOME | 1         |           |     |         |
| 2   |   |   |   |     | VEL  | 100       |           |     |         |
| 3   |   |   |   |     | TAG  | 1         |           |     |         |
| 4   |   |   |   |     | MOVL | 1         |           |     |         |
| 5   |   |   |   |     | MOVL | 2         |           |     |         |
| 6   |   |   |   |     | GOTO | 1         |           |     |         |
| 7   |   |   |   |     |      |           |           |     |         |

### Position data

| No. | Axis1   |
|-----|---------|
| 1   | 100.000 |
| 2   | 150.000 |
| 3   |         |
| 4   |         |
| 5   |         |
| 6   |         |
| 7   |         |



## 6. Path Operation

### Description

Causes the actuator to move continuously along given four points without stopping (path movement).

The actuator moves along the path shown to the right, without stopping at P2 or P3.

Unlike in operations using a MOVP or MOVL, the actuator need not be positioned at P2 and P3 and thus the tact time of movement can be reduced.

Assume that the following command is executed when the actuator is stopped at P1:

```
PATH 2 4
```

The actuator will move to P1, continue to move along points near P2 and P3, and finally reach P4. (Increasing the acceleration will bring the passing points closer to the specified positions.)



Assume that the following commands are entered successively:

```
PATH 2 3
```

```
PATH 3 4
```

The actuator will perform the same operation it would under the following command:

```
PATH 2 4
```

The actuator will perform a reverse operation ( $P4 \rightarrow P3 \rightarrow P2 \rightarrow P1$ ) if the following command is entered while the actuator is stopped at P4:

```
PATH 4 1
```



## 7. Output Control during Path Movement

### Description

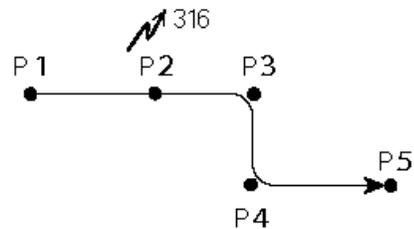
In a coating operation, etc., output control is sometimes required while the robot is moving. The Tabletop Robot can output signals while moving under a PATH command.

### How to Use

Before a PATH command, declare a POTP command to enable signal output during movement. If the output field of the PATH command specifies a given output port or global flag, the output port or flag specified in the output field will turn ON when the actuator, moving via path operation, approaches the position specified in the PATH command.

### Example of Use 1

The actuator moves from P1 to P5, as shown to the right, without stopping. It turns ON output port 316 upon approaching P2.



| Cmd  | Operand 1 | Operand 2 | Pst |
|------|-----------|-----------|-----|
| VEL  | 100       |           |     |
| POTP | 1         |           |     |
| PATH | 1         | 1         |     |
| PATH | 2         | 2         | 316 |
| PATH | 3         | 5         |     |

← A declaration command to enable signal output during path movement.

← Port 316 is turned ON at position P2 specified in this step.

Output ports and flags can only be turned ON using a POTP command. To turn OFF the port or flag that was turned ON during path operation, do so in a subsequent program step (using a BTOF command).

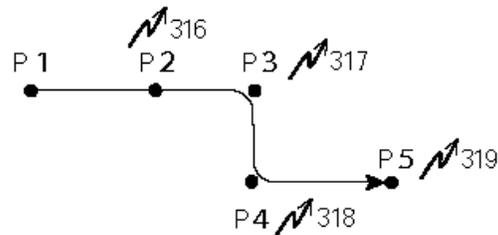
### Example of Use 2

Output ports 310 to 313 can be turned ON successively at positions P2 to P5.

| Cmd  | Operand 1 | Operand 2 | Pst |
|------|-----------|-----------|-----|
| VEL  | 100       |           |     |
| POTP | 1         |           |     |
| PATH | 1         | 1         |     |
| PATH | 2         | 5         | 316 |

← A declaration command to enable signal output during path movement.

← Output ports 316 to 319 are turned ON successively at positions P2 to P5 specified in this step.



## 8. Circular/Arc Operation

### Description

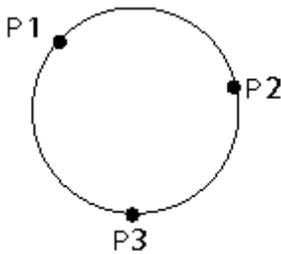
Causes the actuator to move along a two-dimensional circle or arc.

### How to Use

To specify a circle, specify three passing points. To specify an arc, specify three points as the starting point, passing point and ending point.

### Example of Use 1

#### Circle



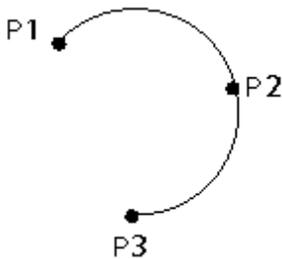
- After the actuator has moved to P1, specify "CIR2 2 3."
- Specifying "CIR2 2 3" based on the positions shown to the left will cause the actuator to move along the circle clockwise.

| E | N | Cnd | Cmd   | Operand 1 | Operand 2 | Pst |
|---|---|-----|-------|-----------|-----------|-----|
|   |   |     | VEL   | 100       |           |     |
|   |   |     | MOV P | 1         |           |     |
|   |   |     | CIR2  | 2         | 3         |     |

- To move the actuator counterclockwise, specify "CIR2 3 2."

### Example of Use 2

#### Arc



- After the actuator has moved to P1, specify "ARC2 2 3."

| E | N | Cnd | Cmd   | Operand 1 | Operand 2 | Pst |
|---|---|-----|-------|-----------|-----------|-----|
|   |   |     | VEL   | 100       |           |     |
|   |   |     | MOV P | 1         |           |     |
|   |   |     | ARC2  | 2         | 3         |     |

### Reference

Circle and arc commands can be used to specify three-dimensional operations (3-axis actuator system), as well as two-dimensional operations (2-axis actuator system).

CIRS --- Three-dimensional circular movement

ARCS --- Three-dimensional arc movement



## 9. Home-return Completion Output

### Description

Causes the actuator to output a signal confirming completion of home return. (Incremental specification)

The Tabletop Robot outputs an all-axis home-return completion signal to the LED (HPS) on the panel window. This section explains how to output a home-return completion signal via programming using a general-purpose output.

Once a general-purpose output turns ON, the output will remain ON even after the current program ends or other program is started. (There are certain conditions where the output turns OFF, such as an actuation of emergency stop. The ON status of the output can be maintained using I/O parameters (I/O parameter Nos. 70 and 71)).

### Example of Use

a. The actuator outputs a home-return completion signal.

| E | N | Cnd | Cmd  | Operand 1 | Operand 2 | Pst |
|---|---|-----|------|-----------|-----------|-----|
|   |   |     | HOME | 11        |           |     |
|   |   |     | BTON | 316       |           |     |

Home return is performed.  
General-purpose output turns ON.  
(A desired output can be set.)

b. Using the home-return completion signal, cause the actuator not to perform home return again if it has already been performed once.

| E | N | Cnd | Cmd  | Operand 1 | Operand 2 | Pst |
|---|---|-----|------|-----------|-----------|-----|
|   | N | 316 | HOME | 11        |           |     |
|   |   |     | BTON | 316       |           |     |

Home return is performed if output 316 is OFF.  
Home -return completion signal is output.

c. Use the output field instead of a BTON command.

| E | N | Cnd | Cmd  | Operand 1 | Operand 2 | Pst |
|---|---|-----|------|-----------|-----------|-----|
|   | N | 316 | HOME | 11        |           | 303 |

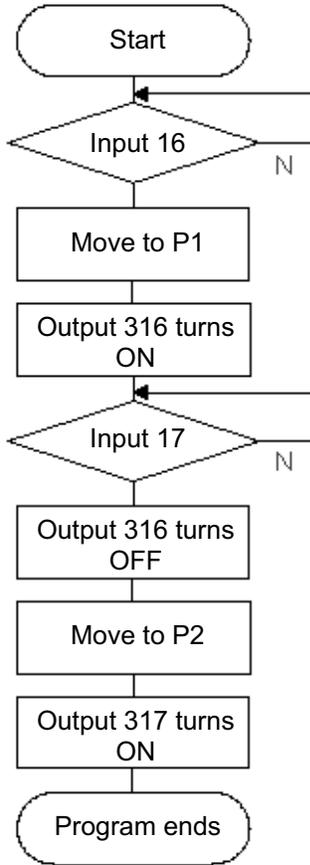
The same processing corresponding to the above two steps is performed.

## 10. Moving an Axis Selectively based on Input and Outputting a Completion Signal

### Description

How to move the actuator selectively based on input and output a processing completion signal

#### Flowchart



### Example of Use

The actuator waits until input port 16 turns ON, upon which it will move to P1.  
 The actuator waits until input port 17 turns ON, upon which it will move to P2.  
 316 is used to issue a signal indicating completion of movement to P1, while 317 is used to issue a signal indicating completion of movement to P2.

### Application program

| E | N | Cnd | Cmdnd | Operand 1 | Operand 2 | Pst | Comment            |
|---|---|-----|-------|-----------|-----------|-----|--------------------|
|   |   |     | VEL   | 100       |           |     | Set Vel to 100mm/s |
|   |   |     | WTON  | 16        |           |     | Wait on Input 16   |
|   |   |     | MOVP  | 1         |           |     | Move to Point 1    |
|   |   |     | BTON  | 316       |           |     | Turn on Output 316 |
|   |   |     | WTON  | 17        |           |     | Wait on Input 17   |
|   |   |     | BTOF  | 316       |           |     | Turn off Outpt 316 |
|   |   |     | MOVP  | 2         |           |     | Move to Point 2    |
|   |   |     | BTON  | 317       |           |     | Turn on Output 317 |
|   |   |     | EXIT  |           |           |     | End Program        |



## 11. Changing the Moving Speed

### Description

Change the moving speed of the actuator.

### How to Use

With the Tabletop Robot, the speed can be set using the following two methods:

- a: Use a VEL command in the application program.
- b: Use a speed set in the position data table.

### Example of Use

#### Application program

| E | N | Cnd | Cmd  | Operand 1 | Operand 2 | Pst |
|---|---|-----|------|-----------|-----------|-----|
|   |   |     | MOYP | 1         |           |     |
|   |   |     | VEL  | 300       |           |     |
|   |   |     | MOYP | 2         |           |     |
|   |   |     | MOYP | 3         |           |     |
|   |   |     | VEL  | 50        |           |     |
|   |   |     | MOYP | 4         |           |     |

#### Position data

| No. | Axis1   | Vel | Acc | Dec1 |
|-----|---------|-----|-----|------|
| 1   | 100.000 | 100 |     |      |
| 2   | 200.000 | 200 |     |      |
| 3   | 300.000 |     |     |      |
| 4   | 400.000 |     |     |      |

Moving speeds in the above program

100-mm position --- Move at 100 mm/sec.

200-mm position --- Move at 200 mm/sec.

300-mm position --- Move at 300 mm/sec.

400-mm position --- Move at 50 mm/sec.

As shown above, if a speed is specified for a given position in the position data table, the setting in the position data table takes precedence over the speed specified for the same position in the application program. In general, speeds are set using a VEL in the application program.

### VEL in the position data table and PATH command

It is possible to change the actuator speed without stopping the actuator, by using a PATH command and VEL in the position data table. (Refer to the next page.)

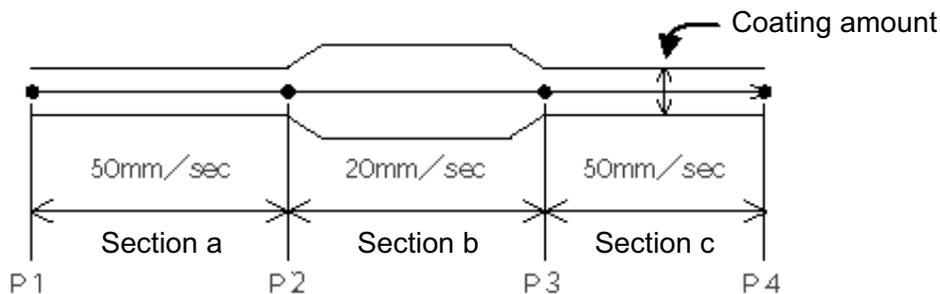
## 12. Changing the Speed during Movement

### Description

Use a PATH command to change the actuator speed while the actuator is moving. This command is useful in a dispensing operation where the coating amount changes during operation.

### Example of Use

The actuator moves at 50 mm/sec in section a, 20 mm/sec in section b and 50 mm/sec in section c, without stopping. (Path operation)



### Position data

| No. | Axis1   | Vel | Acc | Dcl |
|-----|---------|-----|-----|-----|
| 1   | 0.000   | 50  |     |     |
| 2   | 100.000 | 50  |     |     |
| 3   | 200.000 | 20  |     |     |
| 4   | 300.000 | 50  |     |     |

### Application program

“PATH 1 4” is the only movement command needed to implement this operation.

| E | N | Cnd | Cmd  | Operand 1 | Operand 2 | Pst |
|---|---|-----|------|-----------|-----------|-----|
|   |   |     | PATH | 1         | 4         |     |

### Reference

It is also possible to use a CHVL (speed change) command to change the actuator speed from other program. (In the multitasking mode)



### 13. Local/Global Classification of Variables and Flags

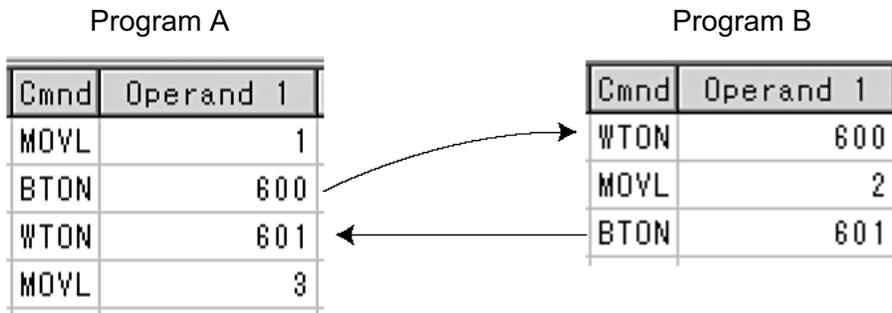
#### Description

The internal variables and flags used in SEL commands are classified into local and global variables/flags.

The shared data range used by all programs is called the global range, while the data range used only by each program is called the local range. To adjust the timings of multiple programs in the multitasking mode or to allow variables to reference one another, the global range must be used.

#### Example of Use

Handshake between programs



As shown in the above example, global flags can be used to perform operations requiring handshake between two programs, such as executing “MOVL 1” in program A, waiting for the actuator to move to the specified position and then executing “MOVL 2” in program B, waiting for the actuator to move to the specified position and then executing “MOVL 3” in program A, and so on.

The variables and flags in the global range are retained until the power is turned off.

The variables and flags in the local range are cleared (to “0” in the case of variables, or turned OFF in the case of flags) when the program is started.



# 14. How to Use Subroutines

## Description

If the same processing is performed multiple times in one program, the applicable operation is defined in a separate group of steps so that these steps can be called every time the operation is required. These steps are called a subroutine. Subroutines are used to shorten and simplify the program steps. Up to 99 subroutines can be used in a single program, and a maximum of 15 subroutine calls can be nested.

## How to Use

Declare/call a subroutine using the following commands:

EXSR: Call a subroutine.

BGSR: Declare the start of a subroutine (declaration of the start of a group of steps).

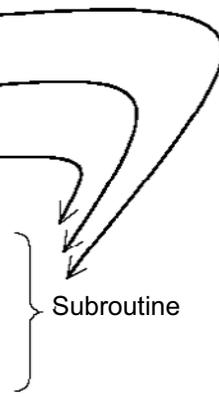
EDSR: Declare the end of a subroutine (declaration of the end of a group of steps).

## Example of Use

| Cmd  | Operand 1 |
|------|-----------|
| VEL  | 100       |
| MOYL | 1         |
| BTON | 316       |
| WTON | 20        |
| BTOF | 316       |
| MOYL | 2         |
| BTON | 316       |
| WTON | 20        |
| BTOF | 316       |
| MOYL | 3         |
| BTON | 316       |
| WTON | 20        |
| BTOF | 316       |
|      |           |
| EXIT |           |

The steps that perform the same operation are defined in a single location.

| Cmd  | Operand 1 |
|------|-----------|
| VEL  | 100       |
| MOYL | 1         |
| EXSR | 1         |
| MOYL | 2         |
| EXSR | 1         |
| MOYL | 3         |
| EXSR | 1         |
|      |           |
| EXIT |           |
| BGSR | 1         |
| BTON | 316       |
| WTON | 20        |
| BTOF | 316       |
| EDSR |           |



## Note

Jumping from within a subroutine to a TAG outside the subroutine using a GOTO command is prohibited.



## 15. Pausing the Operation

### Description

Use a declaration command HOLD to pause the moving axis via an external input.

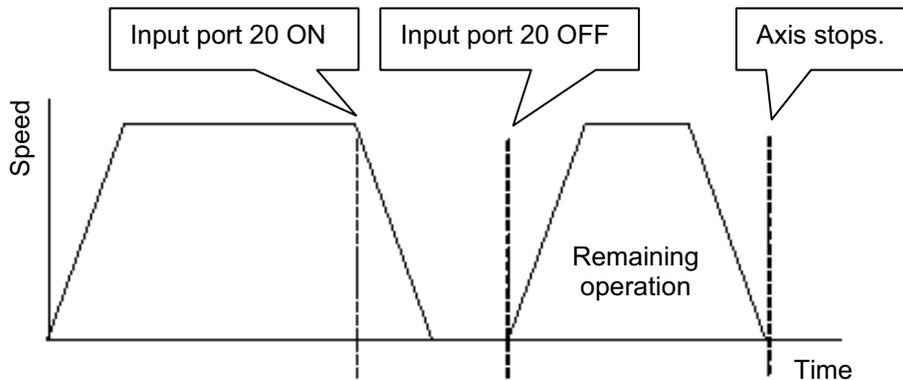
### How to Use

By declaring a HOLD command in the program, the moving axis can be paused (decelerated to a stop) via interruption.

While the HOLD input is ON, the axis is paused (decelerated to a stop) against all movement commands in the program.

### Example of Use

HOLD 20 Declaration of pause when general-purpose input 20 turns ON



### Application

In addition to an input port, a global flag can also be specified in operand 1 of the HOLD command. You can use a global flag to pause the axis from other program.

It is also possible to select the input signal pattern and stopping pattern using operand 2.

- 0 = Contact a (The axis decelerates to a stop) ⇒ Same as when operand 2 is not specified.
- 1 = Contact b (The axis decelerates to a stop)
- 2 = Contact b (The axis decelerates to a stop, after which the servo turns OFF ⇒ The drive power does not turn OFF)

| E | N | Cnd | Cmd  | Operand 1 | Operand 2 | Pst | Comment            |
|---|---|-----|------|-----------|-----------|-----|--------------------|
|   |   |     | HOLD | 20        | 2         |     | Input 20 Servo Off |

### Note

If the actuator is paused during home return, it will repeat the home return sequence from the beginning after the pause input turns OFF.

## 16. Aborting the Operation 1 (CANC)

### Description

Use a declaration command CANC to cause the moving axis to decelerate to a stop and cancel the remaining operation.

### How to Use

While the CANC input is ON, operations of all movement commands in the program are aborted.

### Example of Use

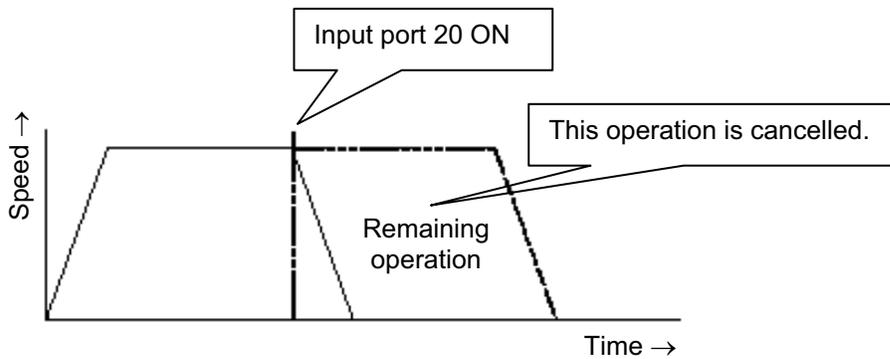
CANC command

```

CANC  20          Abort movement commands when input port 20 turns ON. (Declaration)
:
MOVP  1
MOVP  2
:
WTON  21
:

```

- \* Declare a CANC in a step before the movement command you want to abort.
- \* While the CANC input is ON, operation commands are cancelled successively, while non-operation commands (I/O processing, calculation processing, etc.) are executed successively.



### Note

Using a CANC may cause a situation where the currently executed step in the program can no longer be identified. To prevent this situation, it is recommended that an input wait step be created using a WTON command.

### Application

The input signal pattern can be selected using operand 2 of the CANC command.

0 = Contact a (The axis decelerates to a stop) ⇒ Same as when operand 2 is not specified.

1 = Contact b (The axis decelerates to a stop)

| E | N | Cnd | Cmdnd | Operand 1 | Operand 2 | Pst | Comment         |
|---|---|-----|-------|-----------|-----------|-----|-----------------|
|   |   |     | CANC  | 20        | 2         |     | Cancel Input 20 |



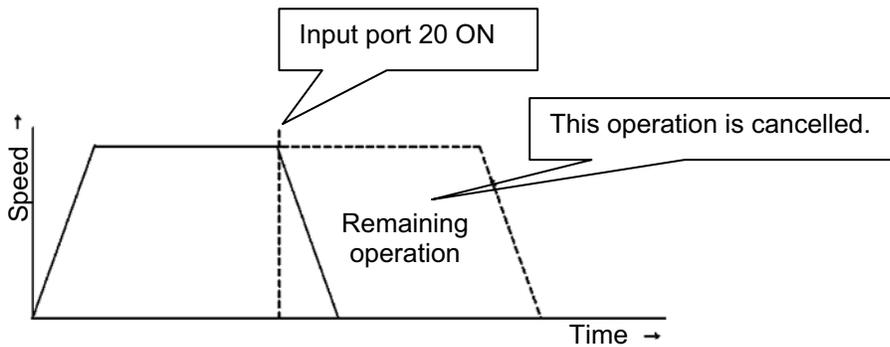
# 17. Aborting the Operation 2 (STOP)

## Description

Causes the moving axis to decelerate to a stop and cancel the remaining operation. (STOP)

## How to Use

Use a STOP command to abort the operation from other program. (In the multitasking mode) Specify the axis to abort using the axis pattern.



## Example of Use 1

STOP command

Main program

```

EXPG  n    Aborting program starts.
:
MOVL  1
MOVL  2
:

```

Abort control program

```

WTON 20    Wait for the abort input to turn ON.
STOP 11    Abort axes 1 and 2.

```

If "STOP 11" is executed during "MOVL 1," "MOVL 1" will be cancelled and the actuator operation will continue from "MOVL 2."

## Example of Use 2

Main program

```

EXPG  n    Aborting program starts.
:
MOVP  1
MOVP  2
:

```

Abort control program

```

WTON 20    Wait for the abort input to turn ON.
STOP 10    Abort axis 2.

```

Executing "STOP 10" during "MOVP 1" will only cancel the operation of axis 2 under "MOVP 1." Both axes 1 and 2 will operate under "MOVP 2."

## Note

During a CP operation (interpolation operation) initiated by a MOVL, etc., executing a STOP command will cancel the operations of all axes regardless of the specified axis pattern.





## 18. Moving to a Specified Position Number

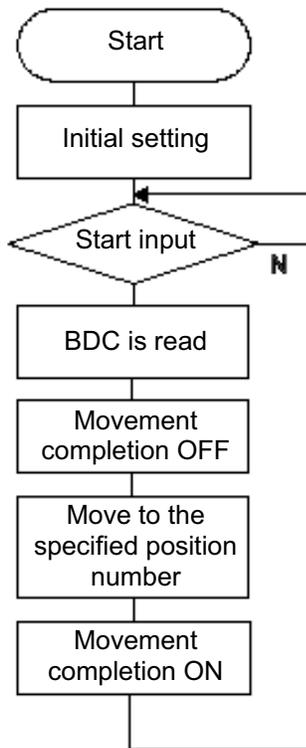
### Description

Read an external BCD code input as a position number and cause the actuator to move to the corresponding position.

### Example of Use

Use an INB command to read a BCD code from an input port as a position number. Up to three digits can be specified as a position number.

#### Flowchart



#### Input assignments

| Port | Description                |
|------|----------------------------|
| 28   | Start input                |
| 16   | Position specification 1   |
| 17   | Position specification 2   |
| 18   | Position specification 4   |
| 19   | Position specification 8   |
| 20   | Position specification 10  |
| 21   | Position specification 20  |
| 22   | Position specification 40  |
| 23   | Position specification 80  |
| 24   | Position specification 100 |
| 25   | Position specification 200 |
| 26   | Position specification 400 |
| 27   | Position specification 800 |

#### Output

303 Completion of movement

#### Application program

| E | N | Cnd | Cmd  | Operand 1 | Operand 2 | Pst | Comment            |
|---|---|-----|------|-----------|-----------|-----|--------------------|
|   |   |     | HOME | 11        |           |     | Home Return        |
|   |   |     | VEL  | 100       |           |     | Velocity Setting   |
|   |   |     | TAG  | 1         |           |     | Jump Marker        |
|   |   |     | WTON | 28        |           |     | Wait on Input 28   |
|   |   |     | INB  | 16        | 3         |     | Ready Position No. |
|   |   |     | BTOF | 303       |           |     | Pos End Signal OFF |
|   |   |     | MOVL | *99       |           |     | Move to position # |
|   |   |     | BTON | 303       |           |     | Pos End Signal ON  |
|   |   |     | GOTO | 1         |           |     | Jump to TAG 1      |

## 19. Conditional Jump

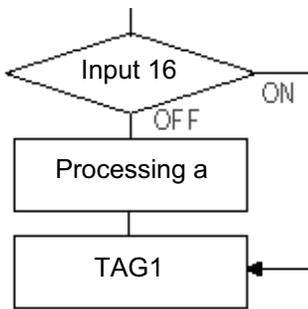
### Description

Select the destination to jump to under a GOTO command, by using an external input, output or internal flag as a condition.

