Instruction Manual

Positioning Module
(with Analog Voltage Output)

Yokogawa Electric Corporation

IM 34M6H58-01E

1st Edition
Applicable Products:

FA-M3 Range-free Multi-controllers
Models: F3NC51-0N and F3NC52-0N
Name: Positioning Module (with Analog Voltage Output)

The document number and document model code for this manual are as follows:
Document number: IM 34M6H58-01E
Document model code: DOCIM

Please refer to the document number in all communications; also refer to the document number or document model code when purchasing additional manuals.
Important

About This Manual

(1) This manual should be passed on to the end user.
(2) Before using the module, read this manual completely to get a thorough understanding of the module.
(3) This manual explains the functions contained in this product, but does not warrant that those will suit the particular purpose of the user.
(4) Under absolutely no circumstances may the contents of this manual be transcribed or copied, in part or in whole, without permission.
(5) The contents of this manual are subject to change without prior notice.
(6) Every effort has been made to ensure accuracy in the preparation of this manual. However, should any errors or omissions come to the attention of the user, please contact the nearest Yokogawa Electric representative or sales office.
Safety Precautions when Using/Maintaining the Product

The following safety symbols are used on the product as well as in this manual.

**CAUTION**

This symbol indicates that the operator must follow the instructions laid out in this manual in order to avoid the risk of personnel injuries or fatalities or damage to the instrument. The manual describes what special care the operator must exercise to prevent electrical shock or other dangers that may result in injury or the loss of life.

Before using the instrument, be sure to ground this terminal.

Function ground terminal
Before using the instrument, be sure to ground this terminal.

 Indicates alternating current.

 Indicates direct current.

(1) The following symbols are used only in the instruction manual.

**WARNING**

Indicates that the operator must refer to the instructions in this manual in order to prevent the instrument (hardware) or software from being damaged, or a system failure from occurring.

**CAUTION**

Draws attention to information essential for understanding the operation and functions.

**TIP**

Gives information that complements the present topic.

**SEE ALSO**

Identifies a source to which to refer.

(2) For the protection and safe use of the product and the system controlled by it, be sure to follow the instructions and precautions on safety stated in this manual whenever handling the product. Take special note that if you handle the product in a manner other than prescribed in these instructions, safety cannot be guaranteed.

(3) If separate protection and/or safety circuits for this product or the system which is controlled by this product are to be installed, ensure that such circuits are installed external to the product.

(4) If component parts or consumables are to be replaced, be sure to use parts specified by the company.

(5) Do not attempt to make modifications or additions internal to the product.
Force Majeure

(1) Yokogawa Electric Corporation (hereinafter referred to as Yokogawa Electric) makes no warranties regarding the product except those stated in the WARRANTY that is provided separately.

(2) Yokogawa Electric assumes no liability to any party for any loss or damage, direct or indirect, caused by the user or any unpredictable defect of the product.

Software Supplied by the Company

(1) Yokogawa Electric makes no other warranties expressed or implied except as provided in its warranty clause for software supplied by the company.

(2) Use the relevant software with one specified computer only. You must purchase another copy of the software for use with each additional computer.

(3) Copying the software for any purpose other than backup is strictly prohibited.

(4) Store the floppy disks (originals) of this software in a safe place.

(5) Reverse engineering, such as decompiling of the software, is strictly prohibited.

(6) No portion of the software supplied by Yokogawa Electric may be transferred, exchanged, or sublet or leased for use by any third party without prior permission by Yokogawa Electric.
General Requirements for Using FA-M3 Controllers

- Avoid installing FA-M3 controllers in the following locations:
  - Where the instrument will be exposed to direct sunlight, or where the operating temperature is outside the range 0°C to 55°C.
  - Where the relative humidity is outside the range 10 to 90%, or where sudden temperature changes may occur and cause condensation.
  - Where corrosive or inflammable gases are present.
  - Where the instrument will be exposed to direct mechanical vibration or shock.

- Securely tighten screws:
  - Securely tighten module mounting screws and terminal screws to avoid problems such as faulty operation.

- Securely fasten connectors of interconnecting cables:
  - Securely fasten connectors of interconnecting cables, and check them thoroughly before turning on the power.

- Interlock with emergency-stop circuitry using external relays:
  - Equipment incorporating the FA-M3 controllers must be furnished with emergency-stop circuitry that uses external relays. This circuitry should be set up to interlock correctly with controller status (stop/run).

- Ground FA-M3 controllers to an independent Japanese Industrial Standard (JIS) Class 3 Ground:
  - Avoid grounding the FG terminal of the FA-M3 controller to the same ground as high-voltage power lines. Ground the terminal to an independent JIS Class 3 ground (ground resistance up to 100 W).

- Observe countermeasures against noise:
  - When assigning inputs/outputs, the user should avoid locating AC-supplied I/O modules in the vicinity of the CPU module.

- Keep spare parts on hand:
  - Stock up on maintenance parts, including spare modules, in advance.

- Discharge static electricity before operating the system:
  - Because static charge can accumulate in dry conditions, first touch grounded metal to discharge any static electricity before touching the system.

- Never use solvents such as paint thinner for cleaning:
  - Gently clean the surfaces of the FA-M3 controllers with a piece of soft cloth soaked in water or a neutral detergent.
  - Do not use solvents such as paint thinner for cleaning, as they may cause deformation, discoloration, or malfunctioning.
Avoid storing the FA-M3 controllers in places with high temperature or humidity:
  • Since the CPU module has a built-in battery, avoid storing it in places with high temperature or humidity.
  • Since the service life of the battery is drastically reduced by exposure to high temperatures, so take special care (storage temperature can be from -20° to 75°C).

Always turn off the power before installing or removing modules:
  • Turn off power to the power supply module when installing or removing modules, otherwise damage may result.

When installing ROM packs and changing switch settings:
  • In some modules you can remove the right-side cover and install ROM packs or change switch settings. While doing this, do not touch any components on the printed-circuit board, otherwise components may be damaged and modules fail.
CONTENTS

- Important ...................................................................................................... ii
- 1. Overview ................................................................................................. 1-1
- 2. Specifications .......................................................................................... 2-1
  2.1 General Specifications .............................................................................. 2-1
  2.2 Operating Environment ............................................................................. 2-2
  2.3 Model and Suffix Codes ............................................................................. 2-2
  2.4 Applicable Encoders ................................................................................. 2-2
  2.5 Components ............................................................................................. 2-3
  2.6 External Dimensions ............................................................................... 2-4
  2.7 Terminal Assignments and Connection ..................................................... 2-5
  2.8 Applicable External Interface Connectors .................................................. 2-6
- 3. Function Overview .................................................................................... 3-1
  3.1 Positioning Operation ............................................................................... 3-2
  3.2 Change in Target Position during Positioning .......................................... 3-3
  3.3 Change in Velocity during Positioning .................................................... 3-4
  3.4 Velocity Control ........................................................................................ 3-5
  3.5 Change in Velocity during Velocity Control ............................................ 3-6
  3.6 Velocity-to-Position Control Mode Switching ......................................... 3-7
  3.7 Jog Stepping ............................................................................................. 3-8
  3.8 Emergency-stop Input .............................................................................. 3-9
  3.9 Contact Inputs .......................................................................................... 3-10
  3.10 Contact Outputs ...................................................................................... 3-11
  3.11 SEN Signal Output ................................................................................... 3-12
  3.12 Origin-search Operation .......................................................................... 3-13
  3.13 Interrupt Function .................................................................................. 3-14
  3.14 Manual Pulser Mode .............................................................................. 3-15
  3.15 Linear-interpolated Operation .................................................................. 3-16
  3.16 On-route Operation ............................................................................... 3-17
  3.17 Arc-interpolated Operation ..................................................................... 3-18
- 4. Parameters ............................................................................................... 4-1
  4.1 List of Parameters .................................................................................... 4-1
    4.1.1 Entry Parameters ............................................................................... 4-4
    4.1.2 Startup Parameters ............................................................................ 4-5
4.1.3 Startup Parameters Related to Origin Search ................................... 4-5
4.1.4 Extended Command Parameters ...................................................... 4-6
4.1.5 Control Mode Switching Parameters ................................................. 4-6
4.1.6 Arc-Interpolation Parameters ............................................................ 4-7

4.2 List of Parameters for Each Command ..................................................... 4-8

4.3 Description of Parameters ........................................................................ 4-10
  4.3.1 Entry Parameters ............................................................................ 4-10
  4.3.2 Startup Parameters ........................................................................ 4-17
  4.3.3 Startup Parameters Related to Origin Search ................................. 4-20
  4.3.4 Extended Command Parameters .................................................... 4-21
  4.3.5 Switch Control Mode Parameter ..................................................... 4-22
  4.3.6 Arc-interpolation Parameters .......................................................... 4-22

4.4 Example of Entry Parameters Setting .............................................................. 4-25

5. Status ........................................................................................................ 5-1
  5.1 List of Status ........................................................................................ 5-1
  5.2 Description of Status ............................................................................. 5-3

6. List of Input/Output Relays ........................................................................ 6-1
  6.1 Output Relays ........................................................................................ 6-1
  6.2 Input Relays ............................................................................................ 6-2

7. Accessing Modules ................................................................................... 7-1
  7.1 Accessing from Sequence CPU .............................................................. 7-1
    7.1.1 Reading the Module Status ............................................................... 7-3
    7.1.2 Set Parameter ................................................................................. 7-4
    7.1.3 Error Reset ...................................................................................... 7-7
    7.1.4 Jog Stepping .................................................................................... 7-9
    7.1.5 Origin-search .................................................................................. 7-12
    7.1.6 Write Current Position .................................................................... 7-17
    7.1.7 Position Control Mode Operation .................................................... 7-19
    7.1.8 Velocity Control Mode Operation .................................................... 7-28
    7.1.9 Switching Velocity to Position Control .......................................... 7-36
    7.1.10 Request to Decelerate and Stop ..................................................... 7-39
    7.1.11 Request to Stop Immediately .......................................................... 7-41
    7.1.12 Extended Command Instructions .................................................. 7-43
    7.1.13 On-route Operation ....................................................................... 7-56
    7.1.14 Arc-interpolation Operation ........................................................... 7-60
  7.2 Accessing from BASIC CPU ................................................................. 7-63
    7.2.1 Reading the Module Status ............................................................... 7-65
    7.2.2 Set Parameter ................................................................................. 7-66
    7.2.3 Error Reset ...................................................................................... 7-67
    7.2.4 Jog Stepping .................................................................................... 7-68
    7.2.5 Origin-search .................................................................................. 7-69
    7.2.6 Write Current Position .................................................................... 7-72
1. Overview

The Models F3NC51-0N and F3NC52-0N are advanced positioning modules (hereinafter simply referred to as the modules or positioning modules) used to control servo drivers and thereby the speed and position of pulse-driven motors. When in use, the positioning modules are attached to the base module of an FA-M3 controller. According to commands from the CPU module of the FA-M3 controller, the positioning modules generate paths for finding positions, carry out positioning loop computations based on feedback by an external position detector, and issue velocity-control commands in the form of analog voltages.

Features

- Just one module can control different types of servo motors/drivers, including uniaxial (F3NC51-0N module) and biaxial (F3NC52-0N module) servo motors. It can also deal with input from incremental and absolute encoders.
- Provided with multiaxial simultaneous control capabilities. Driven by commands from the CPU module, the modules can perform smooth and versatile position control based on control factors, such as multiaxial linear interpolation, velocity control, and control for switching between the velocity- and position-control modes.
- Can come into action quickly and operate in synchronization with peripheral equipment thanks to the shorter startup time (6 ms maximum), “on-route” operation, the conditional startup capability which uses internal/external triggers, and the end-of-position-detection/positioning interrupt function.

![Figure 1.1 Operating Principle of Positioning Module (with Analog Voltage Output)](image-url)
**WARNING**

- In order for the modules to detect any faulty behavior of a motor, the following entry parameters must be set at their correct values.
  - Positive-direction limit value
  - Negative-direction limit value
  - Velocity limit value
  - Over-limit velocity detection
  - Over-limit acceleration detection
  - Deviation error detection

- In order to ensure safety if the positioning module or a servo driver fails or if any of the signal lines breaks down, be SURE to take necessary safety measures in your system design. Such measures should include manually shutting down the power to a motor if the motor runs in excess of the normal operating limits specified.

---

**CAUTION**

For a servo driver to be connected to the positioning module, choose a velocity-control driver. Position-control and torque-control drivers cannot be connected to the module.
## 2. Specifications

### 2.1 General Specifications

<table>
<thead>
<tr>
<th>Item</th>
<th>Specifications</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Number of axes</strong></td>
<td>F3NC51-0N Module</td>
</tr>
<tr>
<td></td>
<td>F3NC52-0N Module</td>
</tr>
<tr>
<td><strong>Control</strong></td>
<td>One</td>
</tr>
<tr>
<td><strong>Method</strong></td>
<td>Semi-closed-loop control based on feedback by an encoder</td>
</tr>
<tr>
<td><strong>Output</strong></td>
<td>-10 to 10 V</td>
</tr>
<tr>
<td><strong>Encoder</strong></td>
<td>• Incremental encoder (A- and B-phases)</td>
</tr>
<tr>
<td></td>
<td>Maximum of 2 Mpps for RS422-based differential input (when fourfold multiplication applies)</td>
</tr>
<tr>
<td></td>
<td>• Absolute encoder</td>
</tr>
<tr>
<td><strong>Control mode</strong></td>
<td>Position control, velocity control and control for switching between position-control and velocity-control modes</td>
</tr>
<tr>
<td><strong>Position control</strong></td>
<td>Interpolation method</td>
</tr>
<tr>
<td></td>
<td>Axis-by-axis independent interpolation</td>
</tr>
<tr>
<td></td>
<td>Multiaxial linear interpolation (set from CPU module)</td>
</tr>
<tr>
<td></td>
<td>Biaxial arc interpolation (set from CPU module)</td>
</tr>
<tr>
<td><strong>Pulse range</strong></td>
<td>-134,217,728 to 134,217,727 pulses</td>
</tr>
<tr>
<td><strong>Pulse rate range</strong></td>
<td>0.1 to 2,000,000 pulses/sec.</td>
</tr>
<tr>
<td><strong>Functionality</strong></td>
<td>Selection between absolute/relative positions</td>
</tr>
<tr>
<td></td>
<td>On-route operation</td>
</tr>
<tr>
<td></td>
<td>Change in target position/velocity during operation Axis stepping using a manual pulser</td>
</tr>
<tr>
<td><strong>Velocity control</strong></td>
<td>Pulse rate range</td>
</tr>
<tr>
<td></td>
<td>-2,000,000 to 2,000,000 pulses/s</td>
</tr>
<tr>
<td><strong>Functionality</strong></td>
<td>Change in velocity during operation</td>
</tr>
<tr>
<td><strong>Acceleration/deceleration</strong></td>
<td>Acceleration/deceleration method</td>
</tr>
<tr>
<td></td>
<td>Trapezoidal, two-line-segment or S-shape (three-line-segment) tracking</td>
</tr>
<tr>
<td></td>
<td>0 to 32,767 ms each for acceleration/deceleration</td>
</tr>
<tr>
<td><strong>Origin search</strong></td>
<td>Search method</td>
</tr>
<tr>
<td></td>
<td>User-definable by entering an origin setpoint, external trigger or limits</td>
</tr>
<tr>
<td><strong>Search speed</strong></td>
<td>User-definable</td>
</tr>
<tr>
<td><strong>External contact output</strong></td>
<td>Servo ON, Driver reset and Brake OFF contacts</td>
</tr>
<tr>
<td><strong>External contact input</strong></td>
<td>Limit switch, Driver alarm, Origin, External trigger, General-purpose input and Emergency stop contacts</td>
</tr>
<tr>
<td><strong>Data backup</strong></td>
<td>By CPU module</td>
</tr>
<tr>
<td><strong>Startup time</strong></td>
<td>6 ms max.</td>
</tr>
<tr>
<td><strong>Current consumption</strong></td>
<td>390 mA (5 V DC)</td>
</tr>
<tr>
<td></td>
<td>400 mA (5 V DC)</td>
</tr>
<tr>
<td><strong>External power supply</strong></td>
<td>24 V DC, 10 mA</td>
</tr>
<tr>
<td><strong>External wiring</strong></td>
<td>40-pin connector (one unit)</td>
</tr>
<tr>
<td></td>
<td>40-pin connector (two units)</td>
</tr>
<tr>
<td><strong>External dimensions</strong></td>
<td>28.9 (W) \times 100 (H) \times 83.2 (D) (mm) (Note)</td>
</tr>
<tr>
<td><strong>Weight</strong></td>
<td>130 g</td>
</tr>
<tr>
<td></td>
<td>140 g</td>
</tr>
</tbody>
</table>

Note: Excluding protrusions (see the diagram of external dimensions for more details).
2.2 Operating Environment

No restrictions apply to CPU modules with which the positioning modules can be used.

2.3 Model and Suffix Codes

<table>
<thead>
<tr>
<th>Model Code</th>
<th>Suffix Code</th>
<th>Style Code</th>
<th>Option Code</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>F3NC51</td>
<td>-0N</td>
<td>. . . . . .</td>
<td>. . .</td>
<td>Uniaxial, position-loop control; -10 to 10 voltage output; maximum velocity of 2 Mpps</td>
</tr>
<tr>
<td>F3NC52</td>
<td>-0N</td>
<td>. . . . . .</td>
<td>. . .</td>
<td>Biaxial, position-loop control; -10 to 10 voltage output; maximum velocity of 2 Mpps</td>
</tr>
</tbody>
</table>

2.4 Applicable Encoders

1. General-purpose two-phase rotary encoders
2. Serial absolute encoders from YASKAWA Electric Corporation (for example, Σ Series)
3. Serial absolute encoders from Sanyo Denki Co., Ltd. (for example, P Series) or compatible models – Manchester-encoded serial transmission – (for example, MINAS Series from Matsushita Electric Industrial Co., Ltd.)
2.5 Components

- F3NC51-0N module (uniaxial model)

  - RDY indicator: Remains lit when the internal circuitry is in normal operation.
  - ERR1 indicator: Lights up if a given error occurs.
  - 40-pin connector: Connects to external I/O devices such as servo motors and limit switches.

- F3NC52-0N module (biaxial model)

  - RDY indicator: Remains lit when the internal circuitry is in normal operation.
  - ERR1 indicator: Lights up if an error occurs in axis 1.
  - ERR2 indicator: Lights up if an error occurs in axis 2.
  - Axis 1 connector: Connects to external I/O devices such as servo motors and limit switches.
2.6 External Dimensions

Unit: mm
## 2.7 Terminal Assignments and Connection

<table>
<thead>
<tr>
<th>Pin No.</th>
<th>Signal Name</th>
<th>Electrical Specifications</th>
</tr>
</thead>
<tbody>
<tr>
<td>20b</td>
<td>Emergency-stop input*</td>
<td>24V DC, 4.1mA</td>
</tr>
<tr>
<td>19b</td>
<td>External contact input 6*</td>
<td>24V DC, 4.1mA</td>
</tr>
<tr>
<td>19a</td>
<td>External contact input 5 (External trigger)*</td>
<td>24V DC, 4.1mA</td>
</tr>
<tr>
<td>18b</td>
<td>External contact input 4 (Origin)*</td>
<td>24V DC, 4.1mA</td>
</tr>
<tr>
<td>18a</td>
<td>External contact input 3 (Driver alarm)*</td>
<td>24V DC, 4.1mA</td>
</tr>
<tr>
<td>17b</td>
<td>External contact input 2</td>
<td>24V DC, 4.1mA</td>
</tr>
<tr>
<td>17a</td>
<td>External contact input 1</td>
<td>24V DC, 4.1mA</td>
</tr>
<tr>
<td>16b</td>
<td>External contact input (COM)</td>
<td></td>
</tr>
<tr>
<td>16a</td>
<td>Voltage output for speed commands</td>
<td>-10 to 10V, 5mA DC</td>
</tr>
<tr>
<td>15b</td>
<td>Voltage output for speed commands (GND)</td>
<td></td>
</tr>
<tr>
<td>14b</td>
<td>Shield (FG)</td>
<td></td>
</tr>
<tr>
<td>14a</td>
<td>Shield (FG)</td>
<td></td>
</tr>
<tr>
<td>13b</td>
<td>Z-phase encoder input-Z</td>
<td>RS422A-based differential signal terminated with a 220-Ω resistor</td>
</tr>
<tr>
<td>13a</td>
<td>Z-phase encoder input+Z</td>
<td></td>
</tr>
<tr>
<td>12b</td>
<td>B-phase encoder input-B</td>
<td>RS422A-based differential signal terminated with a 220-Ω resistor</td>
</tr>
<tr>
<td>12a</td>
<td>B-phase encoder input+B</td>
<td></td>
</tr>
<tr>
<td>11b</td>
<td>A-phase encoder input-A</td>
<td>RS422A-based differential signal terminated with a 220-Ω resistor</td>
</tr>
<tr>
<td>11a</td>
<td>A-phase encoder input+A</td>
<td></td>
</tr>
<tr>
<td>10b</td>
<td>Encoder/manual pulser signal ground</td>
<td></td>
</tr>
<tr>
<td>10a</td>
<td>Encoder/manual pulser signal ground</td>
<td></td>
</tr>
<tr>
<td>9b</td>
<td>SEN (for YASKAWA Electric’s absolute encoders)</td>
<td>5V DC, 16mA</td>
</tr>
<tr>
<td>9a</td>
<td>SEN_DV (GND)</td>
<td></td>
</tr>
<tr>
<td>8b</td>
<td>reserved</td>
<td></td>
</tr>
<tr>
<td>8a</td>
<td>reserved</td>
<td></td>
</tr>
<tr>
<td>7b</td>
<td>B-phase manual pulser input-B*</td>
<td>RS422A-based differential signal terminated with a 220-Ω resistor</td>
</tr>
<tr>
<td>7a</td>
<td>B-phase manual pulser input+B*</td>
<td></td>
</tr>
<tr>
<td>6b</td>
<td>A-phase manual pulser input-A*</td>
<td>RS422A-based differential signal terminated with a 220-Ω resistor</td>
</tr>
<tr>
<td>6a</td>
<td>A-phase manual pulser input-A*</td>
<td></td>
</tr>
<tr>
<td>5b</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5a</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4b</td>
<td>External contact output (Brake OFF)</td>
<td>24V DC, 5.1A</td>
</tr>
<tr>
<td>3b</td>
<td>External contact output (Driver reset)</td>
<td>24V DC, 5.1A</td>
</tr>
<tr>
<td>3a</td>
<td>External contact output (Servo ON)</td>
<td>24V DC, 5.1A</td>
</tr>
<tr>
<td>2b</td>
<td>External contact output (COM)</td>
<td></td>
</tr>
<tr>
<td>2a</td>
<td>External contact output (24 V)</td>
<td></td>
</tr>
<tr>
<td>1b</td>
<td>Power supply input for external contact output (0 V-IN)</td>
<td>24V DC, 0.1A</td>
</tr>
<tr>
<td>1a</td>
<td>Power supply input for external contact output (24 V-IN)</td>
<td>24V DC, 0.1A</td>
</tr>
</tbody>
</table>

*1 The emergency-stop input is for use with the axis-1 connector only; the axis-2 connector is not internally wired for this input. Emergency-stop signals to the axis-1 connector are, therefore, shared by the two axes.

*2 External contact inputs 1 to 6 can also be used as general-purpose inputs if configured accordingly.

*3 The manual pulser input is for use with the axis-1 connector only; the axis-2 connector is not internally wired for this input. Manual pulser signals to the axis-1 connector are, therefore, shared by the two axes.
2.8 Applicable External Interface Connectors

<table>
<thead>
<tr>
<th>Connection</th>
<th>Applicable Connector</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Soldered</td>
<td>FCN-361J040-AU connector and FCN-360C040-B connector cover (Fujitsu Limited)</td>
<td>Purchase the desired connector kit separately when ordering the module.</td>
</tr>
<tr>
<td>Crimp-on</td>
<td>FCN-363J040 housing, FCN-363J-AU contacts and FCN-360C040-B connector cover (Fujitsu Limited)</td>
<td></td>
</tr>
<tr>
<td>Pressure-welded</td>
<td>FCN-367J040-AU or -F connector (Fujitsu Limited)</td>
<td></td>
</tr>
</tbody>
</table>
3. Function Overview

This chapter explains the major functions of the positioning modules. For details on how to use each function, see Chapter 7. Table 3.1 summarizes the functions discussed in this chapter.

<table>
<thead>
<tr>
<th>Major Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Positioning operation</td>
<td>Carries out positioning in the normal position-control mode.</td>
</tr>
<tr>
<td>Change in target position during</td>
<td>Changes the target position while positioning is in progress.</td>
</tr>
<tr>
<td>positioning</td>
<td></td>
</tr>
<tr>
<td>Change in velocity during</td>
<td>Changes the velocity of rotation while positioning is in progress.</td>
</tr>
<tr>
<td>positioning</td>
<td></td>
</tr>
<tr>
<td>Velocity control</td>
<td>A function that works in the velocity-control mode. This function keeps the</td>
</tr>
<tr>
<td></td>
<td>motor rotating in the same direction.</td>
</tr>
<tr>
<td>Change in velocity during</td>
<td>Changes the velocity of rotation while velocity control is in progress.</td>
</tr>
<tr>
<td>velocity control</td>
<td></td>
</tr>
<tr>
<td>Velocity-to-position control mode</td>
<td>Switches to position control while velocity control is in progress.</td>
</tr>
<tr>
<td>switching</td>
<td></td>
</tr>
<tr>
<td>Jog stepping</td>
<td>Allows a motor to be rotated manually when, for example, teaching the</td>
</tr>
<tr>
<td></td>
<td>position data.</td>
</tr>
<tr>
<td>Emergency-stop input</td>
<td>Brings a motor to an immediate stop using an external contact input.</td>
</tr>
<tr>
<td>External contact inputs</td>
<td>Accept such external contact signals as limit-switch signals or ORIGIN</td>
</tr>
<tr>
<td></td>
<td>contact signals.</td>
</tr>
<tr>
<td>External contact outputs</td>
<td>Deliver the Servo ON, Brake OFF, and Driver reset signals.</td>
</tr>
<tr>
<td>SEN signal output</td>
<td>Requests absolute-value data to be sent when YASKAWA Electric's absolute</td>
</tr>
<tr>
<td></td>
<td>encoder is in use.</td>
</tr>
<tr>
<td>Origin-search operation</td>
<td>Searches for the origin using an external contact input.</td>
</tr>
<tr>
<td>Interrupt function</td>
<td>Asks the CPU module for permission to interrupt when for example,</td>
</tr>
<tr>
<td></td>
<td>positioning is complete.</td>
</tr>
<tr>
<td>Manual pulser mode</td>
<td>Enables a motor to be operated manually using a manual pulser.</td>
</tr>
<tr>
<td>Linear-interpolated operation</td>
<td>Carries out multi-axial, linear-interpolated operation.</td>
</tr>
<tr>
<td>On-route operation</td>
<td>Carries out on-route operation (path operation) in which the tracking</td>
</tr>
<tr>
<td></td>
<td>path under control passes by the vicinity of a given target position.</td>
</tr>
<tr>
<td>Arc-interpolated operation</td>
<td>Carries out biaxial, arc-interpolated operation.</td>
</tr>
</tbody>
</table>
3.1 Positioning Operation

To initiate the positioning operation:
- first write the target velocity, target position, acceleration time, deceleration time and other necessary parameters from the CPU module, and then
- change the state of the output relay labeled “Start Operation Command” from OFF to ON.

When the positioning operation is complete, the input relay labeled “End of Positioning” changes to the ON state. As the target position, you can specify either an absolute position, a relative position based on the encoder position, or a relative position based on the previous target position.

For the acceleration/deceleration curve, you can select a trapezoidal, two-line-segment or S-shape (three-line-segment) trace, where the acceleration/deceleration times are set separately. Specify the end-of-positioning judgment region and the time-out interval of positioning to determine whether or not positioning is complete.

In addition to the normal startup (where operation begins immediately after startup), you can set such a startup mode in which the module waits for an internal/external trigger before it begins the positioning operation.

![Diagram of velocity and acceleration/deceleration times in trapezoidal and trigonometric drives](image)

**Figure 3.1** Velocity and Acceleration/Deceleration Times in Trapezoidal and Trigonometric Drives

![Diagram of determination of end of positioning](image)

**Figure 3.2** Determination of End of Positioning
3.2 Change in Target Position during Positioning

To make the module change the target position:
- first write the parameters necessary for another positioning operation, and then
- change the state of the output relay labeled “Request to Change Target Position” from OFF to ON while the positioning operation is in progress.

The speed of rotation can also be changed at the same time you change the target position. The module can also perform such change-in-target-position operations when the direction in which the motor rotates changes. (In such an application, the module quickly slows the motor to a complete stop and begins at the positioning operation to find the new target position.)

Figure 3.3 Behavior When the Target Position Is Changed
3.3 Change in Velocity during Positioning

The module will perform the change-in-velocity operation when you:
- write the new target velocity and other necessary parameters, and then
- change the state of the output relay labeled “Request to Change Velocity” from OFF to ON while the positioning operation is in progress.

Figure 3.4 Behavior When the Velocity Is Changed
3.4 Velocity Control

To initiate the velocity-control operation:
- first write the target velocity (value will have a minus sign if rotating the motor in the negative direction), acceleration time, deceleration time and other necessary parameters from the CPU module, and then
- change the state of the output relay labeled “Start Operation Command” from OFF to ON.

The velocity-control operation continues until the output relay labeled “Request to Decelerate and Stop” or “Request to Stop Immediately” is turned on.

For the acceleration/deceleration curve, you can select a trapezoidal, two-line-segment or S-shape (three-line-segment) trace, where the acceleration/deceleration times are set separately. Specify the end-of-positioning judgment region and the time-out interval of positioning to determine whether or not positioning is complete.

In addition to the normal startup (where operation begins immediately after startup), you can set such a startup mode in which the module waits for an internal/external trigger before it begins the positioning operation.

Velocity-control operation is not practicable if an absolute encoder is used.
3.5 Change in Velocity during Velocity Control

To make the module change the velocity during velocity control:
- first write the new target velocity, and then
- change the state of the output relay labeled “Request to Change Velocity” from OFF to ON while the velocity-control operation is in progress.

The change-in-velocity operation does not allow you to change the velocity in such a manner that the direction in which the motor rotates changes. To change the direction, decelerate the motor until it reaches a stop and then initiate the velocity-control operation after setting a new target velocity.

![Figure 3.5 Velocity Control and Change-in-Velocity Operation](image)
3.6 Velocity-to-Position Control Mode Switching

The module switches to position control while it is in the velocity-control operation and then enters the positioning operation with the position at which the switch was made set as position "0." This action takes place when you:

- first write the target velocity, target position, acceleration time, deceleration time and other necessary parameters from the CPU module, and then

- change the control mode from velocity to position control while the velocity-control operation is in progress.

For the acceleration/deceleration curve, you can select a trapezoidal, two-line-segment or S-shape (three-line-segment) trace, where the acceleration/deceleration times are set separately. Specify the end-of-positioning judgment region and the time-out interval of positioning to determine whether or not positioning is complete.

In addition to the normal switching (where switching takes place immediately after the given command is executed), you can set a switching mode such that the module waits for an external trigger before it switches to position control. To switch to position control after detecting a Z-phase input signal, specify the polarity of the Z-phase, as well as the frequency of Z-phase pulse counting.

• Cases where no Z-phase pulse counting is specified

![Diagram](image1)

• Cases where Z-phase pulse counting is specified (twice during the rise time)

![Diagram](image2)

Figure 3.6 Behavior When Switching from Velocity to Position Control
3.7 Jog Stepping

To carry out jog stepping:
- first write the target velocity, acceleration time, deceleration time and other necessary parameters from the CPU module, and then
- change the state of the output relay labeled “Positive-direction Jog Stepping” or “Negative-direction Jog Stepping” from OFF to ON.

To quit the jog-stepping operation, change the ON-state output relay to an OFF state.

