User's Manual
For
PCL6025
Pulse Control LSI

NPM

Nippon Pulse Motor Co., Ltd.

[Preface]

Thank you for considering our pulse control LSI, the "PCL6025."
To learn how to use the PCL6025, read this manual to become familiar with the product.
The handling precautions for installing this IC are described at the end of this manual. Make sure to read them before installing the IC.

[Cautions]

(1) Copying all or any part of this manual without written approval is prohibited.
(2) The specifications of this IC may be changed to improve performance or quality without prior notice.
(3) Although this manual was produced with the utmost care, if you find any points that are unclear, wrong, or have inadequate descriptions, please let us know.
(4) We are not responsible for any results that occur from using this IC, regardless of item (3) above.

- Explanation of the descriptions in this manual

1. The "x" and "y" of terminal names and bit names refer to the X axis and Y axis, respectively.
2. Terminals with a bar over the name (ex. $\overline{RST}$) are negative logic. Their logic cannot be changed.
   Terminals without a bar over the name are positive logic. Their output logic can be changed.
3. When describing the bits in registers, "n" refers to the bit position. A "0" means that the bit is in position 0, and that it is prohibited to write to any bit other than the "0" bit. Finally, this bit will always return a "0" when read.

INDEX

1. Outline and Features..1
   1-1.Outline..1
   1-2. Features..1

2. Specifications..5

3. Terminal Assignment Diagram..6

4. Terminal Pin Numbers, Names and Functions..7

5. Block Diagram..12

6. CPU Interface..13
   6-1. Setting up connections to a CPU..13
   6-2. Precautions for designing hardware..13
   6-3. CPU interface circuit block diagram..14
   6-4. Address map..16
     6-4-1. Axis arrangement map..16
     6-4-2. Internal map of each axis..16
   6-5. Description of the map details..18
     6-5-1. Write the command code and axis selection (COMW, COMB)..18
     6-5-2. Write to an output port (OTPW, OTPB)..18
     6-5-3. Write/read the input/output buffer (BUFW, BUFB)..18
     6-5-4. Reading the main status (MSTSW, MSTSB)..19
     6-5-5. Reading the sub status and input/output port. (SSTSW, SSTSB, IOPB)..19

7. Commands (Operation and Control Commands)..20
   7-1. Operation commands..20
     7-1-1. Procedure for writing an operation command..20
     7-1-2. Start command..20
     7-1-3. Speed change command..21
     7-1-4. Stop command..21
     7-1-5. NOP (do nothing) command..21
   7-2. General-purpose output bit control commands..22
   7-3. Control command..22
     7-3-1. Software reset command..22
     7-3-2. Counter reset command..23
     7-3-3. ERC output control command..23
     7-3-4. Pre-register control command..23
     7-3-5. PCS input command..23
     7-3-6. LTCH input (counter latch) command..23
   7-4. Register control command..24
     7-4-1. Procedure for writing data to a register..24
     7-4-2. Procedure for reading data from a register..24
     7-4-3. Table of register control commands..25
   7-5. General-purpose output port control..26
     7-5-1. Control data writing procedures..26
     7-5-2. Control data bit allocation..26

8. Registers 27
8-1. Table of registers 27
8-2. Pre-registers 28
8-3. Description of the registers 29
8-3-1. RMV (PRMV) registers (Feed amount, target position) 29
8-3-2. RFL (PRFL) registers (Initial speed) 29
8-3-3. RFH (PRFH) registers (Operation speed) 29
8-3-4. RUR (PRUR) registers (Acceleration rate) 29
8-3-5. RDR (PRDR) registers (Deceleration rate) 30
8-3-6. RMG (PRMG) registers (Speed magnification rate) 30
8-3-7. RDP (PRDP) registers (Ramping-down point) 30
8-3-8. RMD (PRMD) registers (Operation mode) 31
8-3-9. RIP (PRIP) registers (Circular interpolation center position, master axis feed amount) 33
8-3-10. RUS (PRUS) registers (S-curve acceleration range) 33
8-3-11. RDS (PRDS) registers (S-curve deceleration range) 33
8-3-12. RFA register (Speed at amount correction) 33
8-3-13. RENV1 register (Environment setting 1) 34
8-3-14. RENV2 register (Environment setting 2) 36
8-3-15. RENV3 register (Environment setting 3) 38
8-3-16. RENV4 register (Environment setting 4) 40
8-3-17. RENV5 register (Environment setting 5) 42
8-3-18. RENV6 register (Environment setting 6) 43
8-3-19. RENV7 register (Environment setting 7) 43
8-3-20. RCUN1 register (COUNTER1 <command position>) 44
8-3-21. RCUN2 register (COUNTER2 <machine position>) 44
8-3-22. RCUN3 register (COUNTER3 <deflection counter>) 44
8-3-23. RCUN4 register (COUNTER4 <general-purpose counter>) 44
8-3-24. RCMP1 register (Comparison data for comparator 1) 45
8-3-25. RCMP2 register (Comparison data for comparator 2) 45
8-3-26. RCMP3 register (Comparison data for comparator 3) 45
8-3-27. RCMP4 register (Comparison data for comparator 4) 45
8-3-28. RCMP5 register (Comparison data for comparator 5) 45
8-3-29. RIRQ register (Specify event interruption cause) 46
8-3-30. RLTC1 register (COUNTER1 latch data) 47
8-3-31. RLTC2 register (COUNTER2 latch data) 47
8-3-32. RLTC3 register (COUNTER3 latch data) 47
8-3-33. RLTC4 register (COUNTER4 latch data) 47
8-3-34. RSTS register (Extension status) 48
8-3-35. REST register (Error INT status) 49
8-3-36. RIST register (Event INT status) 50
8-3-37. RPLS register (Number of residual pulses to feed) 51
8-3-38. RSPD register (EZ counter, current speed monitor) 51
8-3-39. RSDC register (Automatically calculated ramping-down point) 51
8-3-40. RCI (PRCI) registers (Number of steps for interpolation) 51
8-3-41. RIPS register (Interpolation status) 52

9. Operation Mode 53
9-1. Continuous operation mode using command control 53
9-2. Positioning operation mode 53
9-2-1. Positioning operation (specify a target position using an incremental value) 53
9-2-2. Return the specified position to zero 53
9-2-3. Return the machine position to zero 53
9-2-4. One pulse operation 53
9-2-5. Timer operation 54
9-3. Pulsar (PA/PB) input mode 55
9-3-1. Continuous operation using a pulsar input 56
9-3-2. Positioning operations using a pulsar input 56
9-3-3. Command position zero return operation using a pulsar input 57
9-3-4. Machine position zero return operation using a pulsar input 57
9-4. External switch (±DR) operation mode 58
9-4-1. Continuous operation using an external switch 58
9-4-2. Positioning operation using an external switch 59
9-5. Zero position operation mode 60
9-5-1. Zero return operation 61
9-5-2. Leaving the zero position operations 69
9-5-3. Zero search operation 69
9-6. EL or SL operation mode 70
9-6-1. Feed until reaching an EL or SL position 71
9-6-2. Leaving an EL or SL position 71
9-7. EZ count operation mode 71
9-8. Interpolation operations 72
9-8-1.Interpolation operations 72
9-8-2. Interpolation control axis 72
9-8-3. Constant synthetic speed control 73
9-8-4. Linear interpolation 1 73
9-8-5. Linear interpolation 2 74
9-8-6. Circular interpolation 75
9-8-7. Operation during interpolation 76

10. Speed Patterns 78
10-1. Speed patterns 78
10-2. Speed pattern settings 79
10-3. FH correction (eliminate triangle driving) 84
10-4. Example of setting up an acceleration/deceleration speed pattern 88
10-5. Changing speed patterns while in operation 89

11. Description of the Functions 90
11-1. Reset 90
11-2. Position override 91
11-2-1. Target position override 1 91
11-2-2. Target position override 2 (PCS signal) 92
11-3. Output pulse control 93
11-3-1. Output pulse mode 93
11-3-2. Control the output pulse width and operation complete timing 94
11-4. Idling control 95
11-5. Mechanical external input control 96
11-5-1. +EL, -EL signal 96
11-5-2. SD signal 97
11-5-3. ORG, EZ signals 99

11-6. Servomotor I/F 100
11-6-1. INP signal 100
11-6-2. ERC signal 101
11-6-3. ALM signals 102
11-7. External start, simultaneous start 103
11-7-1. $\overline{CSTA}$, $\overline{STA}$ signal 103
11-7-2. PCS signal 104
11-8. External stop / simultaneous stop 105
11-9. Emergency stop 106
11-10. Counter 107
11-10-1. Counter type and input method 107
11-10-2. Counter reset 108
11-10-3. Latch the counter and count condition 109
11-11. Comparator 111
11-11-1. Comparator types and functions 111
11-11-2. Software limit function 114
11-11-3. Out of step stepper motor detection function 115
11-12. Backlash correction and slip correction 116
11-13. Vibration restriction function 117
11-14. Synchronous starting 118
11-14-1. Start triggered by another axis stopping 119
11-14-2. Starting from an internal synchronous signal 120
11-15. Output an interrupt signal 122

12. Electrical Characteristics 124
12-1. Absolute maximum ratings 124
12-2. Recommended operating conditions 124
12-3. DC characteristics 125
12-4. AC characteristics 1) (reference clock) 125
12-5. AC characteristics 2) (CPU I/F) 126
12-5-1. CPU-I/F 1) (IF1 = H, IF0 = H) Z80 126
12-5-2. CPU-I/F 2) (IF1 = H, IF0 = L) 8086 127
12-5-3. CPU-I/F 3) (IF1 = L, IF0 = L) H8 128
12-5-4. CPU-I/F 4) (IF1 = L, IF0 = L) 68000 129
12-6. Operation timing 130

13. External Dimensions 132

Appendix: List of various items 133
Appendix 1: List of commands 133
Appendix 2: Setting speed pattern 135
Appendix 3: Label list 138

Handling Precautions 146
1. Design precautions 146
2. Precautions for transporting and storing LSIs 146
3. Precautions for installation 146
4. Other precautions 147

1. Outline and Features

1-1. Outline

The PCL6025 is a CMOS LSI designed to provide the oscillating, high-speed pulses needed to drive stepper motors and servomotors (pulse string input types).
It can offer various types of control over the pulse strings and therefore the motor performance. These include continuous feeding, positioning, zero return at a constant speed, linear acceleration/deceleration, and S-curve acceleration/deceleration.
Two axes can be controlled simultaneously by this chip. It can provide linear interpolation, circular interpolation, check of the PCL operation status, and output interrupt signals under lots of conditions. It also integrates an interface for servo control drivers.
These functions can be used with simple commands. The intelligent design philosophy reduces the burden on the CPU units to control motors.

1-2. Features

♦ CPU-I/F
The PCL6025 contains the following CPU interface circuits.
1) 8-bit interface for Z80 CPU.
2) 16-bit interface for 8086 CPU.
3) 16-bit interface for H8 CPU.
4) 16-bit interface for 68000 CPU.

♦ Acceleration/Deceleration speed control
Linear acceleration/deceleration and S-curve acceleration/deceleration are available.
Linear acceleration/deceleration can be inserted in the middle of an S-curve acceleration/deceleration curve. (Specify the S-curve range.)
The S-curve range can specify each acceleration and deceleration independently. Therefore, you can create an acceleration/deceleration profile that consists of linear acceleration and S-curve deceleration, or vice versa.
Acceleration/deceleration can be applied for an interpolation operation. (Only when both acceleration and deceleration are assigned the same pattern.)

♦ Interpolation operation
Feeding with linear interpolation and circular interpolation are both possible.

♦ Speed override
The feed speed can be changed in the middle of any feed operation.

♦ Overriding target position 1) and 2)
1) The target position (feed amount) can be changed while feeding in the positioning mode.
If the current position exceeds the newly entered position, the motor will decelerate, stop (immediate stop when already feeding at a low speed), and then feed in the reverse direction.
2) Starts operation the same as in the continuous mode and, when it receives an external signal, it will stop after the specified number of pulses.