The actuator waits for multiple inputs and performs processing appropriate for the input that turned ON.

### Example of Use 1

If input 10 is ON, the actuator will jump to "TAG 1." If input 10 is OFF, the actuator will perform the next processing.



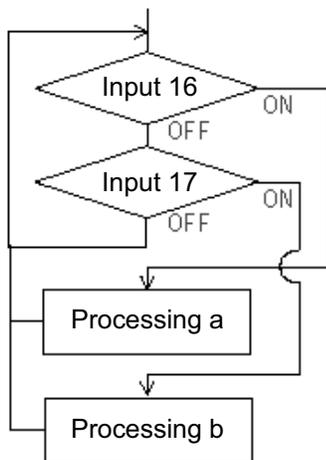
| E            | N | Cnd | Cmd  | Operand 1 |
|--------------|---|-----|------|-----------|
|              |   | 16  | GOTO | 1         |
| Processing a |   |     |      |           |
|              |   |     | TAG  | 1         |
| Processing b |   |     |      |           |

"GOTO 1" is executed if input 16 is ON.

\* If input 16 is ON, the actuator will skip processing a and perform processing b.  
If input 16 is OFF, the actuator will perform processing a and then perform processing b.

### Example of Use 2

The actuator waits for input port 16 or 17 to turn ON. If input 16 turns ON, the actuator will perform processing a. If input 17 turns ON, the actuator will perform processing b.



| E            | N | Cnd | Cmd  | Operand 1 |
|--------------|---|-----|------|-----------|
|              |   |     | TAG  | 1         |
|              |   | 16  | GOTO | 2         |
|              |   | 17  | GOTO | 3         |
|              |   |     | GOTO | 1         |
|              |   |     | TAG  | 2         |
| Processing a |   |     |      |           |
|              |   |     | GOTO | 1         |
|              |   |     | TAG  | 3         |
| Processing b |   |     |      |           |
|              |   |     | GOTO | 1         |

— No input  
 - - - - Input 16 turns ON  
 ——— Input 17 turns ON

If inputs 16 and 17 both turn ON, the actuator will perform processing a.



## 20. Waiting for Multiple Inputs

### Description

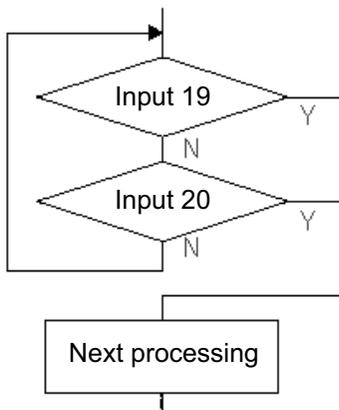
Causes the actuator to wait for multiple inputs and proceed to the next processing when any of these inputs turns ON.

### Point

With a WTON command, the actuator cannot proceed to the next processing until the specified input turns ON. In other words, the actuator cannot wait for multiple inputs.

### Example of Use

Monitor inputs 19 and 20. When either input turns ON (19 "OR" 20), the actuator will move to the next step.



Program a

| E | N | Cnd | Cmd  | Operand 1 |
|---|---|-----|------|-----------|
|   |   |     | TAG  | 1         |
|   |   | 19  |      |           |
| 0 |   | 20  | GOTO | 2         |
|   |   |     | GOTO | 1         |
|   |   |     | TAG  | 2         |

Next processing

Program b

| E | N | Cnd | Cmd  | Operand 1 |
|---|---|-----|------|-----------|
|   |   |     | TAG  | 1         |
|   | N | 19  |      |           |
| A | N | 20  | GOTO | 1         |

Next processing

\* The same processing is performed in both programs a and b.

As shown in the sample, it is possible to cause the actuator to wait for input without using a WTON command. This function also supports operations where multiple input conditions must be combined.



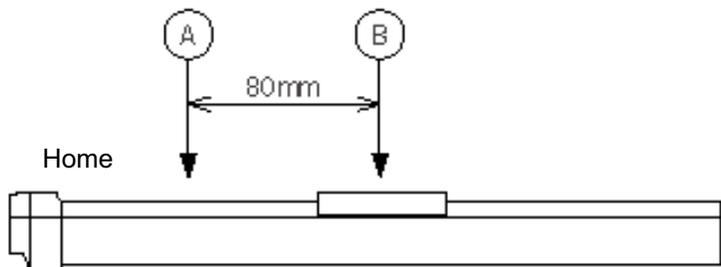
## 21. How to Use Offset

### Description

If you want to move (offset) all teaching points by several millimeters because the actuator has not been installed in the correct position exactly, etc., an offset can be specified for position data using an OFST command.

It is also possible to pitch-feed the actuator using an OFST command. (Refer to 23, “Constant pitch Feed Operation.”)

| E | N | Cnd | Cmdnd | Operand 1 | Operand 2 | Pst | Comment            |
|---|---|-----|-------|-----------|-----------|-----|--------------------|
|   |   |     | VEL   | 100       |           |     |                    |
|   |   |     | MOV P | 1         |           |     | Move to Point A    |
|   |   |     | OFST  | 1         | 80        |     | Axis 1 80mm Offset |
|   |   |     | MOV P | 1         |           |     | Move to Point B    |



### Note

Once an offset is set, all movement commands will be adjusted based on the offset. To cancel the offset, execute an OFST command again with “0” mm specified as the offset amount. The offset will not be reflected in different programs (even in the multitasking mode). To apply an offset to all programs, the offset must be specified in each program.



## 22. Executing an Operation n Times

### Description

Causes the actuator to execute a specific operation n times.

### Example of Use

The actuator moves back and forth between P1 and P2 10 times repeatedly, and then ends the program.

Use a CPEQ command to compare the number of times the operation has actually been repeated, against "10."

Home return is assumed to have been completed.

### Application program

| E | N | Cnd | Cmd  | Operand 1 | Operand 2 | Pst | Comment            |
|---|---|-----|------|-----------|-----------|-----|--------------------|
|   |   |     | VEL  | 100       |           |     | Velocity Setting   |
|   |   |     | LET  | 1         | 0         |     | Initialize Counter |
|   |   |     | TAG  | 1         |           |     | Jump Marker        |
|   |   |     | MOVP | 1         |           |     | Move to Point 1    |
|   |   |     | MOVP | 2         |           |     | Move to Point 2    |
|   |   |     | ADD  | 1         | 1         |     | Add 1 to Counter   |
|   |   |     | CPEQ | 1         | 10        | 900 | Stop at 10 cycles  |
|   | N | 900 | GOTO | 1         |           |     | Loop if not 10 cyc |
|   |   |     | EXIT |           |           |     | End Program        |

### Reference

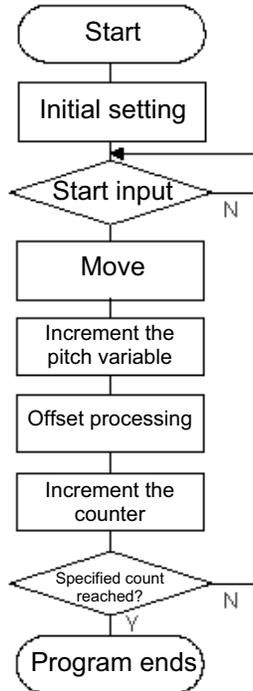
The same operation can also be performed using a DWEQ command.

## 23. Constant-pitch Feed Operation

### Description

Feed the actuator by a specified pitch *n* times from a reference position. The pitch and number of feeds are specified by variables beforehand.

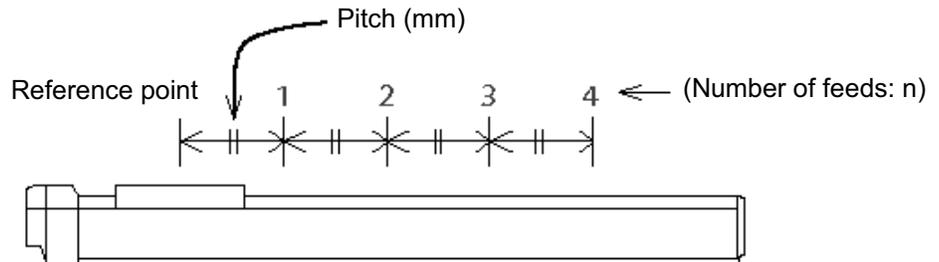
### Flowchart



### Example of Use

Use an OFST command to pitch-feed the actuator. A counter variable is used to count the number of times the actuator has been fed. The X-axis is pitch-fed in the positive direction.

**Point**  
An OFST command applies only to movement commands. Executing an OFST command alone will not move the axis.



### Application program

| E | N | Cnd | Cmd  | Operand 1 | Operand 2 | Pst | Comment            |
|---|---|-----|------|-----------|-----------|-----|--------------------|
|   |   |     | LET  | 1         | 4         |     | Move number, n=4   |
|   |   |     | LET  | 100       | 80        |     | 80mm Pitch Variabl |
|   |   |     | LET  | 2         | 0         |     | Clear Counter_2    |
|   |   |     | LET  | 101       | 0         |     | Clear Offset Var.  |
|   |   |     | HOME | 1         |           |     | Home Axis 1        |
|   |   |     | VEL  | 100       |           |     | Velocity Setting   |
|   |   |     | TAG  | 1         |           |     | Jump Marker 1      |
|   |   |     | WTON | 1         |           |     | Wait on Strt Input |
|   |   |     | MOVP | 1         |           |     | Move to Point 1    |
|   |   |     | ADD  | 101       | *100      |     | Add pitch to Ofst  |
|   |   |     | OFST | 1         | *101      |     | X-Axis Offset      |
|   |   |     | ADD  | 2         | 1         |     | Add 1 to Counter_2 |
|   |   |     | CPGT | 2         | *1        | 900 | Post when 4 cycles |
|   | N | 900 | GOTO | 1         |           |     | Loop if not done   |
|   |   |     | EXIT |           |           |     | End Program        |

### Reference

Pitch feed can also be implemented using MVPI/MVLI commands.



## 24. Jogging

### Description

Move the slider forward or backward while an input is ON or OFF.

In addition to an input, an output or global flag can be used to implement jogging.

If the specified input does not meet the condition when the command is executed, the slider will not perform jogging but proceed to the next step instead.

Once a soft limit is reached, the slider will stop and the next command step will become effective regardless of the input status.

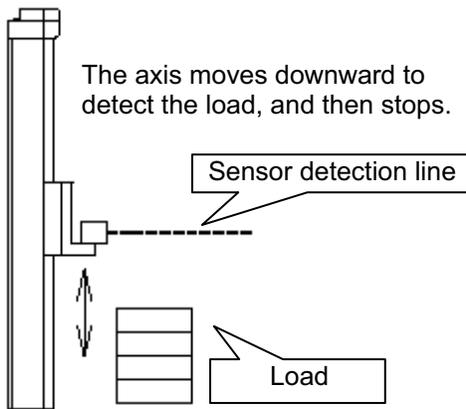
### How to Use

- Explanation of commands

|      |    |    |  |
|------|----|----|--|
| JFWN | 1  | 20 | Axis 1 moves forward while input 20 is ON.   |
| JFWF | 1  | 21 | Axis 1 moves forward while input 21 is OFF.  |
| JBWN | 10 | 22 | Axis 2 moves backward while input 22 is ON.  |
| JBWF | 10 | 23 | Axis 2 moves backward while input 23 is OFF. |

### Example of Use 1

- Stop the axis movement when a sensor input is received.



The axis moves downward to detect the load, and then stops.

```

:
VEL      50      Specify low speed.
JFWF    1      20  Move until a sensor input (20) turns ON.
EXIT
Program ends.

```

### Example of Use 2

- Perform jogging as normally done from the teaching pendant (operation of 2 axes).

### Application program

| E | N | Cnd | Cmd  | Operand 1 | Operand 2 | Pst |
|---|---|-----|------|-----------|-----------|-----|
|   |   |     | TAG  | 1         |           |     |
|   |   |     | JFWN | 1         | 20        |     |
|   |   |     | JBWN | 1         | 21        |     |
|   |   |     | JFWN | 10        | 22        |     |
|   |   |     | JBWN | 10        | 23        |     |
|   | N | 24  | GOTO | 1         |           |     |
|   |   |     | EXIT |           |           |     |

### Note

HOLD, STOP and CANC commands remain effective during jogging.



## 25. Switching Programs

### Description

Switch from one program to another via programming by using an EXPG or ABPG command.

### Example of Use 1

Start program 2 when the processing under program 1 is completed, and end program 1.

```
Program 1  Program 2
:          :
EXPG 2    :
EXIT
```

### Example of Use 2

Start a different program externally and end the current program.

```
Program 1  Program 2
ABPG 2    ABPG 1
:          :
```

If program 2 is started while program 1 is running, program 1 will be aborted.  
If program 1 is started while program 2 is running, program 2 will be aborted.

### Application

Specifying a program number in operand 2 will allow all programs from the one corresponding to the program number specified in operand 1 to one corresponding to the program number specified in operand 2 to be started (EXPG) or aborted (ABPG) simultaneously.

### Note

- The Tabletop Robot supports multitasking. By starting different programs successively while the robot is running a given program, a total of up to 16 programs can be run simultaneously. To use programs different from the 16 programs currently running, close unnecessary programs and then switch to the desired programs.
- If a program is aborted via an ABPG command and the program was executing a movement command, the actuator will immediately decelerate to a stop.



## 26. Aborting a Program

### Description

Abort the program currently running.

In the multitasking mode, execute an ABPG command (abort other program) from other program.

### Note

\* If the aborted program was executing a movement command, the actuator will immediately decelerate to a stop.

### Example of Use

Main program (Prg. 1)

```
EXPG  n   Abort control program starts.
WTON  10
MOVP  1
BTON  303
:
:
```

Abort control program (Prg. n)

```
WTON  20   Wait for the abort input to turn ON.
ABPG  1   Abort Prg 1.
EXIT           Program ends.
```

\* If an ABPG is executed while the actuator is moving via a MOVP command, the actuator will immediately decelerate to a stop and the program will end.



## ◎ How to Use Internal DIOs

### 1. Internal DIs and Dedicated Functions

Internal DI Nos. 001 to 006, 014 and 015 can be assigned as dedicated function ports (software reset input, etc.) by parameter settings. (They are not assigned to dedicated functions when the robot is shipped from the factory.)

To implement a dedicated function, assign it to an internal DI via a parameter and then turn ON/OFF the internal DI. Note that internal DIs cannot be controlled from the I/O connector. A dedicated function can be implemented by turning ON/OFF the internal DO corresponding to the internal DI assigned to that function.

Correspondence of DI port numbers and DO port numbers is shown below.

| DO port No. | DI port No. | Dedicated function        | Parameter No.        |
|-------------|-------------|---------------------------|----------------------|
| 308         | 001         | Software reset            | IO parameter No. 031 |
| 309         | 002         | Servo ON                  | IO parameter No. 032 |
| 310         | 003         | Auto program start        | IO parameter No. 033 |
| 311         | 004         | Software interlock        | IO parameter No. 034 |
| 312         | 005         | Pause reset               | IO parameter No. 035 |
| 313         | 006         | Pause                     | IO parameter No. 036 |
| 314         | 014         | Drive-source cutoff input | IO parameter No. 044 |
| 315         | 015         | Home return, etc.         | IO parameter No. 045 |

For example, executing the following SEL program will turn ON input port No. 1:

```
BTON 308
TIMW 1
EXIT
```

If DI No. 001 is set as a software reset input, a software reset will be implemented (the robot will restart). For details on the dedicated functions, refer to the parameter list.

Note: The parameters are normally set to the above DI and DO port numbers before shipment. Note that even when the input port number assigned to a given input function selection \*\*\* is changed by setting "Physical input port number for input function selection \*\*\*" accordingly, the functions where the ON/OFF statuses of output port Nos. 308 to 315 are reflected in input port Nos. 1 to 6, 14 and 15 will be maintained, as shown in the table above. However, the dedicated functions will be disabled. For example, setting "Input function selection 001" and "Physical input port number for input function selection 001" to "1 (soft reset)" and "16," respectively, and then turning output port No. 308 ON will turn input port No. 1 ON, but soft reset will not be executed. For details, refer to 2.3.2, "Standard Interface (Main Application Version 0.19 or Later)."



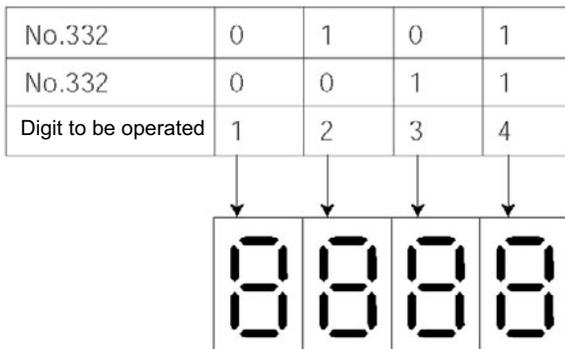
## 2. Showing User SEL Program Data on the 7-segment LED Display

The 7-segment LED display in the panel window on the front panel normally shows information received from the system. This 7-segment LED display can also be set to show data according to the SEL programs created by the user. (In this mode, the LED display shows user program data and system information alternately.)

Internal DO Nos. 332, 333, 337 to 346 are used to show user program data on the 7-segment display.

| Port No. | Function                                   |
|----------|--|
| 332      | 7-segment user display digit specification |
| 333      | 7-segment user display digit specification |
| 334      | For future expansion                       |
| 335      | For future expansion                       |
| 336      | For future expansion                       |
| 337      | 7-segment display refresh                  |
| 338      | 7-segment user/system alternate display    |
| 339      | 7-segment user display specification       |
| 340      | DT0 (7-segment user display bit)           |
| 341      | DT1 (7-segment user display bit)           |
| 342      | DT2 (7-segment user display bit)           |
| 343      | DT3 (7-segment user display bit)           |
| 344      | DT4 (7-segment user display bit)           |
| 345      | DT5 (7-segment user display bit)           |
| 346      | DT6 (7-segment user display bit)           |
| 347      |  |

DO Nos. 332 and 333 are used to specify the segment digit to be operated.  
Port ON/OFF statuses and specified digits (0: OFF, 1: ON)



Note: The parameters are normally set to the above port numbers before shipment.



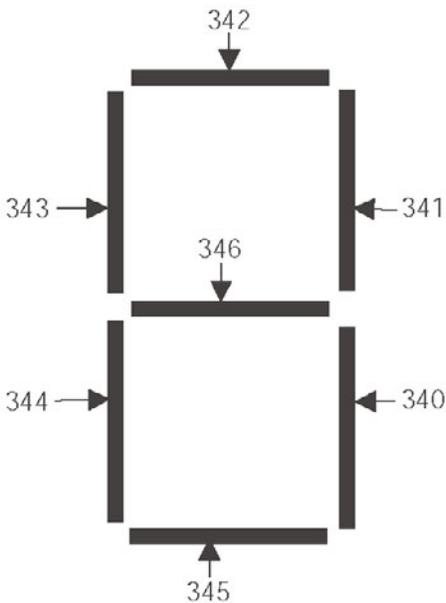
DO No. 339 is used to switch between user program data display and system information display. If DO No. 339 is set to "1," user SEL program data is shown. If DO No. 339 is set to "0," normal system information is shown. Set DO No. 338 to "1." If DO No. 339 is set to "1," user program data and system information are alternated every second.

Note: If DO No. 338 is set to "0"

Only user program data is shown, and the 1-second alternate display is not implemented. If an error of operation-cancellation level or higher has occurred, the applicable system information (error code) will be shown.

If an error of message level or lower has occurred, user program data will be shown continuously and the applicable system information (error code) will not be shown. Since the user has no way of knowing the occurrence of message-level or lower errors, set DO No. 338 to "0."

DO Nos. 340 to 346 correspond to the individual 7-segment display bits.



Data is shown on the 7-segment LED display at an ON edge of DO No. 337 (after a switching sequence of "0," "1" and "0").

(The 7-segment pattern set by DO Nos. 340 to 346 is shown in the digit specified by DO Nos. 332 and 333.)

Note: The parameters are normally set to the above port numbers before shipment.



## How to Use

- [1] Set a display mode using DO Nos. 338 and 339.
- [2] Set the digit to show (refresh) data in, using DO Nos. 332 and 333.
- [3] Set a 7-segment display pattern using DO Nos. 340 to 346.
- [4] Turn DO Nos. 337 (refresh) OFF, ON and then OFF. (Data will be refreshed at an ON edge.)

To display data in a different digit, repeat steps (2) to (4).

The display will continue to show user program data even after the display SEL program ends. To end the user program display mode, turn DO No. 339 OFF.

## Sample program

| No | E | N | Cnd                    | Cmd  | Operand 1 | Operand 2 | Pst | Comment   |
|----|---|---|------------------------|------|-----------|-----------|-----|---|
| 1  |   |   |                        | LET  | 99        | 3         |     |   |
| 2  |   |   |                        | OUT  | 338       | 339       |     | Alternate display of user program data and system information |
| 3  |   |   |                        | BTOF | 337       |           |     |   |
| 4  |   |   | * Set data in digit 1. |      |           |           |     |   |
| 5  |   |   |                        | LET  | 99        | 0         |     |   |
| 6  |   |   |                        | OUT  | 332       | 333       |     | Specify digit 1.  |
| 7  |   |   |                        | LET  | 99        | 3         |     | 3 = Display data "1"  |
| 8  |   |   |                        | OUT  | 340       | 346       |     | 7-segment pattern 1   |
| 9  |   |   |                        | BTON | 337       |           |     | Refresh ON  |
| 10 |   |   |                        | BTOF | 337       |           |     | Refresh OFF   |
| 11 |   |   | * Set data in digit 2. |      |           |           |     |   |
| 12 |   |   |                        | LET  | 99        | 1         |     |   |
| 13 |   |   |                        | OUT  | 332       | 333       |     | Specify digit 2.  |
| 14 |   |   |                        | LET  | 99        | 118       |     | 118 = Display data "2"  |
| 15 |   |   |                        | OUT  | 340       | 346       |     | 7-segment pattern 2   |
| 16 |   |   |                        | BTON | 337       |           |     | Refresh ON  |
| 17 |   |   |                        | BTOF | 337       |           |     | Refresh OFF   |
| 18 |   |   | * Set data in digit 3. |      |           |           |     |   |
| 19 |   |   |                        | LET  | 99        | 2         |     |   |
| 20 |   |   |                        | OUT  | 332       | 333       |     | Specify digit 3.  |
| 21 |   |   |                        | LET  | 99        | 103       |     | 103 = Display data "3"  |
| 22 |   |   |                        | OUT  | 340       | 346       |     | 7-segment pattern 3   |
| 23 |   |   |                        | BTON | 337       |           |     | Refresh ON  |
| 24 |   |   |                        | BTOF | 337       |           |     | Refresh OFF   |
| 25 |   |   | * Set data in digit 4. |      |           |           |     |   |
| 26 |   |   |                        | LET  | 99        | 3         |     |   |
| 27 |   |   |                        | OUT  | 332       | 333       |     | Specify digit 4.  |
| 28 |   |   |                        | LET  | 99        | 75        |     | 75 = Display data "4"   |
| 29 |   |   |                        | OUT  | 340       | 346       |     | 7-segment pattern 4   |
| 30 |   |   |                        | BTON | 337       |           |     | Refresh ON  |
| 31 |   |   |                        | BTOF | 337       |           |     | Refresh OFF   |
| 32 |   |   |                        |      |           |           |     |   |
| 33 |   |   |                        | IXII |           |           |     |   |

Note: The parameters are normally set to the above port numbers before shipment.



## ◎ List of Parameters

If you have any question regarding changing the parameters, please contact IAI's Sales Engineering Section. After changing a parameter, record the new and old parameter settings.

If you have purchased the PC software, we recommend that you back up the parameters immediately after the controller is delivered and when the system incorporating the controller is started. Since a number of customizing settings use parameters, you should back up the parameters regularly as you back up the programs.

To make the new parameters effective, write them to the flash ROM and then execute a software reset or reconnect the power.

The lists below are examples of default values displayed on the PC software. The default parameter settings vary depending on the operating condition and actuators used.

The values in the "Input range" column represent input limitations on the teaching pendant or in PC software. For the actual settings, enter the values defined in the "Remarks" column.

Values other than those defined in the "Remarks" column are for future expansion, even when they are inside the input range.

Therefore, do not enter values other than those defined in the "Remarks" column.