Figure 3.7 Jog-stepping Operation (Positive Direction)
3.8 Emergency-stop Input

The positioning module has one emergency-stop input common to the uniaxial and biaxial models. This input is designed for exclusive use as a type-b contact input. Be SURE to wire the input when using the module. If the input is left open, the module does not operate at all.
3.9  Contact Inputs

The positioning module has six external contact inputs for both the uniaxial and biaxial models. Define the function of each contact in the contact input mode. You can read the state of each contact input using an application program irrespective of their functional definitions. In addition, you can set the polarity of each contact input separately.
3.10 Contact Outputs

The positioning module has three contact outputs labeled “Servo ON,” “Brake OFF,” and “Driver reset,” for both the uniaxial and biaxial models. Each output is enabled by its corresponding command. You can set the polarity of each contact output separately. In addition, you can read the state of each contact output using an application program.
3.11 SEN Signal Output

The SEN signal output is used only when YASKAWA Electric’s serial absolute encoders or drivers equivalent to the encoders are connected to the positioning module. The output is not allowed for any other purposes. When using any other type of driver, leave this output open.
3.12 Origin-search Operation

To start the origin search:
- first write the direction of the search, the search speed, the type of operation when an external contact input is detected (origin-search mode), the direction of the edge for detecting the Z-phase, and other necessary parameters, and then
- change the state of the output relay labeled “Origin Search” from OFF to ON.

If the positioning module detects a change in the state of a preset external contact input after the start of the origin-search operation, the module either stops the motor or checks the Z-phase, depending on the setting of the contact input. When checking the Z-phase, the module detects the preset Z-phase pulse count, and then slows down and stops the motor. The module regards the position where an input defining an origin is detected as being the origin (position “0”), and then decelerates the motor into a complete stop. The position where the motor actually stops is as far from the origin as the distance the motor has traveled during the deceleration time.

There is an application where an origin search is carried out at two different speeds or a change is made to the direction of rotation while checking for an external contact input during the origin search. In such a case, split the origin search process into several cycles while varying the parameters for each cycle, and then do the search. This strategy enables you to customize your origin-search operation to follow your desired search patterns.

### Origin-search Operation in Absolute Encoder System

You can also perform the origin-search operation in absolute encoder systems using the same YASKAWA Electric’s method used in incremental encoder systems. In this case however, the positions (such as the current and command positions) given upon completion of the origin search are irrelevant to the absolute value in question. Even so, the module continues to retain absolute-value data in the Absolute-value Status memory area.

No origin-search operation can be performed in absolute encoder systems using Sanyo Denki’s method.
3.13 Interrupt Function

The positioning module can generate an interruption to the CPU module for the two cases noted below. For details on interruption handling, refer to the Sequence CPU – Instructions manual (publication number IM 34M6P12-03E) or the BASIC Programming Tool M3 (publication number IM34M6Q22-01E).

(1) Position-detecting Interruption
If you initiate the positioning operation after configuring the module for position detection, you can interrupt the CPU module when the command position or encoder position reaches the given setpoint (when the input relay defined as “Notice of Position Detection” changes to the ON state). The delay in detecting the encoder position is 1 ms maximum.

(2) End-of-Positioning Interruption
You can interrupt the CPU module when positioning is complete (when the input relay labeled “End of Positioning” changes to the ON state).
3.14 Manual Pulser Mode

If you turn on the manual pulser mode, you can operate the motor using the manual pulser. Define the relationship of the number of pulses input from the pulser and the amount of motor travel using the pulser’s scale values. Although the manual pulser input is common to both axis 1 and axis 2, set the manual pulser mode separately for axis 1 and axis 2. If you set both axes in the manual pulser mode, you can put the two axes into operation at the same time with just one manual pulser input. When the positioning module is in the manual pulser mode, it is not possible to operate the motor by issuing commands from the CPU module.
3.15 Linear-interpolated Operation

To carry out linear-interpolated operation:
- first write the target velocity, target position, acceleration time, deceleration time and other necessary parameters for each axis from the CPU module, and then
- change the state of the output relay labeled “Start Operation Command” from OFF to ON for both axes simultaneously.

When the positioning operation for each axis is complete, the input relay labeled “End of Positioning” for each axis changes to the ON state. In this operation, set the same acceleration and deceleration times for both axes that are brought into linear-interpolated operation. Calculate and set the ratio between the target velocities of the two axes so that it equals the ratio between the travels of the two axes.

![Diagram showing linear-interpolated operation](image)

Figure 3.8 Multiaxial Linear-interpolated Operation (Example of Biaxial Application)
3.16 On-route Operation

If you initiate another positioning operation while the current positioning operation is in progress, the positioning module begins the new operation before the current operation ends. The module therefore carries out a combination of these two operations until the current operation ends. This mode of the positioning operation is referred to as the on-route operation. The interval at which the two operations overlap is called the on-route interval. Using the on-route operation, you can continue your position finding toward the new target position without stopping at the target position set for the current positioning operation. It is also possible to define the mode of on-route operation where the direction of rotation may be changed.

- Normal positioning operation

- On-route operation

- Example of on-route operation in biaxial linear-interpolated operation

Figure 3.9 Normal Positioning Operation and On-route Operation
### 3.17 Arc-interpolated Operation

Biaxial arc-interpolated operation can be implemented by converting a positioning command coming from the CPU module into a trigonometric function within the positioning module. To bring the positioning module into an arc-interpolated operation:

- first define the center of the X-Y plane, the radius, the starting angle, the angular travel and other necessary parameters from the CPU module, and then

- change the state of the output relay defined as “Start Operation Command” from OFF to ON.

![Figure 3.10  Arc-interpolated Operation](image)
4. Parameters

4.1 List of Parameters

Among the parameters listed in Tables 4.2 through 4.7, the ones with two data position numbers are 2-word data. The data with the smaller number is the low-order word, and the one with the larger number is the high-order word.

Data position numbers are specified for each word. The WRITE and READ instructions used for accessing from a sequence program must be on a word basis. Long-word based instructions cause inappropriate access. You should also use word-based instructions when you access from BASIC programs.

Data whose setting units are [(1/65536) pulse/ms], [(1/65536) pulses/ms/ms], [(1/65536) degrees], or [(1/65536) degrees/ms] (*1 data) are fixed-point data with 1 word integer part (16 bits) and 1 word decimal part (16 bits). The data with the smaller number is the low-order word, and the one with the larger number is the high-order word.

### Fixed-point data

The digits in the integer part of the binary data are sequentially defined as 1, 2, 4 ... , and the digits in the decimal part are defined as 1/2, 1/4, 1/8, and so forth. If both the integer part and the decimal part consist of 16 bits, the least significant bit is 1/65536, which means that it is a 32-bit (long-word) data whose setting unit is 1/65536. Negative numbers are expressed as complements of 2 like ordinary binary data.

<table>
<thead>
<tr>
<th>Bit</th>
<th>31(MSB)</th>
<th>30</th>
<th>...</th>
<th>17</th>
<th>16</th>
<th>15</th>
<th>14</th>
<th>...</th>
<th>1</th>
<th>0(LSB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>Sign bit</td>
<td>16384</td>
<td>...</td>
<td>2</td>
<td>1</td>
<td>1/2</td>
<td>1/4</td>
<td>...</td>
<td>1/32768</td>
<td>1/65536</td>
</tr>
<tr>
<td></td>
<td></td>
<td>High-order word</td>
<td>Low-order word</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(Example of fixed-point data)

When setting 123.45 [pulse/ms] (=123450 [pulses/s]),

\[123.45 \times 65536 = 8090419.2 \ \text{[(1/65536) pulse/ms]}\]

Thus, we should set 8090419 as a long word data. The high-order word of this data is 123, (8090419 divided by 65536). The low-order word is the remainder, i.e., 29491.
An example of sequence programs which convert data in pulses/s into data to be set for a positioning module

Let D0001 (long-word data) be the original data (pulses/s).

1. Divide D0001 by 1000 (long-word division) and put the result into D0011. In this case, since the maximum value of D0001 is 2000000 (2 Mpps) and it is positive, the maximum value of the result is 2000 and thus the high-order word (D0012) is always 0. The low-order word of the result of the division (D0011) is the high-order word (the integer part which is 16 bits long) of the value [(1/65536) pulse/ms] to be calculated.

   The remainder is put into D0013 (the low-order word) and D0014 (the high-order word). Since the maximum value of the remainder is 999 because the divisor is 1000, the high-order word of the remainder (D0014) is always 0.

2. Then, multiply the remainder by 65536 and divide it again by 1000. A useful trick here is this: the remainder is in D0013 and D0012 is 0; thus, if we treat D0012 as a long-word data, its value is already the result of multiplication of the remainder by 6556. Therefore, in order to divide the result of multiplication of the remainder by 65536 by 1000, all we have to do is divide D0012 by 1000 (long-word division). Put the result of this division into D0021.

   D0012 (long word) is 999*65536 at a maximum, and it is divided by 1000, than 65470 is obtained at a maximum. High-order word (D0022) becomes always 0. Thus D0021 is the value [(1/65536) pulses/ms] of low-order word (the decimal part of 16 bits) and the remainder will be discarded as truncation.

3. Now, combine the results D0011 and D0021 into a long-word data [(1/65536) pulses/ms]. To do this, we only have to do long-word division twice and transfer the resulting high-order word and low-order word to D0032 and D0031, respectively. D0011-D0014 and D0021-D0024 are work areas.

   (1) Long-word division

<table>
<thead>
<tr>
<th>D0011</th>
<th>=</th>
<th>D0001</th>
<th>/</th>
<th>1000</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>D0011</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

   Operation results

   D0014  D0013  D0012  D0011
   0      Remainder 0  High-order of [(1/65536)] pulses/ms

   (2) Long-word division

<table>
<thead>
<tr>
<th>D0021</th>
<th>=</th>
<th>D0012</th>
<th>/</th>
<th>1000</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>D0021</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

   Operation results

   D0024  D0023  D0022  D0021
   0      Remainder 0  Low-order of [(1/65536)] pulses/ms
(3) Word Transfer (twice)

```
MOV D0011 D0032
```

```
MOV D0021 D0031
```

<table>
<thead>
<tr>
<th>D0032</th>
<th>D0031</th>
</tr>
</thead>
<tbody>
<tr>
<td>High-order of [(1/65536)] pulses/ms</td>
<td>Low-order of [(1/65536)] pulses/ms</td>
</tr>
</tbody>
</table>

In the case of 123450 [pulses/s]

(1) D0011 = 123450/1000 (long word division)

<table>
<thead>
<tr>
<th>D0014</th>
<th>D0013</th>
<th>D0012</th>
<th>D0011</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>450</td>
<td>0</td>
<td>123</td>
</tr>
</tbody>
</table>

29491200 (450*65536)

(2) D0021 = 29491200/1000 (long word division)

<table>
<thead>
<tr>
<th>D0024</th>
<th>D0023</th>
<th>D0022</th>
<th>D0021</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>200</td>
<td>0</td>
<td>29491</td>
</tr>
</tbody>
</table>

The high-order word of the [1/65536] pulses/ms data is 123, and the low-order word is 29491.
4.1.1 Entry Parameters

Entry parameters are usually set only once after turning the power on. You can set them by writing from the CPU module and then executing the Parameter Set command.

Table 4.2 Entry parameters

<table>
<thead>
<tr>
<th>Data position number</th>
<th>Parameter name</th>
<th>Initial value</th>
<th>Range of values</th>
</tr>
</thead>
<tbody>
<tr>
<td>001/002 201/202</td>
<td>Positive-direction limit value</td>
<td>134217727</td>
<td>-134217728 to 134217727 [pulses]</td>
</tr>
<tr>
<td>003/004 203/204</td>
<td>Negative-direction limit value</td>
<td>-134217728</td>
<td>-134217728 to (positive-direction limit value-1) [pulses]</td>
</tr>
<tr>
<td>005/006 205/206</td>
<td>Velocity limit value</td>
<td>131072000</td>
<td>1 to 131072000 [(1/65535) pulses/ms] *1</td>
</tr>
<tr>
<td>007/008 207/208</td>
<td>Over-limit velocity detection</td>
<td>131072000</td>
<td>1 to 131072000 [(1/65536) pulses/ms] *1</td>
</tr>
<tr>
<td>009/010 209/210</td>
<td>Over-limit acceleration detection</td>
<td>131072000</td>
<td>1 to 131072000 [(1/65536) pulses/ms/ms] *1</td>
</tr>
<tr>
<td>011/012 211/212</td>
<td>Deviation error detection</td>
<td>0134217727</td>
<td>1 to 134217727 [pulses]</td>
</tr>
</tbody>
</table>
| 013 213              | Motor rotation direction               | 0             | 0: positive-direction with positive velocity command voltage  
|                      |                                       |               | 1: negative direction with positive velocity command voltage |
| 014 214              | Contact input polarity                | 0             | Specified in bits for each point                      |
| 015 215              | Contact output polarity                | 0             | Specified in bits for each point                      |
| 016 216              | Position loop gain                    | 200           | 1 to 10000 [0.01Hz]                                  |
| 017 217              | Velocity feedforward factor            | 9500          | 0 to 20000 [1/10000]                                 |
| 018 218              | Specification of encoder              | 0             | 0: General-purpose incremental method  
|                      |                                       |               | 1: Sanyo Denki's Manchester coding absolute  
|                      |                                       |               | 2: YASKAWA Electric's serial                        |
| 019 219              | Encoder multiplication number         | 4             | 1: 1 multiplication  2: 2 multiplication  4: 4 multiplication |
| 020/021 220/221      | Velocity/voltage ratio                | 10240         | 1 to 2000000 [pps/V]                                 |
| 022 222              | (reserved)                            | 0             | 0: Fixed (unchangeable)                              |
| 023/024 223/224      | Absolute offset value                 | 0             | -134217728 to 134217727 [pulses]                     |
| 025/026 225/226      | Absolute pulse count                  | 32768         | 1 to 134217727 [pulses]                              |
| 027 227              | Sanyo Denki's ABS bit length          | 28            | 17 to 28                                             |
| 028 228              | Sanyo Denki's ABS rotation direction  | 0             | 0: Encoder data increase in positive direction  
|                      |                                       |               | 1: Encoder data increase in negative direction       |
| 029 229              | Contact input mode                    | 0             | 0: Normal mode with limits  
|                      |                                       |               | 1: Normal mode without limits  
|                      |                                       |               | 2: Alarm detail mode with limits  
|                      |                                       |               | 3: Alarm detail mode without limits                   |
| 030 230              | (reserved)                            | 0             | 0: Fixed (unchangeable)                              |
| 031 231              | (reserved)                            | 9000          | 9000: Fixed (unchangeable)                           |
| 032 232              | (reserved)                            | 10000         | 10000: Fixed (unchangeable)                          |
4.1.2 Startup Parameters

Startup parameters are referred to when executing commands related to positioning. These parameters do not have initial values. It is necessary to write all the required parameters when executing the commands.

Table 4.3 Startup Parameters

<table>
<thead>
<tr>
<th>Data position number</th>
<th>Parameter name</th>
<th>Range of values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Axis 1</td>
<td>Axis 2</td>
<td></td>
</tr>
<tr>
<td>041/042</td>
<td>Target velocity</td>
<td>1 to 131072000 [(1/65536) pulses/ms] *1 (Position control mode) -131072000 to 131072000 [(1/65536) pulses/ms] *1 (Velocity control mode)</td>
</tr>
<tr>
<td>043/044</td>
<td>Target position</td>
<td>-134217728 to 134217727 [pulses]</td>
</tr>
<tr>
<td>045</td>
<td>Target position mode</td>
<td>0: Absolute position 1: Relative position from the encoder position 2: Relative position from the previous target position</td>
</tr>
<tr>
<td>046</td>
<td>Acceleration time</td>
<td>0 to 32767 [ms]</td>
</tr>
<tr>
<td>047</td>
<td>Acceleration mode selection</td>
<td>Refer to &quot;Description of parameters.&quot;</td>
</tr>
<tr>
<td>048</td>
<td>Acceleration parameter 1</td>
<td>1 to 99 [%]</td>
</tr>
<tr>
<td>049</td>
<td>Acceleration parameter 2</td>
<td>1 to 99 [%]</td>
</tr>
<tr>
<td>050</td>
<td>Deceleration time</td>
<td>0 to 32767 [ms]</td>
</tr>
<tr>
<td>051</td>
<td>Deceleration mode selection</td>
<td>Refer to &quot;Description of parameters.&quot;</td>
</tr>
<tr>
<td>052</td>
<td>Deceleration parameter 1</td>
<td>1 to 99 [%]</td>
</tr>
<tr>
<td>053</td>
<td>Deceleration parameter 2</td>
<td>1 to 99 [%]</td>
</tr>
<tr>
<td>054/055</td>
<td>Positioning judgement range</td>
<td>0 to 134217727 [pulses]</td>
</tr>
<tr>
<td>056</td>
<td>Positioning time-out interval</td>
<td>0 to 32767 [ms]</td>
</tr>
<tr>
<td>057</td>
<td>Interpolation mode selection</td>
<td>0: Normal operation 1: Arc interpolation X axis 2: Arc interpolation Y axis</td>
</tr>
<tr>
<td>058</td>
<td>Startup mode selection</td>
<td>0: Normal startup 1: Wait for external trigger 2: Wait for internal trigger</td>
</tr>
<tr>
<td>059</td>
<td>Position detection mode</td>
<td>0: No detection 1: Command value 2: Encoder position</td>
</tr>
<tr>
<td>060/061</td>
<td>Position detection setpoint</td>
<td>-134217728 to 134217727 [pulses]</td>
</tr>
</tbody>
</table>

4.1.3 Startup Parameters Related to Origin Search

These parameters are referred to when executing commands related to origin search. These parameters do not have initial values. It is necessary to write all the required parameters when executing the commands.

Table 4.4 Startup Parameters Related to Origin Search

<table>
<thead>
<tr>
<th>Data position number</th>
<th>Parameter name</th>
<th>Range of values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Axis 1</td>
<td>Axis 2</td>
<td></td>
</tr>
<tr>
<td>062</td>
<td>Origin search mode</td>
<td>See to Sections 7.1.5 and 7.2.5 for details.</td>
</tr>
<tr>
<td>063</td>
<td>Origin search direction</td>
<td>0: Positive direction 1: Negative direction</td>
</tr>
<tr>
<td>064</td>
<td>Z-phase edge selection</td>
<td>0: OFF → ON edge 1: ON → OFF edge</td>
</tr>
<tr>
<td>065</td>
<td>Z-phase pulse count</td>
<td>0 to 32767 [times]</td>
</tr>
<tr>
<td>066/067</td>
<td>Z-phase search range</td>
<td>0 to 134217727 [pulses]</td>
</tr>
<tr>
<td>068/069</td>
<td>Origin offset value</td>
<td>-134217728 to 134217727 [pulses]</td>
</tr>
</tbody>
</table>
4.1.4 Extended Command Parameters

These parameters are referred to when executing extended commands. These parameters do not have initial values. It is necessary to write all the required parameters when executing the commands.

Table 4.5 Extended Command Parameters

<table>
<thead>
<tr>
<th>Data position number</th>
<th>Parameter name</th>
<th>Range of values</th>
</tr>
</thead>
<tbody>
<tr>
<td>072/073</td>
<td>Steady-state deviation adjust amount</td>
<td>-32768 to 32767 [pulses]</td>
</tr>
<tr>
<td>074</td>
<td>274 Manual pulser scale value</td>
<td>-500 to 500 [times]</td>
</tr>
</tbody>
</table>

4.1.5 Control Mode Switching Parameters

These parameters are referred to when executing the Control Mode Switching command. Since it does not have an initial value, you have to write it before executing the command.

Table 4.6 Control Mode Switching Parameter

<table>
<thead>
<tr>
<th>Data position number</th>
<th>Parameter name</th>
<th>Range of values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Axis 1 081</td>
<td>281 Control Mode Switching parameter</td>
<td>0: Switch to velocity control mode 1: Switch to position control mode 2: Switch to position control mode (wait for Z-phase)</td>
</tr>
</tbody>
</table>
4.1.6 Arc-Interpolation Parameters

These parameters are referred to when executing arc-interpolation commands. These parameters do not have initial values. It is necessary to write all the required parameters when executing the commands.

Table 4.7 Arc-interpolation Parameter

<table>
<thead>
<tr>
<th>Data position number</th>
<th>Parameter name</th>
<th>Range of values</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Axis 1</td>
<td>Axis 2</td>
</tr>
<tr>
<td>082/083</td>
<td>282/283</td>
<td>Center position</td>
</tr>
<tr>
<td>084/085</td>
<td>284/285</td>
<td>Radius</td>
</tr>
<tr>
<td>086/087</td>
<td>286/287</td>
<td>Starting angle</td>
</tr>
<tr>
<td>088/089</td>
<td>288/289</td>
<td>Angular travel</td>
</tr>
<tr>
<td>090/091</td>
<td>290/291</td>
<td>Angular target velocity</td>
</tr>
<tr>
<td>092</td>
<td>292</td>
<td>Acceleration time</td>
</tr>
<tr>
<td>093</td>
<td>293</td>
<td>Deceleration time</td>
</tr>
<tr>
<td>094/095</td>
<td>294/295</td>
<td>Target Position</td>
</tr>
<tr>
<td>096</td>
<td>296</td>
<td>Correction Pulse Range</td>
</tr>
</tbody>
</table>
4.2 List of Parameters for Each Command

You need to write parameters before executing commands for the positioning module from the CPU module. The following table shows a list of parameters for each command. The Set Parameter command is not included in the table, for it changes all entry parameters at once.
Table 4.8 List of Required Parameters for Each Command

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Command Name (Output Relay Name)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Start Operation command</td>
</tr>
<tr>
<td></td>
<td>(positioning control mode)</td>
</tr>
<tr>
<td></td>
<td>Start Operation command</td>
</tr>
<tr>
<td></td>
<td>(arc-interpolated operation)</td>
</tr>
<tr>
<td></td>
<td>Request to Switch Control mode</td>
</tr>
<tr>
<td></td>
<td>(positioning control)</td>
</tr>
<tr>
<td></td>
<td>Request to Change Tgt. Position</td>
</tr>
<tr>
<td></td>
<td>Request to Change velocity</td>
</tr>
<tr>
<td></td>
<td>(during position control)</td>
</tr>
<tr>
<td></td>
<td>Positive-Direction JOG stepping</td>
</tr>
<tr>
<td></td>
<td>Negative-Direction JOG stepping</td>
</tr>
<tr>
<td></td>
<td>Write Current Position</td>
</tr>
<tr>
<td></td>
<td>Request to Stop immediately</td>
</tr>
<tr>
<td></td>
<td>Positional Position</td>
</tr>
<tr>
<td></td>
<td>Internal Trigger</td>
</tr>
<tr>
<td>Target Velocity</td>
<td>●</td>
</tr>
<tr>
<td>Target Position</td>
<td>●</td>
</tr>
<tr>
<td>Target Position Mode</td>
<td>●</td>
</tr>
<tr>
<td>Acceleration Time</td>
<td>●</td>
</tr>
<tr>
<td>Select Acceleration Mode</td>
<td>●</td>
</tr>
<tr>
<td>Acceleration Parameter 1</td>
<td>●</td>
</tr>
<tr>
<td>Acceleration Parameter 2</td>
<td>●</td>
</tr>
<tr>
<td>Deceleration Time</td>
<td>●</td>
</tr>
<tr>
<td>Select Deceleration Mode</td>
<td>●</td>
</tr>
<tr>
<td>Deceleration Parameter 1</td>
<td>●</td>
</tr>
<tr>
<td>Deceleration Parameter 2</td>
<td>●</td>
</tr>
<tr>
<td>Positioning Judgement Range</td>
<td>●</td>
</tr>
<tr>
<td>Positioning Time-out Interval</td>
<td>●</td>
</tr>
<tr>
<td>Select Interpolation Mode</td>
<td>●</td>
</tr>
<tr>
<td>Select Startup Mode</td>
<td>●</td>
</tr>
<tr>
<td>Positioning Detection Mode</td>
<td>●</td>
</tr>
<tr>
<td>Positioning Detection Setpoint</td>
<td>●</td>
</tr>
<tr>
<td>Origin Search Mode</td>
<td>●</td>
</tr>
<tr>
<td>Origin Search Direction</td>
<td>●</td>
</tr>
<tr>
<td>Z-phase Edge Selection</td>
<td>●</td>
</tr>
<tr>
<td>Z-phase Pulse Count</td>
<td>●</td>
</tr>
<tr>
<td>Z-phase Search Range</td>
<td>●</td>
</tr>
<tr>
<td>Origin Offset Value</td>
<td>●</td>
</tr>
<tr>
<td>Extended Command Parameter</td>
<td>●</td>
</tr>
<tr>
<td>Constant Deviation Adjust Amt.</td>
<td>●</td>
</tr>
<tr>
<td>Manual Pulser Scale Value</td>
<td>●</td>
</tr>
<tr>
<td>Control Mode Switching P'meter</td>
<td>●</td>
</tr>
<tr>
<td>Center Position</td>
<td>●</td>
</tr>
<tr>
<td>Radius</td>
<td>●</td>
</tr>
<tr>
<td>Starting Angle</td>
<td>●</td>
</tr>
<tr>
<td>Angular Travel</td>
<td>●</td>
</tr>
<tr>
<td>Angular Target Velocity</td>
<td>●</td>
</tr>
<tr>
<td>Acceleration Time</td>
<td>●</td>
</tr>
<tr>
<td>Deceleration Time</td>
<td>●</td>
</tr>
<tr>
<td>Target Position</td>
<td>●</td>
</tr>
<tr>
<td>Correction Pulse Range</td>
<td>●</td>
</tr>
</tbody>
</table>
4.3 Description of Parameters

4.3.1 Entry Parameters

All the entry parameters are set with initial values when turning on the power. You can write the parameters you need to change in an application program with the Set Parameter command. If the written parameters are not correct, an input relay defined as “Error Notification” is set ON, and all the commands except the Set Parameter command are rejected. Execute the Set Parameter command to set the parameters at their correct values.

**CAUTION**

Errors resulting from the Set Parameter command cannot be reset by the Error Reset command. Write the correct values of the relevant parameters with the input relay defined as “Error Notification” set ON, and execute the Set Parameter command again.

- **Positive-direction limit value**
  
  Range of values: -134217728 to 134217727 [pulses]
  
  Initial value: 134217727

- **Negative-direction limit value**
  
  Range of values: -134217728 to (positive-direction limit value - 1) [pulses]
  
  Initial value: -134217728

These parameters set the positive/negative-direction operation limits as pulse counts from the origin (absolute position). The incremental encoder defines the position when power is turned ON as the origin if Origin Search or Write Current Position is not executed. If the command position exceeds the specified range during operation (during Servo ON), a Servo Error results. If a startup is performed after specifying a target position that exceeded the specified range, an error results and the axes does not move.

The limit values in the positive/negative-direction are not checked during JOG stepping, in Manual Pulser mode, during Origin Search, or in Velocity Control mode. (Any error does not occur.)

(Note) The limit values are usually set within the range of physical end-of-positions.

---

**Diagram:**

- Positive direction limit value
- Negative direction limit value
- The range of physical end-of-positions
- The actual range of positions
- **Velocity limit**
  
  Range of values: 1 to 131072000 \([1/65536] \text{ pulses/ms}\)
  
  Initial value: 131072000
  
  This parameter sets the limit of the target velocity. If startup is performed with a target velocity exceeding this limit, an error results.

  *(Note)*  Velocity Limit is the parameter that sets the limit of the velocity of the path generated by the positioning module. For example, the velocity of the path generated by the positioning module can exceed the velocity set point in an on-route interval of an on-route operation. In this case, the velocity is constrained by the Velocity Limit.

- **Over-limit velocity detection value**
  
  Range of values: 1 to 131072000 \([1/65536] \text{ pulses/ms}\)
  
  Initial value: 131072000
  
  This parameter sets the speed limit of a motor in operation. When the velocity of the motor exceeds this value during operation (Servo On), a servo error results.

  *(Note)*  This is the limit of the velocity detected from encoder inputs, not related to the velocity of the path generated by the positioning module. If an extremely small value is set, a servo error may occur even during End of Positioning. Also, during positioning, due to velocity overshoot and the change in velocity during operation, the velocity detected may be instantaneously larger than the target velocity. Thus, you should allow some buffer for the value of this parameter.

- **Over-limit acceleration detection value**
  
  Range of values: 1 to 131072000 \([1/65536] \text{ pulses/ms/ms}\)
  
  Initial value: 131072000
  
  This parameter sets the acceleration limit (velocity change rate) of a motor in operation. When the acceleration of the motor exceeds this value during operation (Servo On), a servo error results.

  *(Note)*  This is the limit of the acceleration detected from encoder inputs, not related to the acceleration of the path generated by the positioning module. If an extremely small value is set, a servo error may occur even during End of Positioning. Also, the acceleration detected by the system may be greater than the acceleration computable from the target velocity and the acceleration/deceleration time because of ripples during actual positioning. Thus, you should allow some buffer for the value of this parameter.

- **Deviation error detection value**
  
  Range of values: 1 to 134217727 [pulses]
  
  Initial value: 134217727
This parameter sets the limit on the difference (deviation) between the command position and the encoder feedback value of the axis in operation. If the value of this parameter is exceeded during operation (Servo ON), a servo error occurs.

(Note) Problems with position control loops such as the velocity command voltage and the encoder feedback are detected with the value of this parameter. Ensure that you write an appropriate value for this parameter. You can set the appropriate value by checking the maximum value of deviation in a normal operation.

• Rotation direction of the motor

0: Positive direction operation with positive velocity command voltage (B-phase forward)
1: Negative direction operation with positive velocity command voltage (B-phase forward)

Initial value: 0

This parameter sets the relation of the positive/negative position set from the CPU module and the direction of the motor’s rotation.

(Note) Positive/negative direction operation represents the positive/negative direction set from the CPU module, respectively. This does not set the relationship between the positive/negative velocity command voltage and the encoder feedback.

• Contact input polarity

Specified as a bit for each contact.

Initial value: 0

This parameter defines the logic of contact inputs. Refer to the section on contact input mode for the meaning of each contact input. 0 indicates an “a” contact input, and 1 indicates a “b” contact input.

(Note) An “a” contact input is an input that becomes effective when a signal is input, and a “b” contact input is an input that becomes effective when signals are shut off. For example, “b” contact limit input’ means detects a limit when signals are shut off and no limit is detected when a signal is input.
• **Contact output polarity**

   Specified as a bit for each contact.

   **Initial value**: 0

   This parameter defines the polarity of contact outputs. 0 indicates an “a” contact, and 1 indicates a “b” contact.

   (Note) An “a” contact output is an output that becomes effective when the contact is ON, and a “b” contact output is an output that becomes effective when contact is OFF. For example, “a” contact Brake Off output is an output in which the contact becomes ON when brake is OFF.

• **Position loop gain**

   Range of values: 1 to 10000 (0.01 to 100.00 Hz)

   **Initial value**: 200 (2Hz)

   This parameter specifies the servo range of the position control loop in [0.01Hz]. To adjust the value of this parameter, start from something around 200 to 300 (2 to 3 Hz), and gradually make it larger. Set it to the maximum value within the range of causing no oscillation or strange vibration of a motor.

   (Note) Adjust the velocity loop range with the Servo Driver before adjusting the position loop gain. For information on how to adjust the velocity loop range, refer to the instruction manual for the servo driver connected to your system.

• **Velocity feedforward factor**

   Range of value: 0 to 20000 (0.0000 to 2.0000)

   **Initial value**: 9500 (0.9500)

   This parameter specifies the velocity feedforward factor in units of [1/10000]. The range of values is 0 to 20000 (0.0000 to 2.0000).

   (Note) Usually, you do not have to change the initial value. If overshoot for the target position causes a problem, set a smaller value. Note that the motor’s response for the path generated by the positioning module will be reduced if the value of this parameter is decreased.

• **Specification of encoder**

   0: General-purpose Incremental

   1: Manchester-coded Absolute from Sanyo Denki Co., Ltd

   2: Serial Absolute from YASKAWA Electric Corporation

   **Initial Value**: 0
This parameter sets the encoder method. If you are going to use a general two-phase encoder, specify 0. If you are going to use Sanyo Denki's Manchester Coding Serial Absolute encoder, specify 1. And if you are going to use Serial Absolute Encoder from YASKAWA Electric Corporation, specify 2. If 1 or 2 is set, absolute data transmitted via serial transmission from the encoder will be read by the A-phase signal input of encoder signals.

- **Encoder multiplication**
  1: 1 × multiplication
  2: 2 × multiplication
  4: 4 × multiplication
  Initial value: 4

This parameter defines the multiplication of an incremental encoder. Usually, the value is set to 4. If the Operation Range Pulse Count exceeds the possible setting range, change the value to 2 or 1, which sets the operation range pulse count to 1/2 or 1/4.

- **Velocity/voltage ratio**
  Range of value: 1 to 2000000 [pps/V]
  Initial value: 10240

This parameter defines the ratio of the velocity command output voltage [V] and the actual velocity of operation [pps (pulses/s)]. For example, if the rated frequency of rotation is 3000 rpm, the velocity command voltage is 6V, and the encoder pulse count is 4096 pulses/rotation,

\[
3000 \text{ [rpm]} \times 4096 \text{ [pulses/rotation]} \div 60 \text{ [s/min]} \div 6 \text{ [V]} = 34133 \text{ [pps/V]}
\]

Thus, 34133 is to be specified.