♦ FH correction function (prevents triangle driving)
If the LSI will output a small number of pulses while in the positioning mode, it will automatically lower the maximum speed to prevent a triangle profile of acceleration and deceleration curves.

♦ Look ahead function
The next two sets of data (feed amount, initial speed, feed speed, acceleration rate, deceleration rate, speed magnification rate, ramping-down point, operation mode, center of circular interpolation, S-curve range on an acceleration, S-curve range on a deceleration, number of steps for circular interpolation) can be written while executing the current data. The next set of data, and other sets of data, can be written in advance of their execution for checking by the comparator.
When the current operation is complete, the system will immediately execute the next operation.

♦ A variety of counter circuits
The following four counters are available separately for each axis.

Counter	Use or purpose	Counter Input/Output
COUNTER1	28-bit counter for control of the command position	Outputs pulses
COUNTER2	28-bit counter for mechanical position control (Can be used as general-purpose counter)	EA/EB input Outputs pulses PA/PB input
COUNTER3	16-bit counter for controlling the deviation between the command position and the machine's current position	Outputs pulses and EA/EB input Outputs pulses and PA/PB input EA/EB input and PA/PB input
COUNTER4	28-bit counter used to output synchronous signals (Can be used as general-purpose counter)	Outputs pulses EA/EB input PA/PB input 1/2 of reference clock

All counters can be reset by writing a command or by providing a CLR signal.
They can also be latched by writing a command, or by providing an LTC or ORG signal.

♦ Comparator
There are five comparator circuits for each axis. They can be used to compare between target values and internal counter values.
The counter to compare can be selected from COUNTER1 (command position counter), COUNTER2 (mechanical position counter), COUNTER3 (deflection counter), and COUNTER4 (a general-purpose counter).
Comparators 1 and 2 can also be used as software limits (+SL, -SL).

♦ Software limit function
You can set software limits using two of the comparator's circuits.
When the mechanical position approaches the software limit range, the LSI will instruct the motors to stop immediately or to stop by deceleration. After that these axes can only be moved in the direction opposite to their previous travel.

♦ Backlash correction function
The LSI has a backlash correction function. Each time the feed direction is changed, the LSI applies a backlash correction. However, the backlash correction cannot be applied while performing a circular interpolation.

♦ Synchronous signal output function
The LSI can output pulse signals for each specified rate interval.

♦ Simultaneous start function
Multiple axes controlled by the same LSI, or controlled by multiple sets of this LSI, can be started at the same time.

♦ Simultaneous stop function
Multiple axes controlled by the same LSI, or controlled by multiple sets of this LSI, can be stopped at the same time by a command, by an external signal, or by an error stop on any axis.

♦ Vibration restriction function
Specify a control constant in advance and add one pulse each for reverse and forward feed just before stopping. Using this function, vibration can be decreased while stopping.

♦ Manual pulsar input function
By applying manual pulse signals (PA/PB), you can rotate a motor directly.
The input signals can be 90˚ phase difference signals (1x, 2x, or 4x) or up and down signals.
The LSI will stop outputting pulses when a limit signal is received. However, you cannot rotate a motor in the opposite direction using these pulses.

♦ Direct input of operation switch
Positive and negative direction terminals (±DR) are provided to drive a motor with an external operation switch. These switches turn the motor forward (+) and backward (-).

♦ Out-of-step detection function
This LSI has a deflection counter which can be used to compare command pulses and encoder signals (EA/EB). It can be used to detect out-of-step operation and to confirm a position by using a comparator.

♦ Idling pulse output function
This function outputs a preset number of pulses at the self start frequency (FL) before a high-speed start acceleration operation.
When the initial speed is set higher during the acceleration, this function is effective in preventing out-of-step operation.

♦ Operation mode
The basic operations of this LSI are: continuous operation, positioning, zero return, linear interpolation, and circular interpolation. By setting the optional operation mode bits, you can use a variety of operations.
<Examples of the operation modes>
1) Start/stop by command.
2) Continuous operation and positioning operation using PA/PB inputs (manual pulsar).
3) Operate for specified distances or in continuous operation using +DR/-DR signals (drive switch).
4) Zero return operation.
5) Positioning operation using commands.
6) Hardware start of the positioning operation using $\overline{\text{CSTA}}$ or $\overline{\text{STA}}$ input.
7) Change the target position after turning ON the PCS. (Target position override 2)

♦ Variety of zero return sequences (Homing)
The following patterns can be used.
1) Feeds at low speed and stops when the ORG signal is turned ON
2) Feeds at low speed and stops when an EZ signal is received (after the ORG signal is turned ON).
3) Feeds at low speed, reverses when the ORG signal is turned ON, and stops when an EZ signal is received.
4) Feeds at low speed and stops when the EL signal is turned ON. (Normal stop)
5) Feeds at low speed, reverses when the EL signal is turned ON, and stops when an EZ signal is received.
6) Feeds at high speed, decelerates when the SD signal is turned ON, and stops when the ORG signal is turned ON.
7) Feeds at high speed, decelerates when the ORG signal is turned ON, and stops when an EZ signal is received.
8) Feeds at high speed, decelerates and stops after the ORG signal is turned ON. Then, it reverse feeds and stops when an EZ signal is received.
9) Feeds at high speed, decelerates and stops by memorizing the position when the ORG signal is turned ON, and stops at the memorized position.
10) Feeds at high speed, decelerates to the position stored in memory when an EZ signal is received after the ORG signal is turned ON. Then, returns to the memorized position if an overrun occurs.
11) Feeds at high speed, reverses after a deceleration stop triggered by the EL signal, and stops when an EZ signal is received.

♦ Mechanical input signals
The following four signals can be input for each axis.
1) +EL: When this signal is turned ON, while feeding in the positive (+) direction, movement on this axis stops immediately (with deceleration). When this signal is ON, no further movement occurs on the axis in the positive (+) direction. (The motor can be rotated in the negative (-) direction.)
2) –EL: Functions the same as the +EL signal except that it works in the negative (-) direction.
3) SD: This signal can be used as a deceleration signal or a deceleration stop signal, according to the software setting. When this is used as a deceleration signal, and when this signal is turned ON during a high speed feed operation, the motor on this axis will decelerate to the FL speed. If this signal is ON and movement on the axis is started, the motor on this axis will run at the FL low speed. When this signal is used as a deceleration stop signal, and when this signal is turned ON during a high speed feed operation, the motor on this axis will decelerate to the FL speed and then stop.
4) ORG: Input signal for a zero return operation.

The input logic of the +EL and –EL signals can be changed with the ELL terminal.
The input logic of the SD and ORG signals can be changed using software.

♦ Digital servomotor I/F

The following three signals can be used as an interface for each axis

1) INP: Input positioning complete signal that is output by a servomotor driver.
2) ERC: Output deflection counter clear signal to a servomotor driver.
3) ALM: Regardless of the direction of operation, when this signal is ON, movement on this axis stops immediately (deceleration stop). When this signal is ON, no movement can occur on this axis.

The input logic of the INP, ERC, and ALM signals can be changed using software.
The ERC signal is a pulsed output. The pulse length can be set. (12 µs to 104 ms. A level output is also available.)

♦ Output pulse specifications

Output pulses can be set to a common pulse or 2-pulse mode. The output logic can also be selected.

♦ Emergency stop signal ($\overline{CEMG}$) input

When this signal is turned ON, movement on both axes stops immediately. While this signal is ON, no movement is allowed on either axes.

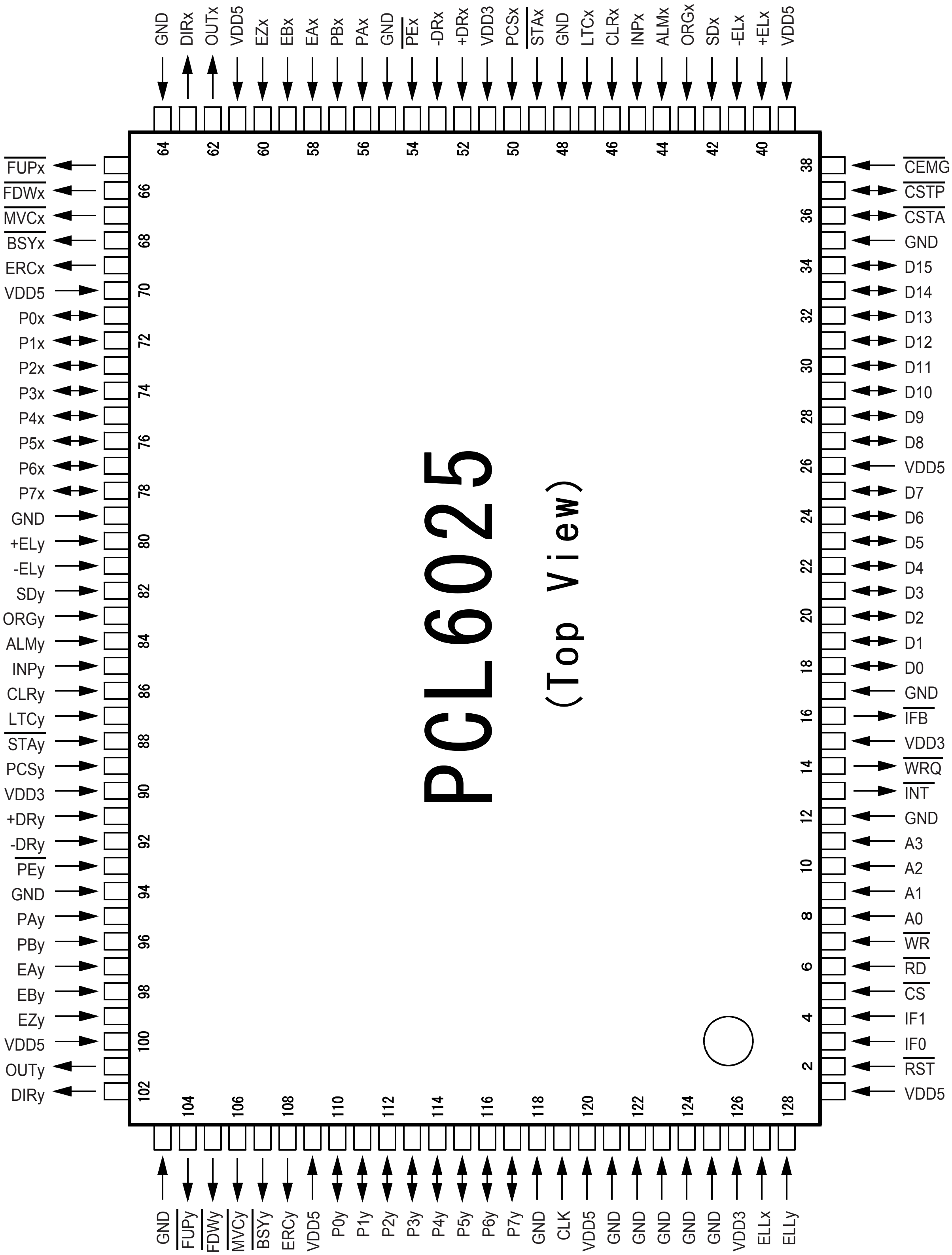
♦ Interrupt signal output

An $\overline{INT}$ signal (interrupt request) can be output for many reasons.
The $\overline{INT}$ terminal output signal can use ORed logic for lots of conditions on each axis.
(When more than one 6025 LSI is used, wired OR connections are not possible.)