## 1. I/O Parameters

| No.      | Parameter name   | Default value (Reference) | Input range    | Unit | Remarks   |
|----------|--|---------------------------|----------------|------|---|
| 1        | I/O port assignment type   | 0                         | Reference only |      | 0: Fixed assignment   |
| 2        | Input port start number with fixed standard I/O1 assignments                       | 000                       | -1 to 599      |      | 0 + (Multiple of 8) (Invalid if a negative value is set)<br>* Keep the default value if the main application version is 0.18 or earlier.<br>(Change prohibited)   |
| 3        | Output port start number with fixed standard I/O1 assignments                      | 300                       | -1 to 599      |      | 300 + (Multiple of 8) (Invalid if a negative value is set)<br>* Keep the default value if the main application version is 0.18 or earlier.<br>(Change prohibited)   |
| 4        | Input port start number with fixed expanded I/O2 assignments                       | 32                        | -1 to 599      |      | 0 + (Multiple of 8) (Invalid if a negative value is set)<br>* Keep the default value if the main application version is 0.18 or earlier.<br>(Change prohibited)   |
| 5        | Output port start number with fixed expanded I/O2 assignments                      | 316                       | -1 to 599      |      | 300 + (Multiple of 8) (Invalid if a negative value is set)<br>* Keep the default value if the main application version is 0.18 or earlier.<br>(Change prohibited)   |
| 6        | Input port start number with fixed expanded I/O1 assignments (Network I/F module)  | -1                        | -1 to 599      |      | 0 + (Multiple of 8) (Invalid if a negative value is set)  |
| 7        | Output port start number with fixed expanded I/O1 assignments (Network I/F module) | -1                        | -1 to 599      |      | 300 + (Multiple of 8) (Invalid if a negative value is set)  |
| 8        | For future expansion   | -1                        | -1 to 599      |      |   |
| 9        | For future expansion   | -1                        | -1 to 599      |      |   |
| 10       | Standard I/O1 error monitor  | 1                         | 0 to 5         |      | 0: Do not monitor<br>1: Monitor<br>2: Monitor (Do not monitor errors relating to 24-V I/O power source)<br>3: Monitor (Monitor only errors relating to 24-V I/O power source)<br>* Some exceptions apply. |
| 11       | Expanded I/O2 error monitor  | 1                         | 0 to 5         |      | 0: Do not monitor<br>1: Monitor<br>2: Monitor (Do not monitor errors relating to 24-V I/O power source)<br>3: Monitor (Monitor only errors relating to 24-V I/O power source)<br>* Some exceptions apply. |
| 12       | Expanded I/O1 error monitor (Network I/F module)                                   | 1                         | 0 to 5         |      | 0: Do not monitor<br>1: Monitor<br>* Some exceptions apply.   |
| 13       | For future expansion   | 1                         | 0 to 5         |      |   |
| 14       | Number of ports using network-I/F-module remote input                              | 0                         | 0 to 240       |      | Refer to the operation manual for each network I/F card (CC-Link, DeviceNet, etc.).   |
| 15       | Number of ports using network-I/F-module remote input                              | 0                         | 0 to 240       |      | Refer to the operation manual for each network I/F card (CC-Link, DeviceNet, etc.).   |
| 16       | (For future expansion = Change prohibited)   | 0                         | 0 to 256       |      | Multiple of 8   |
| 17       | (For future expansion = Change prohibited)   | 0                         | 0 to 256       |      | Multiple of 8   |
| 18 to 19 | (For expansion)  | 0                         |                |      |   |
| 20       | Input filtering periods  | 2                         | 1 to 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 to 9         | msec | Input signal is recognized when the status is held for twice the period set by this parameter.  |
| 22       | For future expansion   | 2000                      | Reference only | msec |   |



## I/O Parameters

| No.      | Parameter name               | Default value (Reference) | Input range    | Unit | Remarks  |
|----------|------------------------------|---------------------------|----------------|------|--|
| 23       | For future expansion         | 0H                        | Reference only |      |  |
| 24       | For future expansion         | 0                         | Reference only |      |  |
| 25 to 29 | (For expansion)              | 0                         |                |      |  |
| 30       | Input function selection 000 | 1                         | 0 to 5         |      | 1: Program start signal (ON edge) (007 to 013: BCD-specified program number)<br>* 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 to 5         |      | 0: General-purpose input<br>1: Software reset signal (1 second ON)<br>If continued operation is specified as the action upon emergency stop, enable the software reset signal (to provide a means of canceling the operation).<br>* The condition of output port No. 308 (internal DIO) is input to input port No. 001 (internal DIO).   |
| 32       | Input function selection 002 | 0                         | 0 to 5         |      | 0: General-purpose input<br>1: Servo ON<br>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)<br>* The condition of output port No. 309 (internal DIO) is input to input port No. 002 (internal DIO).  |
| 33       | Input function selection 003 | 1                         | 0 to 5         |      | 0: General-purpose input<br>1: General-purpose input (Start the auto-start program upon power-ON reset/software reset in the AUTO mode)<br>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.<br>* The condition of output port No. 310 (internal DIO) is input to input port No. 003 (internal DIO). |
| 34       | Input function selection 004 | 0                         | 0 to 5         |      | 0: General-purpose input<br>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)<br>* The condition of output port No. 311 (internal DIO) is input to input port No. 004 (internal DIO).  |
| 35       | Input function selection 005 | 0                         | 0 to 5         |      | 0: General-purpose input, 1: Operation-pause reset signal (ON edge)<br>* The condition of output port No. 312 (internal DIO) is input to input port No. 005 (internal DIO).  |
| 36       | Input function selection 006 | 0                         | 0 to 5         |      | 0: General-purpose input<br>1: Operation-pause reset signal (OFF level) (Valid only during automatic operation)<br>* Cancel pause when an operation-pause reset signal is received.<br>* The condition of output port No. 313 (internal DIO) is input to input port No. 006 (internal DIO).  |



I/O Parameters

| No. | Parameter name                | Default value (Reference) | Input range | Unit | Remarks   |
|-----|-------------------------------|---------------------------|-------------|------|---|
| 37  | Input function selection 007  | 1                         | 0 to 5      |      | 0: General-purpose input<br>1: Program number specified for program start (least significant bit)<br>* Keep the default value if the main application version is 0.18 or earlier.<br>(Change prohibited)  |
| 38  | Input function selection 008  | 1                         | 0 to 5      |      | 0: General-purpose input<br>1: Program number specified for program start<br>* Keep the default value if the main application version is 0.18 or earlier.<br>(Change prohibited)  |
| 39  | Input function selection 009  | 1                         | 0 to 5      |      | 0: General-purpose input<br>1: Program number specified for program start<br>* Keep the default value if the main application version is 0.18 or earlier.<br>(Change prohibited)  |
| 40  | Input function selection 010  | 1                         | 0 to 5      |      | 0: General-purpose input<br>1: Program number specified for program start<br>* Keep the default value if the main application version is 0.18 or earlier.<br>(Change prohibited)  |
| 41  | Input function selection 011  | 1                         | 0 to 5      |      | 0: General-purpose input<br>1: Program number specified for program start<br>* Keep the default value if the main application version is 0.18 or earlier.<br>(Change prohibited)  |
| 42  | Input function selection 012  | 1                         | 0 to 5      |      | 0: General-purpose input<br>1: Program number specified for program start<br>* Keep the default value if the main application version is 0.18 or earlier.<br>(Change prohibited)  |
| 43  | Input function selection 013  | 1                         | 0 to 5      |      | 0: General-purpose input<br>1: Program number specified for program start<br>2: Error reset (On edge)<br>* Keep the default value if the main application version is 0.18 or earlier.<br>(Change prohibited)  |
| 44  | Input function selection 014  | 0                         | 0 to 5      |      | 0: General-purpose input (Cancel cutoff when the drive-source cutoff factor is removed)<br>1: Drive-source cutoff reset input (ON edge) (Valid when the factor has been removed)<br>* The condition of output port No. 314 (internal DIO) is input to input port No. 014 (internal DIO).  |
| 45  | Input function selection 015  | 0                         | 0 to 5      |      | 0: General-purpose input<br>1: Home return of all valid axes (ON edge) (Servo ON must be executed first = I/O parameter No. 32, Axis-specific parameter No. 13)<br>2: Home return of all valid incremental axes (ON edge) (Main application version 0.16 or later) (Servo ON must be executed first = I/O parameter No. 32, Axis-specific parameter No. 13)<br>* The condition of output port No. 315 (internal DIO) is input to input port No. 015 (internal DIO). |
| 46  | Output function selection 300 | 1                         | 0 to 20     |      | 0: General-purpose output<br>1: Output error of operation-cancellation level or higher (ON)<br>2: Output error of operation-cancellation level or higher (OFF)<br>3: Output error of operation - cancellation level or higher + emergency stop (ON)<br>4: Output error of operation - cancellation level or higher + emergency stop (OFF)<br>* Keep the default value if the main application version is 0.18 or earlier.<br>(Change prohibited)                    |



## I/O Parameters

| No. | Parameter name                | Default value (Reference) | Input range | Unit | Remarks  |
|-----|-------------------------------|---------------------------|-------------|------|--|
| 47  | Output function selection 301 | 1                         | 0 to 20     |      | 0: General-purpose output<br>1: READY output (PIO trigger program can be run)<br>2: READY output (PIO trigger program operation permitted AND error of operation-cancellation level or higher not present)<br>3: READY output (PIO trigger program operation permitted AND error of cold-start level or higher not present)<br>* Keep the default value if the main application version is 0.18 or earlier.<br>(Change prohibited) |
| 48  | Output function selection 302 | 1                         | 0 to 20     |      | 0: General-purpose output<br>1: Emergency-stop output (ON)<br>2: Emergency-stop output (OFF)<br>* Keep the default value if the main application version is 0.18 or earlier.<br>(Change prohibited)  |
| 49  | Output function selection 303 | 2                         | 0 to 5      |      | 0: General-purpose output<br>1: AUTO mode output<br>2: Output during automatic operation (Other parameter No. 12)<br>* Keep the default value if the main application version is 0.18 or earlier.<br>(Change prohibited)   |
| 50  | Output function selection 304 | 2                         | 0 to 5      |      | 0: General-purpose output<br>1: Output if all valid axes are at their homes (= 0)<br>2: Output if all valid axes completed home return (coordinates have been confirmed).<br>3: Output if all valid axes are at their home preset coordinates<br>* Keep the default value if the main application version is 0.18 or earlier.<br>(Change prohibited)   |
| 51  | Output function selection 305 | 0                         | 0 to 5      |      | 0: General-purpose output<br>1: For future expansion<br>2: Output when axis-1 servo is ON (System monitor task output)<br>3: For future expansion  |
| 52  | Output function selection 306 | 0                         | 0 to 5      |      | 0: General-purpose output<br>1: For future expansion<br>2: Output when axis-2 servo is ON (System monitor task output)<br>3: For future expansion  |
| 53  | Output function selection 307 | 0                         | 0 to 5      |      | 0: General-purpose output<br>1: For future expansion<br>2: Output when axis-3 servo is ON (System monitor task output)<br>3: For future expansion  |
| 54  | Output function selection 308 | 0                         | 0 to 5      |      | 0: General-purpose output<br>* The condition of output port No. 308 (internal DIO) is input to input port No. 001 (internal DIO).<br>(The parameters are normally set to the above port numbers before shipment.)  |
| 55  | Output function selection 309 | 0                         | 0 to 5      |      | 0: General-purpose output<br>* The condition of output port No. 309 (internal DIO) is input to input port No. 002 (internal DIO).<br>(The parameters are normally set to the above port numbers before shipment.)  |



## I/O Parameters

| No.      | Parameter name   | Default value (Reference) | Input range | Unit | Remarks  |
|----------|--|---------------------------|-------------|------|--|
| 56       | Output function selection 310  | 0                         | 0 to 5      |      | 0: General-purpose output<br>* The condition of output port No. 310 (internal DIO) is input to input port No. 003 (internal DIO).<br>(The parameters are normally set to the above port numbers before shipment.)                              |
| 57       | Output function selection 311  | 0                         | 0 to 5      |      | 0: General-purpose output<br>* The condition of output port No. 311 (internal DIO) is input to input port No. 004 (internal DIO).<br>(The parameters are normally set to the above port numbers before shipment.)                              |
| 58       | Output function selection 312  | 0                         | 0 to 5      |      | 0: General-purpose output<br>* The condition of output port No. 312 (internal DIO) is input to input port No. 005 (internal DIO).<br>(The parameters are normally set to the above port numbers before shipment.)                              |
| 59       | Output function selection 313  | 0                         | 0 to 5      |      | 0: General-purpose output<br>* The condition of output port No. 313 (internal DIO) is input to input port No. 006 (internal DIO).<br>(The parameters are normally set to the above port numbers before shipment.)                              |
| 60       | Output function selection 314  | 0                         | 0 to 5      |      | 0: General-purpose output<br>* The condition of output port No. 314 (internal DIO) is input to input port No. 014 (internal DIO).<br>(The parameters are normally set to the above port numbers before shipment.)                              |
| 61       | Output function selection 315  | 0                         | 0 to 5      |      | 0: General-purpose output<br>* The condition of output port No. 315 (internal DIO) is input to input port No. 015 (internal DIO).<br>(The parameters are normally set to the above port numbers before shipment.)                              |
| 62       | Physical input port number for axis-1 brake forced release                                   | 0                         | 0 to 299    |      | Forcibly unlock the brake when the applicable port is ON (be aware of a falling load).<br>* Invalid if "0" is set (Invalid if input port No. 0 is specified)<br>* The synchro slave axis will follow the synchro master axis.                  |
| 63       | Physical input port number for axis-2 brake forced release                                   | 0                         | 0 to 299    |      | Forcibly unlock the brake when the applicable port is ON (be aware of a falling load).<br>* Invalid if "0" is set (Invalid if input port No. 0 is specified)<br>* The synchro slave axis will follow the synchro master axis.                  |
| 64       | Physical input port number for axis-3 brake forced release                                   | 0                         | 0 to 299    |      | Forcibly unlock the brake when the applicable port is ON (be aware of a falling load).<br>* Invalid if "0" is set (Invalid if input port No. 0 is specified)<br>* The synchro slave axis will follow the synchro master axis.                  |
| 65       | Physical input port number for axis-4 brake forced release                                   | 0                         | 0 to 299    |      | Forcibly unlock the brake when the applicable port is ON (be aware of a falling load).<br>* Invalid if "0" is set (Invalid if input port No. 0 is specified)<br>* The synchro slave axis will follow the synchro master axis.                  |
| 66 to 69 | (For expansion)  | 0                         |             |      |  |
| 70       | Unaffected general-purpose output area number (MIN) when all operations/programs are aborted | 300                       | 0 to 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 | 315                       | 0 to 599    |      |  |



I/O Parameters

| No. | Parameter name  | Default value (Reference) | Input range     | Unit | Remarks   |
|-----|---|---------------------------|-----------------|------|---|
| 72  | Unaffected general-purpose output area number (MIN) when all operations are paused (servo-axis soft interlock + output-port soft interlock) | 300                       | 0 to 599        |      | * Important: Outputs in this area must be operated (including recovery) under the responsibility of user programs including the "I/O processing program at all operations pause." Outputs outside this area will be forcibly turned OFF, reflecting/holding the results of operations performed while all operation pause is effective (only during automatic operation). (Invalid if "0" is set) |
| 73  | Unaffected general-purpose output area number (MAX) when all operations are paused (servo-axis soft interlock + output-port soft interlock) | 599                       | 0 to 599        |      |   |
| 74  | Number of TP user output ports used (hand, etc.)  | 0                         | 0 to 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 to 599        |      | Referenced by TP. (Valid with TP application version 1.05 or later)   |
| 76  | AUTO mode physical output port number   | 0                         | 0 to 599        |      | (Invalid if "0" is set)   |
| 77  | Input port number permitted to receive PC/TP servo movement command   | 0                         | 0 to 299        |      | * Important: Invalid once operation is started. (Invalid if "0" is set)   |
| 78  | Axis pattern permitted to receive PC/TP servo movement command  | 0                         | 0B to 11111111B |      |   |
| 79  | For future expansion  | 0                         | Reference only  |      |   |
| 80  | (PC/TP SIO usage)   | 1                         | 1 to 1          |      | Switching of DIP switches   |
| 81  | (PC/TP SIO station code)  | 153                       | 153 to 153      |      | 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 (AUTO mode)   | 0                         | 0 to 9          |      | 0: Open SEL program<br>1: Open SEL program (Connect PC/TP when both devices are closed = Used exclusively by the manufacturer)<br>2: IAI protocol B (Slave)   |
| 91  | Station code of SIO channel 1 opened to user  | 153                       | 0 to 255        |      | Valid only with IAI protocol.   |
| 92  | Baud rate type of SIO channel 1 opened to user  | 0                         | 0 to 2          |      | 0: 9.6                    1: 19.2                    2: 38.4 kbps   |
| 93  | Data length of SIO channel 1 opened to user   | 8                         | 7 to 8          |      |   |
| 94  | Stop bit length of SIO channel 1 opened to user   | 1                         | 1 to 2          |      |   |
| 95  | Parity type of SIO channel 1 opened to user   | 0                         | 0 to 2          |      | 0: None                    1: Odd                    2: Even  |
| 96  | Receive operation type of SIO channel 1 opened to user  | 0                         | 0 to 1          |      | 0: Forcibly enable receive after send<br>1: Do not forcibly enable receive at send  |



## I/O Parameters

| No.        | Parameter name   | Default value (Reference) | Input range    | Unit | Remarks                       |
|------------|--|---------------------------|----------------|------|-------------------------------|
| 97         | IAI-protocol minimum response delay for SIO channel 1 opened to user | 0                         | 0 to 999       | msec | Valid only with IAI protocol. |
| 98 to 99   | Reservation of SIO channel 1 opened to user                          | 0                         |                |      |                               |
| 100        | SIO for future expansion   | 0                         | Reference only |      |                               |
| 101        | SIO for future expansion   | 0                         | Reference only |      |                               |
| 102        | SIO for future expansion   | 0                         | Reference only |      |                               |
| 103        | SIO for future expansion   | 0                         | Reference only |      |                               |
| 104        | SIO for future expansion   | 0                         | Reference only |      |                               |
| 105        | SIO for future expansion   | 0                         | Reference only |      |                               |
| 106        | SIO for future expansion   | 0                         | Reference only |      |                               |
| 107        | SIO for future expansion   | 0                         | Reference only |      |                               |
| 108        | SIO for future expansion   | 0                         | Reference only |      |                               |
| 109        | SIO for future expansion   | 0                         | Reference only |      |                               |
| 110        | SIO for future expansion   | 0                         | Reference only |      |                               |
| 111        | SIO for future expansion   | 0                         | Reference only |      |                               |
| 112        | SIO for future expansion   | 0                         | Reference only |      |                               |
| 113        | SIO for future expansion   | 0                         | Reference only |      |                               |
| 114        | SIO for future expansion   | 0                         | Reference only |      |                               |
| 115        | SIO for future expansion   | 0                         | Reference only |      |                               |
| 116 to 119 | (For expansion)  | 0                         |                |      |                               |

PC: PC software  
TP: Teaching pendant



I/O Parameters

| No. | Parameter name      | Default value (Reference) | Input range     | Unit | Remarks  |
|-----|---------------------|---------------------------|-----------------|------|--|
| 120 | Network attribute 1 | 1                         | 0H to FFFFFFFFH |      | Bits 0 to 3: CC-Link remote register area H/L byte swap selection (0: Do not swap, 1: Swap)<br>* The number of used ports and number of occupied stations in I/O parameter Nos. 14 and 15 must match.  |
| 121 | Network attribute 2 | 0                         | 0H to FFFFFFFFH |      |  |
| 122 | Network attribute 3 | 0                         | 0H to FFFFFFFFH |      |  |
| 123 | Network attribute 4 | 0                         | 0H to FFFFFFFFH |      | Bits 0 to 3: Ethernet TCP/IP message communication Selection whether to permit 0.0.0.0 (IP address of connection destination can be ignored) as IP address of connection destination on server (0: Do not permit 1: Permit ( <u>not recommended</u> ))<br>* Note: Number of clients that can be connected simultaneously to one server port channel = 1  |
| 124 | Network attribute 5 | 0                         | 0H to FFFFFFFFH |      | Ethernet TCP/IP message communication attribute<br>Ethernet client/server type<br>(0: Not in use<br>1: Client (Automatic assignment of own port number)<br>2: Client (Specification of own port number)<br>→ This setting is <u>not recommended</u> 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.)<br>3: Server (Specification of own port number)<br>* Note: Number of clients that can be connected simultaneously to one server port channel = 1<br><br>Bits 0 to 3: IAI protocol B/TCP (MANU mode)<br>* PC software can be connected only in the case of a client.<br>Bits 4 to 7: IAI protocol B/TCP (AUTO mode)<br>* PC software can be connected only in the case of a client.<br>Bits 8 to 11: Channel 31 opened to user<br>Bits 12 to 15: Channel 32 opened to user<br>Bits 16 to 19: Channel 33 opened to user<br>Bits 20 to 23: Channel 34 opened to user<br><br>* 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 between the IAI protocol B/TCP MANU and AUTO modes, the connection will be cut off when the MANU/AUTO mode is switched. |
| 125 | Network attribute 6 | 1E32H                     | 0H to FFFFFFFFH |      | Bits 0 to 7: Module-initialization check timer setting when Ethernet is used (100 msec)<br>Bits 8 to 15: Module-initialization check timer setting when Ethernet is not used (100 msec)<br>Bits 16 to 23: Increment of "PC/TP reconnection delay at software reset" when Ethernet is used (sec)  |
| 126 | Network attribute 7 | 7D007D0H                  | 0H to FFFFFFFFH |      | Ethernet TCP/IP message communication attribute<br>Bits 0 to 15: Min timeout value (msec)<br>Bits 16 to 31: Mout timeout value (msec)  |



I/O Parameters

| No. | Parameter name   | Default value (Reference) | Input range          | Unit | Remarks  |
|-----|--|---------------------------|----------------------|------|--|
| 127 | Network attribute 8  | 5050214H                  | 0H to FFFFFFFFH      |      | Ethernet TCP/IP message communication attribute<br>Bits 0 to 7: CONNECT_TIMEOUT (Change is prohibited) (Setting of "0" is prohibited) (sec)<br>Bits 8 to 15: Connection retry interval (IAI protocol B/TCP) (sec)<br>Bits 16 to 23: Send timeout value (sec)<br>Bits 24 to 31: IAI protocol B-SIO non-communication check timer setting (sec) (IAI protocol B/TCP connection trigger)  |
| 128 | Network attribute 9  | 0                         | 0H to FFFFFFFFH      |      | Ethernet TCP/IP message communication attribute<br>Bits 0 to 15: SEL server open timeout value (sec) (No timeout check when "0" is set)  |
| 129 | Network attribute 10   | 0                         | 0H to FFFFFFFFH      |      | Ethernet operation requirement<br>Bits 0 to 3: Modbus/TCP (Remote I/O)<br>(0: Not in use<br>1: Use (Disable EXCEPTION status)<br>2: Use (Enable EXCEPTION status (upper two digits of error number))<br>* Refer to the explanation of error levels in the operation manual and perform processing appropriate for each error level.<br>Bits 4 to 7: TCP/IP message communication (0: Not in use, 1: Use)<br>Bits 8 to 31: Reserved (Operation requirement) |
| 130 | Own MAC address (H)  | 0H                        | Reference only (HEX) |      | Only lower two bytes are valid.  |
| 131 | Own MAC address (L)  | 0H                        | Reference only (HEX) |      |  |
| 132 | Own IP address (H)   | 192                       | 1 to 255             |      | *Setting of "0" and "127" is prohibited.   |
| 133 | Own IP address (MH)  | 168                       | 0 to 255             |      |  |
| 134 | Own IP address (ML)  | 0                         | 0 to 255             |      |  |
| 135 | Own IP address (L)   | 1                         | 1 to 254             |      | *Setting of "0" and "255" is prohibited.   |
| 136 | Subnet mask (H)  | 255                       | 0 to 255             |      |  |
| 137 | Subnet mask (MH)   | 255                       | 0 to 255             |      |  |
| 138 | Subnet mask (ML)   | 255                       | 0 to 255             |      |  |
| 139 | Subnet mask (L)  | 0                         | 0 to 255             |      |  |
| 140 | Default gateway (H)  | 0                         | 0 to 255             |      |  |
| 141 | Default gateway (MH)   | 0                         | 0 to 255             |      |  |
| 142 | Default gateway (ML)   | 0                         | 0 to 255             |      |  |
| 143 | Default gateway (L)  | 0                         | 0 to 255             |      |  |
| 144 | IAI protocol B/TCP: Own port number (MANU mode)                          | 64511                     | 1025 to 65535        |      | *Important note: Always set a unique number for each port number. (Duplication of 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 to 65535        |      |  |
| 146 | Channel 32 opened to user (TCP/IP): Own port number                      | 64513                     | 1025 to 65535        |      |  |
| 147 | Channel 33 opened to user (TCP/IP): Own port number                      | 64514                     | 1025 to 65535        |      |  |
| 148 | Channel 34 opened to user (TCP/IP): Own port number                      | 64515                     | 1025 to 65535        |      |  |
| 149 | IAI protocol B/TCP: IP address of connection destination (MANU mode) (H) | 192                       | 0 to 255             |      | *Setting of "0" and "127" is prohibited.   |