(Note) The encoder pulse count per rotation of an encoder becomes 2 times or 4 times when the encoder multiplication is specified as 2 or 4. When calculating the pulse count, be sure to check the defined multiplication of the given encoder pulse count. Furthermore, some servo drivers allow setting the encoder pulse count per rotation using a parameter of the given servo driver. In this case, check the specification of the servo driver to avoid invalid setting for the positioning module.

(Example)
Suppose the pulse count per rotation of an encoder is 2048 with 1 × multiplication. If you are going to use this encoder with 4 × multiplication, the encoder pulse count can be calculated by: \(2048 \times 4 = 8192\)

- **Absolute offset value**
  Range of value: -134217728 to 134217727 [pulses]
  Initial value: 0

This parameter specifies the difference between the value output by the absolute encoder and the current position that the CPU module reads from the positioning module. If the absolute encoder outputs 0, the current position becomes the value of this parameter. This parameter is not relevant if you are not using the absolute encoder.
- **Absolute encoder pulse count**
  
  Range of value: 1 to 134217727 [pulses]
  Initial value: 32768
  
  This parameter defines the pulse count per rotation of YASKAWA Electric’s absolute encoder. Take the encoder multiplication into consideration when setting the value of this parameter. For example, if you are using an absolute encoder, whose pulse counting with $1 \times$ multiplication is 1024 per rotation, with 4 multiplication, set the value to 4069. This parameter is not relevant if you are not using YASKAWA Electric’s absolute encoder.

- **Sanyo ABS bit length**
  
  Range of values: 17 to 28
  Initial value: 28
  
  This parameter specifies the bit length of the position data of Sanyo Denki’s Serial Absolute Encoder. For details, see the applicable encoder or servo driver manuals.

  (Note) Bits other than position data (such as Alarm information) can be read in the status of absolute receive raw data.

- **Sanyo ABS operation direction**
  
  0: encoder data increase in the positive direction
  1: encoder data increase in the negative direction
  Initial value: 0
  
  This parameter specifies the direction of increase/decrease of the encoder data of Sanyo Denki’s Serial Absolute Encoder.

  (Note) The relationship between the positive/negative direction and the polarity of the velocity command voltage output is specified by the Motor Rotation Direction parameter. Therefore, when the direction of the motor’s rotation and this parameter are improperly set, the position control loop becomes a positive feedback loop and will be unable to control the motor.

- **Contact input mode**
  
  0: Normal mode, (with positive- or negative-direction limited)
  1: Normal mode (without positive- or negative-direction limited)
  2: Alarm detail mode, using limit (with positive- or negative-direction limited)
  3: Alarm detail mode (without positive- or negative-direction limited)
  Default value: 0
  
  Selects the meaning of the external contact input depending on the encoder method as shown below.
- External contact input for general-purpose incremental encoder
  (In this case, “0” and “2”, “1” and “3” are the same.)

<table>
<thead>
<tr>
<th>Mode</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>DI1</td>
<td>Positive-direction limit</td>
<td>For general purpose</td>
<td>Positive-direction limit</td>
<td>For general purpose</td>
</tr>
<tr>
<td>DI2</td>
<td>Negative-direction limit</td>
<td>For general purpose</td>
<td>Negative-direction limit</td>
<td>For general purpose</td>
</tr>
<tr>
<td>DI3</td>
<td>ALM</td>
<td>ALM</td>
<td>ALM</td>
<td>ALM</td>
</tr>
<tr>
<td>DI4</td>
<td>For general purpose/ Origin</td>
<td>For general purpose/ Origin</td>
<td>For general purpose/ Origin</td>
<td>For general purpose/ Origin</td>
</tr>
<tr>
<td>DI5</td>
<td>For general purpose/ external trigger</td>
<td>For general purpose/ external trigger</td>
<td>For general purpose/ external trigger</td>
<td>For general purpose/ external trigger</td>
</tr>
<tr>
<td>DI6</td>
<td>For general purpose</td>
<td>For general purpose</td>
<td>For general purpose</td>
<td>For general purpose</td>
</tr>
</tbody>
</table>

- External contact input for YASKAWA Electric’s Serial Absolute encoder

<table>
<thead>
<tr>
<th>Mode</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>DI1</td>
<td>Positive-direction limit</td>
<td>For general purpose</td>
<td>Positive-direction limit</td>
<td>For general purpose</td>
</tr>
<tr>
<td>DI2</td>
<td>Negative-direction limit</td>
<td>For general purpose</td>
<td>Negative-direction limit</td>
<td>For general purpose</td>
</tr>
<tr>
<td>DI3</td>
<td>ALM</td>
<td>ALM</td>
<td>ALM</td>
<td>ALM</td>
</tr>
<tr>
<td>DI4</td>
<td>For general purpose/ Origin</td>
<td>For general purpose/ Origin</td>
<td>ALM code 0</td>
<td>ALM code 0</td>
</tr>
<tr>
<td>DI5</td>
<td>For general purpose/ external trigger</td>
<td>For general purpose/ external trigger</td>
<td>ALM code 1</td>
<td>ALM code 1</td>
</tr>
<tr>
<td>DI6</td>
<td>For general purpose</td>
<td>For general purpose</td>
<td>ALM code 2</td>
<td>ALM code 2</td>
</tr>
</tbody>
</table>

- External contact input for Sanyo Denki’s Manchester Coding Serial Absolute encoder

<table>
<thead>
<tr>
<th>Mode</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>DI1</td>
<td>Positive-direction limit</td>
<td>For general purpose</td>
<td>Positive-direction limit</td>
<td>For general purpose</td>
</tr>
<tr>
<td>DI2</td>
<td>Negative-direction limit</td>
<td>For general purpose</td>
<td>Negative-direction limit</td>
<td>For general purpose</td>
</tr>
<tr>
<td>DI3</td>
<td>ALM</td>
<td>ALM</td>
<td>ALM 1</td>
<td>ALM 1</td>
</tr>
<tr>
<td>DI4</td>
<td>For general purpose</td>
<td>For general purpose</td>
<td>ALM 2</td>
<td>ALM 2</td>
</tr>
<tr>
<td>DI5</td>
<td>For general purpose/ external trigger</td>
<td>For general purpose/ external trigger</td>
<td>ALM 4</td>
<td>ALM 4</td>
</tr>
<tr>
<td>DI6</td>
<td>For general purpose</td>
<td>For general purpose</td>
<td>ALM 8</td>
<td>ALM 8</td>
</tr>
</tbody>
</table>

(Note) The Positive-Direction Limit, Negative-Direction Limit and ALM external contact inputs cause the “positive-direction limit input detected”, “negative-direction limit input detected” and “driver alarm input detected” errors, respectively. For details on the ALM signals when using absolute encoders, see applicable servo driver instruction manuals.
4.3.2 Startup Parameters

Startup parameters do not have initial values. The area where these parameters will be written are set to “0” when power is turned on.

- Target velocity
  Set range: 1 to 131072000 [(1/65536) pulses/ms] (in position control mode)
  -131072000 to 131072000 [(1/65536) pulses/ms] (in velocity control mode)

Sets the operating speed of the positioning as well as velocity control mode.
(Note) The maximum value of the target velocity is limited by the Velocity Limit Value entry parameter.

- Target position
  -134217728 to 134217727 [pulses]

Sets the target position for the positioning.
(Note) The target position and the position calculated with the target position mode must be between the Positive-Direction Limit value and the Negative-Direction Limit value entry parameters.

- Target position mode
  0: Absolute position
  1: Relative position (reference: encoder position)
  2: Relative position (reference: last target position)

Specifies the type of the target position:
1) Absolute position: Setup target position not related to the startup position.
2) Relative position (reference: encoder position): Relative position with the startup encoder feedback position as the reference point.
3) Relative position (reference: last target position): While stopping, relative position with the command position as the reference point. For on-route operation, relative position with the target position of the previous operation as the reference point.
(Note) When operating with the relative position (reference: encoder position), the target position of the positioning is calculated with the encoder feedback position as the reference point. Thus, if there is some stationary deviation at the positioning startup, take note that the setup data and the actual motor operation amount are not exactly the same (the difference is the deviation).

- Acceleration time
  Setup range: 0 to 32767 [ms]

Sets the time interval from starting the operation to reaching the target velocity. It has different meaning for change target position and change velocity. For details, refer to the explanation of each command.
• **Deceleration time**
  Setup range: 0 to 32767 [ms]

  Sets the time interval from starting the operation to stopping deceleration

• **Acceleration mode selection**
  0: Trapezoidal acceleration
  1: 2-stage acceleration
  2: S-form (3-stage) acceleration

  Specifies the acceleration pattern from startup until reaching the target velocity. For details on 2-stage acceleration and S-form acceleration, refer to the acceleration parameters provided below.

• **Deceleration mode selection**
  0: Trapezoidal deceleration
  1: 2-stage deceleration
  2: S-form (3-stage) deceleration

  Specifies the deceleration pattern from the target velocity until stopping. For details on 2-stage deceleration and S-form deceleration, refer to the deceleration parameters provided below.

• **Acceleration parameter 1, acceleration parameter 2, deceleration parameter 1, deceleration parameter 2**

  Parameters which specify the form of the segment lines for 2-stage and S-form (3-stage) acceleration and deceleration. Parameter 1 specifies the percentage of the time for the segment corresponding to low speed (for 2-stage), or the percentage of the time for the segments corresponding to low speed and high speed (S-form). Parameter 2 specifies the percentage of the acceleration (deceleration) amount specified by parameter 1.

  The actual form of acceleration (deceleration) is shown below. Time and velocity for each segment is calculated as follows:

  1) = acceleration time \times (acceleration parameter 1 ÷ 100)
  2) = target velocity \times (acceleration parameter 2 ÷ 100)
  3) = deceleration time \times (deceleration parameter 1 ÷ 100)
  4) = target velocity \times (deceleration parameter 2 ÷ 100)
The setup range for 2-stage acceleration/deceleration is 1 to 99, for S-form acceleration/deceleration, it is 1 to 49.

- **Positioning judgment range**
  Setup range: 0 to 134217727 [pulses]

  Sets the deviation pulse counts for judging the end of positioning. After the positioning path generation ends, when the deviation is less than or equal to this value, positioning ends.

- **Positioning time-out interval**
  Setup range: 0 to 32767 [ms]

  Sets the maximum value of the time interval between the termination of the path generation and end of positioning. If end of positioning does not occur in this interval after the termination of the path generation, a servo error occurs.

  If this parameter is set to “0”, positioning is completed just after end of pulse clear out. (No judgement for the end of positioning.)

- **Interpolation mode selection**
  0: Normal operation
  1: Arc-interpolation X axis
  2: Arc-interpolation Y axis

  Select interpolation mode (0) when executing normal positioning in position control mode. Arc-interpolation is explained in the “Arc-interpolation” item.

- **Startup mode selection**
  0: Normal startup
  1: Wait for external trigger
  2: Wait for internal trigger

  Specifies waiting for an external (internal) trigger at startup. Execution of the command to which the external (internal) trigger wait can be specified is withheld until the input of the external trigger (the internal trigger) is set.

  But the data which are necessary for operation (e.g., target position, acceleration pattern,
deceleration pattern) are calculated only when the command is issued. Also, while waiting for the external (internal) trigger, commands other than “Decelerate and stop”, “Stop immediately”, “Servo OFF”, “Stationary deviation adjust”, “Write current position” and “Position detection notification reset” are not accepted. “Decelerate and stop”, “Stop immediately” and “Servo OFF” reset the trigger wait state. If the system detects an error during operation and stops the motor, the trigger wait state is reset.

(Note) External triggers are external contact inputs which are set to “External trigger” in the contact input mode. Internal triggers are output relays labeled “Internal trigger.”

• Position detection mode
  0: No detection
  1: Command position
  2: Encoder position

Specifies the reference position for specified position detection. If it is set to “0”, no detection is performed.

(Note) To detect the actual motor position, set the position detection mode (2).

• Position detection setpoint
  Setup range: -134217728 to 134217727 [pulses]

When the position detection mode is set to “1” or “2”, the input relay defined as “position detection notification” is set “ON” when the position specified during the positioning operation in position control mode reaches this setpoint. This input relay can be used as an interrupt input in application programs.

4.3.3 Startup Parameters Related to Origin Search

• Origin-search mode
  Set by a bit pattern.

Sets the move of the motor for each contact input using a bit pattern after detecting the edges of each contact input during origin search. For details, refer to origin-search explanation.

• Origin-search direction
  0: Positive direction
  1: Negative direction

Sets the rotation direction of the motor during origin search.
• Select Z-phase edge
  0: OFF to ON edge
  1: ON to OFF edge

Sets the polarity of the Z-phase input for detecting Z-phase when in origin-search or switching the velocity to position control.

• Z-phase pulse count number
  Setup range: 0 to 32767 [times]

Sets which Z-phase pulse is effective for the Z-phase detection when in origin-search or switching the velocity to position control.

• Z-phase search range
  Setup range: 0 to 134217727 [pulses]

Error occurs if Z-phase cannot be detected after operating the number of pulses which was set by this parameter for Z-phase detection when in origin-search or switching the velocity to position control.

Note: This parameter is used to prevent (continuing operation of the motor when Z-phase cannot be detected because of Z-phase signal disconnection, etc. Usually, this is set close to the period of Z-phase.

• Origin-offset value
  Setup range: -134217728 to 134217727 [pulses]

Specifies the value of the position where the input specifying the origin was detected during origin search, and the value of the position where the control was switched from the velocity to position control. Usually, the value “0” is used, but set other values if necessary.

### 4.3.4 Extended Command Parameters

Specifies the type of the extended command. For details, refer to the explanation of each parameter.

Referred to when an extended command is executed.

• Extended command parameter
  0: Servo ON
  1: Servo OFF
  2: Brake ON
  3: Brake OFF
  4: Steady-state deviation adjust
  5: Driver reset
  10: Manual pulser mode ON
  11: Manual pulser mode OFF
• Steady-state deviation adjust range
  Setup range: -32768 to 32767 [pulses]

  Referred to when executing the steady-state deviation adjust.

• Manual pulser scale value
  Setup range: -500 to 500 [times]

  Sets the moving amount [pulses] of axes per one manual pulser input pulse. Effective when manual pulser mode ON is executed. Manual pulser input is calculated by a multiplication of 4.

  The manual pulser input is counted in positive direction if input is done in B phase lead.

  If a negative value is set, then the motor rotates in the opposite direction of the manual pulser input.

4.3.5 Switch Control Mode Parameter

  Specifies the switch control mode switching command.

• Control mode switching parameter
  0: Switch to speed control mode
  1: Switch to position control mode
  2: Switch to position control mode (wait for Z-phase)

  Specifies to which mode to switch when request to switch control mode is executed.

4.3.6 Arc-interpolation Parameters

  These parameters are specific to the arc-interpolation operation.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Center position</td>
<td>-134217728 to 134217727 [pulses]</td>
</tr>
<tr>
<td>Radius</td>
<td>1 to 134217727 [pulses]</td>
</tr>
<tr>
<td>Starting angle</td>
<td>-23592960 to 23592960 [(1/65536) degrees]</td>
</tr>
<tr>
<td>Angular travel</td>
<td>-2123366400 to 2123366400 [(1/65536) degrees]</td>
</tr>
<tr>
<td>Angular Speed Setpoint</td>
<td>1 to 23592960 [(1/65536) degrees]</td>
</tr>
<tr>
<td>Acceleration time</td>
<td>0 to 32767 [ms]</td>
</tr>
<tr>
<td>Deceleration time</td>
<td>0 to 32767 [ms]</td>
</tr>
<tr>
<td>Target position</td>
<td>-134217728 to 134217727 [pulses]</td>
</tr>
<tr>
<td>Correction pulse range</td>
<td>0 to 32767 [pulses]</td>
</tr>
</tbody>
</table>

  When you specify the arc-interpolation X axis or the arc-interpolation X axis in Interpolation Mode selection at the positioning start, the arc-interpolation positioning is performed. The positioning module generates the command position and operates in the following way.

  (Note)  X axis and Y axis which perform the arc interpolation is not related to axis 1 or axis 2 of the positioning module. It is not necessary that X axis and Y axis are in the same positioning module.
The track of the arc interpolation is generated by the following formulae:

\[
\text{X axis} \left( [\text{Central Point Position}] + [\text{Radius}] \right) \times \cos \left( [\text{Start Angle}] + [\text{Angular Velocity Set Point}] \times \text{time} \right)
\]

\[
\text{Y axis} \left( [\text{Central Point Position}] + [\text{Radius}] \right) \times \sin \left( [\text{Start Angle}] + [\text{Angular Velocity Set Point}] \times \text{time} \right)
\]

When \([\text{Angular Velocity Set Point}] \times \text{time}\) becomes equal to \([\text{Angular Travel}]\), it finishes. If Acceleration Time and Deceleration Time are both 0, the actual track is the same as the one defined by the above formulae. Otherwise, as Angular Velocity increases / decreases with time, the term \([\text{Angular Velocity Set Point}] \times \text{time}\) must be changed. But the basic idea is the same. In the case of the arc interpolation, Acceleration Time is the time interval in which the angular velocity changes from 0 to Angular Velocity Set Point when starting. Deceleration Time is the time interval in which the angular velocity changes from Angular Velocity Set Point to 0 when stopping.

The arc interpolation operation is realized by setting the parameters to X axis and Y axis independently, and then starting them simultaneously. So, it is necessary to specify the same value on both axes for Start Angle, Angular Travel, Angular Velocity Set Point, Acceleration Time and Deceleration Time.

Here \([\text{Angle}]\) is defined in the X-Y plane with the positive part of the X axis as 0 degree, and to increase in the counterclockwise direction. For example, the positive part of the Y axis is 90 degrees, and the negative part of the X axis is 180 degrees.

In the following, we will explain the case of X axis. For Y axis, change COS to SIN.

When the system starts, the position to start the arc-interpolation operation is using the above formulae with time being 0 in the positioning module, i.e.,

\[
[\text{Central Point Position}] + [\text{Radius}] \times \cos([\text{Start Angle}])
\]

This position must coincide with the current position of the axis when starting the system. (Set the parameters so that the above condition is satisfied.) Even if a strict coincidence is impossible because of calculation errors, etc., keep the maximum value of this difference under the Correction Pulse Range. If the difference exceeds the Correction Pulse Range, a servo error occurs at startup. (If the difference is under the Correction Pulse Range, the axis moves to the staring position of the arc interpolation operation with Speed Limit Value at startup.)

The position to finish the arc interpolation operation is calculated as follows:

\[
\text{Central Point Position} + \text{Radius} \times \cos(\text{Start Angle} + \text{Angular Travel})
\]

Thus, this position must coincide with Target Position. (Set the parameters so that the above condition is satisfied.) Even if a strict coincidence is impossible because of calculation errors, etc., keep the maximum value of this difference under the Correction Pulse Range. If the difference exceeds the Correction Pulse Range, a servo error occurs when finishing the arc interpolation operation. (If the difference is under the Correction Pulse Range, the axis moves to the Target Position of the arc interpolation operation with Velocity Limit Value when finishing the arc-interpolation operation.)
Arc-interpolation Operation

**CAUTION**

X axis and Y axis move independently also when performing arc interpolation. So when an error occurs at one axis, the other axis continues moving. If it is necessary to stop the other axis, stop the motor by applying the immediate stop command to the moving axis after detecting the error with an application program (and checking the input relay defined as “Error Notification”).

---

**Figure 4.1 Arc Interpolation Operation**
4.4 Example of Entry Parameters Setting

The following example shows the minimum setting of the entry parameters for controlling the motor using the positioning module. The underlined values are set.

- **The motor used**
  - Rated number of revolutions: 3000 rpm
  - Rated Output voltage: 6 V DC
  - Encoder: General-purpose incremental encoder
  - Pulse number: 8192 pulses/rotation (a multiplication of 4)
  - Servo ON, Brake OFF, Driver reset signals: all “a” contacts
  - Driver alarm signals: all “b” contacts

⚠️ **CAUTION**

In some cases, you can set / change these conditions using the servo driver arbitrarily. Then the parameters set to the positioning module must coincide with the setting of the servo driver. So calculate the values of the entry parameters after confirming the setting of the servo motor.

- **Mechanism**
  - Direct shaft driven by ball thread
  - Ball thread pitch: 5 mm/rot
  - Operation Range: -500 mm to +1000 mm (operates in the positive direction with the positive velocity command voltage)
  - Maximum speed: 6000 mm/min (100 mm/s)
  - Acceleration / deceleration limit: equal to or less than 0.1 s (100 ms) when changing from 0 to the maximum speed
  - Contact input: Positive-direction limit (“b” contact), negative-direction limit (“b” contact), origin (“a” contact), ready signal (“a” contact). Others are not used.

- **Calculation of the entry parameters**
  - Positive-direction Limit Value
    
    \[
    \text{Limit Value} = \frac{1000 \text{ [mm]}}{5 \text{ [mm/rot]}} \times 8192 \text{ [pulses/rot]} = 1638400 \text{ [pulses]} 
    \]

  - Negative-direction Limit Value
    
    \[
    \text{Limit Value} = \frac{-500 \text{ [mm]}}{5 \text{ [mm/rot]}} \times 8192 \text{ [pulses/rot]} = -819200 \text{ [pulses]} 
    \]

  - Velocity Limit Value
    
    \[
    \text{Limit Value} = \frac{100 \text{ [mm/s]}}{5 \text{ [mm/rot]}} \times 8192 \text{ [pulses/rot]} = 163840 \text{ [pulses/s]} 
    
    163840 \text{ [pulses/s]} \times 1000 \times 65536 = 10737418 (\frac{1}{65536} \text{ pulses/ms}) 
    \]
• Over Velocity Detection Value
  Set a value larger than the velocity limit value: 10737418 [(1/65536) pulses/ms], e.g.
  the result of multiplying the velocity limit value by 1.1.
  \[10737418 \times 1.1 = 11811859 \times (1/65536) \text{pulses/ms}]\]

• Over Acceleration Detection Value
  \[100 \times 1000 \times 65536 = 107374 \times (1/65536) \text{pulses/ms/ms} \]
  Set a value larger than the value above, e.g., the result of multiplying the value by 2.
  \[107374 \times 2 = 214748 \times (1/65536) \text{pulses/ms/ms} \]

• Deviation-Error-Detection Value
  Depends on the position loop gain. Set a value of about 10% of the operation range.
  \[(1638400 + 819200) \times 0.1 = 245760 \text{ pulses}]\]

• Rotation Direction
  Positive velocity command voltage, positive-direction operation : 0

• Contact-Input Polarity
  $0007$
  Positive-direction limit ("b" contact), negative-direction limit ("b" contact), ALM ("b" contact), origin ("a" contact). Others are not used and are temporarily set to “a” contact.

• Contact-Output Polarity
  $0000$
  All “a” contact

• Position-Loop Range
  Initial value: 200

• Velocity Feedforward Factor
  Initial value: 9500

• Specification of Encoder
  0 (general-purpose incremental method)

• Encoder Multiplication Number
  4 (× 4 multiplication)
- Velocity / Voltage Ratio
  
  \[
  \text{Velocity} = \frac{3000 \text{ rpm} \times 8192 \text{ pulses}}{60 \text{ sec/min}} = 409600 \text{ pulses/s}
  \]
  \[
  \frac{409600 \text{ pulse/s}}{6 \text{ V}} = 68267 \text{ pps/V}
  \]

- Absolute Offset Value
  
  Initial value: 0 (not used)

- Absolute Pulse Number
  
  Initial value: 32768 (not used)

- Sanyo ABS Bit Length
  
  Initial value: 28 (not used)

- Sanyo ABS Operation Direction
  
  Initial value: 0 (not used)

- Contact Input Mode
  
  0 (normal mode, using limit)
5. Status

The status is the data which the CPU module is called up from the positioning module. The positioning module status is checked by this data and input relays.

5.1 List of Status

Status listed with 2 data position numbers are 2 word data. The data with the smaller number is the low-order word, and the one with the larger number is the high-order word.

Data position numbers are specified for each word. The READ instruction used for accessing from a sequence program must be on a word basis. Long-word based instructions cause inappropriate access.

Use word-wise instructions also when accessing from a BASIC program.

Data counted by \( \left( \frac{1}{65536} \right) \) pulses/ms™ (1 data) are fixed-point data with a 1-word (16 bits) integer and 1-word (16 bits) decimal. The data with the smaller number is the low-order word, and the one with the larger number is the high-order word. For more information on fixed-point data, see Section 4.1, “List of Parameters.”

To convert a data in \( \left( \frac{1}{65536} \right) \) pulses/ms unit into one in [pulses/sec] unit, multiply it with 1000 in long word operation, ignore the lowest-order word and the highest-order word and use the remaining two words as long-word data.

(Example)

Let D0001 be a long-word data in \( \left( \frac{1}{65536} \right) \) pulses/ms unit. The operation is as follows:

\[
\begin{array}{c|c|c|c}
\hline
D0011 & = & D0001 & \times 1000 \\
\hline
\end{array}
\]

Operation Result:

\[
\begin{array}{c|c|c|c}
D0014 & D0013 & D0012 & D0011 \\
0 & Resultant long word [pulses/ms] & Truncated Portion \\
\hline
\end{array}
\]
Table 5.1 List of Status

<table>
<thead>
<tr>
<th>Data position number</th>
<th>Status Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>101</td>
<td>Error Status</td>
<td>Error information code when an error occurs</td>
</tr>
<tr>
<td>102</td>
<td>Detailed Error Code</td>
<td>Error information from the absolute encoder</td>
</tr>
<tr>
<td>103</td>
<td>Contact Input Status</td>
<td>States of contact inputs</td>
</tr>
<tr>
<td>104/105</td>
<td>Current Position Status (Command Position)</td>
<td>[pulses]</td>
</tr>
<tr>
<td>106/107</td>
<td>Current Velocity Status (Command Position)</td>
<td>[(1/65536) pulses/ms] *1</td>
</tr>
<tr>
<td>108/109</td>
<td>Current Position Status (Encoder Position)</td>
<td>[pulses]</td>
</tr>
<tr>
<td>110/111</td>
<td>Current Velocity Status (Encoder Position)</td>
<td>[(1/65536) pulses/ms] *1</td>
</tr>
<tr>
<td>112/113</td>
<td>Target Position Status</td>
<td>[pulses]</td>
</tr>
<tr>
<td>114</td>
<td>Extended Status</td>
<td>Operation information such as acceleration, deceleration, overlap, waiting for trigger, and control mode</td>
</tr>
<tr>
<td>115</td>
<td>Remaining Deceleration Time</td>
<td>Remaining time in [ms] until the path generation arrives at the target position</td>
</tr>
<tr>
<td>116</td>
<td>Contact Output Status</td>
<td>States of external contact outputs</td>
</tr>
<tr>
<td>117/118</td>
<td>Absolute Data Status (YASKAWA ABS) Absolute Receive Raw Data (Sanyo ABS)</td>
<td>[pulses] 30 bits of received data</td>
</tr>
</tbody>
</table>

**CAUTION**

When 2-word data are called up from the CPU module, the simultaneity of the high- and low-order words in the 2-word data cannot be assured because of a conflict of the positioning module's data-update period with the CPU module's call-up timing. Each time data are called up from the sequence CPU, read them twice in succession using the READ command to ensure the data match. By doing this, the simultaneity of the high- and low-order words in the 2-word data can be assured. In such a case, do not use the HRD command to match the data. Otherwise, the data simultaneity cannot be assured. If the 2-word data are called up from the BASIC CPU, it requires a longer 2-word data call-up time than the positioning module's data update period, and the simultaneity of the data cannot be assured.
5.2 Description of Status

- **Error Status**
  Read the error code when an error occurs. It is meaningless when the relay defined as “Error Notification” is OFF. For details, see Chapter 8, “List of Error Codes.”

- **Detailed Error Code**
  In YASKAWA Electric Corporation’s Absolute Encoder System, this is the two-byte string of an error code in ASCII codes read from the error information transmitted from the encoder. (The first byte is the high-order word, and the second byte the low-order word.) It is irrelevant when the incremental encoder is used. For information on what each code indicates, see applicable servo driver Instruction manuals.

- **Contact Input Status**
  Reads the state of the external contact input and the emergency stop input. The state of each contact is stored as 1 bit (0: OFF, 1: ON). When a contact is specified as “a” contact, it is represented by “1” if the contact is open. When a contact is specified as “b” contact, it is represented by “1” if the contact is closed.
  For the meaning of each contact input, see the contact input mode in Section 4.3.1 provided earlier in this manual.

  15~                  ~6 5 4 3 2 1 0
  ----------------------x x x x x x x
  - DI1 (external contact input 1)
  - DI2 (external contact input 2)
  - DI3 (external contact input 3)
  - DI4 (external contact input 4)
  - DI5 (external contact input 5)
  - DI6 (external contact input 6)
  - Emergency stop input

- **Current Position Status (Command Position) [pulses]**
  This represents the current position of the command position (the position of the path generated by the positioning module). It is not the actual position of the motor.

- **Current Velocity Status (Command Position) [(1/65536) pulses/ms]**
  This monitors the current velocity at the command position (the position of the path generated by the positioning module). It is not the actual velocity of the motor.
  Regardless of the direction of the motor’s rotation, the value is zero or positive.

- **Current Position Status (Encoder Position) [pulses]**
  This represents the current position in the encoder feedback value. It is the actual position of the motor.

- **Current Velocity Status (Encoder Position) [(1/65536) pulses/ms]**
  This represents the current velocity in the encoder feedback value. It is the actual velocity of the motor. Regardless of the direction of the motor’s rotation, the value is zero or positive.
• Target Position Status  [pulses]
This reads the operation target position during a positioning operation. The target position calculated from the target position and the target position mode which is set at the start of the positioning is stored as the operation target position.

• Extended Status
Reads the data expressing the operation states of the axes in bits. Each corresponding is set to 1.

```
15~  10  9  8  7  6  5  4  3  2  1  0
  x  x  x  0  x  x  x  x  x  x  x  x
```

(0) Accelerating
(1) Moving at constant velocity
(2) Decelerating
(3) Accelerating in the latter operation in the on-route interval
(4) Constant velocity in the latter operation in the on-route interval
(5) Decelerating in the latter operation in the on-route interval
(6) Decelerating during target position change with direction change
(7) Accelerating and decelerating during velocity change
(8) Decelerating during Decelerate-and-Stop command
(9) In manual pulser mode
(10) Origin detected during origin search
(12) Waiting for control mode switching
(13) Waiting for internal relay
(14) Waiting for external relay
(15) In velocity control mode

**CAUTION**
In some cases, the positioning module restricts the execution of commands depending on its state. (Refer to the explanation of each command.) Then it is necessary to get the detailed status of the positioning module with an application program. For that purpose, you can use this extended status. Usually, the data is separated into bits after reading.

• Remaining Deceleration Time [ms]
This represents the time until the end of the path generation after starting deceleration to stop at the target position during positioning. The value is 0 during path generation stop, and -1 during acceleration and constant velocity operation.
This status is the time until the stop at the target position. It does not represent the remaining deceleration time during deceleration by the Request to Decelerate and Stop during positioning or in velocity control mode. This status is useful to decide when to start on-route operation.

- **Contact Output Status** — the status of the external contact outputs
  It monitors the states of the external contact outputs. Relevant bits are set 1.

  15  3  2  1  0
  000000000000 x x x x

  - Servo ON
  - Resetting driver
  - Brake OFF
  - SEN Signal ON

- **Absolute Data Status** [pulses]
  The YASKAWA Denki’s absolute encoder monitors the position based on the absolute position which is read from the encoder. The value is the same as the Current Position Status except when Origin Search or Write Current Position is executed.
  It is irrelevant if you are using other encoders.

- **Absolute Receive Raw Data**
  It monitors all the 30 bits of the received data without any processing when you use Sanyo Denki’s Absolute Encoder. (LSB adjusted. The highest 2 bits are always 0.)
  It is irrelevant if you are using other encoders.

**CAUTION**

The relevant bits in this data can check the alarm codes transmitted with encoder data.
For details, see applicable motor or the servo driver instruction manuals. See also the following example.