2. Specifications

Item	Description
Number of axes	Two axes (X and Y)
Reference clock	Standard: 19.6608 MHz (Max. 20 MHz)
Positioning control range	-134,217,728 to +134,217,727 (28-bit)
Ramping-down point setting range	0 to 16,777,215 (24-bit)
Number of registers used for setting speeds	Three for each axis (FL, FH, and FA (speed correction))
Speed setting step range	1 to 65,535 (16-bits)
Speed magnification range	Multiply by 0.1 to 100 Multiply by 0.1 = 0.1 to 6,553.5 pps Multiply by 1 = 1 to 65,535 pps Multiply by 100 = 100 to 6,553,500 pps (When the reference clock is 19.6608 MHz)
Acceleration/deceleration characteristics	Selectable acceleration/deceleration pattern for both increasing and decreasing speed separately, using Linear and S-curve acceleration/deceleration.
Acceleration rate setting range	1 to 65,535 (16-bit)
Deceleration rate setting range	1 to 65,535 (16-bit)
Ramping-down point automatic setting	While in single-axis operation: Automatic setting within the range of (deceleration time) < (acceleration time x 2) Two-axis interpolation operation: Automatic setting when the same pattern is used for acceleration and deceleration.
FH correction function (prevents triangle driving)	Prevents the triangle shaped pulse driving that may occur with short positioning operations. Triangle curve occurs when the feed distance is very short and the motor has to start decelerating before it completes its acceleration pattern. This may adversely affect the mechanical system. In order to prevent this triangle shaped operation, this function automatically lowers the top speed so that the triangular shaped curve become trapezoids.
Manual operation input	Manual pulsar input, pushbutton switch input
Counter	COUNTER1: Command position counter (28-bit) COUNTER2: Mechanical position counter (28-bit) COUNTER3: Deflection counter (16-bit) COUNTER4: General-purpose counter (28-bit)
Comparators	28-bits x 5 circuits
Interpolation functions	Linear interpolation, circular interpolation
Operating temperature range	-40 to +85°C
Power supply	Power source for internal control: 3.3 V±10% Power source for I/O: +5V±10%
Package	128-pin QFP

3. Terminal Assignment Diagram



4. Terminal Pin Numbers, Names and Functions

<table>
<tr><th>Signal name</th><th>Terminal No.</th><th>Input/ output</th><th>Logic</th><th>Description</th></tr>
<tr><td>GND</td><td>12, 17, 35, 48, 55, 64, 79, 94, 103, 118, 121, 122, 123, 124, 125</td><td>Power source</td><td></td><td>Supply a ground.
Make sure to connect all of these terminals.</td></tr>
<tr><td>VDD5</td><td>1, 26, 39, 61, 70, 100, 109, 120</td><td>Power source</td><td></td><td>Power source for the interface.
Supply +5 VDC.
The allowable power supply range is +5 VDC ±10%.
Make sure to connect all of these terminals.</td></tr>
<tr><td>VDD3</td><td>15, 51, 90, 126</td><td>Power source</td><td></td><td>Power source for the internal circuits.
Supply +3.3 VDC.
The allowable power supply range is +3.3 VDC ±10%.
Make sure to connect all of these terminals.</td></tr>
<tr><td>$\overline{RST}$</td><td>2</td><td>Input</td><td>Negative</td><td>Input reset signal.
Make sure to set this signal LOW after turning ON the power and before starting operation. Input and holding $\overline{RST}$ low for at least 10 cycles of the reference clock.
For details about the chip's status after a reset, see section 11-1, "Reset", in this manual.</td></tr>
<tr><td>CLK</td><td>119</td><td>Input</td><td></td><td>Input a CMOS level reference clock signal.
(All other terminals should have TTL level inputs.)
The reference clock frequency is 19.6608 MHz. The LSI creates output pulses based on the clock input on this terminal.</td></tr>
<tr><td>IF0
IF1</td><td>3
4</td><td>Input</td><td></td><td>Enter the CPU-I/F mode
<table>
<tr><th rowspan="2">IF1</th><th rowspan="2">IF0</th><th rowspan="2">CPU example</th><th colspan="4">CPU signal connected to the terminal</th></tr>
<tr><th>$\overline{RD}$</th><th>$\overline{WR}$</th><th>A0</th><th>$\overline{WRQ}$</th></tr>
<tr><td>L</td><td>L</td><td>68000</td><td>+5 V</td><td>R/$\overline{W}$</td><td>$\overline{LDS}$</td><td>$\overline{DTACK}$</td></tr>
<tr><td>L</td><td>H</td><td>H8</td><td>$\overline{RD}$</td><td>$\overline{HWR}$</td><td>(GND)</td><td>$\overline{WAIT}$</td></tr>
<tr><td>H</td><td>L</td><td>8086</td><td>$\overline{RD}$</td><td>$\overline{WR}$</td><td>(GND)</td><td>READY</td></tr>
<tr><td>H</td><td>H</td><td>Z80</td><td>$\overline{RD}$</td><td>$\overline{WR}$</td><td>A0</td><td>$\overline{WAIT}$</td></tr>
</table></td></tr>
<tr><td>$\overline{CS}$</td><td>5</td><td>Input</td><td>Negative</td><td>When the signal level on this terminal is LOW, the $\overline{RD}$ and $\overline{WR}$ terminals will be valid.</td></tr>
<tr><td>$\overline{RD}$
$\overline{WR}$</td><td>6
7</td><td>Input</td><td>Negative</td><td>Connect the I/F signals to the CPU. The $\overline{RD}$ and $\overline{WR}$ terminals are valid when $\overline{CS}$ terminal is LOW.</td></tr>
<tr><td>A0 to A3</td><td>8 to 11</td><td>Input</td><td>Positive</td><td>Address control signals</td></tr>
<tr><td>$\overline{INT}$</td><td>13</td><td>Input</td><td>Negative</td><td>Outputs an interrupt request signal (IRQ) to an external CPU.
After this terminal is turned ON, the signal will return to OFF when a RESET (error interrupt cause) or RIST (event interrupt cause) signal is received. The output status can be checked with an MSTSW (main status) command signal.
The $\overline{INT}$ signal can be masked.
When more than one 6025 LSI is used, a wired OR connection between $\overline{INT}$ terminals is not allowed.</td></tr>
</table>

Signal name	Terminal No.	Input/ output	Logic	Description
$\overline{\text{WRQ}}$	14	Output	Negative	Outputs a wait request signal to cause a CPU to wait. The 6025 LSI needs 4 reference clock cycles to process each command. If the $\overline{\text{WRQ}}$ signal is not used, make sure that an external CPU does not access this LSI during this interval.
$\overline{\text{IFB}}$	16	Output	Negative	Signal used to indicate that the LSI is processing commands. Use this signal to make connections with a CPU that does not have a wait control input terminal. When the LSI receives a write command from a CPU, this signal will go LOW. When the LSI finishes processing, this signal will go HIGH. The LSI makes sure that this terminal is HIGH and then proceeds to the next step.
D0 to D7	18 to 25	Input/ Output	Positive	Bi-directional data bus. When connecting a 16-bit data bus, connect the lower 8 signal lines here.
D8 to D15	27 to 34	Input/ Output	Positive	Bi-directional data bus. When connecting a 16-bit data bus, connect the upper 8 signal lines here. A Z80-I/F (IF1 = H, IF0 = H) is used. Provide a pull up resistor (5k to 10 K-ohms) on VDD5. (One resistor can be used for all 8 lines.)
$\overline{\text{CSTA}}$	36	Input/ Output*	Negative	Input/Output terminal for simultaneous start. When more than one 6025 LSI is used and you want to start them simultaneously, connect this terminal on each LSI. The terminal status can be checked using an RSTS command signal (extension status).
$\overline{\text{STAx}}$ $\overline{\text{STAy}}$	49 88	Input	Negative	Input terminal for an external start signal. Although these terminals have the same function as $\overline{\text{CSTA}}$, these terminals can provide a separate input for each axis.
$\overline{\text{CSTP}}$	37	Input/ Output*	Negative	Input/Output terminal for a simultaneous stop. When more than one 6025 LSI is used and you want to stop them simultaneously, connect this terminal on each LSI. The terminal status can be checked using an RSTS command signal (extension status).
$\overline{\text{CEMG}}$	38	Input U	Negative	Input for an emergency stop. When this terminal signal goes LOW while in operation, both axes stop immediately. As long as this terminal is LOW, neither axis can be started.
ELLx ELLy	127 128	Input U		Specify the input logic for the ±EL signal. LOW: The input logic on ±EL is positive. HIGH: The input logic on ±EL is negative.
+ELx +ELy	40 80	Input U	Negative%	Input end limit signal in the positive (+) direction. When this signal is ON while in feeding in the positive (+) direction, the motor on that axis will stop immediately or will decelerate and stop. Specify the input logic using the ELL terminal. The terminal status can be checked using an SSTSW command signal (sub status).
-ELx -ELy	41 81	Input U	Negative%	Input end limit signal in the negative (-) direction. When this signal is ON while feeding in negative (-) direction, the motor on that axis will stop immediately, or will decelerate and stop. Specify the input logic using the ELL terminal. The terminal status can be checked using an SSTSW command signal (sub status).
SDx SDy	42 82	Input U	Negative#	Input deceleration signal. Selects the input method: LEVEL or LATCHED inputs. The input logic can be selected using software. The terminal status can be checked using an SSTSW command signal (sub status). A deceleration stop is also possible.

Signal name	Terminal No.	Input/ output	Logic	Description
ORGx ORGy	43 83	Input U	Negative #	Input zero position signal. Used for zero return and other operations. (Edge detection.) The input logic can be selected using software. The terminal status can be checked using an SSTSW command signal (sub status).
ALMx ALMy	44 84	Input U	Negative #	Input alarm signal. When this signal is ON, the motor on that axis stops immediately, or will decelerate and stop. The input logic can be selected using software. The terminal status can be checked using an SSTSW command signal (sub status).
OUTx OUTy	62 101	Output	Negative #	Outputs command pulses for controlling a motor, or outputs direction signals. When Common Pulse mode is selected: Outputs a direction signal. When 2-pulse mode is selected: Outputs pulses in the negative (-) direction. The output logic can be changed using software.
DIRx DIRy	63 102	Output	Negative #	Output command pulses for controlling a motor, or outputs direction signal. When Common Pulse mode is selected: Outputs a direction signal. When 2-pulse mode is selected: Output pulses in the negative (-) direction. The output logic can be changed using software
EAx EBx EAy EBy	58 59 97 98	Input U		Input encoder signals for controlling the mechanical position using an encoder. Input a 90˚ phase difference signal (1x, 2x, 4x) or input positive (+) pulses on EA and negative (-) pulses on EB. When inputting 90˚ phase difference signals, if the EA signal phase is ahead of the EB signal, the LSI will count pulses. The counting direction can be changed using software.
EZx EZy	60 99	Input U	Negative #	Input a marker signal (this signal is output once for each turn of the encoder) when using the marker signal in zero return mode. Use of the EZ signal improves zero return precision. The input logic can be changed using software. The terminal status can be checked using an RSTS command signal (extension status).
$\overline{\text{PEx}}$ $\overline{\text{PEy}}$	54 93	Input U	Negative	Setting these terminals LOW enables PA/PB and +DR/-DR input. By inputting an axis change switch signal, one manual pulsar can be used alternately for two axes.
PAx PBx PAy PBy	56 57 95 96	Input U		Input for receiving external drive pulses, such as manual pulsar. You can input 90˚ phase difference signals (1x, 2x, 4x) or positive (+) pulses (on PA) and negative (-) pulses (on PB). When 90˚ phase difference signals are used, if the signal phase of PA is ahead of the PB signal, the LSI will count up. The counting direction can be changed using software.
+DRx -DRx +Dry -DRy	52 53 91 92	Input U	Negative #	Input for manual operation using an external switch. Specifying the feed length, low-speed continuous feed, and high-speed continuous feed are possible. The input logic can be changed using software. The terminal status can be checked using an RSTS command signal (extension status).
PCSx PCSy	50 89	Input U	Negative #	Using this input signal, a motor can be changed from continuous operation to positioning operation. (Override 2 of the target position.) The input logic can be changed using software. The terminal status can be checked using an RSTS command signal (extension status).