## I/O Parameters

| No.        | Parameter name  | Default value (Reference) | Input range   | Unit | Remarks  |
|------------|---|---------------------------|---------------|------|--|
| 150        | IAI protocol B/TCP: IP address of connection destination (MANU mode) (MH) | 168                       | 0 to 255      |      |  |
| 151        | IAI protocol B/TCP: IP address of connection destination (MANU mode) (ML) | 0                         | 0 to 255      |      |  |
| 152        | IAI protocol B/TCP: IP address of connection destination (MANU mode) (L)  | 100                       | 0 to 254      |      | * Setting of "0" and "255" is prohibited.  |
| 153        | IAI protocol B/TCP: Port number of connection destination (MANU mode)     | 64611                     | 0 to 65535    |      | * "0" can be set in the case of a server.<br>0 = Port number of connection destination is ignored (only the IP address is checked)<br>* "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 to 255      |      | * Setting of "0" and "127" is prohibited.  |
| 155        | IAI protocol B/TCP: IP address of connection destination (AUTO mode) (MH) | 168                       | 0 to 255      |      |  |
| 156        | IAI protocol B/TCP: IP address of connection destination (AUTO mode) (ML) | 0                         | 0 to 255      |      |  |
| 157        | IAI protocol B/TCP: IP address of connection destination (AUTO mode) (L)  | 100                       | 0 to 254      |      | * Setting of "0" and "255" is prohibited.  |
| 158        | IAI protocol B/TCP: Port number of connection destination (AUTO mode)     | 64611                     | 0 to 65535    |      | * "0" can be set in the case of a server.<br>0 = Port number of connection destination is ignored (only the IP address is checked)<br>* "0" cannot be set in the case of a client. |
| 159        | IAI protocol B/TCP: Own port number (AUTO mode)                           | 64516                     | 1025 to 65535 |      | * Important note: Always set a unique number for each port number.<br>(Duplication of port numbers is permitted only in the IAI protocol B/TCP MANU/AUTO modes.)                   |
| 160 to 169 | (For network expansion)   | 0                         |               |      |  |
| 170 to 282 | (For expansion)   | 0                         |               |      |  |
| 283        | Physical input port number for input function selection 000               | -1                        | -1 to 299     |      | * Invalid if a negative value is set. (Input function selection 000 will be specified to input port No. 0.)<br>(Main application version 0.19 or later)                            |
| 284        | Physical input port number for input function selection 001               | -1                        | -1 to 299     |      | * Invalid if a negative value is set. (Input function selection 001 will be specified to input port No. 1.)<br>(Main application version 0.19 or later)                            |
| 285        | Physical input port number for input function selection 002               | -1                        | -1 to 299     |      | * Invalid if a negative value is set. (Input function selection 002 will be specified to input port No. 2.)<br>(Main application version 0.19 or later)                            |
| 286        | Physical input port number for input function selection 003               | -1                        | -1 to 299     |      | * Invalid if a negative value is set. (Input function selection 003 will be specified to input port No. 3.)<br>(Main application version 0.19 or later)                            |
| 287        | Physical input port number for input function selection 004               | -1                        | -1 to 299     |      | * Invalid if a negative value is set. (Input function selection 004 will be specified to input port No. 4.)<br>(Main application version 0.19 or later)                            |
| 288        | Physical input port number for input function selection 005               | -1                        | -1 to 299     |      | * Invalid if a negative value is set. (Input function selection 005 will be specified to input port No. 5.)<br>(Main application version 0.19 or later)                            |



## I/O Parameters

| No. | Parameter name  | Default value (Reference) | Input range | Unit | Remarks  |
|-----|---|---------------------------|-------------|------|--|
| 289 | Physical input port number for input function selection 006 | -1                        | -1 to 299   |      | * Invalid if a negative value is set. (Input function selection 006 will be specified to input port No. 6.)<br>(Main application version 0.19 or later)  |
| 290 | Physical input port number for input function selection 007 | -1                        | -1 to 299   |      | * Invalid if a negative value is set. (Input function selection 007 will be specified to input port No. 7.)<br>* If "start program number" is specified for input function selection 007, specify the next larger input port number immediately adjacent to the LSB side of the start program number.<br>(Main application version 0.19 or later)  |
| 291 | Physical input port number for input function selection 008 | -1                        | -1 to 299   |      | * Invalid if a negative value is set. (Input function selection 008 will be specified to input port No. 8.)<br>* If "start program number" is specified for input function selection 008, specify the next larger input port number immediately adjacent to the LSB side of the start program number.<br>(Main application version 0.19 or later)  |
| 292 | Physical input port number for input function selection 009 | -1                        | -1 to 299   |      | * Invalid if a negative value is set. (Input function selection 009 will be specified to input port No. 9.)<br>* If "start program number" is specified for input function selection 009, specify the next larger input port number immediately adjacent to the LSB side of the start program number.<br>(Main application version 0.19 or later)  |
| 293 | Physical input port number for input function selection 010 | -1                        | -1 to 299   |      | * Invalid if a negative value is set. (Input function selection 010 will be specified to input port No. 10.)<br>* If "start program number" is specified for input function selection 010, specify the next larger input port number immediately adjacent to the LSB side of the start program number.<br>(Main application version 0.19 or later) |
| 294 | Physical input port number for input function selection 011 | -1                        | -1 to 299   |      | * Invalid if a negative value is set. (Input function selection 011 will be specified to input port No. 11.)<br>* If "start program number" is specified for input function selection 011, specify the next larger input port number immediately adjacent to the LSB side of the start program number.<br>(Main application version 0.19 or later) |
| 295 | Physical input port number for input function selection 012 | -1                        | -1 to 299   |      | * Invalid if a negative value is set. (Input function selection 012 will be specified to input port No. 12.)<br>* If "start program number" is specified for input function selection 012, specify the next larger input port number immediately adjacent to the LSB side of the start program number.<br>(Main application version 0.19 or later) |



## I/O Parameters

| No. | Parameter name  | Default value (Reference) | Input range | Unit | Remarks  |
|-----|---|---------------------------|-------------|------|--|
| 296 | Physical input port number for input function selection 013   | -1                        | -1 to 299   |      | * Invalid if a negative value is set. (Input function selection 013 will be specified to input port No. 13.)<br>* If "start program number" is specified for input function selection 013, specify the next larger input port number immediately adjacent to the LSB side of the start program number.<br>(Main application version 0.19 or later) |
| 297 | Physical input port number for input function selection 014   | -1                        | -1 to 299   |      | * Invalid if a negative value is set. (Input function selection 014 will be specified to input port No. 14.)<br>(Main application version 0.19 or later)   |
| 298 | Physical input port number for input function selection 015   | -1                        | -1 to 299   |      | * Invalid if a negative value is set. (Input function selection 015 will be specified to input port No. 15.)<br>(Main application version 0.19 or later)   |
| 299 | Physical output port number for output function selection 300 | 0                         | 0 to 599    |      | * Invalid if "0" is set. (Output function selection 300 will be specified to output port No. 300.)<br>(Main application version 0.19 or later)   |
| 300 | Physical output port number for output function selection 301 | 0                         | 0 to 599    |      | * Invalid if "0" is set. (Output function selection 301 will be specified to output port No. 301.)<br>(Main application version 0.19 or later)   |
| 301 | Physical output port number for output function selection 302 | 0                         | 0 to 599    |      | * Invalid if "0" is set. (Output function selection 302 will be specified to output port No. 302.)<br>(Main application version 0.19 or later)   |
| 302 | Physical output port number for output function selection 303 | 0                         | 0 to 599    |      | * Invalid if "0" is set. (Output function selection 303 will be specified to output port No. 303.)<br>(Main application version 0.19 or later)   |
| 303 | Physical output port number for output function selection 304 | 0                         | 0 to 599    |      | * Invalid if "0" is set. (Output function selection 304 will be specified to output port No. 304.)<br>(Main application version 0.19 or later)   |
| 304 | Physical output port number for output function selection 305 | 0                         | 0 to 599    |      | * Invalid if "0" is set. (Output function selection 305 will be specified to output port No. 305.)<br>(Main application version 0.19 or later)   |
| 305 | Physical output port number for output function selection 306 | 0                         | 0 to 599    |      | * Invalid if "0" is set. (Output function selection 306 will be specified to output port No. 306.)<br>(Main application version 0.19 or later)   |
| 306 | Physical output port number for output function selection 307 | 0                         | 0 to 599    |      | * Invalid if "0" is set. (Output function selection 307 will be specified to output port No. 307.)<br>(Main application version 0.19 or later)   |
| 307 | Physical output port number for output function selection 308 | 0                         | 0 to 599    |      | * Invalid if "0" is set. (Output function selection 308 will be specified to output port No. 308.)<br>(Main application version 0.19 or later)   |
| 308 | Physical output port number for output function selection 309 | 0                         | 0 to 599    |      | * Invalid if "0" is set. (Output function selection 309 will be specified to output port No. 309.)<br>(Main application version 0.19 or later)   |
| 309 | Physical output port number for output function selection 310 | 0                         | 0 to 599    |      | * Invalid if "0" is set. (Output function selection 310 will be specified to output port No. 310.)<br>(Main application version 0.19 or later)   |
| 310 | Physical output port number for output function selection 311 | 0                         | 0 to 599    |      | * Invalid if "0" is set. (Output function selection 311 will be specified to output port No. 311.)<br>(Main application version 0.19 or later)   |



## I/O Parameters

| No. | Parameter name   | Default value (Reference) | Input range | Unit | Remarks  |
|-----|--|---------------------------|-------------|------|--|
| 311 | Physical output port number for output function selection 312          | 0                         | 0 to 599    |      | * Invalid if "0" is set. (Output function selection 312 will be specified to output port No. 312.) (Main application version 0.19 or later)      |
| 312 | Physical output port number for output function selection 313          | 0                         | 0 to 599    |      | * Invalid if "0" is set. (Output function selection 313 will be specified to output port No. 313.) (Main application version 0.19 or later)      |
| 313 | Physical output port number for output function selection 314          | 0                         | 0 to 599    |      | * Invalid if "0" is set. (Output function selection 314 will be specified to output port No. 314.) (Main application version 0.19 or later)      |
| 314 | Physical output port number for output function selection 315          | 0                         | 0 to 599    |      | * Invalid if "0" is set. (Output function selection 315 will be specified to output port No. 315.) (Main application version 0.19 or later)      |
| 315 | Physical output port number for output function selection 300 (area 2) | 0                         | 0 to 599    |      | * Invalid if "0" is set. (No output port will be specified for output function selection 300 (area 2).) (Main application version 0.19 or later) |
| 316 | Physical output port number for output function selection 301 (area 2) | 0                         | 0 to 599    |      | * Invalid if "0" is set. (No output port will be specified for output function selection 301 (area 2).) (Main application version 0.19 or later) |
| 317 | Physical output port number for output function selection 302 (area 2) | 0                         | 0 to 599    |      | * Invalid if "0" is set. (No output port will be specified for output function selection 302 (area 2).) (Main application version 0.19 or later) |
| 318 | Physical output port number for output function selection 303 (area 2) | 0                         | 0 to 599    |      | * Invalid if "0" is set. (No output port will be specified for output function selection 303 (area 2).) (Main application version 0.19 or later) |
| 319 | Physical output port number for output function selection 304 (area 2) | 0                         | 0 to 599    |      | * Invalid if "0" is set. (No output port will be specified for output function selection 304 (area 2).) (Main application version 0.19 or later) |
| 320 | Physical output port number for output function selection 305 (area 2) | 0                         | 0 to 599    |      | * Invalid if "0" is set. (No output port will be specified for output function selection 305 (area 2).) (Main application version 0.19 or later) |
| 321 | Physical output port number for output function selection 306 (area 2) | 0                         | 0 to 599    |      | * Invalid if "0" is set. (No output port will be specified for output function selection 306 (area 2).) (Main application version 0.19 or later) |
| 322 | Physical output port number for output function selection 307 (area 2) | 0                         | 0 to 599    |      | * Invalid if "0" is set. (No output port will be specified for output function selection 307 (area 2).) (Main application version 0.19 or later) |
| 323 | Physical output port number for output function selection 308 (area 2) | 0                         | 0 to 599    |      | * Invalid if "0" is set. (No output port will be specified for output function selection 308 (area 2).) (Main application version 0.19 or later) |
| 324 | Physical output port number for output function selection 309 (area 2) | 0                         | 0 to 599    |      | * Invalid if "0" is set. (No output port will be specified for output function selection 309 (area 2).) (Main application version 0.19 or later) |
| 325 | Physical output port number for output function selection 310 (area 2) | 0                         | 0 to 599    |      | * Invalid if "0" is set. (No output port will be specified for output function selection 310 (area 2).) (Main application version 0.19 or later) |
| 326 | Physical output port number for output function selection 311 (area 2) | 0                         | 0 to 599    |      | * Invalid if "0" is set. (No output port will be specified for output function selection 311 (area 2).) (Main application version 0.19 or later) |



I/O Parameters

| No. | Parameter name   | Default value (Reference) | Input range | Unit | Remarks   |
|-----|--|---------------------------|-------------|------|---|
| 327 | Physical output port number for output function selection 312 (area 2) | 0                         | 0 to 599    |      | * Invalid if "0" is set. (No output port will be specified for output function selection 312 (area 2).)<br>(Main application version 0.19 or later)   |
| 328 | Physical output port number for output function selection 313 (area 2) | 0                         | 0 to 599    |      | * Invalid if "0" is set. (No output port will be specified for output function selection 313 (area 2).)<br>(Main application version 0.19 or later)   |
| 329 | Physical output port number for output function selection 314 (area 2) | 0                         | 0 to 599    |      | * Invalid if "0" is set. (No output port will be specified for output function selection 314 (area 2).)<br>(Main application version 0.19 or later)   |
| 330 | Physical output port number for output function selection 315 (area 2) | 0                         | 0 to 599    |      | * Invalid if "0" is set. (No output port will be specified for output function selection 315 (area 2).)<br>(Main application version 0.19 or later)   |
| 331 | Output function selection 300 (area 2)                                 | 0                         | 0 to 20     |      | 0: General-purpose output<br>1: Error output of operation-cancellation level or higher (ON)<br>2: Error output of operation-cancellation level or higher (OFF)<br>3: Error output of operation-cancellation level or higher + Emergency stop output (ON)<br>4: Error output of operation-cancellation level or higher + Emergency stop output (OFF)<br>(Main application version 0.19 or later) |
| 332 | Output function selection 301 (area 2)                                 | 0                         | 0 to 20     |      | 0: General-purpose output<br>1: READY output (PIO trigger program operation permitted)<br>2: READY output (PIO trigger program operation permitted AND error output of operation-cancellation level or higher not present)<br>3: READY output (PIO trigger program operation permitted AND error output of cold-start level or higher not present)<br>(Main application version 0.19 or later)  |
| 333 | Output function selection 302 (area 2)                                 | 0                         | 0 to 20     |      | 0: General-purpose output<br>1: Emergency stop output (ON)<br>2: Emergency stop output (OFF)<br>(Main application version 0.19 or later)  |
| 334 | Output function selection 303 (area 2)                                 | 0                         | 0 to 5      |      | 0: General-purpose output<br>1: AUTO mode output<br>2: Output during automatic operation (other parameter No. 12)<br>(Main application version 0.19 or later)   |
| 335 | Output function selection 304 (area 2)                                 | 0                         | 0 to 5      |      | 0: General-purpose output<br>1: Output if all valid axes are at their homes (= 0)<br>2: Output if all valid axes completed home return (coordinates have been confirmed)<br>3: Output if all valid axes are at their home preset coordinates  |
| 336 | Output function selection 305 (area 2)                                 | 0                         | 0 to 5      |      | 0: General-purpose output<br>2: Output when the axis 1 servo is ON (system monitor task output)<br>3: Reserved by the system<br>(Main application version 0.19 or later)  |
| 337 | Output function selection 306 (area 2)                                 | 0                         | 0 to 5      |      | 0: General-purpose output<br>2: Output when the axis 2 servo is ON (system monitor task output)<br>3: Reserved by the system<br>(Main application version 0.19 or later)  |
| 338 | Output function selection 307 (area 2)                                 | 0                         | 0 to 5      |      | 0: General-purpose output<br>2: Output when the axis 3 servo is ON (system monitor task output)<br>3: Reserved by the system<br>(Main application version 0.19 or later)  |



## I/O Parameters

| No.           | Parameter name                            | Default value<br>(Reference) | Input range | Unit | Remarks   |
|---------------|---|------------------------------|-------------|------|---|
| 339           | Output function<br>selection 308 (area 2) | 0                            | 0 to 5      |      | 0: General-purpose output<br>(Main application version 0.19 or later) |
| 340           | Output function<br>selection 309 (area 2) | 0                            | 0 to 5      |      | 0: General-purpose output<br>(Main application version 0.19 or later) |
| 341           | Output function<br>selection 310 (area 2) | 0                            | 0 to 5      |      | 0: General-purpose output<br>(Main application version 0.19 or later) |
| 342           | Output function<br>selection 311 (area 2) | 0                            | 0 to 5      |      | 0: General-purpose output<br>(Main application version 0.19 or later) |
| 343           | Output function<br>selection 312 (area 2) | 0                            | 0 to 5      |      | 0: General-purpose output<br>(Main application version 0.19 or later) |
| 344           | Output function<br>selection 313 (area 2) | 0                            | 0 to 5      |      | 0: General-purpose output<br>(Main application version 0.19 or later) |
| 345           | Output function<br>selection 314 (area 2) | 0                            | 0 to 5      |      | 0: General-purpose output<br>(Main application version 0.19 or later) |
| 346           | Output function<br>selection 315 (area 2) | 0                            | 0 to 5      |      | 0: General-purpose output<br>(Main application version 0.19 or later) |
| 347 to<br>400 | (For future expansion)                    | 0                            |             |      |   |



## 2. Parameters Common to All Axes

| No.      | Parameter name  | Default value (Reference) | Input range      | Unit   | Remarks  |
|----------|---|---------------------------|------------------|--------|--|
| 1        | Effective axis pattern                                      | 0000B                     | 00B to 11111111B |        |  |
| 2        | Default override  | 100                       | 1 to 100         |        | Used if not specified in program. (Invalid for SIO operation)  |
| 3 to 8   | (For expansion)   | 0                         |                  |        |  |
| 9        | Deadman-switch enabling physical axis pattern               | 11111111B                 | 00B to 11111111B |        | Not affected by a BASE command. (Always specify 11111111 if all axes are used. If not, the servo may be turned off only for the specified axes without cutting off the drive source (7-segment LED display does not show "DSF").)<br>* In the case of an optional (custom) specification, the optional (custom) specification is given priority over the deadman-switch enabling physical axes, drive-source cutoff specification, servo OFF specification, 7-segment display specification, etc.  |
| 10       | (For expansion)   | 0                         |                  |        |  |
| 11       | Default acceleration  | 20                        | 1 to 200         | 0.01 G | Used if not specified in position data, program or SIO message, etc.   |
| 12       | Default deceleration  | 20                        | 1 to 200         | 0.01 G | Used if not specified in position data, program or SIO message, etc.   |
| 13       | Default speed   | 30                        | 1 to 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 to 5           |        | 0: "Deceleration = Acceleration" when the deceleration in the operation point data is "0"<br>1: "Deceleration = 0" when the deceleration in the operation point data is "0"  |
| 15       | Maximum jog speed when home return is incomplete            | 30                        | 1 to 250         | mm/s   |  |
| 16 to 19 | (For expansion)   | 0                         | ~                |        |  |
| 20       | Maximum operating speed check timing                        | 1                         | 0 to 1           |        | 0: Check at input<br>1: Check at operation<br>* If "Check at operation" is selected, the distribution speed (CP) of specified speed or the specified speed (PTP) will be compared against the maximum operating speed of each axis and clamped at the allowable speed. Accordingly, the system can achieve its maximum performance in accordance with the operation command. However, complete check cannot be performed at input (since the command/operation start position is indeterminable). In the case of CP, the distribution speed will vary depending on the operation start position. Therefore, specifying CP at an unspecified position (first point movement, etc.) will cause the speed to fluctuate depending on where the operation is started. |
| 21       | Maximum operating speed for input value check               | 300                       | 1 to 9999        | mm/s   | If "Input" is selected as the maximum speed check timing, this parameter will be used to check for input error.  |
| 22       | Maximum acceleration  | 100                       | 1 to 999         | 0.01 G |  |
| 23       | Maximum deceleration  | 100                       | 1 to 999         | 0.01 G |  |
| 24       | Minimum emergency deceleration                              | 30                        | 1 to 300         | 0.01 G |  |
| 25       | (Acceleration/deceleration at home return (old))            | 30                        | 1 to 300         | 0.01 G | (Invalid)  |
| 26       | Acceleration/deceleration specification type                | 0                         | 0 to 5           |        | 0: T system, 1: P, M system  |
| 27       | Master axis type  | 0                         | 0 to 5           |        | 0: T system, 1: P system   |



Parameters Common to All Axes

| No.      | Parameter name  | Default value (Reference) | Input range     | Unit       | Remarks  |
|----------|---|---------------------------|-----------------|------------|--|
| 28       | Selection of inching → jog auto-switching prohibition | 0                         | 0 to 5          |            | 0: Execute auto-switching (Continuous button ON timer), 1: Prohibited<br>* Referenced by the PC/TP. (Handy terminal automatic switching function is not available.)  |
| 29       | All-axis setting bit pattern 1                        | 0                         | 0H to FFFFFFFFH |            | Bits 0 to 3: Selection of use of last PC/TP inching distance (0: Do not use, 1: Use)<br>* Referenced by the PC/TP (Excluding ANSI-compatible TP) (PC software version 2.0.0.42 or later or TP application version 1.09 or later).<br>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)<br>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) |
| 30       | Default division angle                                | 150                       | 0 to 1200       | 0.1 degree | ("0" can be input in PC software version 1.1.1.0 or later or TP application version 1.06 or later)   |
| 31       | Default division distance                             | 0                         | 0 to 10000      | mm         | ("0" can be input in PC software version 1.1.1.0 or later or TP application version 1.06 or later)   |
| 32       | Arch-trigger start-point check type                   | 0                         | 0 to 5          |            | 0: Check operation amount and actual position, 1: Check operation amount only  |
| 33       | Safety speed in manual mode                           | 250                       | 1 to 250        | mm/s       | * This parameter is treated as a value equivalent to or below the minimum value set in "Axis-specific parameter No. 29, VLMX speed" for all valid axes.  |
| 34 to 50 | (For expansion)                                       | 0                         | ~               |            |  |

PC: PC software  
TP: Teaching pendant



### 3. Axis-Specific Parameters

| No     | Parameter name  | Default value (Reference) | Input range           | Unit        | Remarks  |
|--------|---|---------------------------|-----------------------|-------------|--|
| 1      | Axis operation type   | 0                         | 0 to 1                |             | 0: Linear movement axis, 1: Rotational movement axis (Angle control)   |
| 2 to 5 | (For expansion)   | 0                         | ~                     |             |  |
| 6      | Coordinate/physical-operation direction selection           | 1                         | 0 to 1                |             | 0: Motor CCW → Positive direction on the coordinate system<br>1: Motor CCW → Negative direction on the coordinate system   |
| 7      | Soft limit +  | 50000                     | -99999999 to 99999999 | 0.001 mm    | Fixed to 359.999 degrees internally in the index mode. Invalid in the infinite-stroke mode.  |
| 8      | Soft limit -  | 0                         | -99999999 to 99999999 | 0.001 mm    | Fixed to 0 degree internally in the index mode. Invalid in the infinite-stroke mode.   |
| 9      | Soft-limit actual position margin                           | 2000                      | 0 to 9999             | 0.001 mm    | Actual position margin in the positioning boundary critical zone in the infinite-stroke mode   |
| 10     | Home-return method  | 0                         | 0 to 5                |             | 0: Search phase Z after end search<br>1: Current position 0 home (This parameter can be specified only with an incremental encoder. Pay attention to contact.),<br>2: Current position = Preset home (This parameter can be specified only with an incremental encoder. Pay attention to contact.) |
| 11     | Home-return end-search direction selection                  | 0                         | 0 to 1                |             | 0: Negative end of the coordinate system<br>1: Positive end of the coordinate system   |
| 12     | Home preset value   | 0                         | -99999999 to 99999999 | 0.001 mm    | (Refer to axis-specific parameter No. 76)  |
| 13     | SIO/PIO home-return order                                   | 0                         | 0 to 16               |             | Executed from the smallest one.  |
| 14     | For future expansion (Change prohibited)                    | 0                         | Reference only        |             |  |
| 15     | For future expansion (Change prohibited)                    | 0                         | Reference only        |             |  |
| 16     | For future expansion (Change prohibited)                    | 0                         | Reference only        |             |  |
| 17     | For future expansion (Change prohibited)                    | 10                        | Reference only        | mm/sec      |  |
| 18     | For future expansion (Change prohibited)                    | 100                       | Reference only        | mm/sec      |  |
| 19     | End search speed at home return                             | 20                        | 1 to 100              | mm/sec      |  |
| 20     | Phase-Z search speed at home return                         | 3                         | 1 to 10               | mm/sec      | Exercise caution, since limitations apply depending on the read/encoder pulse count.   |
| 21     | Offset travel distance at home return                       | 2500                      | -99999999 to 99999999 | 0.001 mm    | Offset travel distance from the ideal phase-Z position (Positive value = Applied in the direction of moving away from the end) (Refer to axis-specific parameter No. 76)   |
| 22     | Error check tolerance for phase-Z position at home return   | 0                         | 0 to 99999999         | 0.001 mm    | Minimum allowable actual distance of "End (mechanical or LS) – Phase Z," in the case of a rotary encoder   |
| 23     | Phase-Z count per encoder revolution                        | 1                         | 1 to 8                |             | Only "1" can be set, in the case of an absolute encoder.   |
| 24     | Push stop check time at home return                         | 1500                      | 1 to 5000             | msec        | Used to check the push motion during home return.  |
| 25     | Push stop check time at positioning                         | 500                       | 1 to 5000             | msec        | Used to check the push motion during PUSH command operation.   |
| 26     | (Phase-Z evacuation distance at absolute home return (old)) | 1000                      | 0 to 99999            | 0.001 mm    | Evacuation distance from the actual phase-Z position (Positive value = Applied in the direction of moving away from the end) (Phase-shift prevention margin) (Refer to axis-specific parameter No. 76)   |
| 27     | Maximum motor speed   | 5000                      | Reference only        | rpm, mm/sec | Rpm value in the case of a rotary encoder (Change prohibited)  |
| 28     | Maximum operating speed of each axis                        | 300                       | 1 to 9999             | mm/s        |  |
| 29     | VLMX speed  | 300                       | 1 to 9999             | mm/s        | During VLMX operation, the maximum operating speed of each axis or VLMX speed, whichever is lower, is used as the maximum speed of the applicable axis.  |
| 30     | Servo ON check time   | 20                        | 0 to 5000             | msec        | Brake equipped: Time after receiving a servo-ON start response until start of brake unlocking<br>Brake not equipped: Time after receiving a servo ON start response until transition to an operation-enabled status  |