### Example of Received Raw Data

<table>
<thead>
<tr>
<th>BIT</th>
<th>MSB</th>
<th>. . .</th>
<th>. . .</th>
<th>. . .</th>
<th>. . .</th>
<th>. . .</th>
<th>. . .</th>
<th>. . .</th>
<th>. . .</th>
<th>. . .</th>
<th>. . .</th>
<th>. . .</th>
<th>LSB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low-order</td>
<td>D15</td>
<td>D14</td>
<td>D13</td>
<td>D12</td>
<td>D11</td>
<td>D10</td>
<td>D9</td>
<td>D8</td>
<td>D7</td>
<td>D6</td>
<td>D5</td>
<td>D4</td>
<td>D3</td>
</tr>
<tr>
<td>high-order</td>
<td>0</td>
<td>0</td>
<td>BA</td>
<td>BW</td>
<td>0</td>
<td>0</td>
<td>D25</td>
<td>D24</td>
<td>D23</td>
<td>D22</td>
<td>D21</td>
<td>D20</td>
<td>D19</td>
</tr>
</tbody>
</table>

D0 to D25: absolute data (26 bits)
BA: battery alarm
BW: battery warning
6. List of Input/Output Relays

The positioning module has 32 output relays and 32 input relays as the interface to the FA-M3 CPU module. For details on input/output relays, see applicable sections in Chapter 7, “Accessing Modules.”

**CAUTION**

NEVER set the output relays of axis 2 (Y49 to Y64) of the F3NC51-0N. Further, the input relays of axis 2 (X17 to X32) are meaningless.

---

6.1 Output Relays

<table>
<thead>
<tr>
<th>Output relay number</th>
<th>Operation when ON</th>
</tr>
</thead>
<tbody>
<tr>
<td>Axis 1</td>
<td>Axis 2</td>
</tr>
<tr>
<td>Y33</td>
<td>Y49</td>
</tr>
<tr>
<td>Y34</td>
<td>Y50</td>
</tr>
<tr>
<td>Y35</td>
<td>Y51</td>
</tr>
<tr>
<td>Y36</td>
<td>Y52</td>
</tr>
<tr>
<td>Y37</td>
<td>Y53</td>
</tr>
<tr>
<td>Y38</td>
<td>Y54</td>
</tr>
<tr>
<td>Y39</td>
<td>Y55</td>
</tr>
<tr>
<td>Y40</td>
<td>Y56</td>
</tr>
<tr>
<td>Y41</td>
<td>Y57</td>
</tr>
<tr>
<td>Y42</td>
<td>Y58</td>
</tr>
<tr>
<td>Y43</td>
<td>Y59</td>
</tr>
<tr>
<td>Y44</td>
<td>Y60</td>
</tr>
<tr>
<td>Y45</td>
<td>Y61</td>
</tr>
<tr>
<td>Y46</td>
<td>Y62</td>
</tr>
<tr>
<td>Y47</td>
<td>Y63</td>
</tr>
<tr>
<td>Y48</td>
<td>Y64</td>
</tr>
</tbody>
</table>

(Note) Insert the FA-M3 slot number where this module is installed in □□□□.
# 6.2 Input Relays

<table>
<thead>
<tr>
<th>Input relay number</th>
<th>Operation when ON</th>
</tr>
</thead>
<tbody>
<tr>
<td>Xnnn01</td>
<td>Start Operation Command ACK</td>
</tr>
<tr>
<td>Xnnn02</td>
<td>Extended command ACK</td>
</tr>
<tr>
<td>Xnnn03</td>
<td>Request to Decelerate and Stop ACK</td>
</tr>
<tr>
<td>Xnnn04</td>
<td>Request to Stop Immediately ACK</td>
</tr>
<tr>
<td>Xnnn05</td>
<td>End-of-Origin Search</td>
</tr>
<tr>
<td>Xnnn06</td>
<td>Switch Control Mode ACK</td>
</tr>
<tr>
<td>Xnnn07</td>
<td>During Positive-direction Operation</td>
</tr>
<tr>
<td>Xnnn08</td>
<td>During Negative-direction Operation</td>
</tr>
<tr>
<td>Xnnn09</td>
<td>Change Target Position ACK</td>
</tr>
<tr>
<td>Xnnn10</td>
<td>Change Velocity ACK</td>
</tr>
<tr>
<td>Xnnn11</td>
<td>Write Current Position ACK</td>
</tr>
<tr>
<td>Xnnn12</td>
<td>Error Notification</td>
</tr>
<tr>
<td>Xnnn13</td>
<td>Position Detected Notification</td>
</tr>
<tr>
<td>Xnnn14</td>
<td>End-of-Positioning</td>
</tr>
<tr>
<td>Xnnn15</td>
<td>End-of-Pulse Clear Out</td>
</tr>
<tr>
<td>Xnnn16</td>
<td>Parameter Set ACK</td>
</tr>
</tbody>
</table>

(Note) Insert the FA-M3 slot number where this module is installed in nnn.
7. Accessing Modules

In the program examples shown in this chapter, the positioning module is installed in the 4th slot (slot no. 004) of the main unit; when only one axis is used in these examples, they will be axis 1.

7.1 Accessing from Sequence CPU

The following are the instructions to access from the sequence CPU with the ladder sequence program. The details on these instructions are explained in the Sequence CPU Instruction Manual – Instructions (publication number IM 34M6P12-03E).

Reading/Writing Parameters and Status

The instructions should be of the word unit. The long-word unit instructions cannot be used.

• Special Module READ instruction (READ instruction)

<table>
<thead>
<tr>
<th>READ</th>
<th>SL</th>
<th>n1</th>
<th>D</th>
<th>k</th>
</tr>
</thead>
</table>
SL: Slot number where the module is installed
n1: First data position number for reading data
D: First device to store the read data
k: Number of data (in word unit) to be read

• Special Module WRITE instruction (WRITE instruction)

<table>
<thead>
<tr>
<th>WRITE</th>
<th>S</th>
<th>SL</th>
<th>n2</th>
<th>k</th>
</tr>
</thead>
</table>
S: Device to store the write data
SL: Slot number where the module is installed
n2: First data position number for the write data
k: Number of data (in word unit) to be written

• Special Module High-Speed READ instruction (HRD instruction)

<table>
<thead>
<tr>
<th>HRD</th>
<th>SL</th>
<th>n1</th>
<th>D</th>
<th>k</th>
</tr>
</thead>
</table>
SL: Slot number where the module is installed
n1: First data position number for reading data
D: First device to store the read data
k: Number of data (in word unit) to be read

• Special Module High-Speed WRITE instruction (HWR instruction)

<table>
<thead>
<tr>
<th>HWR</th>
<th>S</th>
<th>SL</th>
<th>n2</th>
<th>k</th>
</tr>
</thead>
</table>
S: Device to store the write data
SL: Slot number where the module is installed
n2: First data position number for the write data
k: Number of data (in word unit) to be written
Interrupt Processing

The input relay defined as “End of Positioning” and “Specified Position-Detection Notification” can be used for interrupt processing in the positioning module. The interrupt program between the INTP and IRET instructions is to be executed when the interrupt input changes from OFF to ON.

<table>
<thead>
<tr>
<th>INTP</th>
<th>S</th>
</tr>
</thead>
<tbody>
<tr>
<td>IRET</td>
<td></td>
</tr>
</tbody>
</table>

S: Input relay to be the interrupt reason
7.1.1 Reading the Module Status

This section explains how to read the status of the positioning module.

Items to Note:

- There is no special item to note here. The status of the positioning module can be read at any time.

Program Example:

In the example below, all of the status is read at once with the READ instruction.

- List of devices used

<table>
<thead>
<tr>
<th>Device</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D0101</td>
<td>Error Status</td>
</tr>
<tr>
<td>D0102</td>
<td>Detailed Error Code</td>
</tr>
<tr>
<td>D0103</td>
<td>Contact Input Status</td>
</tr>
<tr>
<td>D0104/D0105</td>
<td>Current Position Status (Command Position)</td>
</tr>
<tr>
<td>D0106/D0107</td>
<td>Current Velocity Status (Command Position)</td>
</tr>
<tr>
<td>D0108/D0109</td>
<td>Current Position Status (Encoder Position)</td>
</tr>
<tr>
<td>D0110/D0111</td>
<td>Current Velocity Status (Encoder Position)</td>
</tr>
<tr>
<td>D0112/D0113</td>
<td>Target Position Status</td>
</tr>
<tr>
<td>D0114</td>
<td>Extended Status</td>
</tr>
<tr>
<td>D0115</td>
<td>Remaining deceleration time</td>
</tr>
<tr>
<td>D0116</td>
<td>Contact Output Status</td>
</tr>
<tr>
<td>D0117/D0118</td>
<td>Absolute Value Data Status</td>
</tr>
<tr>
<td>D0121/D0122</td>
<td>Current Velocity (Command Position) [pulses/s]</td>
</tr>
<tr>
<td>D0123/D0124</td>
<td>Current Velocity (Encoder Position) [pulses/s]</td>
</tr>
<tr>
<td>D0131 - D0134</td>
<td>(Calculation Work Area)</td>
</tr>
<tr>
<td>D0135 - D0138</td>
<td>(Calculation Work Area)</td>
</tr>
<tr>
<td>I00001 - I00016</td>
<td>Bit Data of Contact Input</td>
</tr>
<tr>
<td>I00017 - I00032</td>
<td>Bit Data of Contact Output</td>
</tr>
<tr>
<td>I00033 - I00048</td>
<td>Bit Data of Extended Status</td>
</tr>
</tbody>
</table>

Module Status Reading Program

```
M0033
0001: READ 4101 D0101 18
0002: MOV D0103 I00001
0003: MOV D0116 I00017
0004: MOV D0114 I00033
0005: D0131 = D0106 * 1000
0006: MOV D0132 D0121
0007: D0135 = D0110 * 1000
0008: MOV D0136 D0123
```

Figure 7.1 Module Status Reading Program
7.1.2 Set Parameter

This sets the entry parameters.

Items to Note:

- This command is accepted only if all the following conditions are satisfied: (1) servo is OFF and (2) no other commands are being executed. If these conditions are not satisfied, the command will then be ignored.

- Errors cannot be reset with the Error Reset command. If an error occurs with the Set Parameter command (error code 20xx), execute the Set Parameter command again with proper data. If other errors occur, they should be reset beforehand with the Error Reset instruction.

Procedures:

1) Write parameters on the positioning module with the WRITE instruction.

2) Set the Set Parameter output relay.

3) Reset Set Parameter output relay after confirming that Set Parameter ACK input relay is set. (If there is any error in the parameters, the Set Parameter ACK input relay is not set, but the Error Notification input relay is set. In this case, reset the Set Parameter output relay, go back to step 1 and retry.)

4) Check that Set Parameter ACK is reset.

Program Example:

In this example, parameters are set in the data register in advance. Since all entry parameters (including reserved parameters) are written at once with the WRITE instruction, it is necessary to set values of all parameters, including those whose values need not be changed from the initial value.
### List of devices used

<table>
<thead>
<tr>
<th>Device</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D0001/D0002</td>
<td>Positive Direction Limit Value</td>
</tr>
<tr>
<td>D0003/D0004</td>
<td>Negative Direction Limit Value</td>
</tr>
<tr>
<td>D0005/D0006</td>
<td>Velocity Limit Value</td>
</tr>
<tr>
<td>D0007/D0008</td>
<td>Over-limit velocity detection</td>
</tr>
<tr>
<td>D0009/D0010</td>
<td>Over-limit acceleration detection</td>
</tr>
<tr>
<td>D0011/D0012</td>
<td>Deviation error detection</td>
</tr>
<tr>
<td>D0013</td>
<td>Direction of motor’s rotation</td>
</tr>
<tr>
<td>D0014</td>
<td>Contact Input polarity</td>
</tr>
<tr>
<td>D0015</td>
<td>Contact output polarity</td>
</tr>
<tr>
<td>D0016</td>
<td>Position loop range</td>
</tr>
<tr>
<td>D0017</td>
<td>Velocity feedforward factor</td>
</tr>
<tr>
<td>D0018</td>
<td>Encoder method specification</td>
</tr>
<tr>
<td>D0019</td>
<td>Encoder multiplication number</td>
</tr>
<tr>
<td>D0020/D0021</td>
<td>Velocity/voltage ratio</td>
</tr>
<tr>
<td>D0022</td>
<td>(reserved)</td>
</tr>
<tr>
<td>D0023/D0024</td>
<td>Absolute offset value</td>
</tr>
<tr>
<td>D0025/D0026</td>
<td>Absolute pulse count</td>
</tr>
<tr>
<td>D0027</td>
<td>Sanyo Denki’s ABS bit length</td>
</tr>
<tr>
<td>D0028</td>
<td>Sanyo Denki’s ABS rotation direction</td>
</tr>
<tr>
<td>D0029</td>
<td>Contact input mode</td>
</tr>
<tr>
<td>Y00448</td>
<td>Set Parameter (Output Relay)</td>
</tr>
<tr>
<td>X00416</td>
<td>Set Parameter ACK (Input Relay)</td>
</tr>
<tr>
<td>I00099</td>
<td>Command Execution Prohibit Condition (set in another part)</td>
</tr>
<tr>
<td>I00101</td>
<td>Request to Execute command</td>
</tr>
<tr>
<td>I00102</td>
<td>Request to Execute command (Differentiate Up)</td>
</tr>
<tr>
<td>I00103</td>
<td>Waiting for ACK</td>
</tr>
<tr>
<td>I00104</td>
<td>Executing Command</td>
</tr>
<tr>
<td>I00105</td>
<td>Forced Release of Set Parameter</td>
</tr>
</tbody>
</table>

---

**Figure 7.2 Set Parameter Program**

```
00018
00020
00030
00037

0010
0011
0012
0013
0014
0015
0016
0017
0018

0101
0102
0103
0104
0105
0106
0107
0108

DIFU
WRITE
SET
RST

I00101
I00102
I00099
I00102
I00099

I00103
I00103
I00103

I00104
I00104
I00104

I00105
I00105
I00105

Y00448
Y00448
Y00448

X00416
X00416
X00416

X00105
X00105
X00105

D0001
D0001
D0001

4
1
29

```

IM 34M6H58-01E  1st Edition : Feb.01,1999-00
Figure 7.3  Set Parameter Program Time Chart
7.1.3 Error Reset

This resets the error status of the positioning module.

Items to Note:

- The positioning module ignores any commands other than the Error Reset command and Set Parameter command in error status (i.e., status where the Error Notification input relay is set). So be sure to execute the Error Reset command in error status.

  However, when an error occurs because of a parameter error in the Set Parameter command, the error cannot be reset using the Error Reset command.

- If the Error Reset command is executed for driver alarm detection error (error code “12”), the driver reset signal is set to ON for 500 ms and then switched off. Thereafter, the system waits for the reset of the driver alarm for a maximum length of 500 ms.

- Despite execution of the Error Reset command, if the cause of the error is not removed, the Error Notification will not be OFF.

Procedures:

1) Set the Error Reset output relay.
2) Check that the Error Notification relay is reset. Then reset the Error Reset output relay.

Program Example:

This example assumes that the Error Reset operation is done manually. All output relays of the positioning module are reset at the time of the Error Reset operation. The error code is preserved until the completion of Error Reset.

List of devices used

- Error Notification (Input Relay)
- Error Reset (Output Relay)
- Request to Reset Error (Manually Operated Signal)
- Request to Reset Error (Differentiate Up)
- Request to Reset Error (Differentiate Down)
- Error Code Storage Device

Figure 7.4 Error Reset Program
Figure 7.5 Error Reset Program Time Chart
7.1.4 Jog Stepping

When the Positive (Negative)-direction Jog Stepping output relay is ON, rotates the motor in the positive (negative) direction.

Items to Note:

- This command is accepted only if all of the following conditions are satisfied: (1) not in error state, (2) during servo ON and in End-of-positioning state, (3) in position control mode and (4) no other command executing. Otherwise, it is ignored.
- During jog stepping, you can stop the motor with the Stop Immediately command, but not with the Decelerate-and-Stop command. Terminate jog stepping first when decelerate and stop is desired.

Procedures:

1) Write required parameters of the Jog Stepping command on the positioning module.

2) When the Positive-direction (Negative-direction) Jog Stepping output relay is set, the motor rotates according to the parameters, the Operating-in Positive-direction (Negative-direction) input relay is set, and the End-of-Positioning input relay is reset, the End-of-Pulse Clear Out and End-of-Positioning input relays are set.

3) Reset the Positive-direction (Negative-direction) Jog Stepping output relay. The motor will be decelerated and stopped according to the parameters set at the start of the jog stepping. After the positioning module terminates the startup generation, the End-of-Pulse Clear Out input relay is set and the Operating-in-positive-direction (or negative-direction) input relay is reset. Then the End-of-positioning input relay is set when the difference between the encoder feedback value and the specified position value is less than or equal to the positioning judgement range.

Program Example:

In this example, the jog stepping starts with the Request for Jog Stepping, and ends with the release of the request. Jog stepping mode is reset automatically if any error occurs during jog stepping. This example shows only the positive-direction case. Required parameters are set in advance in the data registers.

The following program also occupies area for non-required parameters because all the parameters are written at once.
- List of devices used

<table>
<thead>
<tr>
<th>Device Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D0201/D0202</td>
<td>Target Velocity</td>
</tr>
<tr>
<td>D0203 - D0205</td>
<td>(unnecessary area)</td>
</tr>
<tr>
<td>D0206</td>
<td>Acceleration Time</td>
</tr>
<tr>
<td>D0207</td>
<td>Acceleration Mode Selection</td>
</tr>
<tr>
<td>D0208</td>
<td>Acceleration Parameter 1</td>
</tr>
<tr>
<td>D0209</td>
<td>Acceleration Parameter 2</td>
</tr>
<tr>
<td>D0210</td>
<td>Deceleration Time</td>
</tr>
<tr>
<td>D0211</td>
<td>Deceleration Mode Selection</td>
</tr>
<tr>
<td>D0212</td>
<td>Deceleration Parameter 1</td>
</tr>
<tr>
<td>D0213</td>
<td>Deceleration Parameter 2</td>
</tr>
<tr>
<td>D0214/D0215</td>
<td>Positioning Judgement Range</td>
</tr>
<tr>
<td>D0216</td>
<td>Positioning Time Out Interval</td>
</tr>
<tr>
<td>Y00442</td>
<td>Positive-direction Jog Stepping (Output Relay)</td>
</tr>
<tr>
<td>X00414</td>
<td>End-of-positioning (Input Relay)</td>
</tr>
<tr>
<td>X00412</td>
<td>Error Notification (Input Relay)</td>
</tr>
<tr>
<td>I00099</td>
<td>Command Execution Prohibit Condition (set in another part)</td>
</tr>
<tr>
<td>I00121</td>
<td>Jog Stepping Request</td>
</tr>
<tr>
<td>I00122</td>
<td>Jog Stepping Request (Differentiate Up)</td>
</tr>
<tr>
<td>I00123</td>
<td>Jog Stepping Request (Differentiate Down)</td>
</tr>
<tr>
<td>I00124</td>
<td>Executing Positive-direction Jog Stepping</td>
</tr>
</tbody>
</table>

Figure 7.6 Jog Stepping Program
Figure 7.7 Jog Stepping Program Time Chart
### 7.1.5 Origin-search

This executes origin-search. There are four external contact inputs related to origin-search. The origin-search mode specifies the action when individual rising and falling edges of these four inputs (eight in total) are detected during the origin-search using bit patterns (2 bits for each edge, so 16 bits in total).

In the example below, the condition of the search is as follows: The direction of the origin-search is negative, (1) Stop immediately when the rising edge of the Negative-Direction Limit is detected, (2) Decelerate and stop when the rising edge of the Origin input is detected, and (3) Shift to Z phase search when the falling edge of the Origin-Search is detected. In this case, the Origin Search mode is set in the following way:

```
15~                                     ~0
1 1 0 0 0 0 0 0 0 0 1 0 0 1 (=C009)
```

![Figure 7.8 Example of Origin-search Operation](image-url)
Origin-search Mode in detail:
The mode is set in bit units according to the list below. There are four ways of setting using two bits for the rising and falling edges of one external contact input.

<p>| | | | | | | | | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>14</td>
<td>13</td>
<td>12</td>
<td>11</td>
<td>10</td>
<td>9</td>
<td>8</td>
<td>7</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>01</td>
<td>10</td>
<td>11</td>
<td>00</td>
<td>01</td>
<td>10</td>
<td>11</td>
<td>00</td>
<td>01</td>
<td>10</td>
<td>11</td>
<td>00</td>
<td>01</td>
<td>10</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>Fall in origin input, ignore</td>
<td>Fall in origin input, Z-phase search</td>
<td>Fall in origin input, decelerate and stop</td>
<td>Fall in origin input, stop immediately</td>
<td>Rise in origin input, ignore</td>
<td>Rise in origin input, Z-phase search</td>
<td>Rise in origin input, decelerate and stop</td>
<td>Rise in origin input, stop immediately</td>
<td>Fall in external trigger input, ignore</td>
<td>Fall in external trigger input, Z-phase search</td>
<td>Fall in external trigger input, decelerate and stop</td>
<td>Fall in external trigger input, stop immediately</td>
<td>Rise in external trigger input, ignore</td>
<td>Rise in external trigger input, Z-phase search</td>
<td>Rise in external trigger input, decelerate and stop</td>
<td>Rise in external trigger input, stop immediately</td>
<td>Fall in positive direction limit input, ignore</td>
</tr>
</tbody>
</table>

Items to Note
This command is accepted only if all of the following conditions are satisfied: (1) not in error state, (2) servo-ON and during End-of-positioning, (3) in position control mode, and (4) no other command executing. Otherwise, it is ignored.

- The origin-search ends when a specified external contact input is detected and the search stopped. Change the parameters and re-execute the origin-search after it ends if you want to continue the search in another direction or with a different speed.

- In the following cases, an error occurs and the search is automatically stopped:
  - The limit input of the origin-search direction is set as ignored but the limit input is detected.
  - After shifting to the Z phase search, the limit input of the origin-search direction is detected (causes error regardless of the setting).
  - After shifting to the Z-phase search, Z-phase cannot be detected within the Z-phase search range.

- If you want to change the setup values according to the status of the external contact input at the beginning of the origin search, read the state with the Contact Input Status and then execute the origin-search.

- When the origin-search mode is “0”, the program does not check the External Contact input and Z-phase search is performed from operation startup.
Procedures:

1) Write parameters required by the Origin-Search command to the positioning module.

2) The motor rotates following the parameters when the output relay defined as “Start Origin-search” is set. The motor continues to rotate until a specified external contact input is detected.

3) When Z-pulse is detected for the specified number of times after shifting to Z-phase search (if this number is specified as “0”, then without shifting to Z-phase), current position is specified as “Position 0” (or the Origin-Offset value if it is defined) and the operation is decelerated and stopped. The End-of-Positioning and End-of-Origin-search input relays are set. After that, the End-of-Origin-Search is reset if the Start-Origin-Search is reset. However, if the Start-Origin-Search is reset before End-of-Origin-Search is set, then the End-of-Origin-Search will not be set. (The End-of-Origin-search input relay is set when there was no shift to Z-phase search and the operation is stopped with the Decelerate-and-Stop command or with the Stop Immediately command.)

- Bit 10 (End-of-Origin Search during the origin-search operation) of the extended status is set only when the operation is stopped after shifting to Z-phase search and detecting the Z-phase.

This bit is reset when the origin-search starts. It is “1” when the main switch is turned on.

Program Example:

This program starts the origin-search using the Request to Start Origin Search and stops it using the specified external contact input.

If any error is detected during the search, the search mode is automatically reset.

The following program also occupies area for non-required parameters because all the parameters are written at once.

All required parameters are set in advance in the data register.
- List of devices used

<table>
<thead>
<tr>
<th>Device Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D0301/D0302</td>
<td>Target velocity</td>
</tr>
<tr>
<td>D0303 - D0305</td>
<td>(unnecessary area)</td>
</tr>
<tr>
<td>D0306</td>
<td>Acceleration Time</td>
</tr>
<tr>
<td>D0307</td>
<td>Acceleration Mode Selection</td>
</tr>
<tr>
<td>D0308</td>
<td>Acceleration Parameter 1</td>
</tr>
<tr>
<td>D0309</td>
<td>Acceleration Parameter 2</td>
</tr>
<tr>
<td>D0310</td>
<td>Deceleration Time</td>
</tr>
<tr>
<td>D0311</td>
<td>Deceleration Mode Selection</td>
</tr>
<tr>
<td>D0312</td>
<td>Deceleration Parameter 1</td>
</tr>
<tr>
<td>D0313</td>
<td>Deceleration Parameter 2</td>
</tr>
<tr>
<td>D0314/D0315</td>
<td>Positioning Judgement Range</td>
</tr>
<tr>
<td>D0316</td>
<td>Positioning Time Out interval</td>
</tr>
<tr>
<td>D0317 - D0321</td>
<td>(unnecessary area)</td>
</tr>
<tr>
<td>D0322</td>
<td>Origin-search Mode</td>
</tr>
<tr>
<td>D0323</td>
<td>Origin-search Direction</td>
</tr>
<tr>
<td>D0324</td>
<td>Z-phase Edge Selection</td>
</tr>
<tr>
<td>D0325</td>
<td>Z-phase Pulse Count</td>
</tr>
<tr>
<td>D0326/D0327</td>
<td>Z-phase Search Range</td>
</tr>
<tr>
<td>D0328/D0329</td>
<td>Origin-Offset Value</td>
</tr>
<tr>
<td>Y00437</td>
<td>Start Origin-search (Output Relay)</td>
</tr>
<tr>
<td>X00405</td>
<td>End of Origin-search (Input Relay)</td>
</tr>
<tr>
<td>X00412</td>
<td>Error Notification (Input Relay)</td>
</tr>
<tr>
<td>I00099</td>
<td>Command Execution Prohibit Condition (set in another part)</td>
</tr>
<tr>
<td>I00131</td>
<td>Requesting Origin-search</td>
</tr>
<tr>
<td>I00132</td>
<td>Requesting Origin-search (Differentiate Up)</td>
</tr>
<tr>
<td>I00133</td>
<td>During Origin-search Operation</td>
</tr>
<tr>
<td>I00134</td>
<td>Origin-search during Execution</td>
</tr>
</tbody>
</table>

Figure 7.9 Origin-search Program
Figure 7.10 Origin-search Program Time Chart
7.1.6 Write Current Position

When the servo is ON, this operation changes the current position of the axis at end-of-positioning status in the positioning control mode.

When the servo is OFF, it changes the encoder feedback position.

Items to Note:

- This command is accepted only if all of the following conditions are satisfied: (1) not in error state, (2) servo OFF or during end-of-positioning with servo ON, (3) in position control mode, and (4) not executing other commands. Otherwise, it is ignored.

- The change position should fall in the range from the positive-direction limit to the negative-direction limit. Error occurs when requesting to change beyond the limits of the range.

- It is possible to write the current position in YASKAWA Electric absolute-value encoder system. In this case, positions such as the current position or the command position after this operation are not related to the absolute value, though the absolute-value data is in the absolute-value status.

- It is impossible to write current position for Sanyo type absolute-value encoder system.

Procedures:

1) Write the desired position after the change in the Target Position parameter.
2) Set the output relay defined as “Request to Write Current Position.”
3) Reset the relay defined as “Request to Write Current Position” after confirming that the input relay defined as “Current Position Write ACK” is set. If there is a parameter error, the input relay defined as “Error Notification” is set and the Current Position Write ACK is not set.

Program Example:

This is an example of writing the current position. All required parameters are set in advance in the data register.

- List of devices used

<table>
<thead>
<tr>
<th>Device</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D0401/D0402</td>
<td>Target Position (Current Position to change)</td>
</tr>
<tr>
<td>Y00444</td>
<td>Request to Write Current Position (Output Relay)</td>
</tr>
<tr>
<td>X00411</td>
<td>Current Position Write ACK (Input Relay)</td>
</tr>
<tr>
<td>I00099</td>
<td>Command Execution Prohibit Condition (set in another part)</td>
</tr>
<tr>
<td>I00141</td>
<td>Request to Write Current Position</td>
</tr>
<tr>
<td>I00142</td>
<td>Request to Write Current Position (Differentiate Up)</td>
</tr>
<tr>
<td>I00143</td>
<td>Waiting for ACK</td>
</tr>
<tr>
<td>I00144</td>
<td>Executing Command</td>
</tr>
</tbody>
</table>
Figure 7.11 Current Position Writing Program

Figure 7.12 Current Position Writing Program Time Chart
7.1.7 Position Control Mode Operation

(1) Start Operating Instruction (Positioning Operation)

When executed during servo ON and End-of-Positioning, normal positioning operation starts. When executed during positioning operation, two overlapping positioning operations (called on-route operation) start. The latter case will be explained in the section on on-route operation.

Items to Note:

- This command is accepted only if all of the following conditions are satisfied: (1) not in error state, (2) not executing other commands, (3) during servo ON, (4) not during origin search, and (5) not during jog stepping. Otherwise, it is ignored. Further, in the case of normal positioning, there are 2 additional conditions, (6) in End-of-Positioning status, (7) in position control mode. If Items (6) and (7) above are not satisfied, an error operation results.

- If an invalid value is set for a startup parameter, a parameter error results and the motor does not move. The servo is not switched OFF if a parameter error occurs.

Procedures:

1) Write the required parameters in the positioning module.

2) Set the output relay defined as “Request to Start Operation.”

3) After confirming that the Start Operation Command ACK input relay is set, reset the Start Operation relay. The motor starts according to timing when the Start Operation ACK is set. (If there is a parameter error, Start Operation ACK input relay is not set and Error Notification input relay is set.)

4) When the path generated by the positioning module reaches the target position, the End-of-Pulse-Clear-Out Input relay is set. After this, if the difference between the target position (command position) and the encoder feedback position is less than or equal to the Positioning Judgement Pulse Count, the End-of-Positioning input relay is set. However, if the time from the End-of-Pulse-clear-out to End-of-Positioning exceeds the specified value for the Positioning Timeout Interval, an error results.

Program Example:

This is an example of a simple positioning operation. All required parameters are set in advance in the data register.
• List of devices used

<table>
<thead>
<tr>
<th>Device Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D0501/D0502</td>
<td>Target Velocity</td>
</tr>
<tr>
<td>D0503/D0504</td>
<td>Target Position</td>
</tr>
<tr>
<td>D0505</td>
<td>Target Position Mode</td>
</tr>
<tr>
<td>D0506</td>
<td>Acceleration Time</td>
</tr>
<tr>
<td>D0507</td>
<td>Acceleration Mode Selection</td>
</tr>
<tr>
<td>D0508</td>
<td>Acceleration Parameter 1</td>
</tr>
<tr>
<td>D0509</td>
<td>Acceleration Parameter 2</td>
</tr>
<tr>
<td>D0510</td>
<td>Deceleration Time</td>
</tr>
<tr>
<td>D0511</td>
<td>Deceleration Mode Selection</td>
</tr>
<tr>
<td>D0512</td>
<td>Deceleration Parameter 1</td>
</tr>
<tr>
<td>D0513</td>
<td>Deceleration Parameter 2</td>
</tr>
<tr>
<td>D0514/D0515</td>
<td>Positioning Judgement Range</td>
</tr>
<tr>
<td>D0516</td>
<td>Positioning Time Out Interval</td>
</tr>
<tr>
<td>D0517</td>
<td>Interpolation Mode Selection</td>
</tr>
<tr>
<td>D0518</td>
<td>Startup Mode Selection</td>
</tr>
<tr>
<td>D0519</td>
<td>Position-detection Mode</td>
</tr>
<tr>
<td>D0520/D0521</td>
<td>Position-detection Value</td>
</tr>
<tr>
<td>Y00433</td>
<td>Start Operation Command (Output Relay)</td>
</tr>
<tr>
<td>X00401</td>
<td>Start Operation Command ACK (Input Relay)</td>
</tr>
<tr>
<td>X00414</td>
<td>End-of-positioning (Input Relay)</td>
</tr>
<tr>
<td>X00415</td>
<td>End-of-pulse-clear-out (Input Relay)</td>
</tr>
<tr>
<td>I00099</td>
<td>Command Execution Prohibit Condition (set in another part)</td>
</tr>
<tr>
<td>I00201</td>
<td>Request to Start Operation Command</td>
</tr>
<tr>
<td>I00202</td>
<td>Request to Start Operation Command (Differentiate Up)</td>
</tr>
<tr>
<td>I00203</td>
<td>Waiting for Command ACK</td>
</tr>
<tr>
<td>I00204</td>
<td>Positioning In Progress</td>
</tr>
</tbody>
</table>

Figure 7.13  Positioning Operation Program
Application Function:

- Startup Mode Selection

When startup mode “1” is selected, the motor does not start moving until the external trigger input becomes ON after command execution. This mode is useful when it is desirable to wait for an external-synthesizing signal before starting.

When startup mode “2” is selected, the motor does not start moving until the output relay of an internal trigger is set after command execution.