Signal name	Terminal No.	Input/ output	Logic	Description
INPx INPy	45 85	Input U	Negative #	Input servo driver positioning complete signal (in-position signal). Input logic can be changed using software. The terminal status can be checked using an RSTS command signal (extension status).
CLRx CLRy	46 86	Input U	Negative #	Reset a specified counter from COUNTER1 to 4. The input logic can be changed using software. The terminal status can be checked using an RSTS command signal (extension status).
LTCx LTCy	47 87	Input U	Negative #	Latch counter value of specified counters (available on more than one) from COUNTER1 to 4. The input logic can be changed using software. The terminal status can be checked using an RSTS command signal (extension status).
$\overline{FUPx}$ $\overline{FUPy}$	65 104	Output	Negative	Outputs a LOW signal while accelerating.
$\overline{FDWx}$ $\overline{FDWy}$	66 105	Output	Negative	Outputs a LOW signal while decelerating.
$\overline{MVCx}$ $\overline{MVCy}$	67 106	Output	Negative	Outputs a LOW signal while performing a low speed feed.
$\overline{BSYx}$ $\overline{BSYy}$	68 107	Output	Negative	Outputs a LOW signal while feeding.
ERCx ERCy	69 108	Output	Negative #	Outputs a deflection counter clear signal to a servo driver as a pulse. The output logic and pulse width can be changed using software. A LEVEL signal output is also available. The terminal status can be checked using an RSTS command signal (extension status).
P0x/FUPx P0y/FUPy	71 110	Input/ Output*	Positive	Common terminal for general purpose I/O and FUP. (See Note 5.) As a general purpose I/O terminal, three possibilities can be specified: input terminal, output terminal, and one shot pulse output terminal. As an FUP terminal, it outputs a LOW signal while accelerating. The usage, output logic of the FUP and one shot pulse parameters can be changed using software.
P1x/FDWx P1y/FDWy	72 111	Input/ Output*	Positive	Common terminal for general purpose I/O and FDW. (See Note 5.) As a general purpose I/O terminal, three possibilities can be specified: input terminal, output terminal, and one shot pulse output terminal. As an FDW terminal, it outputs a LOW signal while accelerating. The usage, output logic of the FDW and one shot pulse parameters can be changed using software.
P2x/MVCx P2y/MVCy	73 112	Input/ Output*	Positive	Common terminal for general purpose I/O and MVC. (See Note 5.) When used for as an MVC terminal, it outputs a signal while performing a low speed feed. The usage, and output logic of the MVC can be changed using software.
P3x/CP1x/ (+SLx) P3y/CP1y/ (+SLy)	74 113	Input/ Output*	Positive	Common terminal for general purpose I/O and CP1 (+SL). (See Note 5.) When used as a CP1 (+SL) terminal, it outputs a signal while establishing the conditions (within + SL) of comparator 1. The output logic of CP1 (+SL) as well the selection of input or output functions can be changed using software.
P4x/CP2x/ (-SLx) P4y/CP2y/ (-SLy)	75 114	Input/ Output*	Positive	Common terminal for general purpose I/O and CP2 (-SL). (See Note 5.) When used as a CP2 (-SL) terminal, it outputs a signal while establishing the conditions (within - SL) of comparator 2. The output logic of CP2 (-SL) as well as the selection of input or output functions can be changed using software.

Signal name	Terminal No.	Input/ output	Logic	Description
P5x/CP3x P5y/CP3y	76 115	Input/ Output*	Positive	Common terminal for general purpose I/O and CP3. (See Note 5.) When used as a CP3 terminal, it outputs a signal while establishing the conditions of comparator 3. The output logic of CP3 as well as the selection of input or output functions can be changed using software.
P6x/CP4x P6y/CP4y	77 116	Input/ Output*	Positive	Common terminal for general purpose I/O and CP4. (See Note 5.) When used as a CP4 terminal, it outputs a signal while establishing the conditions of comparator 4. The output logic of CP4 as well as the selection of input or output functions can be changed using software.
P7x/CP5x P7y/CP5y	78 117	Input/ Output*	Positive	Common terminal for general purpose I/O and CP5. (See Note 5.) When used as a CP5 terminal, it outputs a signal while establishing the conditions of comparator 5. The output logic of CP5 as well as the selection of input or output functions can be changed using software.

Note 1: "Input U" refers to an input with a pull up resistor. The internal pull up resistance (30 K to 144 K-ohms) is only used to keep a terminal from floating. If you want to use the LSI with an open collector system, an external pull up resistor (5k to 10 K-ohms) is required.
As a noise prevention measure, pull up unused terminals to VDD5 using an external resistor (5 k to 10 K-ohms), or to connect them directly to VDD5.

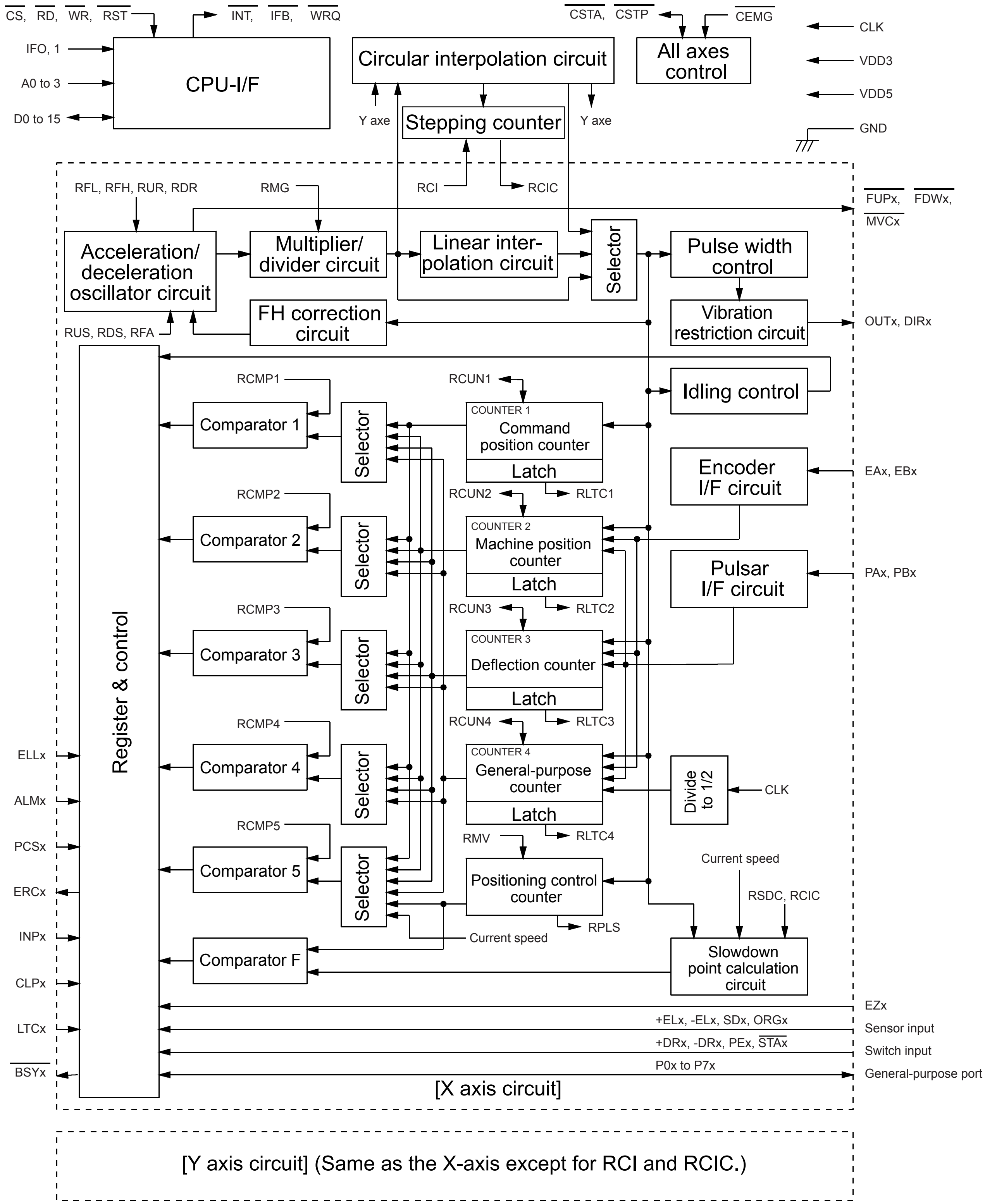
Note 2: "Input/Output *" refers to a terminal with a pull up resistor. The internal pull up resistor (30 K to 144 K-ohms) is only used to keep a terminal from floating. If it is connected in a wired OR circuit, an external pull up resistor (5 k to 10 K-ohms) is required.
As a noise prevention measure, pull up unused terminals to VDD5 using an external resistor (5 k to 10 K-ohms).

Note 3: If an output terminal is not being used, leave it open.

Note 4: "Positive" refers to positive logic. "Negative" refers to negative logic. "#" means that the logic can be changed using software. "%" means that the logic can be changed by the setting on another terminal. The logic shown refers only to the initial status of the terminal. The DIR terminal is initially in a 2-pulse mode.

Note 5: When P0 to P7 are set up as output terminals, they can be controlled simultaneously as 8 bits or one bit at a time using output bit control commands, depending on what is written to the output port (OTPB).
When P0 and P1 are set up as one shot pulse output terminals, they will output a one shot signal (T = Approx. 26 ms) using the output bit control command.

5. Block Diagram



6. CPU Interface

6-1. Setting up connections to a CPU

This LSI can be connected to four types of CPUs by changing the hardware settings.
Use the IF0 and IF1 terminals to change the settings and connect the CPU signal lines as follows.

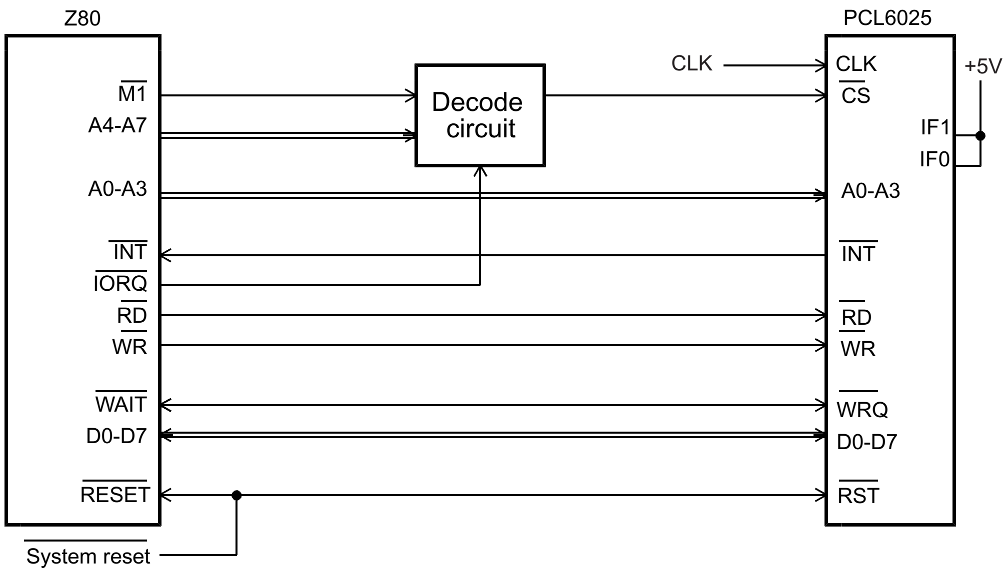
Setting status		CPU type	CPU signal to connect to the 6025 terminals			
IF1	IF0		$\overline{RD}$ terminal	$\overline{WR}$ terminal	A0 terminal	$\overline{WRQ}$ terminal
L	L	68000	+5V	R/$\overline{W}$	$\overline{LDS}$	$\overline{DTACK}$
L	H	H8	$\overline{RD}$	$\overline{HWR}$	(GND)	$\overline{WAIT}$
H	L	8086	$\overline{RD}$	$\overline{WR}$	(GND)	READY
H	H	Z80	$\overline{RD}$	$\overline{WR}$	A0	$\overline{WAIT}$