## Axis-Specific Parameters

| No       | Parameter name   | Default value (Reference) | Input range           | Unit                      | Remarks  |
|----------|--|---------------------------|-----------------------|---------------------------|--|
| 31       | Offset travel speed at home return   | 3                         | 1 to 500              | mm/sec                    |  |
| 32       | Actual distance between phase Z and end  | 0                         | -1 to 99999           | 0.001 mm                  | Absolute distance from the end (mechanical or LS). Obtained automatically if the distance is a negative value. When multiple actuators are combined, it is recommended to write the flash ROM after automatic acquisition. (Refer to axis-specific parameter No. 76) |
| 33       | Ideal distance between phase Z and end   | 0                         | 0 to 99999            | 0.001 mm                  | Absolute distance from the end (mechanical or LS). (Refer to axis-specific parameter No. 76)   |
| 34       | Brake equipment specification  | 0                         | 0 to 1                |                           | 0: Not equipped, 1: Equipped   |
| 35       | Brake unlock check time  | 10                        | 0 to 3000             | msec                      | Time after receiving a brake-unlock start response until transition to an operation-enabled status   |
| 36       | Brake lock check time  | 10                        | 0 to 1000             | msec                      | Time after receiving a brake-lock start response until start of servo OFF  |
| 37       | Change prohibited  | 0                         | 0 to 1                |                           | 0: Rotary encoder  |
| 38       | Encoder ABS/INC type   | 0                         | 0 to 1                |                           | 0: INC, 1: ABS   |
| 39       | Change prohibited  | 1                         | 0 to 1                |                           |  |
| 40       | Pole-sense initial tryout direction selection (For future expansion = Change prohibited) | 0                         | 0 to 1                |                           | 0: Negative end of the coordinate system<br>1: Positive end of the coordinate system   |
| 41       | Pole sense speed (For future expansion = Change prohibited)                              | 25                        | 1 to 100              | DRVVR                     |  |
| 42       | Encoder resolution   | 800                       | 0 to 99999999         | Pulse/rev, 0.001 μm/pulse | Pulses (before division)/rev, in the case of a rotary encoder  |
| 43       | Encoder division ratio   | 0                         | -7 to 7               |                           | Pulses are multiplied by ("n"th power of 1/2).   |
| 44       | Length measurement correction  | 0                         | -99999999 to 99999999 | 0.001 mm/1M               | Valid only for linear movement axes. (Coordinates other than the encoder reference Z point will change proportionally.)  |
| 45 to 46 | (For expansion)  | 0                         |                       |                           |  |
| 47       | Screw lead   | 6000                      | 1 to 99999999         | 0.001 mm                  | Valid only for linear movement axes.   |
| 48 to 49 | (For expansion)  | 0                         |                       |                           |  |
| 50       | Gear ratio numerator   | 1                         | 1 to 99999999         |                           |  |
| 51       | Gear ratio denominator   | 1                         | 1 to 99999999         |                           |  |
| 52       | (For expansion)  | 0                         |                       |                           |  |
| 53       | Setting bit pattern 1 of each axis   | 0                         | 0H to FFFFFFFFH       |                           |  |
| 54       | Travel distance for push stop detection at home return                                   | 20                        | 1 to 99999            | 0.001 mm                  | Used to check the push motion during home return.  |
| 55       | Travel distance for push stop detection at positioning                                   | 30                        | 1 to 99999            | 0.001 mm                  | Used to check the push motion during PUSH command operation.   |
| 56       | Push-abort deviation ratio at home return  | 5000                      | 1 to 99999            |                           | Deviation is compared against "Steady-state deviation of push speed + Push-speed pulse speed x Abort deviation ratio."   |
| 57       | Push-abort deviation ratio at positioning  | 5000                      | 1 to 99999            |                           | Deviation is compared against "Steady-state deviation of push speed + Push-speed pulse speed x Abort deviation ratio."   |
| 58       | Positioning band   | 100                       | 1 to 9999             | 0.001 mm                  |  |
| 59       | Allowable deviation error ratio (Maximum speed pulse ratio)                              | 300                       | 1 to 9999             |                           | Deviation is compared against "Steady-state deviation of maximum operating speed of each axis + Pulse speed of maximum operating speed of each axis x Allowable deviation error ratio."  |
| 60       | Position gain  | 45                        | 1 to 9999             | /s                        |  |



## Axis-Specific Parameters

| No | Parameter name   | Default value (Reference) | Input range           | Unit     | Remarks  |
|----|--|---------------------------|-----------------------|----------|--|
| 61 | FF gain  | 0                         | 0 to 500              | %        |  |
| 62 | Synchro FB gain  | 77                        | 0 to 1000             |          |  |
| 63 | Stop special output range                                      | 0                         | 0 to 9999             | Pulse    | Invalid if "0" is set.   |
| 64 | Stop special output value                                      | 0                         | 0 to 999              | DRVVR    |  |
| 65 | Mating synchro-axis number                                     | 0                         | 0 to 8                |          | Must be input for both axes. (Of the axis pair, the axis with the smaller axis number becomes the master axis. Both axes must have the same resolution characteristics. Commands cannot be issued to the slave axis.) (Invalid if "0" is set)  |
| 66 | Mode selection for rotational movement axis                    | 0                         | 0 to 5                |          | 0: Normal, 1: Index mode   |
| 67 | Short-cut control selection for rotational movement axis       | 0                         | 0 to 5                |          | 0: Do not select, 1: Select (Valid only in the index mode AND when an incremental encoder is used)   |
| 68 | Mode selection for linear movement axis                        | 0                         | 0 to 5                |          | 0: Normal, 1: Infinite-stroke mode (Note: Positioning boundary applies. This setting can be specified only when an incremental encoder is used.)   |
| 69 | (For expansion)  | 0                         | ~                     |          |  |
| 70 | DRVVR value at maximum motor speed                             | 32767                     | Reference only        | DRVVR    | For adjustment by the manufacturer   |
| 71 | DRVVR value at 3x motor torque                                 | 32767                     | Reference only        | DRVVR    | For adjustment by the manufacturer   |
| 72 | DRVVR + offset   | 1                         | Reference only        | DRVVR    | For adjustment by the manufacturer   |
| 73 | DRVVR - offset   | 0                         | Reference only        | DRVVR    | For adjustment by the manufacturer   |
| 74 | DRVVR MAX  | 32436                     | Reference only        | DRVVR    | For adjustment by the manufacturer   |
| 75 | DRVVR MIN  | -32435                    | Reference only        | DRVVR    | For adjustment by the manufacturer   |
| 76 | Home-adjustment parameter set selection                        | 1                         | Reference only        |          | (Change prohibited)<br>0: P21 = Phase-Z evacuation distance at incremental home return<br>P12 = Ideal phase-Z position coordinate<br>1: P33 = Automatically loaded even when "0"; set to "actual distance" when P33 = "0"<br>P21 = Offset travel distance at home return<br>P12 = Coordinates after offset travel at home return<br>P26 is invalid.<br>(For simplification of adjustment)  |
| 77 | Synchro S pulse  | 3                         | 0 to 99999            | Pulse    |  |
| 78 | Maximum takeoff command amount                                 | 0                         | -3000 to 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)<br>* Important: Input using the same sign as the rising coordinate direction. (0.100 mm to 0.500 mm in absolute value as a guideline)<br>* 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 to 3000             | 0.001 mm | Absolute value input   |
| 80 | Maximum forced-feed range                                      | 0                         | 0 to 9999             | 0.001 mm | For reduction of settling time. (Invalid range if "0" is set) (Approx. 1.000 mm as a guideline)  |
| 81 | Minimum forced-feed range                                      | 200                       | 0 to 9999             | 0.001 mm |  |
| 82 | Medium forced-feed range                                       | 600                       | 0 to 9999             | 0.001 mm |  |
| 83 | Absolute synchro slave-axis initialization cancellation        | 0                         | 0 to 5                |          | Valid only with a synchro slave axis.  |
| 84 | Maximum synchronization correction speed of synchro slave axis | 5                         | 0 to 100              | mm/sec   | Maximum travel speed for synchronization position correction of slave axis. Valid only with a synchro slave axis.<br>* Note: Not limited by the safety speed.  |
| 85 | Home-return acceleration/ deceleration                         | 15                        | 1 to 300              | 0.01 G   |  |
| 86 | Zone 1 MAX   | 0                         | -99999999 to 99999999 | 0.001 mm | Valid only when MAX > MIN. * Must be inside the range for at least 3 msec.   |



## Axis-Specific Parameters

| No        | Parameter name       | Default value (Reference) | Input range           | Unit     | Remarks   |
|-----------|----------------------|---------------------------|-----------------------|----------|---|
| 87        | Zone 1 MIN           | 0                         | -99999999 to 99999999 | 0.001 mm | Valid only when MAX > MIN. * Must be inside the range for at least 3 msec.  |
| 88        | Zone 1 output number | 0                         | 0 to 899              |          | Physical output port or global flag (Output is invalid if "0" is input; multiple specification is invalid) ("0" can be input in PC software version 1.0.0.0 or later or TP application version 1.06 or later) |
| 89        | Zone 2 MAX           | 0                         | -99999999 to 99999999 | 0.001 mm | Valid only when MAX > MIN. * Must be inside the range for at least 3 msec.  |
| 90        | Zone 2 MIN           | 0                         | -99999999 to 99999999 | 0.001 mm | Valid only when MAX > MIN. * Must be inside the range for at least 3 msec.  |
| 91        | Zone 2 output number | 0                         | 0 to 899              |          | Physical output port or global flag (Output is invalid if "0" is input; multiple specification is invalid) ("0" can be input in PC software version 1.0.0.0 or later or TP application version 1.06 or later) |
| 92        | Zone 3 MAX           | 0                         | -99999999 to 99999999 | 0.001 mm | Valid only when MAX > MIN. * Must be inside the range for at least 3 msec.  |
| 93        | Zone 3 MIN           | 0                         | -99999999 to 99999999 | 0.001 mm | Valid only when MAX > MIN. * Must be inside the range for at least 3 msec.  |
| 94        | Zone 3 output number | 0                         | 0 to 899              |          | Physical output port or global flag (Output is invalid if "0" is input; multiple specification is invalid) ("0" can be input in PC software version 1.0.0.0 or later or TP application version 1.06 or later) |
| 95        | Zone 4 MAX           | 0                         | -99999999 to 99999999 | 0.001 mm | Valid only when MAX > MIN. * Must be inside the range for at least 3 msec.  |
| 96        | Zone 4 MIN           | 0                         | -99999999 to 99999999 | 0.001 mm | Valid only when MAX > MIN. * Must be inside the range for at least 3 msec.  |
| 97        | Zone 4 output number | 0                         | 0 to 899              |          | Physical output port or global flag (Output is invalid if "0" is input; multiple specification is invalid) ("0" can be input in PC software version 1.0.0.0 or later or TP application version 1.06 or later) |
| 98 to 120 | (For expansion)      | 0                         | ~                     |          |   |

PC: PC software  
TP: Teaching pendant



## 4. Driver Card Parameters

| No | Parameter name   | Default value (Reference) | Input range    | Unit                   | Remarks  |
|----|--|---------------------------|----------------|------------------------|--|
| 1  | Type (upper) (Manufacturing information)   | Space                     | Reference only |                        | For adjustment by the manufacturer   |
| 2  | Type (middle) (Manufacturing information)  | Space                     | Reference only |                        | For adjustment by the manufacturer   |
| 3  | Type (lower) (Manufacturing information)   | Space                     | Reference only |                        | For adjustment by the manufacturer   |
| 4  | Manufacturing data 4 (Manufacturing information)   | Space                     | Reference only |                        | For adjustment by the manufacturer   |
| 5  | Manufacturing data 5 (Manufacturing information)   | Space                     | Reference only |                        | For adjustment by the manufacturer   |
| 6  | Manufacturing data 6 (Manufacturing information)   | Space                     | Reference only |                        | For adjustment by the manufacturer   |
| 7  | Manufacturing data 7 (Manufacturing information)   | Space                     | Reference only |                        | For adjustment by the manufacturer   |
| 8  | Board type (Function information)  | 30                        | Reference only |                        | For adjustment by the manufacturer   |
| 9  | Function information 01 (hard): Encoder support information (upper word)                                       | 0000H                     | Reference only | Encoder ID bit pattern | For adjustment by the manufacturer   |
| 10 | Function information 02 (hard): Encoder support information (lower word)                                       | 0001H                     | Reference only | Encoder ID bit pattern |  |
| 11 | Function information 03 (hard): Hardware support information word 0  | 0004H                     | Reference only |                        | Bit 0: Brake support specification bit (1: Supported, 0: Not supported)<br>Bit 1: For future expansion<br>Bit 2: Motor capacity specification bit (1: □42/□56 motor, 0: □20/□28 motor) |
| 12 | Function information 04 (hard): For future expansion   | 0000H                     | Reference only |                        | For adjustment by the manufacturer   |
| 13 | Function information 05 (hard): For future expansion   | 0000H                     | Reference only |                        | For adjustment by the manufacturer   |
| 14 | Function information 06 (hard): For future expansion   | 0000H                     | Reference only |                        | For adjustment by the manufacturer   |
| 15 | Function information 07 (soft): Motor support information (upper word)   | 0000H                     | Reference only | Motor ID bit pattern   | For adjustment by the manufacturer   |
| 16 | Function information 08 (soft): Motor support information (lower word)   | FFFFH                     | Reference only | Motor ID bit pattern   |  |
| 17 | Function information 09 (soft): Encoder support information (upper word)                                       | 0000H                     | Reference only | Encoder ID bit pattern | For adjustment by the manufacturer   |
| 18 | Function information 10 (soft): Encoder support information (lower word)                                       | 0001H                     | Reference only | Encoder ID bit pattern |  |
| 19 | Function information 11 (soft): Software support information word 0 (For future expansion = Change prohibited) | 0000H                     | Reference only |                        | Bit 0: For future expansion  |
| 20 | Function information 12 (soft): Software version information   | 0000H                     | Reference only |                        |  |
| 21 | Function information 13 (soft): For future expansion   | 0000H                     | Reference only |                        | For adjustment by the manufacturer   |
| 22 | Function information 14 (soft): System log control word  | 0000H                     | Reference only |                        | Bits 0 to 4: For future expansion  |
| 23 | Configuration information 01: Configured capacity (rated motor output)   | 0011H                     | Reference only |                        |  |



Driver Card Parameters

| No | Parameter name  | Default value (Reference) | Input range    | Unit                        | Remarks  |
|----|---|---------------------------|----------------|-----------------------------|--|
| 24 | Configuration information 02: Configured voltage (motor voltage)      | 0018H                     | Reference only |                             |  |
| 25 | Configuration information 03: Motor/encoder configuration information | 0500H                     | Reference only | Motor/encoder ID bit number |  |
| 26 | Configuration information 04: For future expansion                    | 0000H                     | Reference only |                             | For adjustment by the manufacturer   |
| 27 | Configuration information 05: Encoder resolution (upper word)         | 0000H                     | Reference only |                             |  |
| 28 | Configuration information 06: Encoder resolution (lower word)         | 0320H                     | Reference only |                             |  |
| 29 | Configuration information 07: Motor/encoder characteristics word      | 0004H                     | Reference only |                             | Bit 0: Change prohibited (0: Rotary)<br>Bit 1: Change prohibited (0: Incremental)<br>Bit 2: Change prohibited (1: Magnetic sensor equipped)<br>Bit 3: Brake equipment bit (1: Equipped, 0: Not equipped)   |
| 30 | Configuration information 08: For future expansion                    | 0000H                     | Reference only |                             | For adjustment by the manufacturer   |
| 31 | Configuration information 09: Control characteristics word            | 0000H                     | Reference only |                             | For adjustment by the manufacturer   |
| 32 | Configuration information 10: Push torque limit at home return        | 40                        | 0 to 150       | %                           |  |
| 33 | Configuration information 11: Push torque limit at positioning        | 70                        | 0 to 70        | %                           |  |
| 34 | Configuration information 12: Control characteristics word 2          | 300H                      | 0000 to FFFF   |                             | Bits 0 to 7: For future expansion<br>Bit 8: Initial moving direction in excitation-phase signal detection operation (0: CW, 1: CCW)<br>Bit 9: Stop mode selection (0: Full servo mode, 1: Complete stop mode)<br><br>* In the case of coating or other application where operation focus is given to the locus, select "0" (full servo mode). (In this case, the complete stop function is disabled.)<br>In all other applications, "1" (complete stop mode) is normally selected. |
| 35 | Configuration information 13: For future expansion                    | 0H                        | Reference only |                             |  |
| 36 | Configuration information 14: For future expansion                    | 0H                        | Reference only |                             |  |
| 37 | Configuration information 15: For future expansion                    | 0H                        | Reference only |                             | For adjustment by the manufacturer   |
| 38 | For future expansion  | 0H                        | Reference only |                             |  |
| 39 | For future expansion  | 0H                        | Reference only |                             |  |
| 40 | For future expansion  | 0H                        | Reference only |                             |  |
| 41 | For future expansion  | 0H                        | Reference only |                             |  |
| 42 | Torque filter time constant   | 0H                        | 0 to 2500      |                             |  |
| 43 | For future expansion  | 0H                        | Reference only |                             |  |
| 44 | Speed-loop proportional gain time constant (upper word)               | 0H                        | 0000H to 0000H |                             | For pulse motor  |
| 45 | Speed loop proportional gain (lower word)                             | 12CH                      | 0000H to 7530H |                             | For pulse motor  |
| 46 | Speed loop integral gain (upper word)                                 | 0H                        | 0000H to 0004H |                             | For pulse motor  |
| 47 | Speed loop integral gain (lower word)                                 | 11F9H                     | 0000H to FFFFH |                             | For pulse motor  |
| 48 | Excitation-phase fixed mode parameter                                 | 0H                        | Reference only |                             | For pulse motor (Percentage of rated motor current)  |



## Driver Card Parameters

| No | Parameter name       | Default value (Reference) | Input range    | Unit | Remarks |
|----|----------------------|---------------------------|----------------|------|---------|
| 49 | For future expansion | 0H                        | Reference only |      |         |
| 50 | For future expansion | 0H                        | Reference only |      |         |
| 51 | For future expansion | 0H                        | Reference only |      |         |
| 52 | For future expansion | 0H                        | Reference only |      |         |
| 53 | For future expansion | 0H                        | Reference only |      |         |
| 54 | For future expansion | 0H                        | Reference only |      |         |
| 55 | For future expansion | 0H                        | Reference only |      |         |
| 56 | For future expansion | 0H                        | Reference only |      |         |
| 57 | For future expansion | 0H                        | Reference only |      |         |
| 58 | For future expansion | 0H                        | Reference only |      |         |
| 59 | For future expansion | 0H                        | Reference only |      |         |
| 60 | For future expansion | 0H                        | Reference only |      |         |
| 61 | For future expansion | 0H                        | Reference only |      |         |
| 62 | For future expansion | 0H                        | Reference only |      |         |
| 63 | For future expansion | 0H                        | Reference only |      |         |
| 64 | For future expansion | 0H                        | Reference only |      |         |
| 65 | For future expansion | 0H                        | Reference only |      |         |
| 66 | For future expansion | 0H                        | Reference only |      |         |
| 67 | For future expansion | 0H                        | Reference only |      |         |
| 68 | For future expansion | 0H                        | Reference only |      |         |
| 69 | For future expansion | 0H                        | Reference only |      |         |
| 70 | For future expansion | 0H                        | Reference only |      |         |
| 71 | For future expansion | 0H                        | Reference only |      |         |
| 72 | For future expansion | 0H                        | Reference only |      |         |
| 73 | For future expansion | 0H                        | Reference only |      |         |
| 74 | For future expansion | 0H                        | Reference only |      |         |
| 75 | For future expansion | 0H                        | Reference only |      |         |
| 76 | For future expansion | 0H                        | Reference only |      |         |
| 77 | For future expansion | 0H                        | Reference only |      |         |
| 78 | For future expansion | 0H                        | Reference only |      |         |
| 79 | For future expansion | 0H                        | Reference only |      |         |
| 80 | For future expansion | 0H                        | Reference only |      |         |
| 81 | For future expansion | 0H                        | Reference only |      |         |
| 82 | For future expansion | 0H                        | Reference only |      |         |
| 83 | For future expansion | 0H                        | Reference only |      |         |
| 84 | For future expansion | 0H                        | Reference only |      |         |
| 85 | For future expansion | 0H                        | Reference only |      |         |
| 86 | For future expansion | 0H                        | Reference only |      |         |
| 87 | For future expansion | 0H                        | Reference only |      |         |
| 88 | For future expansion | 0H                        | Reference only |      |         |
| 89 | For future expansion | 0H                        | Reference only |      |         |
| 90 | For future expansion | 0H                        | Reference only |      |         |
| 91 | For future expansion | 0H                        | Reference only |      |         |
| 92 | For future expansion | 0H                        | Reference only |      |         |
| 93 | For future expansion | 0H                        | Reference only |      |         |
| 94 | For future expansion | 0H                        | Reference only |      |         |
| 95 | For future expansion | 0H                        | Reference only |      |         |
| 96 | For future expansion | 0H                        | Reference only |      |         |
| 97 | For future expansion | 0H                        | Reference only |      |         |



## Driver Card Parameters

|     |  |    |                |                                    |
|-----|--|----|----------------|------------------------------------|
| 98  | Overrun error counter (Query information)                      | 0H | Reference only | For adjustment by the manufacturer |
| 99  | FPGA detection error counter (Query information)               | 0H | Reference only | For adjustment by the manufacturer |
| 100 | Speed-command underrun-count error counter (Query information) | 0H | Reference only | For adjustment by the manufacturer |
| 101 | For future expansion (Query information)                       | 0H | Reference only | For adjustment by the manufacturer |
| 102 | Overload error counter (Query information)                     | 0H | Reference only | For adjustment by the manufacturer |
| 103 | Overspeed error counter (Query information)                    | 0H | Reference only | For adjustment by the manufacturer |
| 104 | Overcurrent error counter (Query information)                  | 0H | Reference only | For adjustment by the manufacturer |
| 105 | Overheat error counter (Query information)                     | 0H | Reference only | For adjustment by the manufacturer |
| 106 | Encoder error counter (Query information)                      | 0H | Reference only | For adjustment by the manufacturer |
| 107 | CPU error counter (Query information)                          | 0H | Reference only | For adjustment by the manufacturer |
| 108 | Phase-U current sense adjustment value (Query information)     | 0H | Reference only | For adjustment by the manufacturer |
| 109 | Phase-W current sense adjustment value (Query information)     | 0H | Reference only | For adjustment by the manufacturer |
| 110 | For future expansion (Query information)                       | 0H | Reference only | For adjustment by the manufacturer |
| 111 | For future expansion (Query information)                       | 0H | Reference only | For adjustment by the manufacturer |
| 112 | For future expansion (Query information)                       | 0H | Reference only | For adjustment by the manufacturer |