- Position-detection Notification Function

If position detection mode “1” or “2” is set, the input relay defined as “Position Detection Notification” is set to notify the application program when the current position arrives at the specified value for Position Detection Notification.

The relay can be used as interrupt input.

The relay is reset when the output relay labeled “Reset Position Detection Notification” is set.

The instructed position is detected when the position detection mode “1” is set, whereas the encoder feedback position is detected when the position detection mode “2” is set.
(2) Change Target Position

When executed during positioning operation, this changes the target position immediately and then performs the positioning operation with the new target position. You can change the target velocity when changing the target position. In this case, the acceleration/deceleration between the velocities before and after the change is expressed by the slope calculated from the new target velocity and the acceleration time/deceleration time.

Items to Note:

• This command is accepted only if all of the following conditions are satisfied: (1) not in error state, (2) not executing other commands, (3) servo is ON in position control mode, (4) not in the on-route operation interval, (5) not under arc interpolation, (6) not under origin-search, and (7) not under jog stepping. Otherwise, it is ignored.

• A parameter error occurs if invalid values are set to the startup parameters. The servo is switched OFF and the motor stops. (The servo is not switched OFF if the command is executed during end-of-positioning.)

• If the target velocity is changed, a two-step or S-shaped acceleration or deceleration will not be obtained. However, the deceleration at the time of stopping at the target point after changing in the target position follows the deceleration mode and deceleration set time.

Procedures:

1) Write the required parameters in the positioning module.
2) Set the output relay defined as “Request to Change Target Position.”
3) Reset the relay defined as “Request to Change Target Position” after confirming that the input relay defined as “Target Position Change ACK” is set. (If there is a parameter error, the input relay defined as “Target Position Change ACK” is not set and the input relay defined as “Error Notification” is set.)
4) The end-of-positioning is judged according to the parameters at the time of target position change.

Program Example:

This is an example of changing the target position during positioning operation. All required parameters are set in advance in the data register.

The following program has areas for unnecessary parameters because all the parameters are written at once.
• List of devices used

<table>
<thead>
<tr>
<th>Device</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D0601/D0602</td>
<td>Target Velocity</td>
</tr>
<tr>
<td>D0603/D0604</td>
<td>Target Position</td>
</tr>
<tr>
<td>D0605</td>
<td>Target Position Mode</td>
</tr>
<tr>
<td>D0606</td>
<td>Acceleration Time</td>
</tr>
<tr>
<td>D0607</td>
<td>Acceleration Mode Selection</td>
</tr>
<tr>
<td>D0608</td>
<td>Acceleration Parameter 1</td>
</tr>
<tr>
<td>D0609</td>
<td>Acceleration Parameter 2</td>
</tr>
<tr>
<td>D0610</td>
<td>Deceleration Time</td>
</tr>
<tr>
<td>D0611</td>
<td>Deceleration Mode Selection</td>
</tr>
<tr>
<td>D0612</td>
<td>Deceleration Parameter 1</td>
</tr>
<tr>
<td>D0613</td>
<td>Deceleration Parameter 2</td>
</tr>
<tr>
<td>D0614/D0615</td>
<td>Positioning Judgement Range</td>
</tr>
<tr>
<td>D0616</td>
<td>Positioning Time Out Interval</td>
</tr>
<tr>
<td>D0617</td>
<td>(unnecessary area)</td>
</tr>
<tr>
<td>D0618</td>
<td>Startup Mode Selection</td>
</tr>
<tr>
<td>D0619</td>
<td>Position detection Mode</td>
</tr>
<tr>
<td>D0620/D0621</td>
<td>Position-detection Value</td>
</tr>
<tr>
<td>Y00439</td>
<td>Request to Change Target Position (Output Relay)</td>
</tr>
<tr>
<td>X00409</td>
<td>Change Target Position ACK (Input Relay)</td>
</tr>
<tr>
<td>I00099</td>
<td>Command Execution Prohibit Condition (set in another part)</td>
</tr>
<tr>
<td>I00211</td>
<td>Request to Start Target Position Change</td>
</tr>
<tr>
<td>I00212</td>
<td>Request to Start Target Position Change (Differentiate Up)</td>
</tr>
<tr>
<td>I00213</td>
<td>Waiting for Command ACK</td>
</tr>
<tr>
<td>I00214</td>
<td>Executing Command</td>
</tr>
</tbody>
</table>

Figure 7.15  Target Point Changing Program
Application Function:

- **Startup Mode Selection**

  When startup mode “1” is selected, the Change Target Position operation is not started until the external trigger input becomes ON after command execution. This mode is useful when it is desirable to wait for an external-synthesizing signal before starting the operation.

  When startup mode “2” is selected, the operation is not started until the output relay of an internal trigger is set after command execution.

- **Position Detection Notification Function**

  The Position Detection Notification value defined for the positioning operation before the Change Target Position operation is changed to the new specified value when the Change Target Position operation is executed. (If no change is necessary, execute the operation after setting the same values.)

- **Change Target Position with a change in operation direction**

  If the new position has already been passed at the time of executing Change Target Position, Decelerate-and-Stop is performed immediately and the positioning operation with the new target position starts. (If the new position has not yet been passed at the time of executing Change Target Position, but will be passed even if the deceleration is started immediately, the same steps occur.)

  When Change Target Position with Change Operation Direction is executed, another Change Target Position command or Change Velocity command in the first deceleration operation cannot be accepted. You can confirm the status by reading the extended status.

- **Change Target Position during End of Positioning**

  You can execute the Change Target Position command during End of Positioning (when motor is not moving). In this case, the operation is the same as the normal positioning operation.
(3) Change Velocity

Changes the operation speed immediately when velocity changes are made during positioning operation.

The acceleration or deceleration from a change in velocity is expressed by the slope determined by a new target velocity and acceleration time. (The acceleration time parameters are also used for deceleration in velocity changes where the new target velocity is slower than the velocity set previously.)

After the velocity changes have been made, a new target velocity is attained following the Deceleration Mode/Deceleration Time specified with the Velocity change command.

Items to Note:

- This command is accepted only if all of the following conditions are satisfied: (1) not in error state, (2) not executing other commands, (3) during servo ON, (4) not in the on-route interval of an on-route operation, (5) not during origin search, and (6) not during jog stepping. Otherwise, it is ignored. In the velocity control mode, the operation will be different.

- A parameter error occurs if invalid values are set to the startup parameters. The servo is switched OFF and the motor stops. (The servo is not switched OFF if the instruction is executed during end-of-positioning.)

Procedures:

1) Write the required parameters to the positioning module.
2) Set the output relay defined as “Request to Change Velocity.”
3) Reset the relay defined as “Request to Change Velocity” after confirming that the input relay defined as “Velocity Change ACK” is set. The motor starts changing velocity when Velocity Change ACK is set. (If there is a parameter error, the input relay defined as “Change Input ACK” is not set and the input relay defined as “Error Notification” is set.)
4) The end-of-positioning is judged according to the parameters specified before the velocity change.

Program Example:

This is an example of changing the velocity during a positioning operation. All required parameters are set in advance in the data register.

The following program also has areas for unnecessary parameters because all the parameters are written at once.
- List of devices used

<table>
<thead>
<tr>
<th>Device</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D0701/D0702</td>
<td>Target velocity</td>
</tr>
<tr>
<td>D0703 - D0705</td>
<td>(unnecessary area)</td>
</tr>
<tr>
<td>D0706</td>
<td>Acceleration Time</td>
</tr>
<tr>
<td>D0707 - D0709</td>
<td>(unnecessary area)</td>
</tr>
<tr>
<td>D0710</td>
<td>Deceleration Time</td>
</tr>
<tr>
<td>D0711</td>
<td>Deceleration Mode Selection</td>
</tr>
<tr>
<td>D0712</td>
<td>Deceleration Parameter 1</td>
</tr>
<tr>
<td>D0713</td>
<td>Deceleration Parameter 2</td>
</tr>
<tr>
<td>D0714 - D0716</td>
<td>(unnecessary area)</td>
</tr>
<tr>
<td>D0717</td>
<td>Interpolation Mode Selection</td>
</tr>
<tr>
<td>D0718</td>
<td>Startup Mode Selection</td>
</tr>
<tr>
<td>Y00440</td>
<td>Request to Change Speed (Output Relay)</td>
</tr>
<tr>
<td>X00410</td>
<td>Change Velocity ACK (Input Relay)</td>
</tr>
<tr>
<td>I00099</td>
<td>Command Execution Prohibit Condition (set in another part)</td>
</tr>
<tr>
<td>I00221</td>
<td>Request to Start Change Speed</td>
</tr>
<tr>
<td>I00222</td>
<td>Request to Start Change Speed (Differentiate Up)</td>
</tr>
<tr>
<td>I00223</td>
<td>Waiting for Command ACK</td>
</tr>
<tr>
<td>I00224</td>
<td>Executing Command</td>
</tr>
</tbody>
</table>

(0073) Velocity Changing Program (during positioning operation)

```
00180
  I00221
  DIFU I00222

00182
  WRITE D0701 4
  SET Y00440
  SET I00223

00192
  RST Y00440
  RST I00223

00201
  RST I00224
```

Figure 7.17 Velocity Changing Program (during positioning operation)
Application Function:

- Startup Mode Selection

When startup mode “1” is selected, the velocity change operation is not started until the external trigger input becomes ON after executing the command. This mode is useful when startup mode it is desirable to wait for an external-synthesizing signal before starting.

When startup mode “2” is selected, the operation is not started until the output relay of an internal trigger is set after executing the command.
7.1.8 Velocity Control Mode Operation

Velocity control mode can be used only in an incremental encoder system. An absolute value encoder system cannot use this mode.

Specified positive-direction and negative-direction limit values are ignored when this mode is in effect.

(1) Switching Position to Velocity Control

The positioning module is in position control mode when the servo is ON. It is necessary to switch the mode to “velocity control” in order to perform velocity control operation.

Items to Note:

- This command is accepted only if all of the following conditions are satisfied: (1) not in error state, (2) not executing other command (3) servo is ON, (4) not in the on-route interval of an on-route operation, (5) not during an arc interpolated operation, (6) not during origin search, and (7) not during jog stepping. Otherwise, it is ignored.

- When switching position to velocity control, no parameters, other than the Switch Control Mode parameter is referred to. The control is switched to velocity control while maintaining the same velocity.

Procedures:

1) Write “0” (switch to velocity control) as the value of the Switch Control Mode parameter to the positioning module.

2) Set the output relay defined as “Request to Switch Control Mode.”

3) Reset the relay defined as “Request to Switch Control Mode” after confirming that the input relay defined as “Switch Control Mode ACK” is set. The motor is switched to speed control mode while maintaining the same velocity.

Program Example:

This is an example of switching position to velocity control.

- List of devices used

<table>
<thead>
<tr>
<th>Device</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y00438</td>
<td>Request to Switch Control Mode (Output Relay)</td>
</tr>
<tr>
<td>X00406</td>
<td>Switch Control Mode ACK (Input Relay)</td>
</tr>
<tr>
<td>I00099</td>
<td>Command Execution Prohibit Condition (set in another part)</td>
</tr>
<tr>
<td>I00301</td>
<td>Request to Switch Control Mode</td>
</tr>
<tr>
<td>I00302</td>
<td>Request to Switch Control Mode (Differentiate Up)</td>
</tr>
<tr>
<td>I00303</td>
<td>Waiting for Command ACK</td>
</tr>
<tr>
<td>I00304</td>
<td>Executing Command</td>
</tr>
</tbody>
</table>
Figure 7.19 Position to Velocity Control Switching Program

Figure 7.20 Position to Velocity Control Switching Program Time Chart
(2) Start Operation Command (Velocity Control Operation)

Velocity control operation starts when you execute the Start Operation command in velocity control mode. This command starts the operation only. To stop a motor in velocity control operation, the Decelerate-and-Stop command or Stop Immediately command must be executed.

Items to Note:

- This command is accepted only if all of the following conditions are satisfied: (1) not in error state, (2) not executing other command, (3) servo is ON, (4) not during origin-search, and (5) not during the jog stepping. Otherwise, it is ignored. When starting during velocity control, the additional required conditions are (6) in End-of-Positioning state, (7) in Velocity Control mode. (If Items (6) and (7) above are not satisfied, an error occurs in the operation.)
- A parameter error occurs if invalid values are set to the startup parameters. The motor does not move. The servo is not switched OFF when there are errors in the startup parameters.
- In velocity control mode, you must write a positive (negative) value as the target set velocity for the positive (negative)-direction operation.

 Procedures:

1) Write required parameters to the positioning module
2) Set the output relay defined as “Start Operation Command.”
3) Reset the relay defined as “Start Operation Command” after confirming that the input relay defined as “Start Operation Command ACK” is set. The motor starts the operation when Start Operation Command ACK is set. (If there is a parameter error, the input relay defined as “Start Operation Command ACK” is not set and the input relay defined as “Error Notification” is set.)

Program Example:

This is an example of starting with the velocity control operation. All required parameters are set in advance in the data register.
- List of devices used

<table>
<thead>
<tr>
<th>Device</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D0801/D0802</td>
<td>Target velocity</td>
</tr>
<tr>
<td>D0803 - D0805</td>
<td>(unnecessary area)</td>
</tr>
<tr>
<td>D0806</td>
<td>Acceleration Time</td>
</tr>
<tr>
<td>D0807</td>
<td>Acceleration Mode Selection</td>
</tr>
<tr>
<td>D0808</td>
<td>Acceleration Parameter 1</td>
</tr>
<tr>
<td>D0809</td>
<td>Acceleration Parameter 2</td>
</tr>
<tr>
<td>D0810</td>
<td>Deceleration Time</td>
</tr>
<tr>
<td>D0811</td>
<td>Deceleration Mode Selection</td>
</tr>
<tr>
<td>D0812</td>
<td>Deceleration Parameter 1</td>
</tr>
<tr>
<td>D0813</td>
<td>Deceleration Parameter 2</td>
</tr>
<tr>
<td>D0814/D0815</td>
<td>Positioning Judgement Range</td>
</tr>
<tr>
<td>D0816</td>
<td>Positioning Time out Interval</td>
</tr>
<tr>
<td>D0817</td>
<td>(unnecessary area)</td>
</tr>
<tr>
<td>D0818</td>
<td>Startup Mode Selection</td>
</tr>
<tr>
<td>Y00433</td>
<td>Start Operation Command (Output Relay)</td>
</tr>
<tr>
<td>X00401</td>
<td>Start Operation Command ACK (Input Relay)</td>
</tr>
<tr>
<td>X00414</td>
<td>End of Positioning (Input Relay)</td>
</tr>
<tr>
<td>I00099</td>
<td>Command Execution Prohibit Condition (set in another part)</td>
</tr>
<tr>
<td>I00311</td>
<td>Request to Start Operation</td>
</tr>
<tr>
<td>I00312</td>
<td>Request to Start Control Mode (Differentiate Up)</td>
</tr>
<tr>
<td>I00313</td>
<td>Waiting for Command ACK</td>
</tr>
<tr>
<td>I00314</td>
<td>Executing Command</td>
</tr>
</tbody>
</table>

Figure 7.21 Start Velocity Control program
Application Function:

- Startup Mode Selection

When startup mode “1” is selected, the velocity control operation is not started until the external trigger input becomes ON after command execution. This mode is useful when it is desirable to wait for an external-synthesizing signal before starting.

When startup mode “2” is selected, the operation does not start until the output relay of an internal trigger is set after command execution.
(3) Change Velocity

Changes the operating speed when velocity changes are made during an operation in velocity control mode.

The acceleration or deceleration from a change in velocity is expressed by the slope determined by a new target velocity and acceleration time. (The acceleration time parameters are also used for deceleration in velocity changes where the new target velocity is slower than the velocity set previously.)

When stopping a motor with a Decelerate and Stop command after changing velocity, the system follows the deceleration mode / deceleration time set at the execution of the Change Velocity command.

Items to Note:

- This command is accepted only if all of the following conditions are satisfied: (1) not in the error state, (2) not executing other command (3) servo is ON, (4) not in the on-route interval of on-route operation, (5) not during origin search, and (6) not during jog stepping. Otherwise, it is ignored. In the position control mode, the operation will be different.

- A parameter error occurs if invalid values are set to the startup parameters. The servo is switched OFF and the motor stops. (The servo is not switched OFF if the command is executed during end-of-positioning.)

- In velocity control mode, you must write a positive or negative value as the target velocity for the positive- or negative-direction operation.

- You cannot change the operation direction with the Velocity Change command. (You cannot change to a negative target velocity during a positive-direction operation, nor change to a positive target velocity during a negative-direction operation.) To change the operation direction in velocity control mode, you must stop the motor with the Deceleration Stop command and then restart the motor in the opposite direction.

Procedures:

1) Write the required parameters to the positioning module.

2) Set the output relay defined as “Request to Change Velocity.”

3) Reset the relay defined as “Request to Change Velocity” after confirming that the input relay defined as “Change Velocity ACK” is set. The motor starts the operation when the Change Velocity ACK is set. (If there is a parameter error, the input relay defined as “Change Velocity ACK” is not set and the input relay defined as “Error Notification” is set.)

4) The end-of-positioning at the stopping time is judged according to the parameters specified before changing velocity.

Program Example:

This is an example of changing velocity during position control operation. All required parameters are set in advance in the data register.

The following program also occupies areas for unnecessary parameters because all the parameters are written at once.
## List of devices used

<table>
<thead>
<tr>
<th>Device Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D0901/D0902</td>
<td>Target velocity</td>
</tr>
<tr>
<td>D0903 - D0905</td>
<td>(unnecessary area)</td>
</tr>
<tr>
<td>D0906</td>
<td>Acceleration Time</td>
</tr>
<tr>
<td>D0907 - D0909</td>
<td>(unnecessary area)</td>
</tr>
<tr>
<td>D0910</td>
<td>Deceleration Time</td>
</tr>
<tr>
<td>D0911</td>
<td>Deceleration Mode Selection</td>
</tr>
<tr>
<td>D0912</td>
<td>Deceleration Parameter 1</td>
</tr>
<tr>
<td>D0913</td>
<td>Deceleration Parameter 2</td>
</tr>
<tr>
<td>D0914 - D0917</td>
<td>(unnecessary area)</td>
</tr>
<tr>
<td>D0918</td>
<td>Startup Mode Selection</td>
</tr>
<tr>
<td>Y00440</td>
<td>Request to Change Velocity (Output Relay)</td>
</tr>
<tr>
<td>X00410</td>
<td>Change Velocity ACK (Input Relay)</td>
</tr>
<tr>
<td>I00321</td>
<td>Request to Change Velocity</td>
</tr>
<tr>
<td>I00322</td>
<td>Request to Change Velocity (Differentiate Up)</td>
</tr>
<tr>
<td>I00323</td>
<td>Waiting for Command ACK</td>
</tr>
<tr>
<td>I00324</td>
<td>Executing Command</td>
</tr>
</tbody>
</table>

### Change Velocity (during Speed Control Operation) Program

```
<table>
<thead>
<tr>
<th>(0101)</th>
<th>Change Velocity (during Speed Control Operation) Program</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0102)</td>
<td>I00321</td>
</tr>
<tr>
<td>(0103)</td>
<td>WRITE D0901 4 41 18</td>
</tr>
<tr>
<td>(0104)</td>
<td>SET Y00440</td>
</tr>
<tr>
<td>(0105)</td>
<td>SET I00323</td>
</tr>
<tr>
<td>(0106)</td>
<td>SET I00324</td>
</tr>
<tr>
<td>(0107)</td>
<td>RST Y00440</td>
</tr>
<tr>
<td>(0108)</td>
<td>RST I00323</td>
</tr>
<tr>
<td>(0109)</td>
<td>RST I00324</td>
</tr>
<tr>
<td>(0110)</td>
<td></td>
</tr>
</tbody>
</table>
```

### Figure 7.23  Change Velocity (during Velocity Control Operation) Program
Application Function:

- Startup Mode Selection

When startup mode “1” is selected, the velocity change operation is not started until the external trigger input becomes ON after command execution. This mode is useful when it is desirable to wait for an external-synthesizing signal before starting.

When startup mode “2” is selected, the velocity change operation does not start until the output relay of an internal trigger is set after command execution.
7.1.9 Switching Velocity to Position Control

When the velocity control is switched to the position control, the module is in position-control mode and starts positioning from scratch following the preset parameters.

When switching control mode with velocity change,

The acceleration or deceleration from a change in velocity is expressed by the slope determined by a new target velocity and acceleration time. (The acceleration time parameters are also used for deceleration in velocity changes where a new target velocity is slower than the velocity set previously.)

When stopping after completing the positioning operation, the deceleration time is that set at the switch control mode execution.

Items to Note:

• This command is accepted only if all of the following conditions are satisfied: (1) not in error state, (2) not executing other command, (3) servo is ON, (4) not in the on-route interval of on-route operation, (5) not during the arc-interpolated operation, (6) not during origin search, and (7) not during jog stepping. Otherwise, it is ignored. When executed in position control mode, the input relay defined as “Switch Control Mode ACK” is set, but the state of the positioning module is not changed.

• A parameter error occurs if invalid values are set to the startup parameters. The motor stops with the servo OFF. (Servo is not switched OFF when operated during the end-of-positioning.)

• When switching velocity control to position control, the positioning is performed with the target velocity and target position effective at the time of the switch. Therefore, when you execute switch control while stopping in Velocity Control Mode at a target position other than “0”, the positioning operation starts.

Procedures:

1) Write the required parameters to the positioning module.

2) Set the output relay defined as “Request to Switch Control Mode.”

3) Reset the relay defined as “Request to Switch Control Mode” after confirming that the input relay defined as “Switch Control Mode ACK” is set. When “Switch Control Mode ACK” is set, the mode is switched to position control mode and the positioning operation starts. (If there is a parameter error, the input relay defined as “Switch Control Mode ACK” is not set and the input relay defined as “Error Notification” is set.)

Program Example:

This is an example of switching velocity control to position control. All required parameters are set in advance in the data register.

The following program also occupies areas for unnecessary parameters because all the parameters are written at once.
- List of devices used

<table>
<thead>
<tr>
<th>Device</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1001/D1002</td>
<td>Target Velocity</td>
</tr>
<tr>
<td>D1003/D1004</td>
<td>Target Position</td>
</tr>
<tr>
<td>D1005</td>
<td>Target Position Mode</td>
</tr>
<tr>
<td>D1006</td>
<td>Acceleration Time</td>
</tr>
<tr>
<td>D1007 - D1009</td>
<td>(unnecessary area)</td>
</tr>
<tr>
<td>D1010</td>
<td>Deceleration Time</td>
</tr>
<tr>
<td>D1011</td>
<td>Deceleration Mode Selection</td>
</tr>
<tr>
<td>D1012</td>
<td>Deceleration Parameter 1</td>
</tr>
<tr>
<td>D1013</td>
<td>Deceleration Parameter 2</td>
</tr>
<tr>
<td>D1014/D1015</td>
<td>Positioning Judgement Range</td>
</tr>
<tr>
<td>D1016</td>
<td>Positioning Time out Interval</td>
</tr>
<tr>
<td>D1017</td>
<td>(unnecessary area)</td>
</tr>
<tr>
<td>D1018</td>
<td>Startup Mode Selection</td>
</tr>
<tr>
<td>D1019</td>
<td>Position Detection Mode</td>
</tr>
<tr>
<td>D1020/D1021</td>
<td>Position Detection Value</td>
</tr>
<tr>
<td>Y00438</td>
<td>Request to Switch control Mode (Output Relay)</td>
</tr>
<tr>
<td>X00406</td>
<td>Switch control Mode ACK (Input Relay)</td>
</tr>
<tr>
<td>X00414</td>
<td>End-of-positioning (Input Relay)</td>
</tr>
<tr>
<td>I00099</td>
<td>Command Execution Prohibit Condition (set in another part)</td>
</tr>
<tr>
<td>I00331</td>
<td>Request to Switch control Mode</td>
</tr>
<tr>
<td>I00332</td>
<td>Request to Switch control Mode (Differentiate Up)</td>
</tr>
<tr>
<td>I00333</td>
<td>Waiting for Command ACK</td>
</tr>
<tr>
<td>I00334</td>
<td>Executing Command</td>
</tr>
</tbody>
</table>

---

**Figure 7.25  Velocity Control to Position Control Switching Program**

```
00280 00282
(0112)  (0113)
I00331  I00332 I0099

(0114)  (0115)  (0116)

(0117)  (0118)
00292 00301
0171
I00333  Y00438 X00406

DIFU  I00332
WRITE  D1001  4  41  21
SET  Y00438
SET  I00333
SET  I00334
RST  Y00438
RST  I00333
RST  I00334
```

---

IM 34M6H58-01E  1st Edition : Feb.01,1999-00
Application Function:

- **Startup Mode Selection**
  When startup mode “1” is selected, the switch to position control operation is not started until the external trigger input becomes ON after command execution. This mode is useful when it is desirable to wait for an external-synthesizing signal before starting the operation.
  
  When startup mode “2” is selected, the operation is not performed until the output relay of an internal trigger is set after command execution.

- **Position-Detection Notification Function**
  If position-detection mode “1” or “2” is set, the input relay defined as “Position Detection Notification” is set to notify the application program when the current position arrives at the specified value for position detection.
  
  The relay can be used as an interrupt input.
  
  The relay is reset when the output relay defined as “Position Detection Reset Notification” is set.
  
  The position detected is the command position when the Position Detection Mode is “1”, and the encoder feedback position when it is “2”.

- **Switching velocity control to position control with operation direction change**
  If the operation direction in velocity control is different from the direction of the target position after switching, “Decelerate-and-Stop” is performed immediately and the positioning operation with the new target position starts. (If the operation direction does not change, but the new target position will be passed if the deceleration is started immediately, the same step occurs.)
7.1.10 Request to Decelerate and Stop

Decelerates and Stops the operating motor during positioning operation, velocity control operation, etc. The slope of deceleration during Decelerate and Stop is determined from the setup values at startup. (When a command that sets the deceleration mode is executed again during operation, then the setup value at that moment is used for the above purpose.)

Items to Note:

- This command is accepted only if all of the following conditions are satisfied: (1) not in error state, (2) not executing other command, (3) servo is ON, and (4) not during jog stepping. Otherwise, it is ignored.
- If the system is in an on-route operation interval, it decelerates and stops after terminating the on-route operation. Execute the Immediate Stop command in order to stop in on-route operation interval.
- You cannot execute the Decelerate and Stop command during jog-stepping operation. Terminate jog stepping, if necessary.
- The positioning module also accepts Decelerate and Stop command during end-of-positioning. (The corresponding ACK is set.)
- A new positioning operation does not start nor is a Target Position Change command accepted while decelerating with the Decelerate and Stop command. This state can be checked by reading the extended status.

Procedures:

1) Set the output relay defined as “Request to Decelerate and Stop.”
2) Reset the relay defined as “Request to Decelerate and Stop” after confirming that the input relay defined as “Deceleration and Stop ACK” is set. The motor starts decelerating when Deceleration and Stop ACK is set.
3) The input relay defined as “End of Positioning” is set when the motor stops.

Program Example:

This is an example of request for decelerating stop.

- List of devices used

<table>
<thead>
<tr>
<th>Device</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y00435</td>
<td>Request to Decelerat and Stop (Output Relay)</td>
</tr>
<tr>
<td>X00403</td>
<td>Deceleration and Stop ACK (Input Relay)</td>
</tr>
<tr>
<td>X00414</td>
<td>End of Positioning (Input Relay)</td>
</tr>
<tr>
<td>I00099</td>
<td>Command Execution Prohibit Condition (set in another part)</td>
</tr>
<tr>
<td>I00401</td>
<td>Request to Decelerate and Stop</td>
</tr>
<tr>
<td>I00402</td>
<td>Request to Decelerate and Stop (Differentiate Up)</td>
</tr>
<tr>
<td>I00403</td>
<td>Waiting for Command ACK</td>
</tr>
<tr>
<td>I00404</td>
<td>Waiting for Deceleration and Stop</td>
</tr>
</tbody>
</table>
Request to Decelerate and Stop Program

Figure 7.27 Request to Decelerate and Stop Program

Motor speed

Figure 7.28 Request to Decelerate and Stop Program Time Chart
7.1.11 Request to Stop Immediately

Stops the operating motor immediately without deceleration during positioning operation, velocity control operation, etc.

Items to Note:

- This command is accepted only if the following conditions are met: (1) there is no error and (2) servo is ON. Otherwise, it is ignored.
- If the system, while it is operating at a high speed, is stopped suddenly, the system will likely experience mechanical shock.
- The positioning module also accepts an Immediately Stop command during end-of-positioning. (The corresponding ACK is set.)

Procedures:

1) Set the output relay defined as “Request to Stop Immediately.”
2) Reset the relay defined as “Request to Stop Immediately” after confirming that the input relay defined as “Immediately Stop ACK” is set. The motor stops immediately when the Immediately Stop ACK is set.
3) The input relay defined as “End of Positioning” is set when the motor stops.

Program Example:

This is an example of request to stop immediately.

- List of devices used

<table>
<thead>
<tr>
<th>Y00436</th>
<th>Request to Stop Immediately (Output Relay)</th>
</tr>
</thead>
<tbody>
<tr>
<td>X00404</td>
<td>Stop Immediately ACK (Input Relay)</td>
</tr>
<tr>
<td>X00414</td>
<td>End of Positioning (Input Relay)</td>
</tr>
<tr>
<td>I00411</td>
<td>Request to Stop Immediately</td>
</tr>
<tr>
<td>I00412</td>
<td>Request to Stop Immediately (Differentiate Up)</td>
</tr>
<tr>
<td>I00413</td>
<td>Waiting for Command ACK</td>
</tr>
<tr>
<td>I00414</td>
<td>Executing Command</td>
</tr>
</tbody>
</table>
Figure 7.29  Request to Stop Immediately Program

Figure 7.30  Request to Stop Immediately Program Time Chart
7.1.12 Extended Command Instructions

The following are commands used for various purposes related to setting extended command parameters.

(1) Servo ON

The motor enters servo ON and the brake is switched OFF when this command is executed during servo OFF. The positioning module accepts the command even if the servo is ON. (The corresponding ACK is set.)

Items to Note:

• This command is accepted only if there is no error. Otherwise, it is ignored.
• The following three conditions must be satisfied for proper operations. The positioning module should be connected to the servo driver/motor properly. Entry parameters should be set properly. And, the setting of the servo driver should be done properly. An error may occur and the servo may be set to OFF if this operation is executed without meeting these three conditions. If any error is detected, identify the error code and check the conditions above.

Procedures:

1) Write “0” to the Extended Command Parameter.
2) Set the output relay defined as “Extended Command Instruction.”
3) Reset the relay defined as “Extended Command Instruction” after confirming that the input relay defined as “Extended Command ACK” is set. The servo is switched ON when the Extended Command ACK is set. When the servo turns ON, the input relay defined as “End-of-positioning” is set.

Program Example:

This is an example for executing servo ON. Entry parameters are set properly in advance.

• List of major devices used

<table>
<thead>
<tr>
<th>Device</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y00434</td>
<td>Extended Command Instruction (Output Relay)</td>
</tr>
<tr>
<td>X00402</td>
<td>Extended Command ACK (Input Relay)</td>
</tr>
<tr>
<td>X00414</td>
<td>End-of-positioning (Input Relay)</td>
</tr>
<tr>
<td>I00099</td>
<td>Command execution prohibit condition (set in another part)</td>
</tr>
<tr>
<td>I00501</td>
<td>Request for Servo ON</td>
</tr>
<tr>
<td>I00502</td>
<td>Request for Servo ON (Differentiate Up)</td>
</tr>
<tr>
<td>I00503</td>
<td>Waiting for Command ACK</td>
</tr>
<tr>
<td>I00504</td>
<td>Executing Command</td>
</tr>
</tbody>
</table>
Figure 7.31 Servo ON Program

Figure 7.32 Servo ON Program Time Chart
(2) Servo OFF

Servo becomes OFF and the brake switched ON when this command is executed during servo ON. The positioning module accepts the command even during servo OFF. (The corresponding ACK is set.)

Items to Note:

- This command is accepted only if there is no error. Otherwise, it is ignored.
- Servo Off command can be executed even when the motor is operating.

Procedures:

1) Write “1” to the Extended Command parameter.
2) Set the output relay defined as “Extended Command Instruction.”
3) Reset the relay defined as “Extended Command Instruction” after confirming that the input relay defined as “Extended Command ACK” is set. The servo is switched OFF when the Extended Command ACK is set. When the servo turns OFF, the input relay defined as “End-of-positioning” is set.

Program Example:

This is a program example for executing servo OFF. The time chart below is for when the motor stops (end-of-positioning).