6-2. Precautions for designing hardware

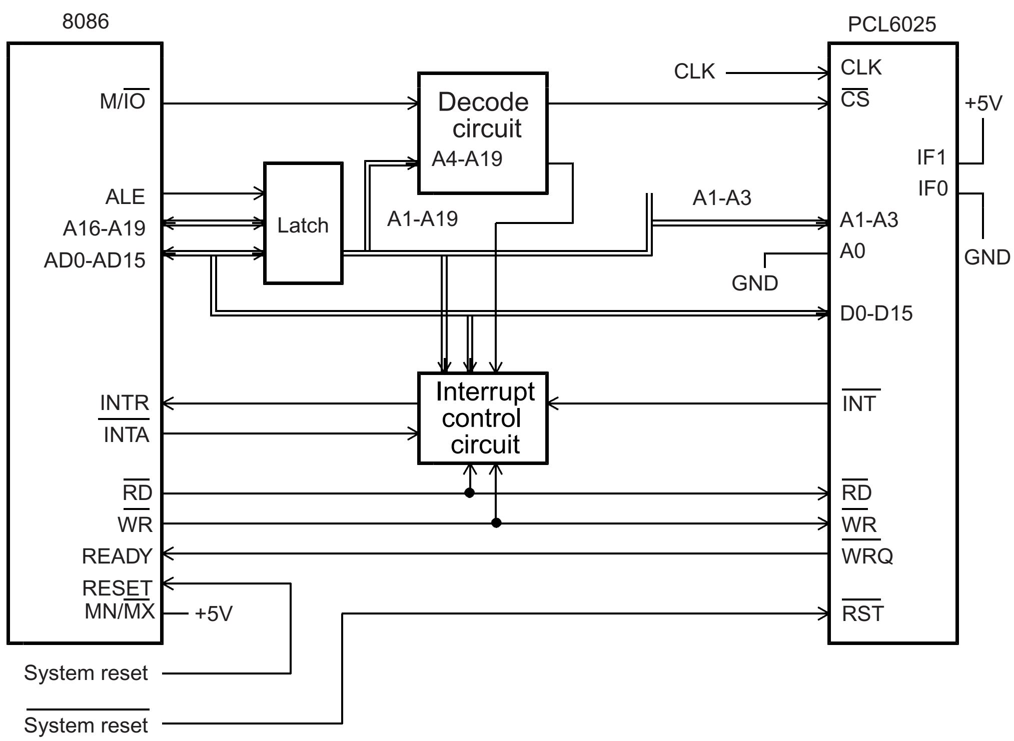
- Apply a CMOS level clock to the CLK terminal.
- To reset the LSI, hold the $\overline{RST}$ signal LOW, and input the CLK signal for at least 10 clock cycles.
- Connect unused P0 to P7 terminals to VDD5 through a pull up resistor (5 k to 10 K-ohms).
- When connecting a CPU with an 8-bit bus, pull up terminals D8 to D15 to VDD5 using an external resistor (5 k to 10 k-ohms). (A single resistor can be used for all 8 lines.)
- Use the ELL terminal to change the ±EL signal input logic.

6-3. CPU interface circuit block diagram

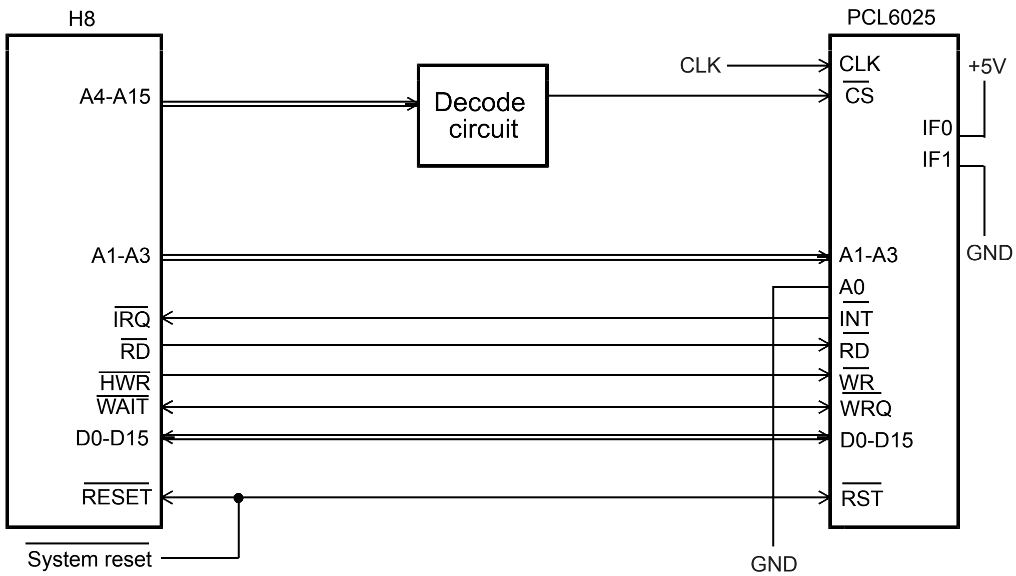
1) Z80 interface



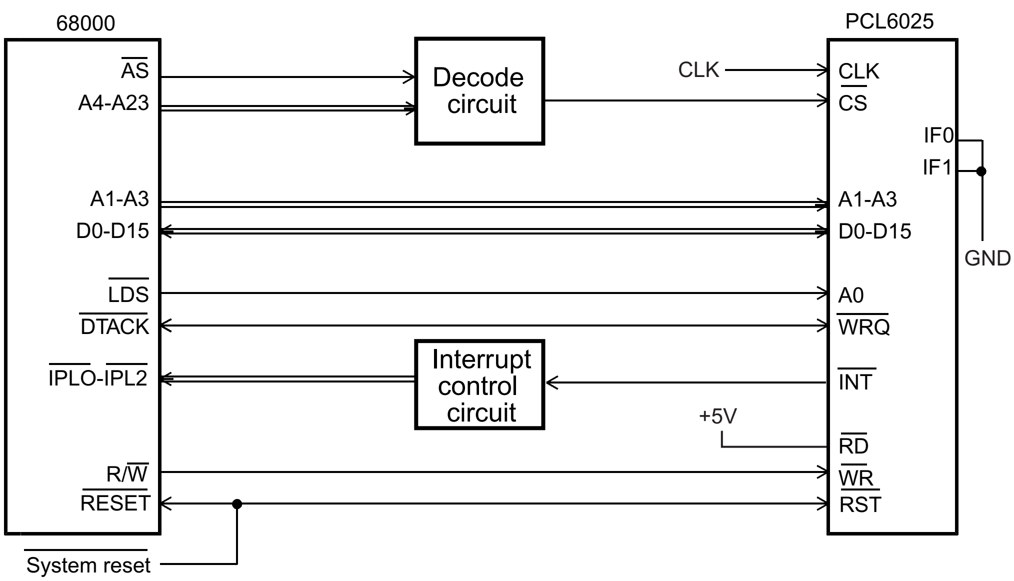
2) 8086 interface



3) H8 interface



4) 68000 interface



6-4. Address map

6-4-1. Axis arrangement map

In this LSI, the control address range for each axis is independent. It is selected by using address input terminal A3, as shown below.

A3	Detail
0	X axis control address range
1	Y axis control address range

6-4-2. Internal map of each axis

The internal map of each axis is defined by A0, A1, and A2 address line inputs.

<When used with the Z80 I/F>

1) Write cycle

A0 to A2	Address signal	Processing detail
000	COMB0	Write a control command
001	COMB1	Assign the axis (specify a control command for execution)
010	OTPB	Change the status of the general-purpose output port (only bits assigned as outputs are effective)
011		(Invalid)
100	BUFB0	Write to the input/output buffer (bits 0 to 7)
101	BUFB1	Write to the input/output buffer (bits 8 to 15)
110	BUFB2	Write to the input/output buffer (bits 16 to 23)
111	BUFB3	Write to the input/output buffer (bits 24 to 31)

2) Read cycle

A0 to A2	Address signal	Processing detail
000	MSTSB0	Read the main status (bits 0 to 7)
001	MSTSB1	Read the main status (bits 8 to 15)
010	IOPB	Read the general-purpose input/output port
011	SSTSB	Read the sub status
100	BUFB0	Read from the input/output buffer (bits 0 to 7)
101	BUFB1	Read from the input/output buffer (bits 8 to 15)
110	BUFB2	Read from the input/output buffer (bits 16 to 23)
111	BUFB3	Read from the input/output buffer (bits 24 to 31)

<When used with the 8086 I/F>

1) Write cycle

A1 to A2	Address signal	Processing detail
00	COMW	Write the axis assignment and control command
01	OTPW	Change the status of the general-purpose output port (only bits assigned as outputs are effective)
10	BUFW0	Write to the input/output buffer (bits 0 to 15)
11	BUFW1	Write to the input/output buffer (bits 16 to 31)

2) Readout cycle

A1 to A2	Address signal	Processing detail
00	MSTSW	Read the main status (bits 0 to 15)
01	SSTSW	Read the sub status or general-purpose input/output port
10	BUFW0	Read from the input/output buffer (bits 0 to 15)
11	BUFW1	Read from the input/output buffer (bits 16 to 31)

<When used with the H8 or 68000 I/F>

1) Write cycle

A1 to A2	Address signal	Processing detail
11	COMW	Write the axis assignment and control command
10	OTPW	Change the status of the general-purpose output port (only bits assigned as outputs are effective)
01	BUFW0	Write to the input/output buffer (bits 0 to 15)
00	BUFW1	Write to the input/output buffer (bits 16 to 31)

2) Readout cycle

A1 to A2	Address signal	Processing detail
11	MSTSW	Read the main status (bits 0 to 15)
10	SSTSW	Read the sub status or general-purpose input/output port
01	BUFW0	Read from the input/output buffer (bits 0 to 15)
00	BUFW1	Read from the input/output buffer (bits 16 to 31)

6-5. Description of the map details

6-5-1. Write the command code and axis selection (COMW, COMB)

Write the commands for reading and writing to registers and the start and stop control commands for each axis.

COMB0: Set the command code. For details, see 7. "Command (Operation and Control commands)."

SELx, y: Select an axis for executing the command. If all of the bits are 0, only this axis (selected by A3) is selected. To write the same command to more than one axis, set the bits of the selected axes to 1.
When you write to a register, the details of the input/output buffer are written into the register for each axis.
When you read from a register, the details in the register are written into the input/output buffer for each axis.

COMW

COMB1								COMB0							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	SELy	SELx								

6-5-2. Write to an output port (OTPW, OTPB)

Specify output terminal status from the general purpose I/O terminals P0 to P7.
Bits corresponding to terminals not set as outputs are ignored.
When writing a word, the upper 8 bits are ignored. However, they should be set to 0 for future compatibility.

OTP0 to 7: Specify the status of output terminals P7n to P0n (n = xy).
A HIGH is output when the bit is set to 1.

OTPW

								OTPB							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	OTP7	OTP6	OTP5	OTP4	OTP3	OTP2	OTP1	OTP0

6-5-3. Write/read the input/output buffer (BUFW, BUFB)

When you want to write data into a register, after placing the data in the input/output buffer, write a "register write command" into COMB0. The data in the input/output buffer will be copied into the register.
When you want to write data into the input/output buffer, write a "register read command" into COMB0. The data in the register will be copied to the input/output buffer. Then you can read the data from the input/output buffer.
The order for writing and reading buffers BUFW0 to 1 (BUFB0 to 3) is not specified. The data written in the input/output buffer can be read at any time.