## 5. Encoder Parameters

| No       | Parameter name  | Default value (Reference) | Input range    | Unit                               | Remarks                            |
|----------|---|---------------------------|----------------|------------------------------------|------------------------------------|
| 1        | Type (upper) (Manufacturing information)  | Space                     | Reference only |                                    | For adjustment by the manufacturer |
| 2        | Type (middle) (Manufacturing information)                                       | Space                     | Reference only |                                    | For adjustment by the manufacturer |
| 3        | Type (lower) (Manufacturing information)  | Space                     | Reference only |                                    | For adjustment by the manufacturer |
| 4        | Manufacturing data 4 (Manufacturing information)                                | Space                     | Reference only |                                    | For adjustment by the manufacturer |
| 5        | Manufacturing data 5 (Manufacturing information)                                | Space                     | Reference only |                                    | For adjustment by the manufacturer |
| 6        | Manufacturing data 6 (Manufacturing information)                                | Space                     | Reference only |                                    | For adjustment by the manufacturer |
| 7        | Manufacturing data 7 (Manufacturing information)                                | Space                     | Reference only |                                    | For adjustment by the manufacturer |
| 8        | Board type (Function information)   | 0                         | Reference only |                                    | For adjustment by the manufacturer |
| 9        | Function information 01: Configured capacity (rated motor output)               | 0000H                     | Reference only | W                                  | For adjustment by the manufacturer |
| 10       | Function information 02: Configured voltage (motor voltage)                     | 0000H                     | Reference only | V                                  | For adjustment by the manufacturer |
| 11       | Function information 03: Motor/encoder configuration information                | 0000H                     | Reference only | Motor/encoder ID bit number        | For adjustment by the manufacturer |
| 12       | Function information 04: Encoder resolution (upper word)                        | 0000H                     | Reference only |                                    | For adjustment by the manufacturer |
| 13       | Function information 05: Encoder resolution (lower word)                        | 0000H                     | Reference only |                                    | For adjustment by the manufacturer |
| 14       | Function information 06: Motor/encoder characteristics word                     | 0000H                     | Reference only |                                    | For adjustment by the manufacturer |
| 15       | Function information 07: Motor/encoder control word 1 (Also applicable to nX-E) | 0000H                     | Reference only | 0.1 N (Kelvin = Temperature scale) | For adjustment by the manufacturer |
| 16       | Function information 08: Motor/encoder control word 2 (Also applicable to nX-E) | 0000H                     | Reference only |                                    | For adjustment by the manufacturer |
| 17       | Function information 09: Motor/encoder control word 3 (Also applicable to nX-E) | 0000H                     | Reference only |                                    | For adjustment by the manufacturer |
| 18       | Function information 10: Motor/encoder control word 4 (Also applicable to nX-E) | 0000H                     | Reference only |                                    | For adjustment by the manufacturer |
| 19       | Function information 11: (For future expansion)                                 | 0000H                     | Reference only |                                    | For adjustment by the manufacturer |
| 20       | Function information 12: (For future expansion)                                 | 0000H                     | Reference only |                                    | For adjustment by the manufacturer |
| 21       | Function information 13: (For future expansion)                                 | 0000H                     | Reference only |                                    | For adjustment by the manufacturer |
| 22       | Function information 14: (For future expansion)                                 | 0000H                     | Reference only |                                    | For adjustment by the manufacturer |
| 23 to 30 | Card parameter (by board type)  | 0000H                     | Reference only |                                    | For adjustment by the manufacturer |



## 6. I/O-Slot Card Parameters

| No        | Parameter name                                   | Default value (Reference) | Input range    | Unit | Remarks                            |
|-----------|--|---------------------------|----------------|------|------------------------------------|
| 1         | Type (upper) (Manufacturing information)         | Space                     | Reference only |      | For adjustment by the manufacturer |
| 2         | Type (middle) (Manufacturing information)        | Space                     | Reference only |      | For adjustment by the manufacturer |
| 3         | Type (lower) (Manufacturing information)         | Space                     | Reference only |      | For adjustment by the manufacturer |
| 4         | Manufacturing data 4 (Manufacturing information) | Space                     | Reference only |      | For adjustment by the manufacturer |
| 5         | Manufacturing data 5 (Manufacturing information) | Space                     | Reference only |      | For adjustment by the manufacturer |
| 6         | Manufacturing data 6 (Manufacturing information) | Space                     | Reference only |      | For adjustment by the manufacturer |
| 7         | Manufacturing data 7 (Manufacturing information) | Space                     | Reference only |      | For adjustment by the manufacturer |
| 8         | Board type (Function information)                | 0                         | Reference only |      | For adjustment by the manufacturer |
| 9         | Function information 01 (by board type)          | 0000H                     | Reference only |      | For adjustment by the manufacturer |
| 10        | Function information 02 (by board type)          | 0000H                     | Reference only |      | For adjustment by the manufacturer |
| 11        | Function information 03 (by board type)          | 0000H                     | Reference only |      | For adjustment by the manufacturer |
| 12        | Function information 04 (by board type)          | 0000H                     | Reference only |      | For adjustment by the manufacturer |
| 13        | Function information 05 (by board type)          | 0000H                     | Reference only |      | For adjustment by the manufacturer |
| 14        | Function information 06 (by board type)          | 0000H                     | Reference only |      | For adjustment by the manufacturer |
| 15        | Function information 07 (by board type)          | 0000H                     | Reference only |      | For adjustment by the manufacturer |
| 16        | Function information 08 (by board type)          | 0000H                     | Reference only |      | For adjustment by the manufacturer |
| 17        | Function information 09 (by board type)          | 0000H                     | Reference only |      | For adjustment by the manufacturer |
| 18        | Function information 10 (by board type)          | 0000H                     | Reference only |      | For adjustment by the manufacturer |
| 19        | Function information 11 (by board type)          | 0000H                     | Reference only |      | For adjustment by the manufacturer |
| 20        | Function information 12 (by board type)          | 0000H                     | Reference only |      | For adjustment by the manufacturer |
| 21        | Function information 13 (by board type)          | 0000H                     | Reference only |      | For adjustment by the manufacturer |
| 22        | Function information 14 (by board type)          | 0000H                     | Reference only |      | For adjustment by the manufacturer |
| 23 to 112 | Card parameter (by board type)                   | 0000H                     | Reference only |      | For adjustment by the manufacturer |



## 7. Other Parameters

| No     | Parameter name   | Default value (Reference) | Input range | Unit | Remarks  |
|--------|--|---------------------------|-------------|------|--|
| 1      | Auto-start program number                                    | 0                         | 0 to 64     |      | (Invalid if "0" is set)  |
| 2      | I/O processing program number at operation/program abort     | 0                         | 0 to 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.)<br>(Invalid if "0" is set) * If the setting is valid, the number of user program tasks that can be used will decrease by 1. Other programs cannot be started from this program.  |
| 3      | I/O processing program number at all operation pause         | 0                         | 0 to 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 to 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, all programs other than the "I/O processing program at operation/program abort" will be cancelled.)<br>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 to 5      |      | 0: When all-operation-cancellation factor has generated (Only when a program is running)<br>1: When all-operation-cancellation factor has generated (Always)<br>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)<br>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 to 99999  | msec | * The setting will become effective after the controller, PC software or TP is shut down and restarted.  |
| 7 to 8 | (For expansion)  | 0                         |             |      |  |
| 9      | Deadman-switch recovery type                                 | 0                         | 0 to 2      |      | 0: Abort operations/programs<br>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.)  |

PC: PC software

TP: Teaching pendant



## Other Parameters

| No       | Parameter name  | Default value (Reference) | Input range    | Unit | Remarks   |
|----------|---|---------------------------|----------------|------|---|
| 10       | Emergency-stop recovery type                            | 0                         | 0 to 4         |      | <p>0: Abort operations/programs</p> <p>1: Recovery after reset</p> <p>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.)</p> <p>3: Abort operations/programs (Software reset when the emergency stop is reset. The home-return completion status of incremental-encoder axes will be reset (EG approximation swap).)</p> <p>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 home-return completion status of incremental-encoder axes will be retained.)</p> |
| 11       | For future expansion                                    | 0                         | Reference only |      |   |
| 12       | Automatic operation recognition type                    | 0                         | 0 to 3         |      | <p>0: Program is running AND all-operation-cancellation factor is not present</p> <p>1: [Program is running OR in AUTO mode] AND all-operation-cancellation factor is not present</p>   |
| 13 to 19 | (For expansion)   | 0                         |                |      |   |
| 20       | System-memory backup battery installation function type | 0                         | 0 to 2         |      | <p>0: Not installed (SEL global data/error lists cannot be recovered from the flash ROM)</p> <p>1: For future expansion (Setting prohibited)</p> <p>2: For future expansion (Setting prohibited)</p> <p>* If "0" is set, the SEL global data and error lists will not be retained after the power is turned off. However, the error lists will be retained after a software reset. (Main application version 0.10 or later)</p> <p>* After the power is turned on with the system-memory backup battery not installed, the point data can be copied from the flash memory.</p>  |

PC: PC software

TP: Teaching pendant



Other Parameters

| No       | Parameter name  | Default value (Reference) | Input range             | Unit | Remarks  |
|----------|---|---------------------------|-------------------------|------|--|
| 21       | Manual mode type  | 0                         | 0 to 5                  |      | 0: Always enable edit and SIO/PIO start (Initial condition after connection = With safety speed)<br>1: Select edit and start (with password) (EU, etc.)<br>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)<br>* Referenced by the PC/TP.  |
| 22       | Control use region  | 0                         | 0 to 99                 |      | 0: J, 1: E, 2: EU  |
| 23       | PSIZ command function type  | 0                         | 0 to 5                  |      | 0: Maximum number of point data areas<br>1: Number of point data used  |
| 24       | Local variable number for storing SEL communication command return code | 99                        | 1 to 99<br>1001 to 1099 |      |  |
| 25 to 29 | (For expansion)   | 0                         |                         |      |  |
| 30       | Option Password 00  | 0H                        | 0H to FFFFFFFFH         |      | HOME command option (Change prohibited)<br>* Change is prohibited unless instructed by the manufacturer.   |
| 31       | Option Password 01  | 0H                        | 0H to FFFFFFFFH         |      | Reserved (Change prohibited)<br>* Change is prohibited unless instructed by the manufacturer.  |
| 32       | Option Password 02  | 0H                        | 0H to FFFFFFFFH         |      | Reserved (Change prohibited)<br>* Change is prohibited unless instructed by the manufacturer.  |
| 33 to 35 | (For expansion)   | 0H                        | 0H to FFFFFFFFH         |      |  |
| 36       | PC/TP data protect setting (Program)                                    | 0H                        | 0H to FFFFFFFFH         |      | Bits 0 to 3: Protect type (0: Read/write, 1: Read only, 2: No read/write)<br>Bits 4 to 7: Protect release method (0: Special operation)<br>Bits 8 to 11: Protect range maximum number (1's place, BCD)<br>Bits 12 to 15: Protect range maximum number (10's place, BCD)<br>Bits 16 to 19: Protect range minimum number (1's place, BCD)<br>Bits 20 to 23: Protect range minimum number (10's place, BCD)<br>* 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 to FFFFFFFFH         |      | Bits 0 to 3: Protect type (0: Read/write, 1: Read only, 2: No read/write)<br>Bits 4 to 7: Protect release method (0: Special operation)<br>Bits 8 to 11: Protect range maximum number (10's place, BCD)<br>Bits 12 to 15: Protect range maximum number (100's place, BCD)<br>Bits 16 to 19: Protect range maximum number (1000's place, BCD)<br>Bits 20 to 23: Protect range minimum number (10's place, BCD)<br>Bits 24 to 27: Protect range minimum number (1000's place, BCD)<br>* The value in the 1's place is considered "0" for both the protect range maximum/minimum numbers.<br>* Referenced by the PC/TP (PC software version 2.0.0.42 or later and TP application version 1.09 or later) |

PC: PC software

TP: Teaching pendant



Other Parameters

| No. | Parameter name                                 | Default value (Reference) | Input range     | Unit | Remarks  |
|-----|--|---------------------------|-----------------|------|--|
| 38  | PC/TP data protect setting (Symbol, parameter) | 0H                        | 0H to FFFFFFFFH |      | Bits 0 to 3: Protect type (Parameter) (0: Read/write, 1: Read only, 2: No read/write)<br>Bits 4 to 7: Protect release method (Parameter) (0: Special operation)<br>Bits 8 to 11: Protect type (Symbol) (0: Read/write, 1: Read only, 2: No read/write)<br>Bits 12 to 15: Protect release method (Symbol) (0: Special operation)<br>* Referenced by the PC/TP (PC software version 2.0.0.42 or later and TP application version 1.09 or later)  |
| 39  | (For future expansion)                         | 0H                        | 0H to FFFFFFFFH |      |  |
| 40  | EEPROM information check type                  | 3H                        | Reference only  |      | For adjustment by the manufacture  |
| 41  | Hardware information check type                | E0H                       | Reference only  |      | For adjustment by the manufacture  |
| 42  | Hardware test type                             | 7H                        | Reference only  |      | For adjustment by the manufacture  |
| 43  | Special monitor type                           | 0H                        | 0H to FFFFFFFFH |      | Change strictly prohibited unless specified by the manufacturer.   |
| 44  | (For expansion)                                | 0                         |                 |      |  |
| 45  | Special start condition setting                | 0                         | 0H to FFFFFFFFH |      | Bits 0 to 3: Enable start from PC/TP in AUTO mode = Used exclusively by the manufacturer (0: Do not enable, 1: Enable)<br><br>Bits 4 to 7: PIO program start (Input port 000)<br>Single start selection (0: Normal, 1: Single start)<br>* When single start is selected, the next PIO program start (input port 000) will not be accepted as long as a program with the same program number as the one started by the last PIO program start (input port 000) is running.<br><br>Bits 8 to 11: Permission of auto program start when all-operation-cancellation factor is present<br>(0: Do not permit, 1: Permit)<br><br>Bits 12 to 15: Permission of ON edge acceptance for PIO program start (input port 000) when all-operation-cancellation factor is present<br>(0: Do not permit, 1: Permit)<br>* 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. |

PC: PC software  
 TP: Teaching pendant



## Other Parameters

| No.      | Parameter name              | Default value (Reference) | Input range     | Unit | Remarks   |
|----------|-----------------------------|---------------------------|-----------------|------|---|
| 46       | Other setting bit pattern 1 | 2001H                     | 0H to FFFFFFFFH |      | <p>Bits 0 to 3:<br/>Variable-value format type in response message to real-number/variable query<br/>(0: Big endian with four upper/lower binary-converted bytes reversed, 1: Big endian)</p> <p>Bits 4 to 7:<br/>Decimal-place rounding selection for real-number → integer-variable assignment in LET/TRAN commands (Main application version 0.53 or later) (0: Do not round, 1: Round)</p> <p>Bits 8 to 11:<br/>For future expansion<br/>* May be affected by hardware compatibility.<br/>* Change strictly prohibited unless specified by the manufacturer.</p> <p>Bits 12 to 15:<br/>Selection of processing to be performed when subroutine first step input condition is not specified when TPCD command = 1<br/>(0: Do not execute, 1: Execute, 2: Error)</p> <p>Bits 16 to 19:<br/>Selection of effective period for CHVL command speed<br/>(0: Effective only while the current main packet is active, 1: Effective during continuous handling of packets)<br/>* If "1" is selected, the speed specified by the CHVL command will be retained during the subsequent operations executed by continuous movement commands such as PATH (commands that require input of continuous program steps).<br/>Also take note of the following limitations:</p> <ul style="list-style-type: none"> <li>• If the speed specified by the CHVL command is greater than the actual speed of the command operation immediately before, the actual speed of the last command operation will apply.</li> <li>• If the CHVL command is executed during a series of continuous movement commands and the command execution timing coincides with the connection of a position movement packet, the current speed may change to the specified speed in two stages.</li> <li>• The connection speed of each position movement packet may increase in proportion to the deceleration.<br/>(Example: The packet connection speed may increase by 9.8 mm/sec at a deceleration of 1.0 G or by 4.9 mm/sec at a deceleration of 0.5 G.)</li> </ul> <p>For other items that require attention, refer to Notes in the CHVL command section of the operation manual.</p> |
| 47 to 50 | (For expansion)             | 0                         |                 |      |   |



## 8. Manual Operation Types

The selectable operation types will vary depending on the setting of the “Manual operation type” parameter (Other parameter No. 21).

### (1) PC software

#### [1] Setting = 0 (Always enable edit and SIO/PIO start)

| Operation type       | Password      | Functions |              |                            |                   |                   |
|----------------------|---------------|-----------|--------------|----------------------------|-------------------|-------------------|
|                      |               | Edit      | Safety speed | Jog, move, continuous move | SIO program start | PIO program start |
| With safety speed    | Not required. | ○         | ○            | ○                          | ○                 | ○                 |
| Without safety speed | Not required. | ○         |              | ○                          | ○                 | ○                 |

#### [2] Setting = 1 (Select edit and start (with password))

| Operation type                   | Password      | Functions |              |                            |                   |                   |
|----------------------------------|---------------|-----------|--------------|----------------------------|-------------------|-------------------|
|                                  |               | Edit      | Safety speed | Jog, move, continuous move | SIO program start | PIO program start |
| Edit and jog                     | Not required. | ○         | ○            | ○                          |                   |                   |
| SIO start and jog (safety speed) | 1817 (*1)     |           | ○            | ○                          | ○                 |                   |
| SIO start and jog                | 1818 (*1)     |           |              | ○                          | ○                 |                   |
| SIO/PIO start and jog            | 1819 (*1)     |           |              | ○                          | ○                 | ○                 |

(\*1) PC software version 0.0.6.0 or later (“0000” in versions 0.0.0.0 through 0.0.5.x)

### (2) Teaching pendant

#### [1] Setting = 0 (Always enable edit and SIO/PIO start)

| Safety-speed enable selection | Password      | Functions |              |                            |                   |                   |
|-------------------------------|---------------|-----------|--------------|----------------------------|-------------------|-------------------|
|                               |               | Edit      | Safety speed | Jog, move, continuous move | SIO program start | PIO program start |
| Enable                        | Not required. | ○         | ○            | ○                          | ○                 | ○                 |
| Disable                       | Not required. | ○         |              | ○                          | ○                 | ○                 |

#### [2] Setting = 1 (Select edit and start (with password))

| Safety-speed enable selection | Password      | Functions |              |                            |                   |                   |
|-------------------------------|---------------|-----------|--------------|----------------------------|-------------------|-------------------|
|                               |               | Edit      | Safety speed | Jog, move, continuous move | SIO program start | PIO program start |
| Enable                        | Not required. | ○         | ○            | ○                          | ○                 | (*3)              |
| Disable                       | 1818 (*1)     | ○         |              | ○                          | ○                 | (*3)              |

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

(\*1) Teaching pendant application version 0.02 or later (not supported by version 0.01 or earlier)

(\*2) PIO program start is enabled only in modes other than the edit mode.

(\*3) In accordance with the “PIO start prohibition selection” setting.

(\*4) In accordance with the “Safety-speed enable” setting.

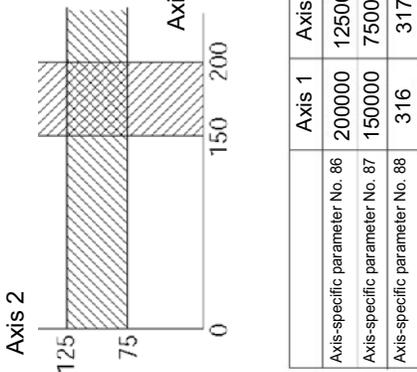


### 9. Use Examples of Key Parameters

You can assign functions in addition to those available under the factory settings or change the factory settings or change the factory-set functions, 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   |
|--|--|---|--|
| <p>Want to prevent errors relating to the standard I/O board and field network board (DeviceNet, CC-Link, etc.). (Want to perform trial operation when wiring is not yet done, etc.)</p>   | <p>I/O error monitor can be disabled to prevent errors from occurring.</p>   | <p>Set "0" in the I/O parameter corresponding to the I/O board whose error monitor you want to disable.<br/>Standard I/O1: I/O parameter No. 10 = 0<br/>Expanded I/O2: I/O parameter No. 11 = 0<br/>Field network: I/O parameter No. 12 = 0</p>   | <p>Set "0" in I/O parameter Nos. 10 and 11 to disable error monitor for the standard I/O board.<br/>Note: To operate a disabled I/O board, be sure to revert the parameter setting to "1."</p>   |
| <p>Want to retain output status while emergency-stop signal is input or the safety gate is open.</p>   | <p>Minimum and maximum port numbers indicating the output ports you wish to retain can be set.</p>   | <p>I/O parameter No. 70 = Output port number MIN<br/>I/O parameter No. 71 = Output port number MAX<br/><br/>Setting example) To retain output ports from port Nos. 316 through 331, set as follows:<br/>I/O parameter No. 70 = 316<br/>I/O parameter No. 71 = 331</p>   | <p>← The status of output port Nos. 316 through 331 will be retained while emergency-stop signal is input or the safety gate is open.<br/><br/><div style="border: 1px solid black; padding: 5px;">Note: The parameters are normally set to the above output port numbers before shipment.</div></p>                       |
| <p>Want to start programs while emergency-stop signal is input or the safety gate is open.<br/>Programs to be started are I/O processing or calculation programs that do not command actuator operation (PIO processing programs).</p> | <p>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.</p> | <p>Other parameter No. 2 = PIO processing program number<br/>I/O parameter No. 70 = Output port number MIN<br/>I/O parameter No. 71 = Output port number MAX<br/><br/>Setting example) To start program No. 5 that involves processing at output port Nos. 316 through 331, set as follows:<br/>Other parameter No. 2 = 5<br/>I/O parameter No. 70 = 316<br/>I/O parameter No. 71 = 331</p> | <p>← Program No. 5 will start while emergency-stop signal is input or the safety gate is open. Output port Nos. 316 through 331 can be used for processing.<br/><br/><div style="border: 1px solid black; padding: 5px;">Note: The parameters are normally set to the above output port numbers before shipment.</div></p> |



| Description   | Action   | Parameter setting   | Manipulation/operation  |        |        |                                |        |          |                                |        |         |                                |     |     |  |
|---|--|---|---|--------|--------|--------------------------------|--------|----------|--------------------------------|--------|---------|--------------------------------|-----|-----|--|
| 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<br>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<br>I/O parameter No. 33 = 1<br>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 output signal when the actuator enters a specified area (zone).   | <p>A desired actuator zone can be set for each axis. A desired output port to turn ON when the axis enters the zone can be set for each axis. A maximum of four zones can be set (zones 1 to 4).</p> <p>Max. value of zone 1:<br/>Axis-specific parameter No. 86</p> <p>Min. value of zone 1:<br/>Axis-specific parameter No. 87</p> <p>Zone 1 output port number:<br/>Axis-specific parameter No. 88</p> <p>Max. value of zone 2:<br/>Axis-specific parameter No. 89</p> <p>Min. value of zone 2:<br/>Axis-specific parameter No. 90</p> <p>Zone 2 output port number:<br/>Axis-specific parameter No. 91</p> <p>Max. value of zone 3:<br/>Axis-specific parameter No. 92</p> <p>Min. value of zone 3:<br/>Axis-specific parameter No. 93</p> <p>Zone 3 output port number:<br/>Axis-specific parameter No. 94</p> <p>Max. value of zone 4:<br/>Axis-specific parameter No. 95</p> <p>Min. value of zone 4:<br/>Axis-specific parameter No. 96</p> <p>Zone 4 output port number:<br/>Axis-specific parameter No. 97</p> | <p>Setting example) Set the area illustrated below as zone 1:</p> <p>Axis 1: Output port No. 316 will turn ON when the axis enters the area between 150 and 200 mm.</p> <p>Axis 2: Output port No. 317 will turn ON when the axis enters the area between 75 and 125 mm.</p>  <table border="1" data-bbox="1047 695 1177 1102"> <thead> <tr> <th></th> <th>Axis 1</th> <th>Axis 2</th> </tr> </thead> <tbody> <tr> <td>Axis-specific parameter No. 86</td> <td>200000</td> <td>125000 *</td> </tr> <tr> <td>Axis-specific parameter No. 87</td> <td>150000</td> <td>75000 *</td> </tr> <tr> <td>Axis-specific parameter No. 88</td> <td>316</td> <td>317</td> </tr> </tbody> </table> <p>*: Max. and min. values are input in units of 0.001 mm.</p> |   | Axis 1 | Axis 2 | Axis-specific parameter No. 86 | 200000 | 125000 * | Axis-specific parameter No. 87 | 150000 | 75000 * | Axis-specific parameter No. 88 | 316 | 317 | <p>For the output signal to be processed, the axes must stay for at least 3 msec in the zone. Duplicate output port numbers cannot be specified.</p> <p>← : Output port No. 316 turns ON.<br/>← : Output port No. 317 turns ON.</p> <p>Note: Before shipment, the parameters are normally set so that the above output port numbers can be used.</p> |
|   | Axis 1   | Axis 2  |   |        |        |                                |        |          |                                |        |         |                                |     |     |  |
| Axis-specific parameter No. 86  | 200000   | 125000 *  |   |        |        |                                |        |          |                                |        |         |                                |     |     |  |
| Axis-specific parameter No. 87  | 150000   | 75000 *   |   |        |        |                                |        |          |                                |        |         |                                |     |     |  |
| Axis-specific parameter No. 88  | 316  | 317   |   |        |        |                                |        |          |                                |        |         |                                |     |     |  |

Before changing a parameter, be sure to read the corresponding section in the List of Parameters.