- List of major devices used

<table>
<thead>
<tr>
<th>Device</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y00434</td>
<td>Extended Command Instruction (Output Relay)</td>
</tr>
<tr>
<td>X00402</td>
<td>Extended Command ACK (Input Relay)</td>
</tr>
<tr>
<td>I00099</td>
<td>End-of-positioning (Input Relay)</td>
</tr>
<tr>
<td>I00511</td>
<td>Request for Servo ON</td>
</tr>
<tr>
<td>I00512</td>
<td>Request for Servo ON (Differentiate Up)</td>
</tr>
<tr>
<td>I00513</td>
<td>Waiting for Command ACK</td>
</tr>
<tr>
<td>I00514</td>
<td>Executing Command</td>
</tr>
</tbody>
</table>
Figure 7.33 Servo OFF Program

Figure 7.34 Servo OFF Program Time Chart
(3) Brake ON

Turns the brake ON.

The positioning module accepts the command even if the brake is ON. (The corresponding ACK is set.)

Items to Note:

• This command is accepted only if the following conditions are satisfied: (1) there is no error and (2) servo is OFF. Otherwise, it is ignored.

• The positioning module operates the brake automatically depending on the state of the servo (ON / OFF). Thus, there is no need to operate the brake under normal conditions. This command is useful when it is necessary to operate the brake while the servo is OFF (e.g., for maintenance).

Procedures:

1) Write “2” to the Extended Command parameter.
2) Set the output relay defined as “Extended Command Instruction.”
3) Reset the relay defined as “Extended Command Instruction” after confirming that the input relay defined as “Extended Command ACK” is set.
(4) Brake OFF

Turns the brake OFF.

The positioning module accepts the command even if the brake is OFF. (The corresponding ACK is set.)

Items to Note:

- This command is accepted only if there is no error. Otherwise, it is ignored.
- The positioning module operates the brake automatically depending on the state of the servo (ON / OFF). Thus, there is no need to operate the brake under normal conditions. This command is useful when it is necessary to operate the brake while the servo is OFF (e.g., for maintenance).

Procedures:

1) Write “3” to the Extended Command parameter.
2) Set the output relay defined as “Extended Command Instruction.”
3) Reset the relay defined as “Extended Command Instruction” after confirming that the input relay defined as “Extended Command ACK” is set.
(5) Stationary Deviation Adjust

A deviation is produced between the specified position at the time of stopping and the encoder feedback position when there is some error in the speed instruction voltage output and the servo driver input of the positioning module (both values should be 0). It is called the stationary deviation. When the precision of the stopping position is important, you can use this command to offset the deviation by setting an offset which cancels the stationary deviation to the velocity instruction voltage output of the positioning module.

Items to Note:

- This command is accepted only if the following conditions are satisfied: (1) there is no error and (2) no other commands are being executed. Otherwise, it is ignored.
- This command sets the offset for adjusting stationary deviation, it does not guarantee that the deviation at the time of stopping is zero.
- If the servo driver you are using has a function to adjust the offset of the velocity instruction voltage input (0 V error), use that function instead of this command.

Procedures:

1) Before executing the Stationary Deviation Adjust command, read (1) Current Position status (instruction position) and (2) Current Position status (encoder position) during servo ON and end-of-positioning, then calculate the difference between (1) and (2) above.

   Note: To re-execute the command, calculate the latest difference by re-executing with the Stationary Deviation Adjust amount set to “0”. You can adjust properly only with a value obtained in the above way. The Stationary Deviation Adjust amount is set to “0” initially: until this command is executed after power-on.

2) Write the calculated result to the Stationary Deviation Adjust parameter and write “4” to the Extended Command parameter.

3) Set the output relay defined as “Extended Command Instruction.”

4) Reset the relay defined as “Extended Command Instruction” after confirming that the input relay defined as “Extended Command ACK” is set. Then, the Current Position status (instruction position) and the Current Position status (encoder position) will be almost the same, and the stationary deviation will be close to zero. When not all the parameters are proper, the input relay defined as “Extended Command ACK” is not set, and the input relay defined as “Error Notification” is set.

Program Example:

This is an example of executing the Stationary Deviation Adjust command. The Stationary Deviation Adjust amount has already been calculated.

- List of major devices used:

<table>
<thead>
<tr>
<th>Device</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1101/D1102</td>
<td>Stationary Deviation Adjust Amount</td>
</tr>
<tr>
<td>Y00434</td>
<td>Extended Command Instruction (Output Relay)</td>
</tr>
<tr>
<td>X00402</td>
<td>Extended Command ACK (Input Relay)</td>
</tr>
<tr>
<td>I00099</td>
<td>Command Execution Prohibit Condition (set in another part)</td>
</tr>
<tr>
<td>I00541</td>
<td>Request for Stationary Deviation Adjust</td>
</tr>
<tr>
<td>I00542</td>
<td>Request for Stationary Deviation Adjust (Differentiate Up)</td>
</tr>
<tr>
<td>I00543</td>
<td>Waiting for Command ACK</td>
</tr>
<tr>
<td>I00544</td>
<td>Executing Command</td>
</tr>
</tbody>
</table>
Stationary Deviation Adjust Program

Figure 7.35 Stationary Deviation Adjust Program

Figure 7.36 Stationary Deviation Adjust Program Time Chart
(6) Driver Reset

When this command is executed during servo OFF, it turns the driver reset signal ON for 500 ms and then it OFF.

It is used for unconditional driver reset, e.g., when the module is powered on.

Items to Note:

- This command is accepted only if all of the following conditions are satisfied: (1) there is no error, (2) no other commands are being executed and (3) servo is OFF. Otherwise, it is ignored.
- This command does not recognize a driver alarm signal. When the command is executed, it outputs the driver reset signal regardless of the state of the driver alarm input signal. No error occurs when the driver alarm signal stays ON after the driver reset signal is output.
- The command cannot be accepted when a positioning module is in error state (when error notification is set). Therefore, execute the normal Error Reset command when a driver alarm occurs during servo ON or an error occurs as a result of setting servo ON during driver alarm.

Procedures:

1) Write “5” to the Extended Command parameter.
2) Set the output relay defined as “Extended Command Instruction.”
3) Reset the relay defined as “Extended Command Instruction” after confirming that the input relay defined as “Extended Command ACK” is set. The driver reset signal is switched on for 500 ms when the Extended Command ACK is set.

Program Example:

This is an example to execute driver reset.

- List of major devices used

<table>
<thead>
<tr>
<th>Device</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y00434</td>
<td>Extended Command Instruction (Output Relay)</td>
</tr>
<tr>
<td>X00402</td>
<td>Extended Command ACK (Input Relay)</td>
</tr>
<tr>
<td>I00099</td>
<td>Command Execution Prohibit Condition (set in another part)</td>
</tr>
<tr>
<td>I00551</td>
<td>Request for Driver Reset</td>
</tr>
<tr>
<td>I00552</td>
<td>Request for Driver Reset (Differentiate Up)</td>
</tr>
<tr>
<td>I00553</td>
<td>Waiting for Command ACK</td>
</tr>
<tr>
<td>I00554</td>
<td>Executing Command</td>
</tr>
</tbody>
</table>
Figure 7.37  Driver Reset Program

Figure 7.38  Driver Reset Program Time Chart
(7) Manual Pulser Mode ON / OFF

When the Manual Pulser Mode ON is executed during servo ON in position control mode, the module is switched to manual pulser mode and the motor moves according to the input pulse of the manual pulser input. The ratio of the manual pulser input to the motor rotation number is specified using the Manual Pulser Scale Value parameter.

When Manual Pulser Mode OFF is executed, the system returns to normal position control mode.

A positioning module accepts the command even if the manual pulser mode ON is executed when in manual pulser mode, or if the the manual pulser mode OFF is executed when not in manual pulser mode. (The corresponding ACK is set.)

Items to Note:

- This command is accepted only if all of the following conditions are satisfied: (1) not in error state, (2) not executing other commands, (3) servo is ON and during end-of-positioning and (4) in position control mode. Otherwise, it is ignored.
- The motor cannot be operated by commands from the CPU module during manual pulser mode.
- During manual pulser mode, the manual pulser mode is reset when the servo becomes OFF due to some errors or execution of the servo OFF command.

Procedures:

- Manual Pulser Mode ON
  1) Write “10” to the Extended Command parameter. Write a value to the Pulser Scale Value parameter.
  2) Set the output relay defined as “Extended Command Instruction.”
  3) Reset the relay defined as “Extended Command Instruction” after confirming that the input relay defined as “Extended Command ACK” is set. The mode is switched to manual pulser when the Extended Command ACK is set. (When not all the parameters are proper, the input relay defined as “Switch Control Mode ACK” is not set, and the input relay defined as “Error Notification” is set.)

- Manual Pulser Mode OFF
  1) Write “11” to the Extended Command parameter.
  2) Set the output relay defined as “Extended Command Instruction.”
  3) Reset the relay defined as “Extended Command Instruction” after confirming that the input relay defined as “Extended Command ACK” is set. The manual pulser mode is reset when the Extended Command ACK is set.

Program Example:

This is an example of executing Manual Pulser Mode ON / OFF. All required parameters are set in advance in the data register.
7-54

List of major devices used

<table>
<thead>
<tr>
<th>Device Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1111</td>
<td>Manual Pulser Scale Value</td>
</tr>
<tr>
<td>Y00434</td>
<td>Extended Command Instruction (Output Relay)</td>
</tr>
<tr>
<td>X00402</td>
<td>Extended Command ACK (Input Relay)</td>
</tr>
<tr>
<td>I00099</td>
<td>Command execution prohibit condition (set in another part)</td>
</tr>
<tr>
<td>I00561</td>
<td>Request for Manual Pulser Mode ON</td>
</tr>
<tr>
<td>I00562</td>
<td>Request for Manual Pulser Mode ON (Differentiate Up)</td>
</tr>
<tr>
<td>I00563</td>
<td>Request for Manual Pulser Mode ON (Waiting for Command ACK)</td>
</tr>
<tr>
<td>I00564</td>
<td>Request for Manual Pulser Mode ON (Executing Command)</td>
</tr>
<tr>
<td>I00565</td>
<td>Manual Pulser Mode</td>
</tr>
<tr>
<td>I00571</td>
<td>Request for Manual Pulser Mode OFF</td>
</tr>
<tr>
<td>I00572</td>
<td>Request for Manual Pulser Mode OFF (Differentiate Up)</td>
</tr>
<tr>
<td>I00573</td>
<td>Request for Manual Pulser Mode OFF (Waiting for Command ACK)</td>
</tr>
<tr>
<td>I00574</td>
<td>Request for Manual Pulser Mode OFF (Executing Command)</td>
</tr>
</tbody>
</table>

Figure 7.39 Manual Pulser Mode ON / OFF Program
Figure 7.40 Manual Pulser Mode ON / OFF Program Time Chart
7.1.13 On-route Operation

When a new positioning operation is started during a positioning operation (before end-of-positioning), an on-route operation will start. On-route operation is used with linear interpolated operation. The system does not stop at each target position, but passes near the target positions (route points) and operates continuously. The startup of positioning operation for the target positions (route points) after the first one is done while checking the status of the remaining deceleration time.

### Remaining deceleration time status

This status reads the time of deceleration for stopping at the target position during the execution of the positioning operation. Its value is “0” during end-of-positioning, and “-1” during acceleration or while operating at the target velocity.

During the on-route operation, start the next positioning after checking that the above value is not larger than the preset values for each axis. This assures that the positioning for the next target position is started before reaching the intermediate target positions (route points). (This time lag is set in ms.)

**Items to Note:**

- For on-route operations, do not set data that will cause any positioning operation to finish earlier than previously started-up data. Otherwise, the operation cannot be guaranteed.

**Procedures:**

1. Write the operation parameters of each axis which are necessary to move to the first target position to the positioning module.
2. For each axis, set the output relay defined as “Start Operation Command” simultaneously (in the same scan).
3. Reset the relay defined as “Start Operation Command” of each axis after confirming that the input relays defined as “Start Operation Command ACK” of all axes are set. (If there is a parameter error, the input relay defined as “Start Operation Command ACK” is not set and the input relay defined as “Error Notification” is set.)
4. Write the startup parameters of each axis which are necessary to move to the next target position on the positioning module after confirming that the input relay defined as “Start Operation Command ACK” is set.
5. For each axis, set the output relay defined as “Start Operation Command” simultaneously after confirming that the remaining deceleration time is not larger than the specified value for all axes.
6. Reset the relay defined as “Start Operation Command” of each axis after confirming that the input relays defined as “Start Operation Command ACK” of all axes are set. The relay defined as “Start Operation Command ACK” is not reset until the positioning which is started first finishes. (Start Operation command ACK” remains set during the on-route operation.)
7. Repeat 4-6 until the final target position is reached.
8. The relay defined as “End of Positioning” is set when the final target position is reached.
Program Example:

This is an example of an on-route operation with two axes and two target positions. This is also applicable to cases with more than two target positions by changing the values in the final interval decision part. All required parameters are set in advance in the data register.

In this program, when error occurs at one axis, the internal relays of the other axis for executing Stop Immediately and servo OFF are set. (This is not included in the time chart.)

- List of devices used:

<table>
<thead>
<tr>
<th>Device Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1201~</td>
<td>Axis 1 Startup Parameters storage area</td>
</tr>
<tr>
<td>D1301~</td>
<td>Axis 2 Startup Parameters storage area</td>
</tr>
<tr>
<td>D1299</td>
<td>Axis 1 Remaining Deceleration Time Value Read</td>
</tr>
<tr>
<td>D1399</td>
<td>Axis 2 Remaining Deceleration Time Value Read</td>
</tr>
<tr>
<td>V01</td>
<td>Startup Parameters Storage Area Pointer</td>
</tr>
<tr>
<td>time</td>
<td>Time set point of on-route interval</td>
</tr>
<tr>
<td>Y00433</td>
<td>Axis 1 Start Operation Command (Output Relay)</td>
</tr>
<tr>
<td>Y00449</td>
<td>Axis 2 Start Operation Command (Output Relay)</td>
</tr>
<tr>
<td>X00401</td>
<td>Axis 1 Start Operation Command ACK (Input Relay)</td>
</tr>
<tr>
<td>X00417</td>
<td>Axis 2 Start Operation Command ACK (Input Relay)</td>
</tr>
<tr>
<td>I00099</td>
<td>Command Execution Prohibit Condition (set in another part)</td>
</tr>
<tr>
<td>I00098</td>
<td>On-route Operation Stop Condition (set in another part)</td>
</tr>
<tr>
<td>I00601</td>
<td>Request to Start</td>
</tr>
<tr>
<td>I00602</td>
<td>Request to Start (Differentiate Up)</td>
</tr>
<tr>
<td>I00603</td>
<td>Waiting for Start Enable (Remaining Deceleration Time Wait)</td>
</tr>
<tr>
<td>I00604</td>
<td>Start Timing</td>
</tr>
<tr>
<td>I00605</td>
<td>Waiting for Start Operation Command ACK</td>
</tr>
<tr>
<td>I00606</td>
<td>Waiting for Start Operation Command ACK Reset</td>
</tr>
<tr>
<td>I00607</td>
<td>Parameter Write Timing (for interval 2 or later)</td>
</tr>
<tr>
<td>I00608</td>
<td>Continuing On-route Operation</td>
</tr>
<tr>
<td>I00609</td>
<td>End of Start for Final Interval</td>
</tr>
<tr>
<td>I00698</td>
<td>Request for Error Handling (for axis 1)</td>
</tr>
<tr>
<td>I00699</td>
<td>Request for Error Handling (for axis 2)</td>
</tr>
</tbody>
</table>
On-route operation program

```
00508
I00601
I00602
I00607
V01 > 32
SET I00609.

RST I00608.

WRITE D1201 4 41 21
WRITE D1301 4 241 21

READ D1299 4 115
READ D1399 4 315

D1299 >= 0 D1299 < time

D1399 >= 0 D1399 < time

Update startup parameter
End on-route operation

Write axis 1 startup parameter
Write axis 2 startup parameter

Read axis 1 deceleration remaining time
Read axis 2 deceleration remaining time

Starting axis 1
Starting axis 2

Waiting for end of on-route interval

Error processing (axis 1)
Error processing (axis 2)

End of biaxial positioning
```

Figure 7.41  On-route Operation Program
**Application Function:**

You can check the states of accelerating, decelerating and moving with constant velocity, during on-route operation, or the like, by reading the extended status.
7.1.14 Arc-Interpolation Operation

Performs the biaxial arc-interpolated operation. For details on the algorithm for calculating the parameters for an arc-interpolated operation, see Section 4.3, “Description of Parameters” provided earlier in this manual.

Items to Note:

- X axis and Y axis move independently also when performing arc interpolation. So when an error occurs at one axis, the other axis continues moving. If it is necessary to stop the other axis, execute the Servo OFF command on the moving axis and stop the motor after detecting the error with an application program (and checking the input relay defined as “Error Notification”).
- If the speed (the actual speed, not the angular velocity) of each axis exceeds the speed limit value, only the axis that exceeds the limit runs within the limit. So the arc interpolation will be an incorrect operation.

Procedures:

1) Write parameters (startup parameter and arc-interpolation parameter) for two axes to execute arc interpolation.
2) Set the output relays defined as “Start Operation Command” for both axes simultaneously (in the same scan).
3) Reset the relay defined as “Start Operation Command” of both axes after confirming that the input relays defined as “Start Operation Command ACK” of both axes are set. The arc-interpolated operation starts when the Start Operation Command ACK is set.

Program Example:

This is an example of arc-interpolated operation with two axes. All required parameters are set in advance in the data register.

In this program, internal relays are included to stop one axis preemptly if an error occurs in the other axis. (This is not included in the time chart.)

- List of devices used:

<table>
<thead>
<tr>
<th>Device</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1401~D1405</td>
<td>X Axis Startup Parameters</td>
</tr>
<tr>
<td>D1411~D1425</td>
<td>X Axis Arc-Interpolation Parameters</td>
</tr>
<tr>
<td>D1501~D1505</td>
<td>Y Axis Startup Parameters</td>
</tr>
<tr>
<td>D1511~D1525</td>
<td>Y Axis Arc-Interpolation Parameters</td>
</tr>
<tr>
<td>Y00433</td>
<td>X Axis Start Operation Command (Output Relay)</td>
</tr>
<tr>
<td>Y00449</td>
<td>Y Axis Start Operation Command (Output Relay)</td>
</tr>
<tr>
<td>X00401</td>
<td>X Axis Start Operation Command ACK (Input Relay)</td>
</tr>
<tr>
<td>X00417</td>
<td>Y Axis Start Operation Command ACK (Input Relay)</td>
</tr>
<tr>
<td>I00099</td>
<td>Command Execution Prohibit Condition (set in another part)</td>
</tr>
<tr>
<td>I00701</td>
<td>Request to Start Arc-Interpolation</td>
</tr>
<tr>
<td>I00702</td>
<td>Request to Start Arc Interpolation (Differentiate Up)</td>
</tr>
<tr>
<td>I00703</td>
<td>Waiting for Command ACK</td>
</tr>
<tr>
<td>I00704</td>
<td>During Arc-interpolated operation</td>
</tr>
<tr>
<td>I00708</td>
<td>Request for Error Handling (for x axis)</td>
</tr>
<tr>
<td>I00709</td>
<td>Request for Error Handling (for y axis)</td>
</tr>
</tbody>
</table>
Figure 7.43 Arc Interpolation Program
Figure 7.44  Arc-Interpolation Program Time Chart
## 7.2 Accessing form BASIC CPU

The following are the instructions to access from the BASIC CPU. For details on each instruction, see the BASIC Programming Tool M3 (publication number IM 34M6Q22-01E).

<table>
<thead>
<tr>
<th>Function</th>
<th>Statement Format</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Module Use Declaration</td>
<td>ASSIGN NC12=SL</td>
<td>Declares use of the module and CPU module.</td>
</tr>
<tr>
<td>Reading Parameters and Status</td>
<td>Enter SL,n NOFORMAT;I SL: Slot number n: Data position number I: Integer-type, Variable name or array variable name to store read data</td>
<td>Reads parameters and status of data position number (n) of the module installed in the slot number (SL), and stores them in the variable (I).</td>
</tr>
<tr>
<td>Writing Parameters</td>
<td>OUTPUT SL,n NOFORMAT;I SL: Slot number n: Data position number I: Integer-type, Variable name or array variable name to store writing data</td>
<td>Replaces parameters of data position number (n) of the module installed in slot number SL with data stored in the variable (I).</td>
</tr>
<tr>
<td>Reading Input Relay (*1)</td>
<td>STATUS SL,n;P SL: Slot number n: Data position number (101/102) P: Integer variable name</td>
<td>Reads the input relay of the module installed in the slot number SL, and stores it in the variable (P).</td>
</tr>
<tr>
<td>Writing Output Relay (*2)</td>
<td>CONTROL SL,n;P,M SL: Slot number n: Data position number (101/102) P: Output data M: Mask pattern</td>
<td>Replaces the output relay of the module installed in the slot number SL with the value stored in the variable (P). Mask pattern (M) allows overwriting to only the specified output relays.</td>
</tr>
<tr>
<td>Interrupt Declaration</td>
<td>ON INT SL,nn GOSUB label CALL subprogram name GOTO label SL: Slot number nn: Input relay number</td>
<td>Declares to branch to specified process when an interrupt is requested.</td>
</tr>
<tr>
<td>Reset Interrupt Declaration</td>
<td>OFF INT SL,nn SL: Slot number nn: Input relay number</td>
<td>Resets the ON INT statement.</td>
</tr>
</tbody>
</table>

*1: For input relays, data position number 101 is for axis 1 (X01 to X16), 102 is for axis 2 (X17 to X32).

*2: For output relays, data position number 101 is for axis 1 (Y33 to Y48), 102 is for axis 2 (Y49 to Y64).
How To Handle 2-Word Data:

It is necessary to convert data from long-word to 2-word integer data when parameters are written to the positioning module. It is also necessary to convert data to long-word after reading the data as two integer variables when 2-word data is read from the positioning module. The following is an example for the conversion.

LDAT : Long-word integer variable (before conversion)
IDL, IDU: Integer variable to store the data (after conversion)

100 IDL = VAL ("$" + RIGHT$ (LHEX$ (LDAT), 4))
110 IDU = VAL ("$" + LEFT$ (LHEX$ (LDAT), 4))

ISL, ISU : Integer variable to store the 2-word data read.
LST : Long-word integer variable (after conversion)

100 LST = VAL (HEX$ (ISU) + HEX$ (ISL))
7.2.1 Reading the Module Status
Explanes how to read the status of the positioning module.

Items to Note:

- There is no particular caution here. The status of the positioning module can be read at any time. Use the ENTER statement to read.
- 2-word data status, such as current position and velocity, can be read by separating them into two integer variables.
7.2.2 Set Parameter

This sets the entry parameters.

Items to Note:

- This command is accepted only if all the following conditions are satisfied: (1) servo is OFF and (2) not executing other commands. If these conditions are not satisfied, then the command will be ignored.

- Errors cannot be reset with the Error Reset command when the error occurs with the Set Parameter command (error code 20xx). Run the Set Parameter command again with proper data. If other errors occur, they should be reset beforehand with the Error Reset command.

Procedures:

1) Write parameters on the positioning module with the WRITE instruction.
2) Set the Set Parameter output relay.
3) Reset Set Parameter output relay after confirming that Set Parameter ACK input relay is set. (If there is any error in the parameters, the Set Parameter ACK input relay is not set, but the Error Notification input relay is set. In this case, reset the Set Parameter output relay, go back to step 1 and retry.)
4) Check that Set Parameter ACK is reset.

Program Example:

In this example, parameters are set in advance in integer type variables (I1 -).

```
• Set Parameter Program
100 OUTPUT SL, 1 NOFORMAT ; I1
110 OUTPUT SL, 2 NOFORMAT ; I2
.
.
200 CONTROL SL, 101 ; $8000, $8000
210 LOOP1@
220 STATUS SL, 101 ; P
230 IF BIT (P, 15) = 0 THEN LOOP1@
240 CONTROL SL, 101 ; $0000, $8000
250 LOOP2@
260 STATUS SL, 101 ; P
270 IF BIT (P, 15) = 1 THEN LOOP2@
```
7.2.3 Error Reset

This resets the error status of the positioning module.

Items to Note:

- The positioning module ignores any commands other than the Error Reset command and Set Parameter command in error status (i.e. status where the Error Notification input relay is set). So be sure to execute the Error Reset command in error status. However, when an error occurs because of a parameter error in the Set Parameter command, the error cannot be reset using the Error Reset command.
- If the Error Reset command is executed for driver alarm detection error (error code “12”), the driver reset signal is set to ON for 500 ms and then switched off. Thereafter, the system waits for the reset of the driver alarm for a maximum length of 500 ms.
- Despite execution of the Error Reset command, if the cause of the error is not removed, the Error Notification will not become OFF.

Procedures:

1) Set the Error Reset output relay.
2) Check that the Error Notification relay is reset. Then reset the Error Reset output relay.

Program Example:

In this example, all output relays of the positioning module are reset at the time of the Error Reset operation. The error code is preserved after the Error Reset.

- Error Reset Program
  100 STATUS SL, 101 ; P
  110 IF BIT (P, 11) = 0 THEN EXIT@
  120 ENTER SL, 101 NOFORMAT ; ERCODE
  130 CONTROL SL, 101 ; $1000
  140 LOOP1@
  150 STATUS SL, 101 ; P
  160 IF BIT (P, 11) = 1 THEN LOOP1 @
  170 CONTROL SL, 101 ; $0000
  180 EXIT@
7.2.4 Jog Stepping

When the Positive- or Negative-direction Jog Stepping output relay is ON, rotates the motor in the positive (or negative) direction.

Items to Note:

- This command is accepted only if all of the following conditions are satisfied: (1) not in error state, (2) during servo ON and in End-of-positioning state, (3) in position control mode and (4) no other command executing. Otherwise, it is ignored.
- During jog stepping, you can stop the motor with the Immediately Stop command, but not with the Decelerate-and-Stop command. Terminate jog stepping first when decelerate and stop is desired.

Procedures:

1) Write required parameters of the Jog Stepping command on the positioning module.
2) When the Positive- or Negative-direction Jog Stepping output relay is set, the motor rotates according to the parameters, the Operating-in Positive-or Negative-direction input relay is set, and the End-of-Positioning input relay is reset, the End of Pulse Clear Out and End of Positioning input relays are set.
3) Reset the Positive-or Negative-direction Jog Stepping output relay. The motor will be decelerated and stopped according to the parameters set at the start of the jog stepping. After the positioning module terminates the startup generation, the End of Pulse Clear Out input relay is set and the Operating-in-positive-direction (or negative-direction) input relay is reset. Then the End-of-positioning input relay is set when the difference of the encoder feedback value and the specified position value is less than or equal to the positioning judgement range.

Program Example:

This is a program example for Start Jog Stepping (JOGST@) and End-of Jog Stepping (JOGEND@). It is only for positive-direction. Required parameters are set in advance in integer type variables (I41~).

- Jog Stepping Program

100 JOGST@  
110 OUTPUT SL, 41 NOFORMAT ; I41  
120 OUTPUT SL, 42 NOFORMAT ; I42  
:  
200 CONTROL SL, 101 ; $0200, $0200  
:  
300 JOGEND@  
310 CONTROL SL, 101 ; $0000, $0200  
320 LOOP1@  
330 STATUS SL, 101 ; P  
340 IF BIT (P, 13) =0 THEN LOOP1@
7.2.5 Origin-search

This executes origin-search. There are four external contact inputs related to origin-search. The origin-search mode specifies the action when each rising and falling edge of these four inputs (eight in total) are detected during the origin-search using bit patterns. (2 bits for each edge, so 16 bits in total.)

In the origin-search in the negative direction as given below, application requirements where (1) Stop immediately when the rising edge of the Negative-Direction Limit is detected, (2) Decelerate and stop when the rising edge of the Origin input is detected, and (3) Shift to Z phase search when the falling edge of the Origin-Search is detected, set, the Origin search mode in the following way:

```
15~  ~0
1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 (= $C009)
```

![Figure 7.45 Example of Origin-search Operation](image)

[Origin-search Mode in detail]

The mode is set in bits according to the list below. There are four ways of setting using two bits for the rising and falling edges of one external contact input.
Items to Note:

This command is accepted only if all of the following conditions are satisfied: (1) not in error state, (2) servo-ON and during End-of-positioning, (3) in position control mode, and (4) no other command executing. Otherwise, it is ignored.

- The origin-search ends when a specified external contact input is detected and the search stopped. Change the parameters and re-execute the origin-search after it ends if you want to continue the search in another direction or with a different speed.
- In the following cases, an error occurs and the search is automatically stopped:
  - The limit input of the origin-search direction is set as ignored, but the limit input is detected.
  - After shifting to the Z phase search, the limit input of the origin-search direction is detected (causes error regardless of the setting).
  - After shifting to the Z-phase search, Z-phase cannot be detected within the Z-phase search range.
  - If you want to change the setup values according to the status of the external contact input at the beginning of the origin search, read the state with the Contact Input Status and then execute the origin-search.
  - When the origin-search mode is "0", the program does not check the External Contact input and Z-phase search is performed from operation startup.

Procedures:

1) Write parameters required by the Origin-Search command to the positioning module.
2) The motor rotates following the parameters when the output relay defined as “Start Origin-search” is set. The motor continues to rotate until a specified external contact input is detected.

3) When Z-pulse is detected for the specified number of times after shifting to Z-phase search (if this number is specified as “0”, then without shifting to Z-phase), current position is specified as “Position 0” (or the Origin Offset value if it is defined) and the operation is decelerated and stopped. The End-of-Positioning and End-of-Origin-search input relays are set. After that, the End-of-Origin-Search is reset if the Start Origin Search is reset. However, if the Start-Origin-Search is reset before End-of-Origin-Search is set, then the End-of-Origin-Search will not be set. (The End-of-Origin-search input relay is set when there was no shift to Z-phase search and the operation is stopped with the Decelerate-and-Stop command, or with the Immediate Stop command.)

- Bit 10 (End-of-Origin Search during the origin-search operation) of the extended status is set only when the operation is stopped after shifting to Z-phase search and detecting the Z-phase. This bit is reset when the origin-search starts. It is “1)” when the main switch is turned on.

Program Example:

This program starts the origin-search and ends it using the specified external contact input. Required parameters are set in integer variables (I41 ~ ).

- Origin-search Program

```
100 OUTPUT SL, 41 NOFORMAT ; I41
110 OUTPUT SL, 42 NOFORMAT ; I42

200 CONTROL SL, 101 ; $0010, $0010
210 LOOP1@
220 STATUS SL, 101 ; P
230 IF BIT (P, 4) = 0 THEN LOOP1@
240 CONTROL SL, 101 ; $0000, $0010
250 LOOP2@
260 STATUS SL, 101 ; P
270 IF BIT (P, 4) =1 THEN LOOP2@
```
7.2.6 Write Current Position

When servo is ON, this operation changes the current position of the axis at end-of-positioning status in the positioning control mode.

When servo is OFF, it changes the encoder feedback position.

Items to Note:

- This command is accepted only if all of the following conditions are satisfied: (1) not in error state, (2) servo OFF or during end-of-positioning with servo ON (3) not executing other commands. Otherwise, it is ignored.

- The change position should fall in the range from the positive-direction limit to the negative-direction limit. Error occurs when requesting to change beyond the limits of the range.

- It is possible to write the current position in YASKAWA Electric absolute value encoder system. In this case, positions such as the current position or the command position after this operation are not related to the absolute value, though the absolute value data is in the absolute value status.

- It is impossible to write current position for SANYO absolute value encoder system.

Procedures:

1) Write the desired position after the change in the Target Position parameter.

2) Set the output relay defined as “Request to Write Current Position.”

3) Reset the relay defined as “Request to Write Current Position” after confirming that the input relay defined as “Current Position Write ACK” is set. If there is a parameter error, the input relay defined as “Error Notification” is set and the Current Position Write ACK is not set.

Program Example:

This is an example of writing the current position. All required parameters are set in advance in the integer variables (I43, I44).

- Write Current Position Program

  100 OUTPUT SL, 43 NOFORMAT ; I43
  110 OUTPUT SL, 44 NOFORMAT ; I44
  200 CONTROL SL, 101 ; $0800, $0800
  210 LOOP1@
  220 STATUS SL, 101 ; P
  230 IF BIT (P, 10) = 0 THEN LOOP1@
  240 CONTROL SL, 101 ; $0000, $0800
  250 LOOP2@
  260 STATUS SL, 101 ; P
  270 IF BIT (P, 10) =1 THEN LOOP2@
7.2.7 Position Control Mode Operation

(1) Start Operation Command (Positioning Operation)

If executed during servo ON and End-of-Positioning, this executes the normal positioning operation. If executed during positioning operation, it executes two overlapping positioning operations (called on-route operation). The latter case will be explained in Section 7.2.13, “on-route Operation.”