BUFW1																BUFW0															
BUFB3								BUFB2								BUFB1								BUFB0							
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

6-5-4. Reading the main status (MSTSW, MSTSB)

MSTSW

MSTSB1								MSTSB0							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPDF	SPRF	0	SCP5	SCP4	SCP3	SCP2	SCP1	SSC1	SSC0	SINT	SERR	SEND	0	SRUN	SSCM

Bit	Bit name	Details
0	SSCM	Set to 1 by writing a start command. Set to 0 when the operation is stopped.
1	SRUN	Set to 1 by the start pulse output. Set to 0 when the operation is stopped.
2	Not defined	(Always 0)
3	SEND	Set to 0 by writing start command. Set to 1 when the operation is stopped.
4	SERR	Set to 1 when an error interrupt occurs. Set to 0 by reading the RESET.
5	SINT	Set to 1 when an error interrupt occurs. Set to 0 by reading the RIST.
6 to 7	SSC0 to 1	Sequence number for execution or stopping.
8	SCP1	Set to 1 when the CMP1 comparison conditions are met.
9	SCP2	Set to 1 when the CMP2 comparison conditions are met.
10	SCP3	Set to 1 when the CMP3 comparison conditions are met.
11	SCP4	Set to 1 when the CMP4 comparison conditions are met.
12	SCP5	Set to 1 when the CMP5 comparison conditions are met.
13	Not defined	(Always 0)
14	SPRF	Set to 1 when the pre-register for the subsequent operation data is full.
15	SPDF	Set to 1 when the pre-register for comparator 5 is full.

6-5-5. Reading the sub status and input/output port. (SSTSW, SSTSB, IOPB)

SSTSW

SSTSB								IOPB							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSD	SORG	SMEL	SPEL	SALM	SFC	SFD	SFU	IOP7	IOP6	IOP5	IOP4	IOP3	IOP2	IOP1	IOP0

Bit	Bit name	Description
0 to 7	IOP0 to 7	Read the status of P0 to 7 (0: L level, 1: H level)
8	SFU	Set to 1 while accelerating.
9	SFD	Set to 1 while decelerating.
10	SFC	Set to 1 while feeding at low speed.
11	SALM	Set to 1 when the ALM input is ON.
12	SPEL	Set to 1 when the +EL input is ON.
13	SMEL	Set to 1 when the –EL input is ON.
14	SORG	Set to 1 when the ORG input is ON.
15	SSD	Set to 1 when the SD input is ON. (Latches the SD signal.) Set to 0 by the next start or a release the SD signal latch operation.

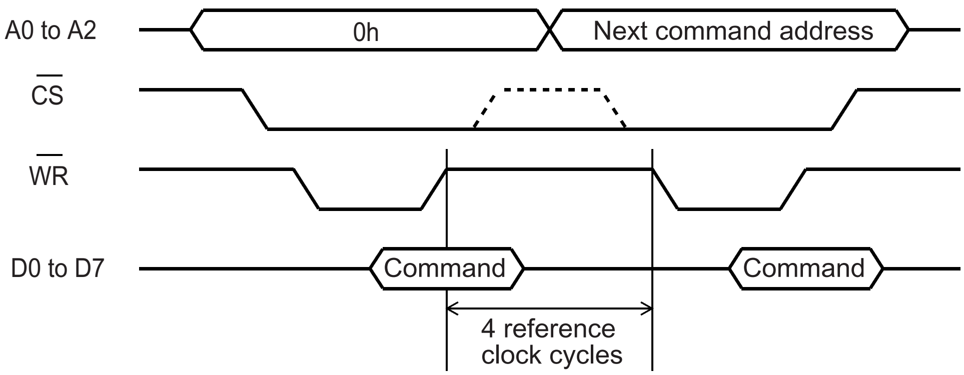
7. Commands (Operation and Control Commands)

7-1. Operation commands

After writing the axis assignment data to COMB1 (address 1 when a Z80 I/F is used), write the command to COMB0 (address 0 when a Z80 I/F is used), the LSI will start and stop, as well as change the speed of the output pulses. When an 8086, H8, or 68000 I/F is used, write 16-bit data, which combines the axis assignment and operation command data.

7-1-1. Procedure for writing an operation command (the axis assignment is omitted)

Write a command to COMB0 (address 0 when a Z80 I/F is used).
Before continuing on to the next command, wait at least 4 reference clock cycles (approx. 0.2 µs when the CLK is 19.6608 MHz). The $\overline{WRQ}$ terminal will output a wait request signal.



7-1-2. Start command

1) Start command

If this command is written while stopped, the motor will start rotating. If this command is written while the motor is operating, it is taken as the next start command.

COMB0	Symbol	Description
50h	STAFL	FL low speed start
51h	STAFH	FH low speed start
52h	STAD	High speed start 1 (FH low speed -> deceleration stop) Note. 1
53h	STAUD	High speed start 2 (Acceleration -> FH low speed -> Deceleration stop) Note. 1

Note 1: For details, see section 10-1, "Speed patterns."

2) Residual pulses start command

Write this command after the motor is stopped on the way of a positioning, it will continue movement for the number of pulses left in the deflection counter.

COMB0	Symbol	Description
54h	CNTFL	Residual pulses FL low speed start
55h	CNTFH	Residual pulses FH low speed start
56h	CNTD	Residual pulses high speed start (FH low speed -> Deceleration stop)
57h	CNTUD	Residual pulses high speed start (Acceleration -> FH low speed -> Deceleration stop)

3) Simultaneous start command

By setting the RMD register, the LSI will send an $\overline{STA}$ signal ($\overline{CSTA}$ or $\overline{STA}$) to the axis which is waiting.

COMB0	Symbol	Description
06h	CMSTA	Output one shot of the start pulse from the $\overline{CSTA}$ terminal.
2Ah	SPSTA	Only this axis will process the command, the same as when the $\overline{STA}$ signal is input.

7-1-3. Speed change command

Write this command while the motor is operating, the motor on that axis will change its feed speed. If this command is written while stopped it will be ignored.

COMB0	Symbol	Description
40h	FCHGL	Change to the FL speed immediately.
41h	FCHGH	Change to the FH speed immediately.
42h	FSCHL	Decelerate and change to the FL speed.
43h	FSCHH	Accelerate and change to the FH speed.

7-1-4. Stop command

1) Stop command

Write this command to stop feeding while operating.

COMB0	Symbol	Description
49h	STOP	Write this command while in operation to stop immediately.
4Ah	SDSTP	Write this command while feeding at FH low speed or high speed, the motor on that axis will decelerate to the FL low speed and stop. If this command is written while the axis is being fed at FL low speed, the motor on that axis will stop immediately.

2) Simultaneously stop command

Stop the motor on any axis whose $\overline{\text{CSTP}}$ input stop function has been enabled by setting the RMD register.

COMB0	Symbol	Description
07h	CMSTP	Outputs one shot of pulses from the $\overline{\text{CSTP}}$ terminal to stop movement on that axis.

3) Emergency stop command

Stops an axis in an emergency

COMB0	Symbol	Description
05h	CMEMG	Emergency stop (same as a $\overline{\text{CEMG}}$ signal input)

7-1-5. NOP (do nothing) command

COMB0	Symbol	Description
00h	NOP	This command does not affect the operation.

7-2. General-purpose output bit control commands

These commands control the individual bits of output terminals P0 to P7.
When the terminals are designated as outputs, the LSI will output signals from terminals P0 to P7. Commands which require terminals that have not been designated as outputs are ignored.
The write procedures are the same as for the Operation commands.
In addition to this command, by writing to a general-purpose output port (OTPB: Address 2 when a Z80 I/F is used), you can set 8 bits as a group. See section 7-5, "General-purpose output port control."

COMB0	Symbol	Description	COMB0	Symbol	Description
10h	P0RST	Make P0 LOW.	18h	P0SET	Make P0 HIGH.
11h	P1RST	Make P1 LOW.	19h	P1SET	Make P1 HIGH.
12h	P2RST	Make P2 LOW.	1Ah	P2SET	Make P2 HIGH.
13h	P3RST	Make P3 LOW.	1Bh	P3SET	Make P3 HIGH.
14h	P4RST	Make P4 LOW.	1Ch	P4SET	Make P4 HIGH.
15h	P5RST	Make P5 LOW.	1Dh	P5SET	Make P5 HIGH.
16h	P6RST	Make P6 LOW.	1Eh	P6SET	Make P6 HIGH.
17h	P7RST	Make P7 LOW.	1Fh	P7SET	Make P7 HIGH.

The P0 and P1 terminals can be set for one shot output (T = approx. 26 msec.) using the RENV2 (Environment setting 2) register, and the output logic can be selected.
To use them as one shot outputs, set the P0 terminal to P0M (bits 0 and 1) = 11, or, set the P1 terminal to P1M (bits 2 and 3) = 11. To change the output logic, set P0L (bit 16) on the P0 terminal and P1L (bit 17) on the P1 terminal.
In order to perform a one-shot output from the P0 and P1 terminals, a bit control command should be written. However, the command you need to write will vary, depending on the output logic selected. See the table below for the details.

Terminal	Logic setting	Bit control command	Terminal	Logic setting	Bit control command
P0	Negative logic (P0L = 0)	P0RST (10h)	P1	Negative logic (P1L = 0)	P1RST (11h)
	Positive logic (P0L = 1)	P0SET (18h)		Negative logic (P1L = 1)	P1SET (19h)

While set for one-shot output, the P0 and P1 terminals will not change when writing to the output port (OTPB: Address 2 when a Z80 I/F is used).

7-3. Control command

Set various controls, such as the reset counter.
The procedures for writing are the same as the operation commands.

7-3-1. Software reset command

Used to reset this LSI.

COMB0	Symbol	Description
04h	SRST	Software reset. (Same function as making the $\overline{RST}$ terminal LOW.)

7-3-2. Counter reset command

Reset counters to zero.

COMB0	Symbol	Description
20h	CUN1R	Reset COUNTER1 (command position).
21h	CUN2R	Reset COUNTER2 (mechanical position).
22h	CUN3R	Reset COUNTER3 (deflection counter).
23h	CUN4R	Reset COUNTER4 (general-purpose counter).

7-3-3. ERC output control command

Control the ERC signal using commands.

COMB0	Symbol	Description
24h	ERCOUT	Outputs the ERC signal.
25h	ERCRST	Resets the output when the ERC signal output is specified to a level type output.

7-3-4. Pre-register control command

Cancel the pre-register settings and transfer the pre-register data to a register.

COMB0	Symbol	Description
26h	PRICAN	Cancel the operation pre-register. Note 1, 2.
27h	PCPCAN	Cancel the RCMP5 operation pre-register (PRCP5).
2Bh	PRISHF	Shift the operation pre-register data. Note 3.
2Ch	PCPSHF	Shift the RCMP5 operation pre-register data.

Note 1: The Operation pre-registers refer to the RMV, RFL, RFH, RUR, RDR, RMG, RDP, RMD, RIP, RUS, RDS, and RCI registers, except for the pre-register for RCMP5, and the pre-registers for the start commands. For details about the pre-registers, see section 8-2, "Pre-register."

Note 2: This cancel command cancels the start commands for the 1st and 2nd pre-registers.

Note 3: The shift command transfers data in the following order: 1st pre-register data → register, 2nd pre-register data → 1st pre-register.

7-3-5. PCS input command

Entering this command has the same results as inputting a signal on the PCS terminal.

COMB0	Symbol	Description
28h	STAON	Alternative to a PCS terminal input.

7-3-6. LTCH input (counter latch) command

Entering this command has the same result as inputting a signal on the LTC terminal.

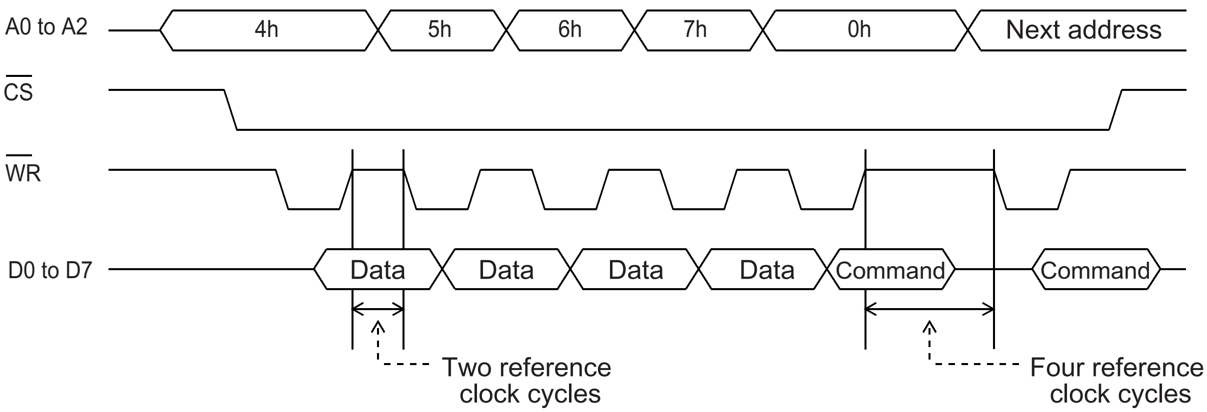
COMB0	Command symbol	Description
29h	LTCH	Alternative to an LTC (latch counter) terminal input.