◎ Error Level Control

| Error level                  | System error assignment source | Error No. (HEX) | Display (7-segment display, etc.) | Error list (Application only) | Error LED output (MAIN only)   | Program run (Application only) |                           | Error reset (Application only) | Remarks   |
|------------------------------|--------------------------------|-----------------|-----------------------------------|-------------------------------|--|--------------------------------|---------------------------|--------------------------------|---|
|                              |                                |                 |                                   |                               |  | Other parameter No. 4 = 0      | Other parameter No. 4 = 1 |                                |   |
| Secret level                 | MAIN application               | 800 to 88F      |                                   | ○                             |  |                                |                           |                                | Special error level provided for maintenance purposes   |
|                              | MAIN core                      | 890 to 8AF      |                                   |                               |  |                                |                           |                                |   |
|                              | PC                             | 8B0 to 8DF      |                                   |                               |  |                                |                           |                                |   |
|                              | TP                             | 8E0 to 8FF      |                                   |                               |  |                                |                           |                                |   |
| Message level                | MAIN application               |                 |                                   |                               |  |                                |                           |                                | Status display, input error, etc.   |
|                              | MAIN core                      |                 |                                   |                               |  |                                |                           |                                |   |
|                              | PC                             |                 |                                   |                               |  |                                |                           |                                |   |
|                              | PC (Update tool)               |                 |                                   |                               |  |                                |                           |                                |   |
|                              | TP                             |                 |                                   |                               |  |                                |                           |                                |   |
|                              | MAIN application               | 200 to 24F      |                                   |                               |  |                                |                           |                                |   |
|                              | MAIN core                      |                 |                                   |                               |  |                                |                           |                                |   |
|                              | PC                             | 250 to 29F      |                                   |                               | △ (Battery and fieldbus errors will be registered in an error list.) |                                |                           |                                |   |
|                              | PC (Update tool)               | 2A0 to 2CF      |                                   |                               |  |                                |                           |                                |   |
|                              | TP                             | 2D0 to 2FF      |                                   |                               |  |                                |                           |                                |   |
|                              | MAIN application               | 900 to 93F      |                                   | ○                             |  |                                |                           |                                |   |
|                              | MAIN core                      | 940 to 97F      |                                   |                               |  |                                |                           |                                |   |
|                              | PC                             | 980 to 9AF      |                                   |                               |  |                                |                           |                                |   |
|                              | PC (Update tool)               | 9B0 to 9BF      |                                   |                               |  |                                |                           |                                |   |
|                              | TP                             | 9C0 to 9FF      |                                   |                               |  |                                |                           |                                |   |
|                              | MAIN application               | A00 to A6F      |                                   |                               |  |                                |                           |                                |   |
| MAIN core                    | A70 to A9F                     |                 |                                   |                               |  |                                |                           |                                |   |
| PC                           | AA0 to ACF                     |                 |                                   |                               |  |                                |                           |                                |   |
| TP                           | AD0 to AFF                     |                 |                                   |                               |  |                                |                           |                                |   |
| Operation-cancellation level | MAIN application               |                 |                                   |                               |  |                                |                           |                                | Errors affecting operation. The system will attempt to reset minor errors below this level using an auto-reset function via external active command (SIO/PIO) (application only). |
|                              | MAIN core                      |                 |                                   |                               |  |                                |                           |                                |   |
|                              | PC                             |                 |                                   |                               |  |                                |                           |                                |   |
|                              | PC (Update tool)               |                 |                                   |                               |  |                                |                           |                                |   |
|                              | TP                             |                 |                                   |                               |  |                                |                           |                                |   |
|                              | MAIN application               | 400 to 4CF      |                                   |                               |  |                                |                           |                                |   |
|                              | MAIN core                      |                 |                                   |                               |  |                                |                           |                                |   |
|                              | PC                             | 4D0 to 4DF      |                                   |                               |  |                                |                           |                                |   |
|                              | PC (Update tool)               | 4E0 to 4EF      |                                   |                               |  |                                |                           |                                |   |
|                              | TP                             | 4F0 to 4FF      |                                   |                               |  |                                |                           |                                |   |



| Error level                  | System error assignment source | Error No. (HEX) | Display (7-segment display, etc.) | Error list (Application only) | Error LED output (MAIN only) | Program run (Application only)   |  | Error reset (Application only) | Remarks   |
|------------------------------|--------------------------------|-----------------|-----------------------------------|-------------------------------|------------------------------|--|--|--------------------------------|---|
|                              |                                |                 |                                   |                               |                              | Other parameter No. 4 = 0  | Other parameter No. 4 = 1  |                                |   |
| Operation-cancellation level | MAIN application               | B00 to B9F      |                                   |                               |                              | The program in which the error generated will be cancelled.<br>(Except for axis errors, a cancellation factor is present only for the moment the error occurs.)<br>* However, in the case of an error requiring servo OFF or all-axis servo OFF, all programs other than the "/I/O processing program at operation/program abort" will be cancelled. | All programs other than the "/I/O processing program at operation/program abort" will be cancelled. (Except for axis errors, a cancellation factor is present only for the moment the error occurs.) | Enabled.                       | Errors affecting operation. The system will attempt to reset minor errors below this level using an auto-reset function via external active command (SIO/PIO) (application only). |
|                              | MAIN core                      | BA0 to BBF      |                                   |                               |                              |  |  |                                |   |
|                              | PC                             | BC0 to BDF      |                                   |                               |                              |  |  |                                |   |
|                              | TP                             | BE0 to BFF      |                                   |                               |                              |  |  |                                |   |
|                              | MAIN application               | C00 to CCF      |                                   |                               |                              |  |  |                                |   |
|                              | MAIN core                      | CD0 to CDF      |                                   |                               |                              |  |  |                                |   |
|                              | PC                             | CE0 to CFF      |                                   |                               |                              |  |  |                                |   |
|                              | TP                             | CF0 to CFF      |                                   |                               |                              |  |  |                                |   |
|                              | MAIN application               |                 |                                   |                               |                              |  |  |                                |   |
|                              | MAIN core                      |                 |                                   |                               |                              |  |  |                                |   |
| Cold-start level             | PC (Update tool)               |                 |                                   |                               |                              |  |  |                                |   |
|                              | TP                             |                 |                                   |                               |                              |  |  |                                |   |
|                              | MAIN application               | 600 to 6CF      |                                   |                               |                              | The program in which the error generated will be cancelled.<br>* However, in the case of an error requiring drive-source cutoff, servo OFF or all-axis servo OFF (initialization error, power error, etc.), all programs other than the "/I/O processing program at operation/program abort" will be cancelled.                                      | All programs other than the "/I/O processing program at operation/program abort" will be cancelled.  | Not enabled.                   | The controller power must be reconnected (MAIN only). (The CPU and OS will run properly.)   |
|                              | MAIN core                      | ---             |                                   |                               |                              |  |  |                                |   |
|                              | PC                             | 6D0 to 6DF      |                                   |                               |                              |  |  |                                |   |
|                              | PC (Update tool)               | 6E0 to 6EF      |                                   |                               |                              |  |  |                                |   |
|                              | TP                             | 6F0 to 6FF      |                                   |                               |                              |  |  |                                |   |
|                              | MAIN application               | D00 to D8F      |                                   |                               |                              |  |  |                                |   |
|                              | MAIN core                      | D90 to DAF      |                                   |                               |                              |  |  |                                |   |
|                              | PC                             | DB0 to DCF      |                                   |                               |                              |  |  |                                |   |
| System-down level            | PC (Update tool)               | DD0 to DDF      |                                   |                               |                              |  |  |                                |   |
|                              | TP                             | DE0 to DFF      |                                   |                               |                              |  |  |                                |   |
|                              | MAIN application               | E00 to E8F      |                                   |                               |                              |  |  |                                |   |
|                              | MAIN core                      | E90 to EBF      |                                   |                               |                              |  |  |                                |   |
|                              | PC                             | EC0 to EDF      |                                   |                               |                              |  |  |                                |   |
|                              | TP                             | EE0 to EFF      |                                   |                               |                              |  |  |                                |   |
|                              | MAIN application               |                 |                                   |                               |                              |  |  |                                |   |
|                              | MAIN core                      |                 |                                   |                               |                              |  |  |                                |   |
|                              | PC                             |                 |                                   |                               |                              |  |  |                                |   |
|                              | PC (Update tool)               |                 |                                   |                               |                              |  |  |                                |   |
| TP                           |                                |                 |                                   |                               |                              |  |  |                                |   |
| MAIN application             | FF0 to FBF                     |                 |                                   |                               |                              |  |  |                                |   |
| MAIN core                    | FC0 to FCF                     |                 |                                   |                               |                              |  |  |                                |   |
| PC                           | FD0 to FDF                     |                 |                                   |                               |                              |  |  |                                |   |
| TP                           | FE0 to 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.   |
|-----------|--|---|
| 207       | Update file name error (IAI protocol)                              | The name of the update program file selected in the update mode is invalid. Select the correct file and repeat the update procedure from the beginning.   |
| 20E       | Motorola S byte count error  | The update program file is invalid. Check the file.   |
| 20F       | Update target specification error (Received by the application)    | The application's system received an update target specification command. To update the program, restart the controller and repeat the update procedure from the beginning.   |
| 630       | Update system code error (Detected by the application)             | The update system code is invalid.  |
| 631       | Update unit code error (Detected by the application)               | The update unit code is invalid.  |
| 632       | Update device number error (Detected by the application)           | The update device number is invalid.  |
| 685       | I/O function selection physical port number error                  | The I/O port number setting specified for a given I/O function selection is invalid. Check the settings of I/O parameter Nos. 62 to 65, 76, 77, 283 to 330, etc.  |
| 801       | SCIF overrun status (IAI protocol reception)                       | Communication failure. Check for noise, connected equipment and communication setting.  |
| 802       | SCIF receive ER status (IAI protocol reception)                    | Communication failure. Check for noise, shorted/disconnected communication cable, connected equipment and communication setting. This error will also occur when establishing communication with the PC/TP wrongly connected to SIO-CH1 being opened to the user. |
| 803       | Receive timeout status (IAI protocol reception)                    | The transfer interval after the first received byte is too long. Possible causes include disconnected communication cable and error in the connected equipment.   |
| 804       | SCIF overrun status (SEL reception)                                | Communication failure. Check for noise, connected equipment and communication setting.  |
| 805       | SCIF receive ER status (SEL reception)                             | Communication failure. Check for noise, shorted/disconnected communication cable, connected equipment and communication setting.  |
| 806       | SCIF receive ER status due to other factor (SEL reception)         | Communication failure. Take the same action specified for error No. 804 or 805.   |
| 807       | Drive-source cutoff relay ER status                                | The motor-drive power ON status remains ON even when the drive source is cut off. The drive-source cut-off relay contacts may have been melted.   |
| 808       | Power OFF status during slave parameter write                      | The power was turned off while writing slave parameters. (This error can be detected only when a backup battery is used.)   |
| 809       | Power OFF status during data write to flash ROM                    | The power was turned off while writing data to the flash ROM. (This error can be detected only when a backup battery is used.)  |
| 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. Check for noise, shorted/disconnected communication cable, connected equipment and communication setting.  |
| 80E       | Expanded-SIO receive buffer overflow status (SEL reception)        | Communication failure. Take the same action specified for error No. 80A, 80B or 80C.  |
| 80F       | Ethernet control status 1  | The receive buffer overflowed. Excessive data was received from outside.  |
| 810       | Ethernet control status 2  | Ethernet control information (for analysis)   |
| 811       | Maintenance information 1  | Ethernet control information (for analysis)   |
| 812       | Maintenance information 2  | Maintenance information (for analysis)  |



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

| Error No. | Error name   | Description, action, etc.  |
|-----------|--|--|
| 813       | Maintenance information 3  | Maintenance information (for analysis)   |
| 814       | Maintenance information 4  | Maintenance information (for analysis)   |
| 815       | Maintenance information 5  | Maintenance information (for analysis)   |
| 900       | Blank step shortage error  | There are not enough blank steps to save step data. Provide enough blank steps needed to save step data.   |
| 901       | Step number error  | The step number is invalid.  |
| 902       | Symbol-definition table number error                               | The symbol-definition table number is invalid.   |
| 903       | Point number error   | The point number is invalid.   |
| 904       | Variable number error  | The variable number is invalid.  |
| 905       | Flag number error  | The flag number is invalid.  |
| 906       | I/O port/flag number error   | The I/O port/flag number is invalid.   |
| 910       | Command error (IAI protocol HT reception)                          | The command ID is not supported or invalid. (For future expansion)   |
| 911       | Message conversion error (IAI protocol HT reception)               | The transmitted message does not match the message format or contains invalid data. (For future expansion)   |
| 912       | PC/TP servo-movement command acceptance-enable input OFF error     | Any axis movement command issued to the axis specified in I/O parameter No. 78 from the PC/TP will not be accepted while the input port specified in I/O parameter No. 77 is OFF. (Important: The acceptance-enable input port will become invalid once the operation is started.) |
| A01       | System-memory backup battery voltage-low warning                   | The voltage of the system-memory backup battery is low. Replace the battery. (Above the minimum data-backup voltage)   |
| A02       | Abnormal system-memory backup battery voltage                      | The voltage of the system-memory backup battery is low. Replace the battery. (Below the minimum data-backup voltage)   |
| A03       | Absolute-data backup battery voltage-low warning (Driver analysis) | The voltage of the absolute-data backup battery is low. Check the battery connection or replace the battery.   |
| A04       | System mode error at core update                                   | An update command was received when the system was not in the core update mode. Before updating the core, confirm that a chip resistance for setting core update mode is provided on the board. (For maintenance)  |
| A05       | Motorola S record format error                                     | The update program file is invalid. Check the file.  |
| A06       | Motorola S checksum error  | The update program file is invalid. Check the file.  |
| A07       | Motorola S load address error                                      | The update program file is invalid. Check the file.  |
| A08       | Motorola S write address over error                                | The update program file is invalid. Check the file.  |
| A09       | Flash-ROM timing limit over error (Write)                          | Error writing the flash ROM  |
| A0A       | Flash-ROM timing limit over error (Erase)                          | Error erasing the flash ROM  |
| A0B       | Flash-ROM verify error   | Error erasing/writing the flash ROM  |



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

| Error No. | Error name   | Description, action, etc.  |
|-----------|--|--|
| A0C       | Flash-ROM ACK timeout  | Error erasing/writing the flash ROM  |
| A0D       | Head sector number specification error                           | Error erasing the flash ROM  |
| A0E       | Sector count specification error                                 | Error erasing the flash ROM  |
| A0F       | Write-destination offset address error (Odd-numbered address)    | Error writing the flash ROM  |
| A10       | Write-source data buffer address error (Odd-numbered address)    | Error writing the flash ROM  |
| A11       | Invalid core-code sector block ID error                          | The core program already written to the flash ROM is invalid.  |
| A12       | Core-code sector block ID erase count over                       | The number of times the flash ROM can be erased was exceeded.  |
| A13       | Flash-ROM write request error when erase is incomplete           | When updating, a flash-ROM write command was received before a flash-ROM erase command. Check the update program file and perform update again.  |
| A14       | Busy-status reset timeout error at EEPROM write                  | A busy-status reset timeout occurred after executing EEPROM write.   |
| A15       | EEPROM write request error due to no-EEPROM in target            | An EEPROM write request was received for a driver or other unit with CPU not equipped with EEPROM.   |
| A16       | EEPROM read request error due to no-EEPROM in target             | An EEPROM read request was received for a driver or other unit with CPU not equipped with EEPROM.  |
| A17       | Message checksum error (IAI protocol reception)                  | The checksum in the received message is invalid.   |
| A18       | Message header error (IAI protocol reception)                    | The header in the received message is invalid. Invalid header position (message is 9 bytes or less) is suspected, among other reasons.   |
| A19       | Message station number error (IAI protocol reception)            | The station number in the received message is invalid.   |
| A1A       | Message ID error (IAI protocol reception)                        | The ID in the received message is invalid.   |
| A1C       | Message conversion error   | The transmitted message does not match the message format or contains invalid data. Check the transmitted message.   |
| A1D       | Start mode error   | A start not permitted in the current mode (MANU/AUTO) was attempted.   |
| A1E       | Start condition non-satisfaction error                           | Start was attempted when the start condition was not satisfied, such as when an all-operation-cancellation factor (see the 7-segment display: Drive-source cutoff, mode switching, error, auto-start switch OFF edge, deadman switch, safety gate, emergency stop, etc.) was present or the flash ROM was being written. |
| A1F       | Axis duplication error (SIO · PIO)                               | The applicable axis is currently in use.   |
| A20       | Servo-control-right acquisition error (SIO · PIO)                | The servo control right is not available.  |
| A21       | Servo-control-right duplicate-acquisition error (SIO · PIO)      | The servo control right has already been acquired.   |
| A22       | Servo-control-right non-acquisition error (SIO · PIO)            | An attempt to retain the servo control right has failed.   |
| 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.  |



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

| Error No. | Error name   | Description, action, etc.   |
|-----------|--|---|
| A26       | Program count specification error                            | The specified number of programs is invalid.  |
| A27       | Program non-registration error                               | The applicable program is not registered.   |
| A28       | Reorganization disable error during program run              | A program-area reorganization operation was attempted while a program was running. End all active programs first. |
| A29       | Active-program edit disable error                            | An edit operation was attempted to a program currently not running. End the applicable program first.             |
| A2A       | Program inactive error                                       | The specified program is not running.   |
| A2B       | Program-run command refusal error in AUTO mode               | Programs cannot be run from the TP/PC software connector in the AUTO mode.  |
| A2C       | Program number error   | The program number is invalid.  |
| A2D       | Inactive program resumption error                            | A resumption request was received for a program currently not running.  |
| A2E       | Inactive program pause error                                 | A pause request was received for a program currently not running.   |
| A2F       | Breakpoint error   | The step number specified as a breakpoint is invalid.   |
| A30       | Breakpoint setting-count specification error                 | The number of breakpoints to be set exceeds the limit value.  |
| A31       | Parameter change value error                                 | The value of parameter changed is invalid.  |
| A32       | Parameter type error   | The parameter type is invalid.  |
| A33       | Parameter number error                                       | The parameter number is invalid.  |
| A34       | Card-parameter buffer read error                             | Error reading the card-parameter buffer   |
| A35       | Card-parameter buffer write error                            | Error writing the card-parameter buffer   |
| A36       | Parameter change refusal error during operation              | Parameters cannot be changed during operation (program is running, servo is in use, etc.).                        |
| A37       | Card manufacturing/function information change refusal error | The card manufacturing/function information cannot be changed.  |
| A38       | Parameter change refusal error during servo ON               | An attempt was made to change a parameter whose change is not permitted while the servo is ON.                    |
| A39       | Non-acquired card parameter change error                     | An attempt was made to change a parameter for a card not recognized at reset.                                     |
| A3A       | Device number error  | The device number is invalid.   |
| A3C       | Memory initialization type specification error               | The specified memory initialization type is invalid.  |
| A3D       | Unit type error  | The unit type is invalid.   |
| A3E       | SEL write data type specification error                      | The specified SEL write data type is invalid.   |
| A3F       | Flash-ROM write refusal error during program run             | The flash ROM cannot be written while a program is running.   |
| 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.  |



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

| Error No. | Error name   | Description, action, etc.   |
|-----------|--|---|
| A44       | Point-data count specification error                       | The specified number of point data is invalid.  |
| A45       | Symbol-record count specification error                    | The specified number of symbol records is invalid.  |
| A46       | Variable-data count specification error                    | The specified number of variable data is invalid.   |
| A48       | Error-detail query type 1 error                            | Error-detail query type 1 is invalid.   |
| A49       | Error-detail query type 2 error                            | Error-detail query type 2 is invalid.   |
| A4A       | Monitoring data type error                                 | The data type for monitoring data query is invalid.   |
| A4B       | Monitoring-record count specification error                | The specified number of records for monitoring data query is invalid.   |
| A4C       | Monitoring-operation special command register busy error   | The driver special command ACK generated a timeout during monitoring operation.   |
| A4E       | Parameter register busy error at issuance of slave command | The driver special command ACK generated a timeout at issuance of a slave command.  |
| A4F       | Software reset refusal error during operation              | Software reset (SIO) is prohibited during operation (program is running, servo is in use, etc.).  |
| A50       | Drive-source recovery request refusal error                | The drive-source cutoff factor (error, deadman switch, safety gate, emergency stop, etc.) has not been removed.   |
| A51       | Operation-pause reset request refusal error                | The all-operation-pause factor (drive-source cutoff, operation-pause signal, deadman switch, safety gate, emergency stop, etc.) has not been removed.                                   |
| A53       | Refusal error due to servo ON                              | A processing not permitted during servo ON was attempted.   |
| A54       | Refusal error due to unsupported function                  | The function is not supported.  |
| A55       | Refusal error due to exclusive manufacturer function       | A processing not opened to users other than the manufacturer was attempted.   |
| A56       | Refusal error due to invalid data                          | The data is invalid.  |
| A57       | Program start duplication error                            | An attempt was made to start a program currently running.   |
| A58       | BCD error warning  | The BCD value being read may be invalid, or the value being written (variable 99) may be a negative value, among other reasons.   |
| A59       | IN/OUT command port flag error warning                     | The number of I/O ports (flags) may have exceeded 32, among other reasons. Check the I/O port (flag) specifications.  |
| A5B       | Character-string → value conversion error warning          | The specified number of converting characters is invalid or characters that cannot be converted to value are included.  |
| A5C       | Copying-character count error warning with SCPY command    | The specified number of copying characters is invalid.  |
| A5D       | SCIF open error in non-AUTO mode                           | The channel was opened in a non-AUTO mode. In the MANU mode, the PC/TP connection must be forcibly disconnected before opening the serial channel opened to the user. Exercise caution. |
| 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.  |



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

| Error No. | Error name                                     | Description, action, etc.   |
|-----------|--|---|
| 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 monitor type supported by the standard DIO board or based on the capacity of FROM on the main CPU board does not match the monitor type on the PC software side (selected on the monitor screen). |



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

| Error No. | Error name   | Description, action, etc.  |
|-----------|--|--|
| B00       | SCHA setting error   | The setting of SCHA command is invalid.  |
| B01       | TPCD setting error   | The setting of TPCD command is invalid.  |
| B02       | SLEN setting error   | The setting of SLEN command is invalid.  |
| B03       | Home-return method error                                   | The setting of "Axis-specific parameter No. 10, Home-return method" is invalid. (Not incremental encoder AND current position 0 home is specified, etc.)                         |
| B04       | 1-shot-pulse output excessive simultaneous use error       | The number of BTPN and BTPF timers operating in one program simultaneously exceeds the upper limit (16).   |
| B05       | Estimate-stroke over error at home return                  | The operation at home return exceeded the estimate stroke. The home sensor or creep sensor may be faulty, among other reasons.   |
| 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       | Home-sensor pull-out timeout error                         | Pull-out from the home sensor cannot be confirmed. Check for operation restriction, wiring, motor, home sensor, etc.   |
| B12       | Storage variable number error for SEL command return code  | The variable number specified for storing SEL command's return code is invalid. 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 non-closed socket open error                      | An attempt was made to open a socket without closing it first.   |
| B1C       | Ethernet in-use-by-other-task error                        | An attempt was made to open a channel already opened by other task.  |
| B1D       | Ethernet non-open error                                    | An attempt was made to use a channel not opened by own task.   |
| B1E       | Ethernet multiple WRIT execution error                     | WRIT commands were executed simultaneously by multiple tasks for the same channel.   |



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

| Error No. | Error name                                   | Description, action, etc.  |
|-----------|--|--|
| 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.<br>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:<br>IP_H ≤ 0 or IP_H = 127 or IP_H > 255<br>or IP_MH < 0 or IP_MH > 255<br>or IP_ML < 0 or IP_ML > 255<br>or IP_L ≤ 0 or IP_L ≥ 255<br>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.<br>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.                              |
| C02       | Executable program count over error          | Execution requests were received for programs exceeding the number that can be executed simultaneously.  |
| C03       | Non-registered program specification error   | The specified program is not registered.   |
| C04       | Program entry point non-detection error      | A request was made to execute a program number for which no program steps are registered.  |
| C05       | Program first-step BGSR error                | The program specified for execution starts with BGSR.  |
| C06       | Executable step non-detection error          | The program specified for execution does not contain executable program steps.   |
| C07       | Subroutine non-definition error              | The subroutine specified for call is not defined.  |
| C08       | Subroutine duplicate-definition error        | The same subroutine number is defined at multiple locations.   |
| C0A       | Tag duplicate-definition error               | The same tag number is defined at multiple locations.  |
| C0B       | Tag non-definition error                     | The tag specified as the jump destination of a GOTO statement is not defined.  |



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

| Error No. | Error name   | Description, action, etc.  |
|-----------|--|--|
| 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       | Expansion-condition code error   | Input program step error. The expansion condition code is invalid.   |
| C18       | Expansion-condition LD simultaneous processing over error              | The number of LDs processed simultaneously exceeds the limit value.  |
| C19       | Expansion-condition LD shortage error 1                                | There is not enough LD when expansion condition A or O is used.  |
| C1A       | Expansion-condition LD shortage error 2                                | There is not enough LD when expansion condition AB or OB is used.  |
| C1C       | Unused-LD detection error  | An attempt was made to execute a command based on multiple LD condition that has been saved, without using it in expansion condition AB or OB.   |
| C1F       | Input-condition CND shortage error                                     | The necessary input condition is not found when an expansion condition is used.  |
| C21       | Input-condition use error with input-condition prohibited command      | Input-condition prohibited commands prohibit the use of input conditions.  |
| C22       | Invalid command position error with input-condition prohibited command | A command for which input condition is prohibited cannot be included in an input condition nest.   |
| C23       | Invalid operand error  | Program step error. The necessary operand data is invalid.   |



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

| Error No. | Error name   | Description, action, etc.   |
|-----------|--|---|
| 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.  |
| C34       | Zone number error  | The zone number is invalid.   |
| C35       | Point number error   | The point number is invalid.  |
| C36       | I/O port/flag number error                                   | The I/O port/flag number is invalid.  |
| C37       | Flag number error  | The flag number is invalid.   |
| C38       | Tag number error   | The tag number is invalid.  |
| C39       | Subroutine number error                                      | The subroutine number is invalid.   |
| C3A       | User-open communication channel number error                 | The channel number of the communication channel opened to the user is invalid.  |
| C3B       | Parameter number error                                       | The parameter number is invalid.  |
| C3C       | Variable number error  | The variable number is invalid.   |
| C3D       | String number error  | The string number is invalid.   |
| C3E       | String-variable data count specification error               | The specified number of string variables exceeds the area, etc.   |
| C40       | String-variable delimiter non-detection error                | Delimiter cannot be detected in the string variable.  |
| C41       | String-variable copy size over error                         | The copy size of string variable is too large.  |
| C42       | Character count non-detection error during string processing | The character-string length is not defined in string processing. Execute a string processing command after defining the length with a SLEN command. |