Items to Note:

- This command is accepted only if all of the following conditions are satisfied: (1) not in error state, (2) not executing other commands, (3) during servo ON, (4) not during origin-search, and (5) not during jog stepping. Otherwise, it is ignored. Further, in the case of normal positioning, there are 2 additional conditions, (6) in End-of-Positioning status, (7) in position control mode. (If Items (6) and (7) above are not satisfied, an error operation results).
- If an invalid value is set for a startup parameter, a parameter error results and the motor does not move. The servo will not be switched OFF by parameter error at startup.

Procedures:

1) Write the required parameters in the positioning module.
2) Set the output relay defined as “Request to Start Operation.”
3) After confirming that the Start Operation Command ACK input relay is set, reset the Start Operation relay. The motor starts according to timing when the Start Operation ACK is set. (If there is a parameter error, Start Operation ACK input relay is not set and Error Notification input relay is set.)
4) When the path generated by the positioning module reaches the target position, the End-of-Pulse-Clear-Out Input relay is set. After this, if the difference between the target position (command position) and the encoder feedback position is less than or equal to the Positioning Judgement Pulse Count, the End of Positioning input relay is set. However, if the time from the End-of-Pulse-clear-out to End-of-Positioning exceeds the specified value for the Positioning Timeout Interval, an error results.
Program Example:

This is an example of a simple-positioning operation. All required parameters are set in advance in the integer variables (I41 ~ ).

- Positioning Operation Program

```
100 OUTPUT SL, 41 NOFORMAT ; I41
110 OUTPUT SL, 42 NOFORMAT ; I42
200 CONTROL SL, 101 ; $0001, $0001
210 LOOP1@
220 STATUS SL, 101 ; P
230 IF BIT (P, 0) = 0 THEN LOOP1@
240 CONTROL SL, 101 ; $0000, $0001
250 LOOP2@
260 STATUS SL, 101 ; P
270 IF BIT (P, 0) = 1 THEN LOOP2@
280 LOOP3@
290 STATUS SL, 101 ; P
300 IF BIT (P, 13) = 0 THEN LOOP3@
```

Application Function:

- Startup Mode Selection

When startup mode “1” is selected, the motor does not start moving until the external trigger input becomes ON after command execution. This mode is useful when it is desirable to wait for an external-synthesizing signal before starting.

When startup mode “2” is selected, the motor does not start moving until the output relay of an internal trigger is set after command execution.

- Position-detection Notification Function

If “1” or “2” is set as the Position Detection Mode, the input relay defined as “Position Detection Notification” is set to notify the application program when the current position during positioning arrives at the specified value for Position Detection Notification.

The relay can be used as interrupt input.

The relay is reset when the output relay defined as “Reset Position Detection Notification” is set.

The position detected is the command position when the Position Detection Mode is “1”, and the encoder feedback position when it is “2”.
(2) Change Target Position

If executed during positioning operation, this changes the target position immediately and then performs the positioning operation with the new target position. You can change the target velocity when changing the target position. In this case, the acceleration/deceleration between the velocities before and after the change is expressed by the slope calculated from the new target velocity and the acceleration time/deceleration time.

Items to Note:

- This command is accepted only if all of the following conditions are satisfied: (1) not in error state, (2) not executing other commands, (3) servo is ON in position control mode, (4) not in the on-route interval of on-route operation, (5) not during arc-interpolated operation, (6) not during origin search, (7) not during jog stepping. Otherwise, it is ignored.

- A parameter error occurs if invalid values are set to the startup parameters. The servo is switched OFF and the motor stops. The servo is not switched OFF if it is operated during end-of-positioning.

- Acceleration/deceleration at the specified velocity change is not performed in a two-step, S-form way, but the deceleration for stopping at the target point after changing follows the deceleration mode and time.

Procedures:

1) Write the necessary parameters to the positioning module.
2) Set the output relay defined as “Request to Change Target Position.”
3) Reset the relay defined as “Request to Change Target Position” after confirming the setting of the input relay defined as “Target Position Change ACK”. The motor starts the operation of the target point changing when the target point changing ACK is set. (If there is a parameter error, the input relay defined as “Target Position Change ACK” is not set and the input relay defined as “Error Notification” is set.)
4) The end-of-positioning is judged according to the parameters at the time of changing.
Program Example:

This is an example of changing the target position during positioning operation. All necessary parameters are set beforehand in the integer variables (I41 -).

- Change Target Position Program
  
  100  OUTPUT SL, 41  NOFORMAT ; I41
  110  OUTPUT SL, 42  NOFORMAT ; I42
  200  CONTROL SL, 101 ; $0040, $0040
  210  LOOP1@
  220  STATUS SL, 101 ; P
  230  IF BIT (P, 8) = 0 THEN LOOP1@
  240  CONTROL SL, 101 ; $0000, $0040
  250  LOOP2@
  260  STATUS SL, 101 ; P
  270  IF BIT (P, 8) = 1 THEN LOOP2@

Application Function:

- Select Startup Mode
  
  When startup mode “1” is selected, the operation (change target position) is not started until the external trigger input becomes ON after command execution. This mode is useful when it is desirable to wait for an external synthesizing signal before starting the operation.

  When startup mode “2” is selected, the operation is not started until the output relay of an internal trigger is set after command execution.

- Position Detection Notification Function
  
  The setting of the position detection notification specified by the positioning operation before changing target point is changed to the new one when the target point changing is executed. (If no change is necessary, execute the operation after setting the same values.)

- Change Target Position with a change in operation direction
  
  If the new position has already been passed at the time of executing Change Target Position, Decelerate-and-Stop is performed immediately and the positioning operation with the new target position starts. (If the new position has not yet been passed at the time of executing Change Target Position, but will be passed even if the deceleration is started immediately, the same steps occur)

  When Change Target Position with Change Operation Direction is executed, do not execute another Change Target Position command or Change Velocity command in the first deceleration operation. Otherwise, the operation cannot be guaranteed. You can confirm the status by reading the extended status.

- Change Target Position during End of Positioning
  
  You can execute the Change Target Position command during End of Positioning (when motor is not moving). In this case, the operation is the same as the normal positioning operation.
(3) Change Velocity

The acceleration or deceleration from a change in velocity is expressed by the slope determined by a new target velocity and acceleration time. (The acceleration time parameters are also used for deceleration in velocity changes where the new target velocity is slower than the velocity set previously.)

After changing the velocity, stopping at the target point follows the specified deceleration mode / deceleration time at the execution of change velocity command.

Items to Note:

- This command is accepted only if all of the following conditions are satisfied: (1) not in error state, (2) not executing other commands, (3) servo is ON, (4) not in the on-route interval of an on-route operation, (5) not during origin search, and (6) not during jog stepping. Otherwise, it is ignored. The operation becomes different in the velocity control mode.

- A parameter error occurs if invalid values are set to the startup parameters. The servo is switched OFF and the motor stops. The servo is not switched OFF if it is operated during end-of-positioning.

Procedures:

1) Write the necessary parameters on the positioning module.
2) Set the output relay defined as “Request to Change Velocity.”
3) Reset the relay defined as “Request to Change Velocity” after confirming the setting of the input relay defined as “Change Velocity ACK.” The motor starts the change speed operation when change velocity ACK is set. (If there is a parameter error, the input relay defined as “Change Input ACK” is not set and the input relay defined as “Error Notification” is set.)
4) The end-of-positioning is judged according to the parameters specified before changing.

Program Example:

This is an example of changing the target position during positioning operation. All necessary parameters are set beforehand in the integer variables (I41 ~ ).

- Change Speed Program (during positioning operation)

100 OUTPUT SL, 41 NOFORMAT ; I41
110 OUTPUT SL, 42 NOFORMAT ; I42

200 CONTROL SL, 101 ; $0080, $0080
210 LOOP1@
220 STATUS SL, 101 ; P
230 IF BIT (P, 9) = 0 THEN LOOP1@
240 CONTROL SL, 101 ; $0000, $0080
250 LOOP2@
260 STATUS SL, 101 ; P
270 IF BIT (P, 9) = 1 THEN LOOP2@
Application Function:

- **Select Startup Mode**

When startup mode “1” is selected, the velocity control operation is not started until the external trigger input becomes ON after command execution. This mode is useful when it is desirable to wait for an external-synthesizing signal before starting the operation.

When startup mode “2” is selected, the motor does not start moving until the output relay of an internal trigger is set after command execution.
7.2.8 Velocity Control Mode

Velocity control mode can be used only at the incremental encoder system. Absolute value encoder system cannot use this mode.

The positive-direction limit value and negative-direction limit value are ignored in velocity control mode.

(1) Switching Position to Velocity Control

The positioning module is in position control mode when servo is ON. It is necessary to switch the mode to “velocity control” in order to perform velocity control operation.

Items to Note:

- This command is accepted only if all of the following conditions are satisfied: (1) not in error state, (2) not executing other command, (3) servo is ON, (4) not in the on-route interval of an on-route operation, (5) not during an arc-interpolated operation, (6) not during origin-search, and (7) not during jog stepping. Otherwise, it is ignored.

- When switching the position to velocity control, no parameters, other than the Switch Control Mode parameter is referred to. The control is switched to velocity control while maintaining the same velocity.

Procedures:

1) Write “0” (switch to the velocity control) to the control mode switch parameter.
2) Set the output relay defined as “Request to Switch Control Mode.”
3) Reset the relay defined as “Request to Switch Control Mode” after confirming that the input relay defined as “Switch Control Mode ACK” is set. The motor is switched to velocity control mode while maintaining the same velocity.

Program Example:

This is an example of switching position to velocity control.

- Switch Position to velocity Control Program

```
100 OUTPUT SL, 81 NOFORMAT ; 0
200 CONTROL SL, 101 ; $0020, $0020
210 LOOP1@
220 STATUS SL, 101 ; P
230 IF BIT (P, 5) = 0 THEN LOOP1@
240 CONTROL SL, 101 ; $0000, $0020
250 LOOP2@
260 STATUS SL, 101 ; P
270 IF BIT (P, 5) = 1 THEN LOOP2@
```
(2) Start Operation Command (Velocity Control Operation)

Speed control operation starts when you execute operation command start in velocity control mode. This command starts the operation only. To stop the motor in velocity control operation, the decelerating stop command or the immediate stop command must be executed.

Items to Note:

- This command is accepted only if all of the following conditions are satisfied: (1) not in error state, (2) not executing other command, (3) servo is ON, (4) not during origin-search, and (5) not during jog stepping. Otherwise, it is ignored. When starting during velocity control, the additional required conditions are (6) in End-of-Positioning state, (7) in velocity Control mode. (If (6) and (7) are not satisfied, an error occurs in the operation).

- A parameter error occurs if invalid values are set to the startup parameters. The motor does not move. The servo is not switched OFF with the parameter error that occurred at the starting.

- In the velocity control mode, you must write a positive (negative) value as the target velocity for the positive (negative)-direction operation.

Procedures:

1) Write required parameters on the positioning module.

2) Set the output relay defined as “Start Operation Command.”

3) Reset the relay defined as “Start Operation Command” after confirming that the input relay defined as “Start Operation Command ACK” is set. The motor starts the operation when Start Operation Command ACK is set. (If there is a parameter error, the input relay defined as “Start Operation Command ACK” is not set and the input relay defined as “Error Notification” is set.)

Program Example:

This is an example of starting with speed control operation. All required parameters are set beforehand in the integer type variants (I41 - ).

- Start Velocity Control Program

```plaintext
100 OUTPUT SL, 41 NOFORMAT ; I41
110 OUTPUT SL, 42 NOFORMAT ; I42

200 CONTROL SL, 101 ; $0001, $0001
210 LOOP1@
220 STATUS SL, 101 ; P
230 IF BIT (P, 0) =0 THEN LOOP1@
240 CONTROL SL, 101 ; $0000, $ 0001
250 LOOP2@
260 STATUS SL, 101 ; P
270 IF BIT (P, 0 ) =1 THEN LOOP2@
```
Application Function:

- Select Startup Mode

When startup mode “1” is selected, the velocity control operation is not started until the external trigger input becomes ON after command execution. This mode is useful when it is desirable to wait for an external synthesizing signal before starting the operation.

When startup mode “2” is selected, the motor does not start moving until the output relay of an internal trigger is set after command execution.
(3) Change Velocity

Changes the Operating speed immediately when the velocity changes are made during positioning operation.

The acceleration or deceleration from a change in velocity is expressed by the slope determined by a new target velocity and acceleration time. (The acceleration time parameters are also used for deceleration in velocity changes where the new target velocity is slower than the velocity set previously.)

After the velocity changes have been made, a new target velocity is attained following the deceleration mode / deceleration time with the Velocity Change command.

Items to Note:

- This command is accepted only if all of the following conditions are satisfied: (1) not in the error state, (2) not executing other command, (3) servo is ON, (4) not in the on-route interval of on-route operation, (4) not during origin-search, and (5) not during jog stepping. Otherwise, it is ignored. In the position control mode, the operation will be different.
- A parameter error occurs if invalid values are set to the startup parameters. The servo is switched OFF and the motor stops. The servo is not switched OFF if it is operated during end-of-positioning.
- In velocity control mode, you must write a positive (negative) value as the target velocity for the positive (negative)-direction operation.
- You cannot change the operation direction with the Change Velocity command. (You cannot change to a negative target velocity during a positive-direction operation, nor change to a positive target velocity during a negative-direction operation.) To change the operation direction in velocity control mode, you must stop the motor with the Decelerate and Stop command and then restart the motor in the opposite direction.

Procedures:

1) Write necessary parameters on the positioning module.
2) Set the output relay defined as “Request to Change Velocity.”
3) Reset the relay defined as “Request to Change Velocity” after confirming the setting of the input relay defined as “Change Velocity ACK.” The motor starts the operation when change velocity ACK is set. (If there is a parameter error, the input relay defined as “Change Velocity ACK” is not set and the input relay defined as “Error Notification” is set.)
4) The end-of-positioning at the stopping time is judged according to the parameters specified before changing velocity.
Program Example:

This is an example of changing velocity during positioning. All required parameters are set beforehand in the integer type variables (I41 ~ ).

- Change Velocity (during Velocity Control Operation) Program

100  OUTPUT  SL,41  NOFORMAT ; I41
110  OUTPUT  SL,42  NOFORMAT ; I42
:
200  CONTROL  SL,101 ; $0080,$0080
210  LOOP1@
220  STATUS  SL,101 ; P
230  IF  BIT(P,9)=0  THEN LOOP1@
240  CONTROL  SL,101 ; $0000,$0080
250  LOOP2@
260  STATUS  SL,101 ; P
270  IF  BIT(P,9)=1  THEN LOOP2@

Application Function:

- Select Start Mode

When startup mode “1” is selected, the velocity control operation is not started until the external trigger input becomes ON after command execution. This mode is useful when it is desirable to wait for an external synthesizing signal before starting the operation.

When startup mode “2” is selected, the motor does not start moving until the output relay of an internal trigger is set after command execution.
7.2.9 Switching Velocity to Position Control

Switches to position control when executed during a velocity control mode operation.

The point at switching is defined as position “0”, and the positioning operation is performed following the parameter values set at the time of switching.

When velocity changes are mode with the control mode selection, the acceleration or deceleration from a change in velocity is expressed by the slope determined by a new target velocity and acceleration time. (The acceleration time parameters are also used for deceleration in velocity changes where the new target velocity is slower than the velocity set previously.)

When stopping after completing the positioning operation, the deceleration time is that set at the switch control mode execution.

Items to Note:

- This command is accepted only if all of the following conditions are satisfied: (1) not in error state, (2) not executing other command, (3) servo is ON, (4) not in the on-route interval of on-route operation, (5) not during the arc-interpolated operation, (6) not during origin-search, and (7) not during jog stepping. Otherwise, it is ignored. When executed in position control mode, the input relay defined as “Switch Control Mode ACK” is set, but the state of the positioning module is not changed.

- A parameter error occurs if improper values are set to the startup parameters. The motor stops with the servo OFF. Servo is not switched OFF when it operated while in the end-of-positioning.

- When switching the velocity to position control, the positioning is performed with the target velocity and target position effective at the time of the switch. Therefore, when you execute switch control while stopping in Velocity Control Mode at a target position other than “0”, the positioning operation starts.

Procedures:

1) Write the required parameters to the positioning module.
2) Set the output relay defined as “Request to Switch Control Mode.”
3) Reset the relay defined as “Request to Switch Control Mode” after confirming that the input relay defined as “Switch Control Mode ACK” is set. When “Switch Control Mode ACK” is set, the mode is switched to position control mode and the positioning operation starts. (If there is a parameter error, the input relay defined as “Switch Control Mode ACK” is not set and the input relay defined as “Error Notification” is set.)

Program Example:

This is an example of switching the velocity to position control. All necessary parameters are set beforehand in the integer type variables (I41 ~ ).

- Switch Velocity Control to Position Control Program

100 OUTPUT SL,41 NOFORMAT ; I41
110 OUTPUT SL,42 NOFORMAT ; I42
:
200 CONTROL SL,101 ; $0020,$0020
210 LOOP1@
Application Function:

- Select Startup Mode

When startup mode “1” is selected, the velocity control operation is not started until the external trigger input becomes ON after command execution. This mode is useful when it is desirable to wait for an external synthesizing signal before starting the operation.

When startup mode “2” is selected, the motor does not start moving until the output relay of an internal trigger is set after command execution.

- Switching using Encoder Z-phase

When this operation is executed with “2” as the control mode switching parameter, the control is switched to positioning control with the position where the specified Z-phase input edge is detected as the position “0” after the command execution. This setting can be used with select startup mode “1” or “2” at the same time. In this case, external / internal trigger is detected first.

- Origin-Offset Setting

When the command is executed with the origin offset other than “0”, the switched position is not “0”, but the specified value as the origin offset. The operation after switching is the positioning starting from that position.

- Position Detection Notification Function

If “1” or “2” is set as position detection mode, the input relay defined as “Position Detection Notification” is set to notify the application program when the current position arrives at the specified value for position detection.

The relay can be used as an interruption input.

The relay is not reset until the output relay defined as “Reset Position Detection Notification” is set.

Detecting position is the specified position when “1” is selected, and the encoder feedback position when “2” is selected.

- Switching velocity control to position control with operation direction change

If the operation direction in velocity control is different from the direction of the target position after switching, Decelerate-and-Stop is performed immediately and the positioning operation with the new target position starts. (If the operation direction does not change, but the new target position will be passed if the deceleration is started immediately, the same step occurs.)
7.2.10 Request to Decelerate and Stop

Decelerates and stops the motor during the positioning operation or, velocity control operation. The slope of deceleration, when either the positioning or velocity control is decelerated and stopped, is determined from the setup values at the time of the module startup or those where a command was executed to reset the deceleration mode.

Items to Note:

- This command is accepted only if all of the following conditions are satisfied: (1) not in error state, (2) not executing other command, (3) servo is ON, and (4) not during jog stepping. Otherwise, it is ignored.
- If the system is in on-route operation interval, it decelerates and stops after terminating the on-route operation. Execute immediate stop in order to stop in on-route operation interval.
- You cannot execute the Decelerate and Stop command during jog stepping operation. Terminate jog stepping, if necessary.
- The positioning module also accepts Decelerate and Stop command during end-of-positioning. (The corresponding ACK is set.)
- Do not execute start of a new positioning operation or Change Target Position command when decelerating in a Decelerate and Stop command. Otherwise, the operation cannot be guaranteed.

Procedures:

1) Set the output relay defined as “Request to Decelerate and Stop.”
2) Reset the relay defined as “Request to Decelerate and Stop” after confirming that the input relay defined as “Decelerate and Stop ACK” is set. The motor starts decelerating when Decelerate and Stop ACK is set.
3) The input relay defined as “End of Positioning” is set when the motor stops.

Program Example:

This is a program example for request to decelerate and stop.

- Request to Decelerate and Stop Program

```
200   CONTROL  SL,101 ; $0004,$0004
210   LOOP1@
220   STATUS   SL,101 ; P
230   IF   BIT(P,2)=0   THEN LOOP1@
240   CONTROL   SL,101 ; $0000,$0004
250   LOOP2@
260   STATUS   SL,101 ; P
270   IF   BIT(P,2)=1   THEN LOOP2@
280   LOOP3@
290   STATUS   SL,101 ; P
300   IF   BIT(P,13)=1   THEN LOOP3@
```
7.2.11 Request to Stop Immediately

Stops the operating motor immediately without deceleration during positioning operation, velocity control operation, etc.

Items to Note:

- This command is accepted only if the following conditions are met: (1) there is no error and (2) servo is ON. Otherwise, it is ignored.
- If the system, while it is operating at a high speed, is stopped suddenly, the system will likely experience mechanical shock.
- The positioning module accepts immediate stop command also during end-of-position. (The corresponding ACK is set.)

Procedures:

1) Set the output relay defined as “Request to Stop Immediately.”
2) Reset the relay defined as “Request to Stop Immediately” after confirming that the input relay defined as “Stop Immediately ACK” is set. The motor stops immediately when Stop Immediately ACK is set.
3) The input relay defined as “End of Positioning” is set when the motor stops.

Program Example:

This is an example for request to stop immediately.

- Request to Stop Immediately Program
  200 CONTROL SL,101 ; $0008,$0008
  210 LOOP1@
  220 STATUS SL,101 ; P
  230 IF BIT(P,3)=0 THEN LOOP1@
  240 CONTROL SL,101 ; $0000,$0008
  250 LOOP2@
  260 STATUS SL,101 ; P
  270 IF BIT(P,3)=1 THEN LOOP2@
  280 LOOP3@
  290 STATUS SL,101 ; P
  300 IF BIT(P,13)=1 THEN LOOP3@
7.2.12 Extended Command Instruction

This is a command used for various purposes related to setting extended command parameters.

(1) Servo ON

The Servo is ON and the brake is switched OFF when this command is executed during servo OFF. The positioning module accepts the command even if the servo is ON. (The corresponding ACK is set.)

Items to Note:

• This command is accepted only if there is no error. Otherwise, it is ignored.
• The following three conditions must be satisfied for proper operations. The positioning module should be connected to the servo driver/motor properly. Entry parameters should be set properly. And, the setting of the servo driver should be done properly. An error may occur and the servo may be set to OFF if this operation is executed without meeting these three conditions. If any error is detected, identify the error code and check the conditions above.

Procedures:

1) Write "0" to the Extended Command Parameter.
2) Set the output relay defined as “Extended Command Instruction.”
3) Reset the relay defined as “Extended Command Instruction” after confirming that the input relay defined as “Extended Command ACK” is set. The servo is switched ON when the Extended Command ACK is set. When the servo turns ON, the input relay defined as “End-of-positioning” is set.

Program Example:

This is an example for executing servo ON. Entry parameters are set properly in advance.

• Servo ON Program
  100  OUTPUT SL, 71 NOFORMAT ; 0
  200  CONTROL SL, 101 ; $ 0002, $ 0002
  210  LOOP1@
  220  STATUS SL, 101 ; P
  230  IF BIT (P, 1) = 0 THEN LOOP1@
  240  CONTROL SL, 101 ; $ 0000, $ 0002
  250  LOOP2@
  260  STATUS SL, 101 ; P
  270  IF BIT (P,1) =1 THEN LOOP2@
(2) Servo OFF

The Servo becomes OFF and the brake is switched ON when this command is executed during servo ON. The positioning module accepts the command even during servo OFF. (The corresponding ACK is set.)

Items to Note:

- This command is accepted only if there is no error. Otherwise, it is ignored.
- Servo Off command can be executed even when the motor is operating.

Procedures:

1) Write “1” to the Extended Command parameter.
2) Set the output relay defined as “Extended Command Instruction.”
3) Reset the relay defined as “Extended Command Instruction” after confirming that the input relay defined as “Extended Command ACK” is set. The servo is switched OFF when the Extended Command ACK is set. When the servo turns OFF, the input relay defined as “End-of-positioning” is set.

Program Example:

This is a program example for executing servo OFF.

- Servo OFF Program
  100 OUTPUT SL, 71 NOFORMAT ; 1
  200 CONTROL SL, 101 ; $0002, $0002
  210 LOOP1@
  220 STATUS SL, 101 ; P
  230 IF BIT (P, 1) = 0 THEN LOOP1@
  240 CONTROL SL, 101 ; $0000, $0002
  250 LOOP2@
  260 STATUS SL, 101 ; P
  270 IF BIT (P, 1) =1 THEN LOOP2@
(3) Brake ON

Turns ON the brake.

The positioning module accepts the command even if the brake is ON. (The corresponding ACK is set.)

Items to Note:

- This command is accepted only if the following conditions are satisfied: (1) there is no error and (2) servo is OFF. Otherwise, it is ignored.
- The positioning module operates the brake automatically depending on the state of the servo (ON / OFF). Thus, there is no need to operate the brake under normal conditions. This command is useful when it is necessary to operate the brake while the servo is OFF (e.g., for maintenance).

Procedures:

1) Write “2” to the Extended Command parameter.
2) Set the output relay defined as “Extended Command Instruction.”
3) Reset the relay defined as “Extended Command Instruction” after confirming that the input relay defined as “Extended Command ACK” is set.
(4) Brake OFF

Turns OFF the brake.

The positioning module accepts the command even if the brake is OFF. (The corresponding ACK is set.)

Items to Note:

- This command is accepted only if there is no error. Otherwise, it is ignored.
- The positioning module operates the brake automatically depending on the state of the servo (ON / OFF). Thus, there is no need to operate the brake under normal conditions. This command is useful when it is necessary to operate the brake while the servo is OFF (e.g., for maintenance).

Procedures: Error! No index entries found.

1) Write “3” to the Extended Command parameter.
2) Set the output relay defined as “Extended Command Instruction.”
3) Reset the relay defined as “Extended Command Instruction” after confirming that the input relay defined as “Extended Command ACK” is set.
(5) Stationary Deviation Adjust

A deviation is produced between the specified position at the time of stopping and the encoder feedback position when there is some error in the speed instruction voltage output and the servo driver input of the positioning module (both values should be 0). It is called the stationary deviation. When the precision of the stopping position is important, you can use this command to offset the deviation by setting an offset which cancels the stationary deviation to the velocity instruction voltage output of the positioning module.

Items to Note:

- This command is accepted only if the following conditions are satisfied: (1) there is no error and (2) no other commands are being executed. Otherwise, it is ignored.
- This command sets the offset for adjusting stationary deviation, it does not guarantee that the deviation at the time of stopping is zero.
- If the servo driver you are using has a function to adjust the offset of the velocity instruction voltage input (0V error), use that function instead of this command.

Procedures:

1) Before executing the Stationary Deviation Adjust command, read (1) Current Position status (instruction position) and (2) Current Position status (encoder position) during servo ON and end-of-positioning, then calculate the difference between (1) and (2) above.

   Note: To re-execute the command, calculate the latest difference by re-executing with the Stationary Deviation Adjust amount set to “0”. You can adjust properly only with a value obtained in the above way. The Stationary Deviation Adjust amount is set to “0” initially: until this command is executed after power-on.

2) Write the calculated result to the Stationary Deviation Adjust parameter and write “4” to the Extended Command parameter.

3) Set the output relay defined as “Extended Command Instruction.”

4) Reset the relay defined as “Extended Command Instruction” after confirming that the input relay defined as “Extended Command ACK” is set. Then, the Current Position status (instruction position) and the Current Position status (encoder position) will be almost the same, and the stationary deviation will be close to zero. When not all the parameters are proper, the input relay defined as “Extended Command ACK” is not set, and the input relay defined as “Error Notification” is set.
Program Example:

This is an example of executing the Stationary Deviation Adjust command. The Stationary Deviation Adjust amount has already been calculated and is stored in integer type variables (I72 and I73).

- Stationary Deviation Adjust Program

100  OUTPUT SL,71 NOFORMAT ; 4
110  OUTPUT SL,72 NOFORMAT ; I72
120  OUTPUT SL,73 NOFORMAT ; I73
:
200  CONTROL SL,101 ; $0002,$0002
210  LOOP1@
220  STATUS SL,101 ; P
230  IF BIT(P,1)=0 THEN LOOP1@
240  CONTROL SL,101 ; $0000,$0002
250  LOOP2@
260  STATUS SL,101 ; P
270  IF BIT(P,1)=1 THEN LOOP2@
(6) Driver Reset

When this command is executed during servo OFF, it turns the driver reset signal ON for 500 ms and then it OFF.

It is used for unconditional driver reset, e.g., when the module is powered on.

Items to Note:

- This command is accepted only if all of the following conditions are satisfied: (1) not in error state, (2) not executing other commands and (3) servo is OFF. Otherwise, it is ignored.

- This command does not recognize a driver alarm signal. When the command is executed, it outputs the driver reset signal regardless of the state of the driver alarm input signal. No error occurs when the driver alarm signal stays ON after the driver reset signal is output.

- The command cannot be accepted when a positioning module is in Error State (when error notification is set). Therefore, execute the normal Error Reset command when a driver alarm occurs during servo ON or an error occurs as a result of setting servo ON during driver alarm.

Procedures:

1) Write “5” to the Extended Command parameter.
2) Set the output relay defined as “Extended Command Instruction.”
3) Reset the relay defined as “Extended Command Instruction” after confirming that the input relay defined as “Extended Command ACK” is set. The driver reset signal is switched on for 500 ms when the Extended Command ACK is set.
(7) Manual Pulser Mode ON/OFF

When the Manual Pulser Mode ON is executed during servo ON in position control mode, the module is switched to manual pulser mode and the motor moves according to the input pulse of the manual pulser input. The ratio of the manual pulser input to the motor rotation number is specified using the Manual Pulser Scale Value parameter.

When Manual Pulser Mode OFF is executed, the system returns to normal position control mode.

A positioning module accepts the command even if the manual pulser mode ON is executed when in manual pulser mode, or if the the manual pulser mode OFF is executed when not in manual pulser mode. (The corresponding ACK is set.)

Items to Note:

- This command is accepted only if all of the following conditions are satisfied: (1) not in error state, (2) not executing other commands, (3) servo is ON and during end-of-positioning and (4) in position control mode. Otherwise, it is ignored.
- The motor cannot be operated by commands from the CPU module during manual pulser mode.
- During manual pulser mode, the manual pulser mode is reset when the servo becomes OFF due to some errors or execution of the servo OFF command.

Procedures:

- Manual Pulser Mode ON
  1) Write “10” to the Extended Command parameter. Write a value to the Pulser Scale Value parameter.
  2) Set the output relay defined as “Extended Command Instruction.”
  3) Reset the relay defined as “Extended Command Instruction” after confirming that the input relay defined as “Extended Command ACK” is set. The mode is switched to manual pulser when the Extended Command ACK is set. (When not all the parameters are proper, the input relay defined as “Switch Control Mode ACK” is not set, and the input relay defined as “Error Notification” is set.)

- Manual Pulser Mode OFF
  1) Write “11” to the Extended Command parameter.
  2) Set the output relay defined as “Extended Command Instruction.”
  3) Reset the relay defined as “Extended Command Instruction” after confirming that the input relay defined as “Extended Command ACK” is set. The manual pulser mode is reset when the Extended Command ACK is set.

Program Example:

This is an example of executing Manual Pulser Mode ON / OFF. All required parameters are set in advance in the integer type variable I74.
• Manual Pulser Mode ON / OFF Program

100  PULSON@
110  OUTPUT  SL,71  NONFORMAT ; 10
120  OUTPUT  SL,73  NONFORMAT ; 174
200  CONTROL  SL,101 ; $0002 ; 0002
210  LOOP1@
220  STATUS  SL, 101 ; P
230  IF  BIT  (P , 1) = 0 THEN LOOP1@
240  CONTROL  SL, 101 ; $0000, $0002
250  LOOP2@
260  STATUS  SL, 101 ; P
270  IF  BIT  (P , 1) =1 THEN LOOP2@

300  PULSOF@
310  OUTPUT  SL, 71 NOFORMAT ; 11
400  CONTROL  SL, 101 ; $0002, $0002
410  LOOP3@
420  STATUS  SL, 101 ; P
430  IF  BIT  (P , 1) =- THEN LOOP3@
440  CONTROL  SL, 101 ; $0000, $0002
450  LOOP4@
460  STATUS  SL, 101 ; P
470  IF  BIT  (P , 1) =1 THEN LOOP4@
7.2.13 On-route Operation

Starts on-route operation when a new positioning operation is started during a positioning operation (before end-of-position). On-route operation is used with linear interpolation operation. The system does not stop at each target position, but passes near the target positions (route points) and operates continuously.