7-4. Register control command

By writing a Register Control command to COMB0 (Address 0 when a Z80 I/F is used), the LSI can copy data between a register and the I/O buffer.
When using the I/O buffer while responding to an interrupt, some precautions are required, such as reading the I/O buffer contents before using it and returning it to its original value after use.

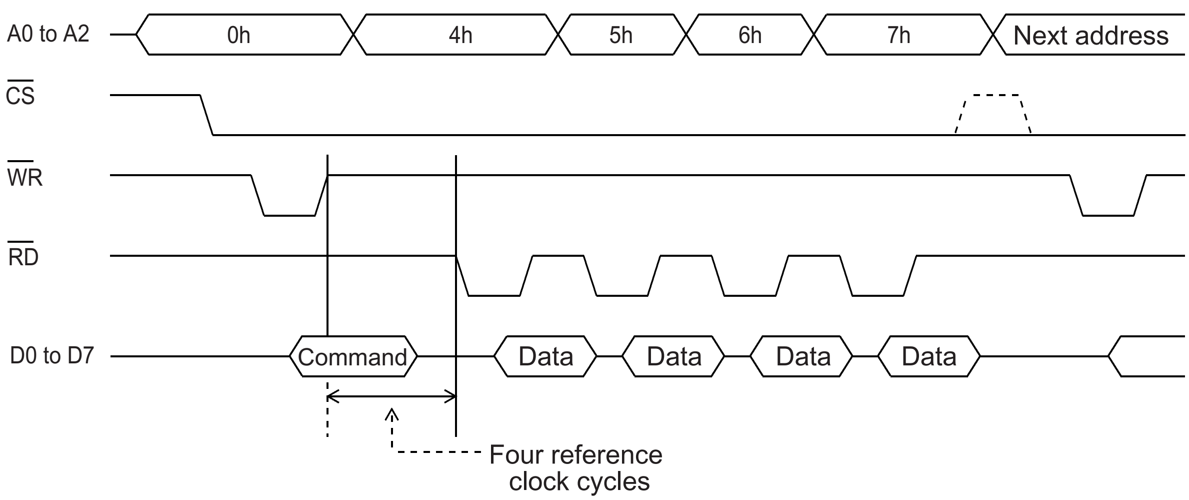
7-4-1. Procedure for writing data to a register (the axis assignment is omitted)

First, write the data that will be written to a register into the I/O buffer (addresses 4 to 7 when a Z80 I/F is used). The order in which the data is written does not matter. Then, write a "register write command" to COMB0 (address 0 when a Z80 I/F is used).
After writing one set of data, wait at least two cycles (approx. 0.1 µs when CLK = 19.6608 MHz) before writing the next set of data. Wait at least four cycles (approx. 0.2 µs when CLK = 19.6608 MHz) between writing one command and the next. The LSI outputs a wait request signal from the $\overline{WRQ}$ terminal.



7-4-2. Procedure for reading data from a register (the axis assignment is omitted)

First, write a "register read out command" to COMB0 (address 0 when a Z80 I/F is used).
Wait at least four reference clock cycles (approx. 0.2 µs when CLK = 19.6608 MHz) for the data to be copied to the I/O buffer. Read the data from the I/O buffer (addresses 4 to 7 when a Z80 I/F is used). The order for reading data from the I/O buffer does not matter.



7-4-3. Table of register control commands

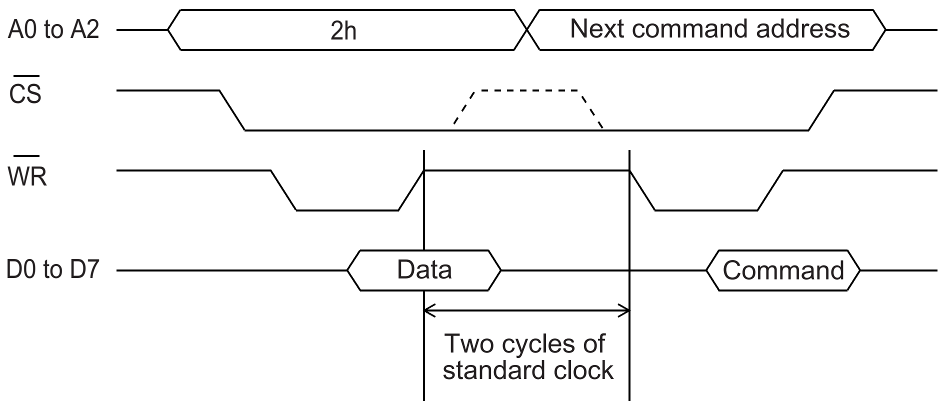
No.	Register	Detail	Read command		Write command		2nd pre-register	Read command		Write command	
			COMB0	Symbol	COMB0	Symbol		COMB0	Symbol	COMB0	Symbol
1	RMV	Feed amount, target position	D0h	RRMV	90h	WRMV	PRMV	C0h	RPRMV	80h	WPRMV
2	RFL	Initial speed	D1h	RRFL	91h	WRFL	PRFL	C1h	RPRFL	81h	WPRFL
3	RFH	Operation speed	D2h	RRFH	92h	WRFH	PRFH	C2h	RPRFH	82h	WPRFH
4	RUR	Acceleration rate	D3h	RRUR	93h	WRUR	PRUR	C3h	RPRUR	83h	WPRUR
5	RDR	Deceleration rate	D4h	RRDR	94h	WRDR	PRDR	C4h	RPRDR	84h	WPRDR
6	RMG	Speed magnification rate	D5h	RRMG	95h	WRMG	PRMG	C5h	RPRMG	85h	WPRMG
7	RDP	Ramping-down point	D6h	RRDP	96h	WRDP	PRDP	C6h	RPRDP	86h	WPRDP
8	RMD	Operation mode	D7h	RRMD	97h	WRMD	PRMD	C7h	RPRMD	87h	WPRMD
9	RIP	Circular interpolation center	D8h	RRIP	98h	WRIP	PRIP	C8h	RPRIP	88h	WPRIP
10	RUS	Acceleration S-curve range	D9h	RRUS	99h	WRUS	PRUS	C9h	RPRUS	89h	WPRUS
11	RDS	Deceleration S-curve range	DAh	RRDS	9Ah	WRDS	PRDS	CAh	RPRDS	8Ah	WPRDS
12	RFA	Feed amount correction speed	DBh	RRFA	9Bh	WRFA					
13	RENV1	Environment setting 1	DCh	RRENV1	9Ch	WRENV1					
14	RENV2	Environment setting 2	DDh	RRENV2	9Dh	WRENV2					
15	RENV3	Environment setting 3	DEh	RRENV3	9Eh	WRENV3					
16	RENV4	Environment setting 4	DFh	RRENV4	9Fh	WRENV4					
17	RENV5	Environment setting 5	E0h	RRENV5	A0h	WRENV5					
18	RENV6	Environment setting 6	E1h	RRENV6	A1h	WRENV6					
19	RENV7	Environment setting 7	E2h	RRENV7	A2h	WRENV7					
20	RCUN1	COUNTER1 (command position)	E3h	RRCUN1	A3h	WRCUN1					
21	RCUN2	COUNTER2 (mechanical position)	E4h	RRCUN2	A4h	WRCUN2					
22	RCUN3	COUNTER3 (deflection counter)	E5h	RRCUN3	A5h	WRCUN3					
23	RCUN4	COUNTER4 (general purpose)	E6h	RRCUN4	A6h	WRCUN4					
24	RCMP1	Data for comparator 1	E7h	RRCMP1	A7h	WRCMP1					
25	RCMP2	Data for comparator 2	E8h	RRCMP2	A8h	WRCMP2					
26	RCMP3	Data for comparator 3	E9h	RRCMP3	A9h	WRCMP3					
27	RCMP4	Data for comparator 4	EAh	RRCMP4	AAh	WRCMP4					
28	RCMP5	Data for comparator 5	EBh	RRCMP5	ABh	WRCMP5	PRCP5	CBh	RPRCP5	8Bh	WPRCP5
29	RIRQ	Event INT setting	ECh	RRIRQ	ACh	WRIRQ					
30	RLTC1	COUNTER1 latched data	EDh	RRLTC1							
31	RLTC2	COUNTER2 latched data	EEh	RRLTC2							
32	RLTC3	COUNTER3 latched data	EFh	RRLTC3							
33	RLTC4	COUNTER4 latched data	F0h	RRLTC4							
34	RSTS	Extension status	F1h	RRSTS							
35	REST	Error INT status	F2h	RREST							
36	RIST	Event INT status	F3h	RRIST							
37	RPLS	Positioning counter	F4h	RRPLS							
38	RSPD	EZ counter, speed monitor	F5h	RRSPD							
39	PSDC	Ramping-down point	F6h	PRSDC							
40	RCI	Circular interpolation stepping number	FCh	RRCI	BCh	WRCI	PRCI	CCh	RPRCI	8Ch	WPRCI
41	RIPS	Interpolation status	FFh	RRIPS							

7-5. General-purpose output port control

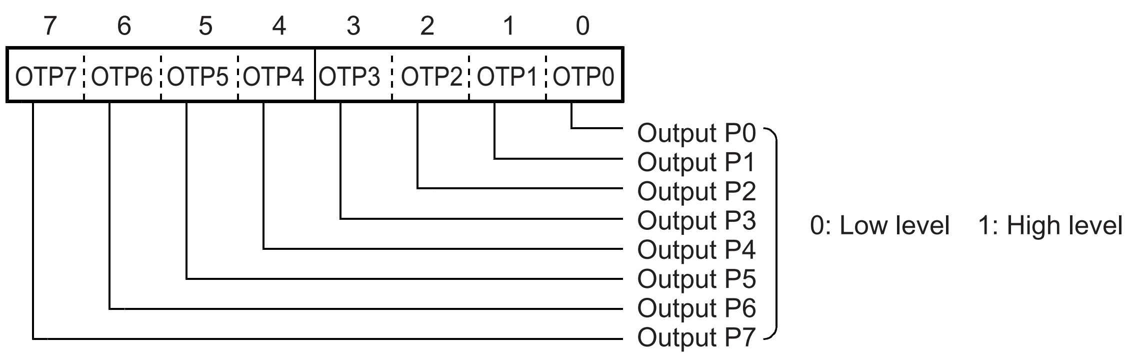
By writing output port control data to the address for the general-purpose output port (OTPB: Address 2 when a Z80 I/F is used), the output status of all 8 bits on the P0 to P7 terminals can be controlled as a group.
When the I/O settings for terminals P0 to P7 are designated as general-purpose outputs, the status of terminals P0 to P7 changes according to the control data written there. Bits corresponding to terminals that are not set up as output terminals, are invalid.
When writing words to the port, the upper 8 bits are discarded. However, they should be set to zero to maintain future compatibility.
The output status of terminals P0 to P7 are latched, even after the I/O setting is changed to input.
The output status for each terminal can be set individually using the bit control command.
The general-purpose ports are assigned to function as general-purpose inputs, general-purpose outputs, or comparators by writing to the Environment Setting 2 register (see section 8-3-14, "RENV2 register"). The status of the general-purpose port can be read from general-purpose I/O port address (IOPB: Address 2 when a Z80 I/F is used).