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

| Error No. | Error name   | Description, action, etc.   |
|-----------|--|---|
| C43       | Character-string length error during string processing           | The character-string length used in string processing is invalid. Check the value of character-string length defined by a SLEN command.         |
| C45       | Symbol definition table number error                             | The symbol definition table number is invalid.  |
| C46       | Blank area shortage error with source-symbol storage table       | There is not enough area to store the source symbols. Check the number of times source symbol can be used.                                      |
| 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 SCH command first.  |
| C4E       | SIO1 invalid usage OPEN error                                    | The usage of serial channel opened to the user does not match the parameter. Check "I/O parameter No. 90, Usage of SIO channel opened to user." |
| C4F       | SEL program/source symbol checksum error                         | The flash ROM data has been destroyed.  |
| C50       | Symbol definition table checksum error                           | The flash ROM data has been destroyed.  |
| C51       | Point data checksum error  | The flash ROM data has been destroyed.  |
| C52       | Backup SRAM data destruction error                               | The backup SRAM data has been destroyed. Check the battery.   |
| C53       | Invalid flash-ROM SEL global data/error list error               | The SEL global data/error lists in the flash ROM are invalid.   |
| C54       | Flash-ROM SEL global data/error list duplication error           | The SEL global data/error lists in the flash ROM are duplicated.  |
| C55       | Flash-ROM erase count over error for SEL global data/error lists | The number of time the flash ROM containing SEL global data/error lists can be erased was exceeded.   |
| C56       | Timing limit over error (Flash ROM erase)                        | Error erasing the flash ROM   |
| C57       | Flash-ROM verify error (Flash ROM erase)                         | Error erasing the flash ROM   |
| C58       | Flash-ROM ACK timeout error (Flash ROM erase)                    | Error erasing the flash ROM   |
| C59       | Head sector number specification error (Flash ROM erase)         | Error erasing the flash ROM   |
| C5A       | Sector count specification error (Flash ROM erase)               | Error erasing the flash ROM   |
| C5B       | Timing limit over error (Flash ROM write)                        | Error writing the flash ROM   |
| C5C       | Flash-ROM verify error (Flash ROM write)                         | Error writing the flash ROM   |
| C5D       | Flash-ROM ACK timeout error (Flash ROM write)                    | Error writing the flash ROM   |
| C5E       | Write-destination offset address error (Flash ROM write)         | Error writing the flash ROM   |



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

| Error No. | Error name   | Description, action, etc.   |
|-----------|--|---|
| 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       | Home-return incomplete error                             | Home return has not completed yet.  |
| C70       | Absolute coordinate non-confirmation error               | Absolute coordinates have not been confirmed. The power must be reconnected.  |
| C71       | Synchro slave-axis command error                         | A command was issued to the synchro slave axis.   |
| C72       | Overrun error  | The overrun sensor was actuated.  |
| C73       | Target-locus soft limit over error                       | The target position or movement locus exceeds a soft limit.   |
| C74       | Actual-position soft limit over error                    | The actual position exceeds a soft limit by the "soft limit/actual position margin" or more.  |
| C75       | Motion-data-packet generation logic error                | The motion-data-packet generation logic is invalid.   |
| C76       | Movement-point count over error                          | Too many packets are generated simultaneously.  |
| C77       | Handling-packet overflow error                           | The servo handling packets overflowed.  |
| C78       | Motion-data-packet overflow error                        | The servo motion data packets overflowed.   |
| C79       | Pole sense operation error                               | Operation is disabled in the pole sense mode.   |
| C7A       | Servo unsupported function error                         | An attempt was made to use an unsupported function.   |



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

| Error No. | Error name   | Description, action, etc.  |
|-----------|--|--|
| 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 servo ready OFF error   | The ready signal for the driver of the applicable axis is OFF.   |
| C87       | SEL unsupported function error   | An attempt was made to use a function not supported by SEL.  |
| C88       | Speed specification error  | The specified speed is invalid.  |
| C89       | Acceleration/deceleration specification error                              | The specified acceleration/deceleration is invalid.  |
| C8B       | Circle/arc calculation logic error   | The arc calculation logic is invalid.  |
| C8D       | Circle/arc calculation error   | Position data that cannot be used in arc movement was specified. Check the position data.  |
| C8E       | Point deletion error during command execution                              | The final point data was deleted while continuous point movement was being calculated.   |
| C8F       | Axis operation type error  | The axis operation type is invalid. Check "Axis-specific parameter No. 1, Axis operation type" and perform operation appropriate for the operation type specified.                     |
| C90       | Spline calculation logic error   | The spline processing logic is invalid.  |
| C91       | Push-motion axis multiple specification error                              | Two or more push-motion axes were specified.   |
| C92       | Push-motion approach distance/speed specification error                    | The specified push-motion approach distance/speed is invalid.  |
| C93       | System output operation error  | The user attempted a system output operation (through the port specified by I/O parameter for output function selection or the zone output port specified by axis-specific parameter). |
| C94       | PIO program number error   | The PIO-specified program number is invalid.   |
| C95       | AUTO program number error  | The setting of "Other parameter No. 1, Auto-start program number" is invalid.  |
| C96       | Start error from operation-abort program                                   | 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       | Home sensor non-detection error                                  | The home sensor cannot be detected. Check the wiring and sensor.  |
| C9A       | Creep sensor non-detection error                                 | The creep sensor cannot be detected. Check the wiring and sensor.   |
| C9B       | Phase Z non-detection error                                      | Phase Z cannot be detected. Check the wiring and encoder.   |
| C9C       | Defective phase-Z position error                                 | The phase-Z position is defective. Normal wear and tear of the mechanical ends and home sensor may also be a reason. Readjustment is necessary.   |
| C9D       | Card parameter write error                                       | Error writing card parameters   |
| C9E       | Servo calculation overflow error                                 | Internal servo calculation error  |
| CA1       | Abnormal absolute-data backup battery voltage (Driver analysis)  | Check the connection of the absolute-data backup battery/replace the battery and/or check the encoder cable connection, and then perform an absolute reset.   |
| CA2       | Abnormal absolute-data backup battery voltage (Main analysis)    | Check the connection of the absolute-data backup battery/replace the battery and/or check the encoder cable connection, and then perform an absolute reset.   |
| CA3       | Slave setting data out-of-range error                            | The data set to the slave is outside the allowable range.   |
| CA4       | Slave error response   | An error response was returned from the slave.  |
| CA5       | Stop deviation overflow error                                    | Movement may have occurred during stopping due to external force or operation may have been restricted during deceleration. This error may also generate when jog operation is restricted (due to contact with an obstacle, contact with a mechanical end before home return, etc.) or when wiring error, faulty encoder or faulty motor is detected during deceleration. |
| CA6       | Palletizing number error   | The specified palletizing number is invalid.  |
| CA7       | Setting error of even-numbered row count for palletizing zigzag  | The set even-numbered row count for palletizing zigzag is invalid.  |
| CA8       | Setting error of palletizing pitches                             | The set palletizing pitches are abnormal.   |
| CA9       | Setting error of placement points in palletizing-axis directions | The set X/Y-axis direction counts for palletizing are invalid.  |
| 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.)

| 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.   |
| CBE       | Target-locus boundary over error   | The target position or movement locus exceeded the positioning boundary in the infinite-stroke mode.   |
| 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.   |



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

| Error No. | Error name  | Description, action, etc.   |
|-----------|---|---|
| CCC       | Arch end-point/trigger reversing error                | The coordinates of end point and end-point arch trigger are reversed during arch motion operation.            |
| CCD       | Drive-source cutoff axis use error                    | An attempt was made to use an axis whose drive source is cut off.   |
| CCE       | Error axis use error                                  | An attempt was made to use an axis currently generating an error.   |
| CCF       | Palletizing reference-point/valid-axis mismatch error | The PX/PY/PZ-axes set by PASE/PCHZ are not valid in the axis pattern of the reference-point data set by PAST. |
| D01       | Encoder EEPROM-write timeout error                    | The encoder is faulty or failure occurred in the encoder communication.                                       |
| D02       | Encoder EEPROM-read timeout error                     | The encoder is faulty or failure occurred in the encoder communication.                                       |
| D03       | Encoder count error                                   | Faulty encoder or defective encoder assembly condition is suspected.  |
| D04       | Encoder one-revolution reset error                    | The encoder is faulty or has turned.  |
| D05       | Encoder-EEPROM write acceptance error                 | The encoder is faulty or failure occurred in the encoder communication.                                       |
| D06       | Encoder received-data error                           | The encoder is faulty or failure occurred in the encoder communication.                                       |
| D07       | Driver logic error                                    | The driver CPU board is in a condition where it cannot operate normally.                                      |
| D08       | Encoder CRC error                                     | The encoder is faulty or failure occurred in the encoder communication.                                       |
| D09       | Driver overspeed error                                | The motor speed exceeded the upper limit.   |
| D0A       | Driver overload error                                 | The power input to the motor exceeded the upper limit.  |
| D0B       | Driver EEPROM data error                              | Failure during write or EEPROM failure  |
| D0C       | Encoder EEPROM data error                             | Failure during write or EEPROM failure  |
| D0E       | Axis sensor error                                     | An error occurred in the axis sensor.   |
| D0F       | Power stage temperature error                         | The power stage board exceeded the upper temperature limit.   |
| D10       | IPM error   | A failure occurred in the motor drive circuit.  |
| D11       | Driver abnormal interruption error                    | The driver CPU board is in a condition where it cannot operate normally.                                      |
| D12       | Encoder disconnection error                           | The encoder cable is disconnected.  |
| 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.   |
| D29       | Excitation detection error                   | An error was detected during excitation communication.                           |
| D2A       | Driver control power overvoltage error       | An overvoltage error was detected in the driver control power.                   |
| D2B       | Driver control power voltage drop error      | A voltage drop was detected in the driver control power.                         |
| D2C       | Drive-power overvoltage error                | An overvoltage error was detected in the motor drive power.                      |
| D2D       | Drive-power voltage drop error               | A voltage drop was detected in the motor drive power.                            |
| D2E       | Synchronous communication error              | A communication failure occurred between the driver board and FPGA (main).       |
| D50       | Fieldbus error (FBMIRQ timeout)              | A FBMIRQ timeout was detected.   |
| D51       | Fieldbus error (FBMIRQ reset)                | A FBMIRQ reset error was detected.   |
| D52       | Fieldbus error (FBMSY)                       | A FBMSY 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 isolation power error    | An Expanded-SIO isolation power error was detected.                              |



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

| Error No. | Error name  | Description, action, etc.   |
|-----------|---|---|
| D61       | Expanded-SIO 1/3 CH isolation power error                 | An Expanded-SIO isolation 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.  |
| 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. |
| D67       | Motor/encoder configuration information mismatch error    | Driver parameter No. 25 "Motor/encoder configuration information" (motor identification number, encoder identification number) does not match encoder parameter No. 11 "Motor/encoder configuration information" (motor identification number, encoder identification number). Check the parameter values, connection of the encoder cable, etc.  |
| D68       | No remote-mode control support board error                | Hardware supporting remote-mode control is not installed, although remote-mode control (AUTO/MANU) is specified in I/O parameter No. 79.  |
| D69       | External terminal block overcurrent or power-supply error | Overcurrent or power-supply error in the external terminal block  |
| 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.   |
| D6E       | Motor drive-source OFF error (MPONSTR-OFF)                | An OFF status of the drive source (MPONSTR-OFF) was detected in a non-shutdown (SHDWNSTR-OFF) state.  |
| D70       | Option use permission error                               | Check, among others, if an option is specified with a system program that does not permit use of options.   |
| 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.  |



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

| Error No. | Error name  | Description, action, etc.  |
|-----------|---|--|
| E12       | SIO-bridge SCIF send-queue overflow error         | The send queue overflowed.   |
| 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.   |
| E1E       | Zone parameter error                              | A value other than an output port/global flag number ("0" is acceptable) or duplicate numbers may be input in axis-specific parameter Nos. 88, 91, 94 and 97, or the output number specified as system output in the I/O parameter for output function selection may be duplicated, among other reasons. |
| 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)   |



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

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



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

| Error No. | Error name   | Description, action, etc.   |
|-----------|--|---|
| E4D       | Driver phase-Z detection logic error                       | A phase-Z detection completion status was notified from the driver in a mode other than the phase-Z detection operation mode. |
| E4E       | Phase-Z count parameter error                              | Check axis-specific parameter Nos. 23, 38, 37, etc.   |
| E4F       | Synchro parameter error                                    | Check axis-specific parameter Nos. 65, 39, all-axis parameter No. 1, etc.   |
| E50       | Driver special command ACK-timeout error                   | ACK cannot be detected for the driver special command.  |
| E51       | Drive unit error (DRVESR)                                  | Error notification from the driver  |
| E52       | Encoder error (DRVESR)                                     | Error notification from the driver  |
| E53       | Driver CPU error (DRVESR)                                  | Error notification from the driver  |
| E54       | Servo control error (DRVESR)                               | Error notification from the driver  |
| E55       | Command error (DRVESR)                                     | Error notification from the driver  |
| E56       | Motor temperature error (DRVESR)                           | Error notification from the driver  |
| E58       | Servo ON/OFF timeout error                                 | Servo ON/OFF cannot be confirmed.   |
| E59       | Brake ON/OFF timeout error                                 | Brake ON/OFF cannot be confirmed.   |
| E5A       | Pole sense non-detection error                             | Motor magnetic pole cannot be detected.   |
| E5B       | Detection OFF error upon pole sense completion             | The motor-magnetic-pole detection status bit (Psenex) is turned OFF after completion of pole sense.                           |
| E5C       | Hold-at-stop servo job error                               | The servo job is invalid.   |
| E5D       | Servo packet error   | The servo packets are invalid.  |
| E5E       | Servo-control-right management array number error          | The servo-control-right management array number is invalid.   |
| E5F       | Length conversion parameter error                          | Check axis-specific parameter Nos. 47, 50, 51, 42, 1, etc.  |
| E60       | Slave maximum receive size over error                      | The slave receive size is too large.  |
| E61       | Slave no normal response reception error                   | Normal response cannot be received from the slave.  |
| E62       | Sending-slave CPU type error                               | The CPU type of the sending slave is invalid.   |
| 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.  |



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

| Error No. | Error name   | Description, action, etc.  |
|-----------|--|--|
| E6A       | Safety-gate open status requiring reset recovery (not error)                   | Close the safety gate and then reconnect the power.  |
| E6B       | Shutdown factor indeterminable error   | Shutdown factor cannot be determined.  |
| E6C       | DO output current error  | The DO output current is abnormal.   |
| E6D       | Drive-source cutoff relay error  | The drive-source cutoff relay may have been melted.  |
| 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.   |
| 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.   |



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

| Error No.  | Error name  | Description, action, etc.   |
|------------|---|---|
| E81        | Priority auto-assignment card non-detection error | Priority auto-assignment card cannot be detected.   |
| E82        | Card mismatch error                               | The combination or positioning of I/O slot cards has a problem.   |
| E83        | I/O slot card error                               | The I/O slot card is invalid.   |
| E84        | Resolution parameter error                        | Check axis-specific parameter Nos. 47, 50, 51, 44, 42, 43, 1, 37, etc.  |
| E85        | Driver ready OFF factor indeterminable error      | Driver ready OFF factor cannot be determined.   |
| E86        | Fieldbus error (FBVCCER)                          | A fieldbus error (FBVCCER) was detected.  |
| E87        | Fieldbus error (FBPOWER)                          | A fieldbus error (FBPOWER) was detected.  |
| E88        | Power error (Other)                               | A power error (Other) was detected. This error also generates when the power OFF → ON interval is short. After the power has been turned off, be sure to wait for at least 5 seconds before turning it back on. Abnormal regenerative resistance temperature is also suspected. |
| E89        | SCIF open error in non-AUTO mode (Servo in use)   | In a mode other than AUTO, opening of the serial 1 channel (also used by the PC software/TP port) from a SEL program is prohibited while the servo is in use (to ensure safety).  |
| E8A        | SEL program flash-ROM status error                | Data is not written to the flash ROM correctly or written in an old, incompatible application version.  |
| E8B        | Symbol definition table flash-ROM status error    | Data is not written to the flash ROM correctly or written in an old, incompatible application version.  |
| E8C        | Point data flash-ROM status error                 | Data is not written to the flash ROM correctly or written in an old, incompatible application version.  |
| E8D        | Parameter flash-ROM status error                  | Data is not written to the flash ROM correctly or written in an old, incompatible application version.  |
| FF0 to F00 | Shutdown error (hi_sysdwn () definition)          | A shutdown error (hi_sysdwn () definition) was detected.  |
| F03 to 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.  |



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

| Error No.  | Error name                                      | Description, action, etc.  |
|------------|---|--|
| 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 to 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.  |



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

| Error No. | Error name   | Description, action, etc.  |
|-----------|--|--|
| A84       | Code sector block ID erase count over                                  | The number of times the flash ROM was erased exceeded the allowable count.   |
| A85       | FROM write request error before erase is complete                      | When updating, a flash-ROM write command was received before a flash-ROM erase command. Confirm that the update program file is valid and then perform update again.   |
| A86       | Absolute-encoder backup battery voltage-low warning (Driver detection) | The voltage of the absolute-data backup battery is low. Check the battery connection or replace the battery.   |
| A87       | Motorola S byte count error (Detected by the core)                     | The update program file is invalid. Check the file.  |
| A88       | Message conversion error (Detected by the core)                        | The received message does not match the message format or contains invalid data. Check the message sent from the host communication device.  |
| A89       | Update target non-specification error (Detected by the core)           | An update command was received before the update target was correctly specified during update processing. Check if an appropriate update PC tool is used and if the target specification and other settings of the update PC tool are correct. |
| A8A       | Update system code error (Detected by the core)                        | The system code in the message of the received update target specification command does not match the controller system. Check the target specification and other settings of the update PC tool.  |
| A8B       | Update unit code error (Detected by the core)                          | The unit code in the message of the received update target specification command does not match the controller unit that can be updated. Check the target specification and other settings of the update PC tool.                              |
| A8C       | Update device number error (Detected by the core)                      | The device number specified in the message of the received update target specification command is not appropriate. Check the target specification, device number and other settings of the update PC tool.                                     |
| A8D       | Flash busy reset timeout (Detected by the core)                        | Error erasing/writing the flash ROM  |
| A8E       | Unit type error (Detected by the core)                                 | The unit type in the received command message is invalid or not supported.   |



(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.  |



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

| Error No. | Error name   | Description, action, etc.  |
|-----------|--|--|
| EAC       | Servo control underrun error                         | A servo control underrun error was detected.   |
| EAD       | Boot error   | A FPGA boot watchdog was detected. The core program may not be running properly.   |
| EAE       | Application-update SCIF receive-queue overflow error | Excessive data is received from outside. (Confirm that a PC and IAI's update tool are used to update the application.)                               |
| EAF       | Installed flash ROM type mismatch (Core)             | The flash ROM type anticipated in the software does not match the flash ROM type actually installed. Check the combination of software and hardware. |
| EB2       | Flash busy reset timeout (Detected by the core)      | Malfunction of the flash ROM. The flash ROM is not reset from the busy mode.   |



## ☉ Troubleshooting of X-SEL Controller

A panel window is provided in the front panel of the Tabletop Robot.

Error numbers will be displayed in this panel window.

When the power is turned on, normally “rdy” or “Ardy” will be displayed. “P01” or other code will be displayed while a program is running.

When an error generates, the panel window will show “EA1D” or other code starting with “E.” (Some errors do not begin with “E.”)

| Status                     | Panel window display |
|----------------------------|----------------------|
| After turning on the power | rdy, Ardy            |
| Program is running         | P01, P64, etc.       |
| Error has generated        | EA1D, ED03, etc.     |

\* Among the alphabets, B and D are shown in lower case.

Depending on the error number, it may be possible to reset the error after removing the cause of the error, or the power must be reconnected to reset the error.

Also, some error numbers are output to the LED display in the panel window, while others are not. For details, see “☉ Error Level Control.”



Troubleshooting (Causes and Countermeasures for Key Errors)

| Error No. | Error name                                | Cause   | Countermeasure  |
|-----------|---|---|---|
| ACF       | AC power cutoff                           | Momentary power failure has occurred or the voltage has dropped.<br>100 V is input while the controller's voltage specification is 200 V. | Check the power-source voltage.<br>If the last digit of the controller's model number is "-1," the power specification is 100 V. If the last digit is "-2," the power specification is 200 V.   |
| ErG       | Emergency stop<br>(This is not an error.) | Emergency-stop signal is input.   | Emergency-stop signal is input in the following condition:<br>1. The emergency-stop button on the teaching pendant is pressed.<br>2. The applicable input terminal in the system connector is turned ON.<br>3. The port switch on the front panel is set to the manual side.<br>(The teaching-pendant/PC-software connector is not connected.)<br>4. The actuator is of sensor specification and the slider is stopped on either end of the slider. |
| dSF       | Deadman switch OFF                        | The switch is set to the manual side even when the teaching-pendant connector or other connector is not connected.                        | Set the switch to the auto side when the teaching-pendant connector or other connector is not connected.  |
| CA5       | Stop deviation overflow error             | Operation is mechanically disabled.<br>If there is no problem in the mechanical function, the power stage board is faulty.                | Check to see if the actuator mounting bolts are contacting inside the axes, or if the slider attachment is contacting any surrounding mechanical parts.<br>Replace the board.   |
| C6b       | Deviation overflow error                  | Operation is mechanically disabled.   | Check to see if the actuator mounting bolts are contacting inside the axes, or if the slider attachment is contacting any surrounding mechanical parts.   |
| d03       | Encoder count error                       | The encoder is faulty or dust is attached.  | Remove the motor cover and apply cleaning air spray for OA equipment, etc., over the cord wheel.<br>If the problem persists, replace/readjust the encoder.  |



| Error No. | Error name               | Cause  | Countermeasure  |
|-----------|--------------------------|--|---|
| d10       | IPM error                | The motor coil is damaged.<br><br>If the motor coil is not damaged, the power stage board (to which the motor power cable is connected) is faulty. | Measure resistances among phases U/V/W. If the resistance values are different, the coil has been burned. Replace the motor.<br>If the resistance values are almost the same, the coil has not been burned.<br>Replace the board. |
| 807       | Shutdown relay ER status | The transistor on the power-supply board (to which the power cable is connected) is damaged.   | Replace the board.  |





## Change History

| Revision Date | Description of Revision  |
|---------------|--|
|               | First edition  |
|               | Second edition   |
|               | Third edition  |
| December 2009 | Fourth edition <ul style="list-style-type: none"><li>• Corrected clerical errors, etc.</li><li>• P. 282 to 287: Added input/output function selections.</li></ul>  |
| May 2010      | Fifth edition <ul style="list-style-type: none"><li>• Added “Before Using This Product” on the first page after the cover.</li><li>• Added “Safety Guide” at the beginning of the main text after the table of contents.</li><li>• Added “Change History” on the last page.</li><li>• Updated the back cover. (Changed the addresses of the head office and sales offices, and indicated that the customer service center Eight is open 24 hours, among others.)</li></ul> |
| October 2010  | Sixth edition <ul style="list-style-type: none"><li>• Warning notes for “Position during servo-on” are added below the servo-on descriptions in pages 140 and 141.</li></ul>   |
| April 2011    | Seventh edition<br>Swapped over the page for CE Marking  |
| March 2012    | Eighth edition <ul style="list-style-type: none"><li>• Contents added and changed in Safety Guide in pages 1 to 7</li><li>• Caution in Handling added in pages 8</li><li>• Contents changed in 13. Warranty in pages 12 to 13</li><li>• Weight added to external dimensions on pages 36 to 43</li><li>• Warning notes added such as in case the grease got into your eye, immediately go to see the doctor for an appropriate care in pages 248 and 249</li></ul>          |
| May 2012      | Ninth edition <ul style="list-style-type: none"><li>• Correction to Ambient Temperature Range on pages 16.<br/>0°C to 40 ° → 5°C to 40°C</li></ul>   |
| Augsut 2012   | Tenth edition <ul style="list-style-type: none"><li>• P. 74 to 83: Clerical error correction of a page.</li></ul>  |





## ***IAI Corporation***

Head Office: 577-1 Obane Shimizu-KU Shizuoka City Shizuoka 424-0103, Japan  
TEL +81-54-364-5105 FAX +81-54-364-2589  
website: [www.iai-robot.co.jp/](http://www.iai-robot.co.jp/)

Technical Support available in USA, Europe and China

## ***IAI America, Inc.***

Head Office: 2690 W. 237th Street, Torrance, CA 90505  
TEL (310) 891-6015 FAX (310) 891-0815  
Chicago Office: 1261 Hamilton Parkway, Itasca, IL 60143  
TEL (630) 467-9900 FAX (630) 467-9912  
Atlanta Office: 1220 Kennestone Circle, Suite 108, Marietta, GA 30066  
TEL (678) 354-9470 FAX (678) 354-9471  
website: [www.intelligentactuator.com](http://www.intelligentactuator.com)

## ***IAI Industrieroboter GmbH***

Ober der Röth 4, D-65824 Schwalbach am Taunus, Germany  
TEL 06196-88950 FAX 06196-889524

## ***IAI (Shanghai) Co., Ltd.***

SHANGHAI JIAHUA BUSINESS CENTER A8-303, 808, Hongqiao Rd. Shanghai 200030, China  
TEL 021-6448-4753 FAX 021-6448-3992  
website: [www.iai-robot.com](http://www.iai-robot.com)