The startup of positioning operation for the target positions (route points) after the first one is done while checking the status of the remaining deceleration time.

---

**Remaining deceleration time status**

This status reads the time of deceleration for stopping at the target position during the execution of the positioning operation. Its value is “0” during end-of-positioning, and “-1” during acceleration or while operating at the target velocity.

During the on-route operation, start the next positioning after checking that the above value is not larger than the preset values for each axis. This assures that the positioning for the next target position is started before reaching the intermediate target positions (route points). (This time lag is set in ms.)

---

**Items to Note:**

Avoid setting data that makes a positioning to finish earlier than the one started later. With such data, we cannot guarantee a proper execution of the on-route operation.

- For on-route operations, do not set data that will cause any positioning operation to finish earlier than previously started-up data. Otherwise, the operation cannot be guaranteed.

**Procedures:**

1) Write the operation parameters of each axis which are necessary to move to the first target position to the positioning module.

2) For each axis, set the output relay labeled “Start Operation Command” simultaneously (in the same scan).

3) Reset the relay defined as “Start Operation Command” of each axis after confirming that the input relays defined as “Start Operation Command ACK” of all axes are set. (If there is a parameter error, the input relay defined as “Start Operation Command ACK” is not set and the input relay defined as “Error Notification” is set.)

4) Write the operation parameters of each axis which are necessary to move to the next target position on the positioning module after confirming the setting of the input relay defined as “Start Operation Command ACK.”

5) For each axis, set the output relay defined as “Start Operation Command” simultaneously after confirming that the remaining deceleration time is not larger than the specified value for all axes.

6) Reset the relay defined as “Start Operation Command” of each axis after confirming that the input relays defined as “Start Operation Command ACK” of all axes are set. The relay defined as “Start Operation Command ACK” is not reset until the positioning which is started first finishes. (Start Operation command ACK” remains set during the on-route operation.)

7) Repeat 4-6 until the final target position is reached.

8) The relay defined as “End of Positioning” is set when the final target position is reached.
Program Example:

This is a program example of on-route operation with two axes. Modification of the positioning parameters for the second and later target positions, and judgement of the end of the on-route operation are not included.

All required parameters are set beforehand in the integer variables (IX41~, IY41~).

- On-route Operation Program

```
100  PASS@
110  OUTPUT SL,41 NONFORMAT ; IX41
120  OUTPUT SL,42 NONFORMAT ; IX42

: 210  OUTPUT SL,241 NONFORMAT ; IY41
220  OUTPUT SL,242 NONFORMAT ; IY42

: 300  LOOP1@
310  ENTER SL,115 NONFORMAT ; TIM1
320  ENTER SL,315 NONFORMAT ; TIM2
330  IF TIM1=-1 THEN LOOP1@
340  IF TIM2=-1 THEN LOOP1@
350  IF TIM1>TPASS THEN LOOP1@
360  IF TIM2>TPASS THEN LOOP1@
370  CONTROL SL,101 ; $0001,$0001
380  CONTROL SL,102 ; $0001,$0001
390  LOOP2@
400  STATUS SL,101 ; P1
410  STATUS SL,102 ; P2
420  IF BIT(P1,0)=0 THEN LOOP2@
430  IF BIT(P2,0)=0 THEN LOOP2@
440  CONTROL SL,101 ; $0001,$0001
450  CONTROL SL,102 ; $0001,$0001
460  LOOP3@
470  STATUS SL,101 ; P1
480  STATUS SL,102 ; P2
490  IF BIT(P1,0)=1 THEN LOOP3@
500  IF BIT(P2,0)=1 THEN LOOP3@
510  IF [terminating condition] THEN @PASEND
520  GOSUB [parameter update]
530  GOTO PASS@
540  PASEND@
550  LOOP4@
560  STATUS SL,101 ; P1
570  STATUS SL,102 ; P2
580  IF BIT(P1,13)=1 THEN LOOP4@
590  IF BIT(P2,13)=1 THEN LOOP4@
```
Application Function:

You can check the states of accelerating, decelerating, moving with constant velocity, during on-route operation, or the like, by reading the extended status.
7.2.14 Arc-Interpolation Operation

Performs the biaxial arc-interpolated operation. For details on the algorithm for calculating the parameters for an arc-interpolated operation, see Section 4.3, "Description of Parameters," provided earlier in this manual.

Items to Note:

- When performing arc-interpolation, the X axis and Y axis move independently. Therefore, when error occurs at one axis, the other axis continues moving. If it is necessary to stop the other axis, stop the motor by applying the servo OFF command to the moving axis after detecting the error with an application program (and checking the input relay defined as "Error Notification").

- If the velocity (the actual velocity, not the angular velocity) of each axis exceeds its limit value, only the axis that exceeds the limit runs within the limit. So the arc-interpolation will be improper.

Procedures:

1) Write parameters (initiate parameter and arc-interpolation parameter) for two axes to execute arc-interpolation.
2) Set the output relays defined as "Start Operation Command" for both axes simultaneously (in the same scan).
3) Reset the relay defined as "Start Operation Command" of both axes after confirming that the input relays defined as "Start Operation Command ACK" of both axes are set. The arc interpolated operation starts when the Start Operation Command ACK is set.

Program Example:

This is an example of arc-interpolation operation with two axes. All required parameters are set in advance in the integer variables (IX82~, IY82~).

- Arc-Interpolation Operation Program

100 OUTPUT SL,82 NONFORMAT ; IX82
:
200 OUTPUT SL,282 NONFORMAT ; IY82
:
300 CONTROL SL,101 ; $0001,$0001
310 CONTROL SL,102 ; $0001,$0001
320 LOOP1@
330 STATUS SL,101 ; P1
340 STATUS SL,102 ; P2
350 IF BIT(P1,0)=0 THEN LOOP1@
360 IF BIT(P2,0)=0 THEN LOOP1@
370 CONTROL SL,101 ; $0000,$0001
380 CONTROL SL,102 ; $0000,$0001
390 LOOP2@
400  STATUS  SL,101 ; P1
410  STATUS  SL,102 ; P2
420  IF  BIT(P1,0)=0  THEN  LOOP2@
430  IF  BIT(P2,0)=0  THEN  LOOP2@
440  LOOP3@
450  STATUS  SL,101 ; P1
460  STATUS  SL,102 ; P2
470  IF  BIT(P1,13)=0  THEN  LOOP3@
480  IF  BIT(P1,13)=0  THEN  LOOP3@
8. List of Error Codes

When an error occurs in the positioning module, the ERR1 indicator (ERR2 indicator for axis 2) on the front of the module lights up to indicate that an error has occurred. The state of an input relay defined as “Error Notification” is set ON and the error code status is set ON. The errors detectable in the positioning module is listed in the table below. If an error occurs, check the error code and execute the Error Reset command. Remove the cause of the error, if necessary.

8.1 Servo Errors

Errors related to motions of motor are called servo errors. If a servo error occurs, the state of the axis at which the error occurs changes to the Servo OFF state and the motor stops.
<table>
<thead>
<tr>
<th>Code</th>
<th>Types of Error</th>
<th>Cause of Errors</th>
</tr>
</thead>
</table>
| 12   | Driver alarm input detected | • Executed Servo ON when the driver alarm signal was ON.  
• The driver alarm signal turned ON during Servo ON. |
| 13   | Positive-direction limit input detected | • An attempt was made to operate the motor in the positive direction when the positive-direction limit input was ON.  
• The positive-direction limit input was turned ON when the motor was operated in the positive direction. |
| 14   | Negative-direction limit input detected | • An attempt was made to operate the motor in the negative direction while the negative-direction limit input was ON.  
• The negative-direction limit input was turned ON while the motor was operated in the negative direction. |
| 17   | Over-limit speed detected | • During Servo ON, the operation speed of the motor exceeded the value of the Over-Limit Speed Detection value parameter. |
| 18   | Over-limit acceleration detected | • During Servo ON, the acceleration of the motor (the rate of velocity change) exceeded the value of the Over-Limit Acceleration Detection value parameter. |
| 19   | Deviation error detected | • During Servo ON, the difference (deviation) between the command position and the encoder feedback value exceeded the value of the Deviation Error Detection Value parameter. |
| 31   | Positive-direction limit error (command position) | • During positioning, the command position exceeded the value of the Positive-Direction Limit parameter.  (If the target position exceeds the value of the Positive-Direction Limit parameter, it results in a parameter error.) |
| 32   | Negative-direction limit error (command position) | • During positioning, the command position exceeded the value of the Negative-Direction Limit parameter.  (If the target position exceeds the value of the Negative-Direction Limit parameter, it results in a parameter error.) |
| 41   | Origin-search is abnormal (Z-phase detection error) | • Z-phase is not detected within the Z-phase search range after shifting to Z-phase search during origin-search. |
| 51   | Absolute encoder error | • Received abnormal data while using a Sanyo Denki’s absolute encoder |
| 52   | Absolute encoder data error | • The received value of the absolute encoder was abnormal (receiving process was normal but the value was abnormal). |
| 53   | Absolute encoder data read error | • Failed to read absolute data while using a YASKAWA Electric’s absolute encoder. |
| 61   | Positioning completion time-out | • The difference between the command position and the encoder feedback position exceeded the value of the Positioning Judgement Range parameter within the interval specified by the Positioning Time-out Interval parameter when the motor stopped while positioning. |
| 62   | Arc-interpolated operation adjust pulse error | • At the start of the arc-interpolation operation, the difference between the starting position calculated using the arc-interpolation parameters and the actual position at which the operation starts exceeded the value of the Arc-Interpolation Adjust Pulse parameter.  
• At the end of the arc interpolated operation, the difference between the ending position calculated using the arc-interpolation parameters and the value of the Target Position parameter exceeded the value of the Arc Interpolation Adjust Pulse parameter. |
| 999  | Internal calculation error | • Obtained a value that cannot be calculated in the positioning module.  (This error does not usually occur, unless you have abnormal combinations of parameters.) |
| 9999 | Emergency stop | • Executed Servo ON in the state of emergency stop.  
• Emergency stop was issued during Servo ON. |
8.2 Parameter Errors

Parameter errors are related to the parameters referred to when executing commands other than Set Parameter commands (e.g., starting the motor). If a parameter error is detected, the ACK input relay corresponding to the command is not set ON. Instead, the input relay defined as “Error Notification” is set ON. If errors occur in more than one parameter, the error code of only one parameter will be stored. When a parameter error occurs while the motor is running (change-in-speed, change-in-target-position, on-route operation, and so on), Servo OFF state results and the motor stops. You will never get the Servo OFF state unless the motor is operating.

Table 8.2 Parameter Errors

<table>
<thead>
<tr>
<th>Codes</th>
<th>Type of Error</th>
<th>Cause of Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>10xx</td>
<td>Parameter error (xx indicates the lower-order two digits of the data position number of a given parameter.)</td>
<td>For information on cause of each error code, refer to the explanation on the corresponding parameters and commands.</td>
</tr>
</tbody>
</table>
8.3 Parameter Setting Errors

A parameter setting error is an error detected while setting parameters. The lower-order two digits of the error code are the same as the lower-order two digits of the data number of the corresponding parameter. If errors occurred for more than one parameter, the error code of the number of the first parameter for which an error was detected is stored. Even if an Error Reset command is executed, the input relay defined as “Error Notification” is not reset. It will be reset only after you write the correct parameters, execute Set Parameter again, and there are no errors in all the entry parameters.

Table 8.3 List of Parameter Setting Errors

<table>
<thead>
<tr>
<th>Code</th>
<th>Type of Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>2001</td>
<td>Positive-direction limit value error</td>
</tr>
<tr>
<td>2003</td>
<td>Negative-direction limit value error</td>
</tr>
<tr>
<td>2005</td>
<td>Velocity limit value error</td>
</tr>
<tr>
<td>2007</td>
<td>Over-limit velocity detection value error</td>
</tr>
<tr>
<td>2009</td>
<td>Over-limit acceleration detection value error</td>
</tr>
<tr>
<td>2011</td>
<td>Deviation error detection error</td>
</tr>
<tr>
<td>2013</td>
<td>Motor rotation direction error</td>
</tr>
<tr>
<td>2016</td>
<td>Position loop range error</td>
</tr>
<tr>
<td>2017</td>
<td>Speed feedforward factor error</td>
</tr>
<tr>
<td>2018</td>
<td>Encoder method specification error</td>
</tr>
<tr>
<td>2019</td>
<td>Encoder’s multiplication error</td>
</tr>
<tr>
<td>2020</td>
<td>Velocity/voltage ratio error</td>
</tr>
<tr>
<td>2022</td>
<td>Reserved parameter error</td>
</tr>
<tr>
<td>2023</td>
<td>Absolute offset value error</td>
</tr>
<tr>
<td>2025</td>
<td>Absolute pulse count error</td>
</tr>
<tr>
<td>2027</td>
<td>Sanyo Denki’s ABS bit-length error</td>
</tr>
<tr>
<td>2028</td>
<td>Sanyo Denki’s ABS operation-direction error</td>
</tr>
<tr>
<td>2029</td>
<td>Contact input mode error</td>
</tr>
<tr>
<td>2030</td>
<td>Reserved parameter error</td>
</tr>
<tr>
<td>2031</td>
<td>Reserved parameter error</td>
</tr>
<tr>
<td>2032</td>
<td>Reserved parameter error</td>
</tr>
</tbody>
</table>

You may have to identify parameter-setting errors for each parameter or with respect to given combinations of parameters. It is possible that an error can occur even if each parameter’s value is within the valid range of setting. For details, refer to the explanation of each parameter.
9. Adjustment of Servo Parameters

The positioning module has two parameters for adjustment: the Position Loop Range parameter and the Velocity Feedforward Factor parameter. This chapter explains the adjustment method for these 2 parameters.

A system that uses a servo motor generally requires the velocity loop and the position loop to be adjusted properly.

If you are using the positioning module, adjust the velocity loop from the servo driver and the position loop from the positioning module. If they are not adjusted properly, the motor performance will be affected.

The larger the values specified for the velocity loop range and the position loop gain, the more responsive the system becomes to your control. However, if the values are too large, problems such as abnormal vibration and speed ripples may occur.

(1) Preparation

Before adjusting the two parameters, refer to section 4.4, “Examples of Entry Parameters Setting” and set the entry parameters properly.

Specifically, refer to the instruction manual for the rated rotation count of the motor, the value of velocity command voltage when the rotation is rated, and the encoder pulse count per rotation.

If the servo driver you are using allows these values to be changed in the setting, be sure to check the contents of the setting and calculate the correct values for the entry parameters of the positioning module. If the values are not appropriate, the positioning module will not operate correctly.

(2) Procedures to adjust the Velocity Loop Range and the Position Loop Gain

When the servo driver can drive the motor without the positioning module being connected:

(1) Follow the explanation in the instruction manual of the servo driver to adjust the velocity loop. You should do the adjustment procedure without connecting the positioning module. If you cannot adjust it properly before connecting the positioning module, the motor will not run normally after you connect it.

(2) Connect the positioning module with the servo driver. Without changing the initial value of the position loop gain, set Servo ON.

(3) Adjust the position loop gain. Set the range with the maximum value which does not cause abnormal vibration of the motor. You need to set Servo OFF before changing the position loop gain.
If the servo driver cannot drive the motor unless the positioning module is connected:

1. Without changing the initial values of the velocity loop range on the servo driver (gain) and the position loop gain on the positioning module (the value at the shipment of the product), set Servo ON. (If you are sure from your experience that the initial values cause problems, change the values as necessary before setting Servo ON.)

2. Adjust the velocity loop range first, and then adjust the position loop gain. First, adjust the velocity loop range according to the instruction manual of the servo driver. To avoid abnormal vibration, try jog stepping to check if the motor rotates smoothly. Set the maximum value of the velocity loop range which does not cause abnormal vibration. For details, refer to the servo driver instruction manual.

3. Next, adjust the position loop gain. The value should be the maximum value which does not cause an abnormal vibration of the motor. Before changing the value of the position loop gain, set Servo OFF.

3) Adjustment of the Velocity Feedforward Factor

You do not usually have to change the initial values. However, adjustment may be necessary in the following cases. In each of the cases, you have to adjust the velocity loop range and the position loop gain before adjusting the velocity feedforward factor.

- The system whose long response time of the motor may cause a problem, such as when you are using a machine tool (the value must be increased).
- The system whose overshoot while positioning may cause a problem (the value must be decreased).

Figure 9.1 Adjusting Servo Parameters
10. External Contact Signals

For a list of the external contact signals, refer to Figure 2.7, External Contacts. This chapter describes the specification of each signal and how to connect it.

10.1 Shield (FG)

Shield (FG) is connected to Frame GND (FG) via the base module. The shield of the velocity-command voltage output and the encoder input signal must be connected here.

![Diagram of Shield (FG) connection](image)

Figure 10.1 Connection of Shield (FG)
10.2 Velocity-command Voltage Output

This is an analog velocity-command voltage output. Be sure to use a shielded twisted-pair cable. The velocity-command voltage output (GND) is connected to the signal GND (SG) inside the module, and is connected to FG in the base module.

- Specifications

<table>
<thead>
<tr>
<th>Item</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output voltage</td>
<td>-10 to +10 V</td>
</tr>
<tr>
<td>Load current</td>
<td>5 mA max.</td>
</tr>
</tbody>
</table>

Figure 10.2 Connecting velocity Command Voltage Output
10.3 Encoder/Manual Pulser Signal GND (SG)

Connect the base potential of the encoder input signal and the manual pulser input signal. They are connected to the GND signal in the module, and is connected to FG via the backboard.
10.4 Encoder Input and Manual Pulser Input

Connect the encoder input signal and the manual pulser input signal. Use a shielded 3-pair (4-pair) twisted cable, and connect the shield line to a shield pin. The signal level is a differential signal conforming to RS422 standards. Connect the base potential of the signal to the SG pin. SG is connected to FG via the backboard. The absolute data is received from the encoder A-phase input.

<table>
<thead>
<tr>
<th>Item</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input</td>
<td>Conforms to RS-422 standards (220 Ω terminating resistor).</td>
</tr>
<tr>
<td>Frequency of input</td>
<td>500 kHz (after 4 x multiplication, 2 MHz) or less</td>
</tr>
<tr>
<td>Disconnection detection</td>
<td>None</td>
</tr>
</tbody>
</table>

The description of the A-phase and the B-phase of the servo driver may correspond inversely to the A-phase and the B-phase of this module. Be sure to confirm the relationship of the encoder signals to the servo driver and the encoders you are connecting. Connect with the B-phase encoder the signal whose phase proceeds when the positive velocity command voltage is impressed. If you connect them in an inverse manner, the position control loop becomes a positive feedback loop and the module does not run properly.

**Figure 10.3 Connecting Encoder Input Signals**

**Figure 10.4 Relationship between the Phases of Velocity Command Voltage Output and Encoder Input Signal**
10.5 External Contact Output

This output is a shared common sink-shaped open collector output. A 24 V DC of external power supply is required. It is insulated from the internal circuitry by a photo coupler. Since the external contact output (24 V), and (COM) are connected to the power supply input for the external contact output (24 V-in) and (0 V-in) in the module, the power supply pin and the servo driver connector pin can be separated. (The two axes share the external power supply.)

<table>
<thead>
<tr>
<th>Item</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Isolation</td>
<td>Photocoupler isolated</td>
</tr>
<tr>
<td>Rated load voltage</td>
<td>24 V DC</td>
</tr>
<tr>
<td>Maximum load current</td>
<td>100 mA/point</td>
</tr>
<tr>
<td>OFF-time leak current</td>
<td>0.1 mA or less</td>
</tr>
<tr>
<td>ON-time residual voltage</td>
<td>1 V DC or less</td>
</tr>
<tr>
<td>Common method</td>
<td>Common to every point (both axis, sink output)</td>
</tr>
<tr>
<td>External power supply</td>
<td>24 V DC, 10 mA (consumed only within the module)</td>
</tr>
</tbody>
</table>

Figure 10.5 Connecting External Contact Outputs
10.6 External Contact Input

This is a shared common 24V DC input compatible both with [+common] and [-common]. It is insulated from the internal circuitry by a photo coupler.

<table>
<thead>
<tr>
<th>Item</th>
<th>Specification of target velocity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Isolation</td>
<td>Photocoupler isolated</td>
</tr>
<tr>
<td>Input impedance</td>
<td>5.6 kΩ</td>
</tr>
<tr>
<td>Rated input voltage (operating voltage range)</td>
<td>24 V DC (20.4 to 26.4 V DC)</td>
</tr>
<tr>
<td>Rated input current</td>
<td>4.1 mA</td>
</tr>
<tr>
<td>ON voltage/current (for &quot;a&quot; contact)</td>
<td>16 V DC min., 3.2 mA min.</td>
</tr>
<tr>
<td>OFF voltage/current (for &quot;a&quot; contact)</td>
<td>5.8 V DC max., 0.9 mA max.</td>
</tr>
<tr>
<td>Common method</td>
<td>Shared common</td>
</tr>
</tbody>
</table>

Figure 10.6 Connecting External Contact Inputs
10.7 Emergency Stop Input

The emergency stop input is a 24 V DC/“b” contact input independent of the external contact inputs. Supply the 24V DC power between “a” and “b” when the emergency stop is released. Only the signal of the axis 1 connector is effective. This input is common to axis 1 and axis 2.

- Specifications

<table>
<thead>
<tr>
<th>Item</th>
<th>Specifications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Isolation</td>
<td>Photocoupler isolated</td>
</tr>
<tr>
<td>Input impedance</td>
<td>5.6 KΩ</td>
</tr>
<tr>
<td>Rated input voltage</td>
<td>24 V DC</td>
</tr>
<tr>
<td>Rated input current</td>
<td>4.1 mA</td>
</tr>
<tr>
<td>Input voltage/current when the emergency stop is released</td>
<td>16 V DC min., 3.2 mA min.</td>
</tr>
<tr>
<td>Input voltage/current during emergency stop</td>
<td>5.8 V DC max., 0.9 mA max.</td>
</tr>
</tbody>
</table>

Figure 10.7 Connecting Emergency Stop Input
10.8 SEN Signals (for YASKAWA Electric’s Absolute Encoders)

This is the SEN signal for YASKAWA Electric’s Absolute Encoders. The output is a 5 V DC signal.

- Specifications

<table>
<thead>
<tr>
<th>Item</th>
<th>Specifications</th>
</tr>
</thead>
<tbody>
<tr>
<td>No-load output voltage</td>
<td>5 V DC</td>
</tr>
<tr>
<td>Maximum load current</td>
<td>10 mA (at 4.5 V DC)</td>
</tr>
<tr>
<td>OFF-time leak current</td>
<td>10 μA max.</td>
</tr>
</tbody>
</table>

![Figure 10.8 Connecting SEN Signal](image)

Figure 10.8 Connecting SEN Signal
11. Examples of Interconnections to Servo Drivers

This chapter presents examples of interconnections of the module with major servo drivers. Note that the figures indicate canonical connections. Other signals may also have to be connected for given purposes.

11.1 Interconnections with YASKAWA Electric's Σ-series Driver

Figure 11.1 is an example of interconnections when using an absolute encoder. If you are not using an absolute encoder, the SEN signal and the BAT (battery) need not be connected.
11. Examples of Interconnections to Servo Drivers

Figure 11.1 Interconnecting to YASKAWA Electric's ∑-series Driver
11.2 Interconnections with Sanyo Denki's P-series Driver (Incremental Encoder)

Figure 11.2 is an example of interconnections when using the PU0-series incremental-encoder servo driver.

![Diagram of interconnections between Positioning Module and PU0 Series driver]

Figure 11.2 Interconnecting Positioning Module to Sanyo Denki's P-series Driver (1)
11.3 Example of Connection with Sanyo Denki's P-series Driver (ABS-R II absolute encoder)

Figure 11.3 is an example of interconnections with the PU0-series ABS-R II absolute encoder servo driver.

![Interconnecting Diagram](image-url)

Figure 11.3  Interconnecting Positioning Module to Sanyo Denki's P-series Driver (II)
### Table 1  Parameter Sheet (for entry parameters)

<table>
<thead>
<tr>
<th>Parameter name</th>
<th>Axis 1 position number</th>
<th>Axis 2 position number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Positive-direction limit</td>
<td>1/2</td>
<td>201/202</td>
</tr>
<tr>
<td>Negative-direction limit</td>
<td>3/4</td>
<td>203/204</td>
</tr>
<tr>
<td>Velocity limit</td>
<td>5/6</td>
<td>205/206</td>
</tr>
<tr>
<td>Over-limit detection</td>
<td>7/8</td>
<td>207/208</td>
</tr>
<tr>
<td>Over-limit acceleration detection</td>
<td>9/10</td>
<td>209/210</td>
</tr>
<tr>
<td>Deviation error detection</td>
<td>11/12</td>
<td>211/212</td>
</tr>
<tr>
<td>Motor rotation direction</td>
<td>13</td>
<td>213</td>
</tr>
<tr>
<td>Contact input polarity</td>
<td>14</td>
<td>214</td>
</tr>
<tr>
<td>Contact output polarity</td>
<td>15</td>
<td>215</td>
</tr>
<tr>
<td>Position loop gain</td>
<td>16</td>
<td>216</td>
</tr>
<tr>
<td>Velocity feedforward factor</td>
<td>17</td>
<td>217</td>
</tr>
<tr>
<td>Encoder method specification</td>
<td>18</td>
<td>218</td>
</tr>
<tr>
<td>Encoder multiplication number</td>
<td>19</td>
<td>219</td>
</tr>
<tr>
<td>Velocity/voltage ratio</td>
<td>20/21</td>
<td>220/221</td>
</tr>
<tr>
<td>(Reserved)</td>
<td>22</td>
<td>222</td>
</tr>
<tr>
<td>Absolute offset value</td>
<td>23/24</td>
<td>223/224</td>
</tr>
<tr>
<td>Absolute pulse count</td>
<td>25/26</td>
<td>225/226</td>
</tr>
<tr>
<td>Sanyo Denki's ABS bit length</td>
<td>27</td>
<td>227</td>
</tr>
<tr>
<td>Sanyo Denki's ABS rotation direction</td>
<td>28</td>
<td>228</td>
</tr>
<tr>
<td>Contact input mode</td>
<td>29</td>
<td>229</td>
</tr>
<tr>
<td>(Reserved)</td>
<td>30</td>
<td>230</td>
</tr>
<tr>
<td>(Reserved)</td>
<td>31</td>
<td>231</td>
</tr>
<tr>
<td>(Reserved)</td>
<td>32</td>
<td>232</td>
</tr>
</tbody>
</table>
### Table 2  Parameter Sheet (other than entry parameters)

<table>
<thead>
<tr>
<th>Parameter name</th>
<th>Axis 1 position number</th>
<th>Axis 2 position number</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>[Startup parameters]</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Target velocity</td>
<td>41/42</td>
<td>241/242</td>
</tr>
<tr>
<td>Target position</td>
<td>43/44</td>
<td>243/244</td>
</tr>
<tr>
<td>Target position mode</td>
<td>45</td>
<td>245</td>
</tr>
<tr>
<td>Acceleration time</td>
<td>46</td>
<td>246</td>
</tr>
<tr>
<td>Acceleration mode selection</td>
<td>47</td>
<td>247</td>
</tr>
<tr>
<td>Acceleration parameter 1</td>
<td>48</td>
<td>248</td>
</tr>
<tr>
<td>Acceleration parameter 2</td>
<td>49</td>
<td>249</td>
</tr>
<tr>
<td>Deceleration time</td>
<td>50</td>
<td>250</td>
</tr>
<tr>
<td>Deceleration mode selection</td>
<td>51</td>
<td>251</td>
</tr>
<tr>
<td>Deceleration parameter 1</td>
<td>52</td>
<td>252</td>
</tr>
<tr>
<td>Deceleration parameter 2</td>
<td>53</td>
<td>253</td>
</tr>
<tr>
<td>Positioning judgement range</td>
<td>54/55</td>
<td>254/255</td>
</tr>
<tr>
<td>Positioning time-out interval</td>
<td>56</td>
<td>256</td>
</tr>
<tr>
<td>Interpolation mode selection</td>
<td>57</td>
<td>257</td>
</tr>
<tr>
<td>Startup mode selection</td>
<td>58</td>
<td>258</td>
</tr>
<tr>
<td>Position detection mode</td>
<td>59</td>
<td>259</td>
</tr>
<tr>
<td>Position detection set point</td>
<td>60/61</td>
<td>260/261</td>
</tr>
<tr>
<td><strong>[Startup parameters related to origin search, etc.]</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Origin-search mode</td>
<td>62</td>
<td>262</td>
</tr>
<tr>
<td>Origin-search direction</td>
<td>63</td>
<td>263</td>
</tr>
<tr>
<td>Z-phase edge selection</td>
<td>64</td>
<td>264</td>
</tr>
<tr>
<td>Z-phase pulse count</td>
<td>65</td>
<td>265</td>
</tr>
<tr>
<td>Z-phase search range</td>
<td>66/67</td>
<td>266/267</td>
</tr>
<tr>
<td>Origin offset value</td>
<td>68/69</td>
<td>268/269</td>
</tr>
<tr>
<td><strong>[Extended command parameters]</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Extended command parameter</td>
<td>71</td>
<td>271</td>
</tr>
<tr>
<td>Stationary deviation adjust amount</td>
<td>72/73</td>
<td>272/273</td>
</tr>
<tr>
<td>Manual pulser scale value</td>
<td>74</td>
<td>274</td>
</tr>
<tr>
<td><strong>[Control mode switching parameters]</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Control mode switching parameter</td>
<td>81</td>
<td>281</td>
</tr>
<tr>
<td><strong>[Arc-interpolation operation parameters]</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Center position</td>
<td>82/83</td>
<td>282/283</td>
</tr>
<tr>
<td>Radius</td>
<td>84/85</td>
<td>284/285</td>
</tr>
<tr>
<td>Starting angle</td>
<td>86/87</td>
<td>286/287</td>
</tr>
<tr>
<td>Angular travel</td>
<td>88/89</td>
<td>288/289</td>
</tr>
<tr>
<td>Angular speed set point</td>
<td>90/91</td>
<td>290/291</td>
</tr>
<tr>
<td>Acceleration time</td>
<td>92</td>
<td>292</td>
</tr>
<tr>
<td>Deceleration time</td>
<td>93</td>
<td>293</td>
</tr>
<tr>
<td>Target position</td>
<td>94/95</td>
<td>294/295</td>
</tr>
<tr>
<td>Adjustment pulse range</td>
<td>96</td>
<td>296</td>
</tr>
<tr>
<td>Status name</td>
<td>Axis 1 position number</td>
<td>Axis 2 position number</td>
</tr>
<tr>
<td>-------------------------------------------------</td>
<td>------------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>Error status</td>
<td>101</td>
<td>301</td>
</tr>
<tr>
<td>Detailed error code</td>
<td>102</td>
<td>302</td>
</tr>
<tr>
<td>Contact input status</td>
<td>103</td>
<td>303</td>
</tr>
<tr>
<td>Current position status (Command position)</td>
<td>104/105</td>
<td>304/305</td>
</tr>
<tr>
<td>Current speed status (Command position)</td>
<td>106/107</td>
<td>306/307</td>
</tr>
<tr>
<td>Current position status (Encoder position)</td>
<td>108/109</td>
<td>308/309</td>
</tr>
<tr>
<td>Current speed status (Encoder position)</td>
<td>110/111</td>
<td>310/311</td>
</tr>
<tr>
<td>Target positions status</td>
<td>112/113</td>
<td>312/313</td>
</tr>
<tr>
<td>Extended status</td>
<td>114</td>
<td>314</td>
</tr>
<tr>
<td>Remaining deceleration time</td>
<td>115</td>
<td>315</td>
</tr>
<tr>
<td>Contact output status</td>
<td>116</td>
<td>316</td>
</tr>
<tr>
<td>Absolute data status (YASKAWA Electric's ABS)</td>
<td>117/118</td>
<td>317/318</td>
</tr>
<tr>
<td>Absolute data status (Sanyo Denki's ABS)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Revision History

<table>
<thead>
<tr>
<th>Edition</th>
<th>Date</th>
<th>Revised Item</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st</td>
<td>Feb. 1999</td>
<td>New publication</td>
</tr>
</tbody>
</table>

Written by Product Marketing Section, PLC Center
Industrial Automation Business Head Quarter.
Yokogawa Electric Corporation

Published by Yokogawa Electric Corporation
2-9-32 Nakacho, Musashino-shi, Tokyo 180-8750, JAPAN

Printed by Yokogawa Graphic Arts Co., Ltd.