7-5-1. Control data writing procedures

Write control data to output port (OTPB: Address 2 when a Z80 I/F is used).
To continue with the next command, the LSI must wait two reference clock cycles (approx. 0.1 µs when CLK = 19.6608 MHz). The $\overline{WRQ}$ terminal outputs a wait request signal.



7-5-2. Control data bit allocation



8. Registers

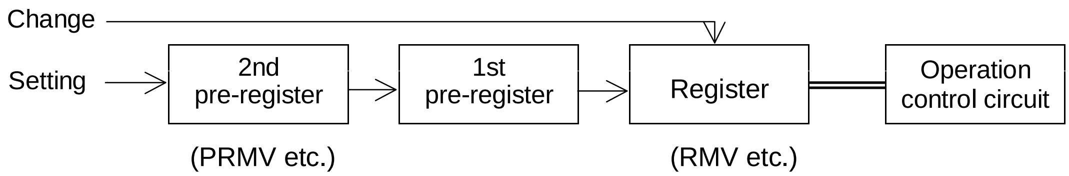
8-1. Table of registers

The following registers are available for each axis.

No.	Register name	Bit length	R/W	Details	2nd pre-register name
1	RMV	28	R/W	Feed amount, target position	PRMV
2	RFL	16	R/W	Initial speed	PRFL
3	RFH	16	R/W	Operation speed	PRFH
4	RUR	16	R/W	Acceleration rate	PRUR
5	RDR	16	R/W	Deceleration rate	PRDR
6	RMG	12	R/W	Speed magnification rate	PRMG
7	RDP	24	R/W	Ramping-down point	PRDP
8	RMD	28	R/W	Operation mode	PRMD
9	RIP	28	R/W	Circular interpolation center position, master axis feed amount with linear interpolation and with multiple chips	PRIP
10	RUS	15	R/W	S-curve acceleration range	PRUS
11	RDS	15	R/W	S-curve deceleration range	PRDS
12	RFA	16	R/W	Speed at amount correction	
13	RENV1	32	R/W	Environment setting 1 (specify I/O terminal details)	
14	RENV2	27	R/W	Environment setting 2 (specify general-purpose port details)	
15	RENV3	32	R/W	Environment setting 3 (specify zero return and counter details)	
16	RENV4	32	R/W	Environment setting 4 (specify details for comparators 1 to 4)	
17	RENV5	22	R/W	Environment setting 5 (specify details for comparator 5)	
18	RENV6	32	R/W	Environment setting 6 (specify details for feed amount correction)	
19	RENV7	32	R/W	Environment setting 7 (specify vibration reduction control details)	
20	RCUN1	28	R/W	COUNTER1 (command position)	
21	RCUN2	28	R/W	COUNTER2 (mechanical position)	
22	RCUN3	16	R/W	COUNTER3 (deflection counter)	
23	RCUN4	28	R/W	COUNTER4 (general-purpose counter)	
24	RCMP1	28	R/W	Comparison data for comparator 1	
25	RCMP2	28	R/W	Comparison data for comparator 2	
26	RCMP3	28	R/W	Comparison data for comparator 3	
27	RCMP4	28	R/W	Comparison data for comparator 4	
28	RCMP5	28	R/W	Comparison data for comparator 5	PRCP5
29	RIRQ	19	R/W	Specify event interruption cause	
30	RLTC1	28	R	COUNTER1 (command position) latch data	
31	RLTC2	28	R	COUNTER2 (mechanical position) latch data	
32	RLTC3	16	R	COUNTER3 (deflection counter) latch data	
33	RLTC4	28	R	COUNTER4 (general-purpose) latch data	
34	RSTS	17	R	Extension status	
35	REST	18	R	Error INT status	
36	RIST	20	R	Event INT status	
37	RPLS	28	R	Positioning counter (number of residual pulses to feed)	
38	RSPD	23	R	EZ counter, current speed monitor	
39	RSDC	24	R	Automatically calculated ramping-down point	
40	RCI	31	R/W	Number of steps for interpolation (only used by the X axis)	PRCI
41	RIPS	24	R	Interpolation status	

8-2. Pre-registers

The following registers and start commands have pre-registers:
RMV, RFL, RFH, RUR, RDR, RMG, RDP, RMD, RIP, RUS, RDS, RCMP5, and RCI.
The term pre-register refers to a register which contains the next set of operation data while the current step is executing. The pre-registers in this LSI have the following two layer design and are processed in FIFO order.



Normally, operation data is written into the 2nd pre-register.
To change the current operation status, such as the feed speed, the LSI will write new data into a register.
Data is transferred from the 2nd pre-register to the 1st pre-register to the execution register where it is executed when a start command is written, and the current operation is complete.

The writing and operation procedures for the pre-registers are shown below.

1) If the 1st and 2nd pre-registers are empty when data is written into the 2nd pre-register, it is transferred to the 1st pre-register, and then to the execution register.

2) When a start command is written, the LSI instructs the motors to operate using the data in the execution register, and then the data in the 1st pre-register is transferred into the execution register.

3) While in operation, the LSI will write the next operation data into the 2nd pre-register. (If the data details are the same as the previous one, this operation can be omitted.)
Since the 1st pre-register has been emptied by transferring its contents to the execution register, the LSI will transfer the operation data in the 2nd pre-register to the 1st pre-register.

4) By writing the start command for the next operation, it is assumed that the 1st pre-register has stored the data.

5) The LSI writes operation data into the 2nd pre-register, one set after another. (Duplicate data can be omitted.)
Since the 1st pre-register has already stored the data, the LSI will only write operation data into the 2nd pre-register one set after another.

6) By writing sequential operation start commands, the 2nd pre-register has stored the data.

7) When the current operation is complete, data is transferred between the registers as follows:
2nd pre-register -> 1st pre-register -> execution register. The LSI then instructs the machine to use the next data.

8) Once the data in the 2nd pre-register has been copied to the 1st pre-register, the 2nd pre-register is empty. It can now store the operation data for the next step.

To check whether there is data in the 2nd pre-register, you can use the MSTSW (main status) command.
By setting the RIRQ (event interrupt cause) register, the LSI can be made to output an $\overline{INT}$ signal when the 2nd preset register is emptied due to the completion of an operation.

Note: When you want the next operation to start automatically using the pre-registers, set the operation completion timing to "cycle completion (METM = 0 on RMD). When "pulse completion" (METM = 0 on RMD), the time between the last pulse and next operation start pulse, will be as little as 14 x T_{CLK} (T_{CKL}: Reference clock cycle).
For details, see 11-3-2. "Control the output pulse width and operation completion timing."

8-3.Description of the registers

The initial value of all the registers and pre-registers is "0."
Please note that with some registers, a value of "0" is outside the allowable setting range.

8-3-1. RMV (PRMV) registers

These registers are used to specify the incremental target position for positioning operations.
PRMV is the 2nd pre-register for RMV.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
&	&	&	&																												

Setting range: -134,217,728 to +134,217,727.
By changing the RMV register while in operation, the feed length can be overridden.

8-3-2. RFL (PRFL) registers

These registers are used to set the initial speed (stop seed) for high speed (with acceleration/deceleration) operation.
PRFL is the 2nd pre-register for RFL.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*																

The setting range is 1 to 65,535. However, the actual speed [pps] may vary with the speed magnification rate setting in the RMG register.

8-3-3. RFH (PRFH) registers

These registers are used to specify the operation speed.
PRFH is the 2nd pre-register for RFH.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*																

The setting range is 1 to 65,535. However, the actual speed [pps] may vary with the speed magnification rate set in the RMG register.

8-3-4. RUR (PRUR) registers

These registers are used to specify the acceleration rate.
PRUR is the 2nd pre-register for RUR.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*																

Setting range is 1 to 65,535.

Note 1: Bits marked with an "*" asterisk will be ignored when written and are 0 when read.
Note 2: Bits marked with an "&" symbol will be ignored when written and will be the same value as the upper most bit among the non-marked bits. (Sign extension)

8-3-5. RDR (PRDR) registers

These registers are used to specify the deceleration rate.
PRDR is 2nd pre-register for RDR.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*																

The normal setting range is 1 to 65,535.
When RDR = 0, the deceleration rate will be the value set by RUR.

8-3-6. RMG (PRMG) registers

These registers are used to set the speed magnification rate.
PRMG is the 2nd pre-register for RMG.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*												

The setting range is 2 to 4,095.
Sets the relationship between the speed register RFL (PRFL), RFH (PRFH), RFA values and the operation speeds.
The actual operation speed [pps] is a product of the speed magnification rate and the speed register setting.

[Setting example when the reference clock is 19.6608 MHz]

Setting	Speed magnification rate	Operation speed setting range [pps]	Setting	Speed magnification rate	Operation speed setting range [pps]
2999	0.1x	0.1 to 6,553.5	59	5x	5 to 327,675
1499	0.2x	0.2 to 13,107.0	29	10x	10 to 655,350
599	0.5x	0.5 to 32,767.5	14	20x	20 to 1,310,700
299	1x	1 to 65,535	5	50x	50 to 3,276,750
149	2x	2 to 131,070	2	100x	100 to 6,553,500

8-3-7. RDP (PRDP) registers

These registers are used to set a ramping-down point (deceleration start point) for positioning operations.
PRDP is the 2nd pre-register for RDP.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
#	#	#	#	#	#	#	#																								

Bits marked with a "#" symbol are ignored when written and change their setting when read according to the setting of MSDP (bit 13) in the RMD register.

MSDP	Setting details	bit #
0	Offset for automatically set values. When this value is positive, the axis decelerates earlier. When this value is negative, the axis decelerates later.	Same as bit 23.
1	When number of pulses left drops to less than a set value, the motor on that axis starts to decelerate.	0

Note 1: Bits marked with an "*" asterisk will be ignored when written and are 0 when read.
Note 2: Bits marked with an "&" symbol will be ignored when written and will be the same value as the upper most bit among the no marked bits. (Sign extension.)

8-3-8. RMD (PRMD) registers

These registers are used to set the operation mode.
PRMD is the 2nd pre-register for RMD.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MIPF	MPCS	MSDP	METM	MCCE	MSMD	MINP	MSDE	0				MOD			

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	MPIE	MADJ	MSP0	MSPE	0	0	MAX1	MAX0	MSY1	MSY0	MSN1	MSN0

Bits	Bit name	Description
Setting basic operation mode		
0 to 6	MOD	Set operation mode. 000 0000 (00h): Continuous positive rotation controlled by command control. 000 1000 (08h): Continuous negative rotation controlled by command control. 000 0001 (01h): Continuous operation controlled by pulsar (PA/PB) input. 000 0010 (02h): Continuous operation controlled by external signal (+DR/-DR) input. 001 0000 (10h): Positive rotation zero return operation. 001 1000 (18h): Negative rotation zero return operation. 001 0010 (12h): Positive feed leaving from the zero position. 001 1010 (1Ah): Negative feed leaving from the zero position. 001 0101 (15h): Zero search in the positive direction 001 1101 (1Dh): Zero search in the negative direction 010 0000 (20h): Feed to +EL or +SL position. 010 1000 (28h): Feed to –EL or –SL position. 010 0010 (22h): Move away from the –EL or –SL position. 010 1010 (2Ah): Move away from the +EL or +SL position. 010 0100 (24h): Feed in the positive direction for a specified number of EZ counts. 010 1100 (2Ch): Feed in the negative direction for a specified number of EZ counts. 100 0001 (41h): Positioning operation (specify the incremental target position) 100 0100 (44h): Zero return of command position (COUNTER1). 100 0101 (45h): Zero return of mechanical position (COUNTER2). 100 0110 (46h): Single pulse operation in the positive direction. 100 1110 (4Eh): Single pulse operation in the negative direction. 100 0111 (47h): Timer operation 101 0001 (51h): Positioning operation controlled by pulsar (PA/PB) input. 101 0100 (54h): Zero return to the specified position controlled by pulsar (PA/PB) input. 101 0101 (55h): Zero return to a mechanical position controlled by pulsar (PA/PB) input. 101 0110 (56h): Positioning operation controlled by external signal (+DR/-DR) input. 110 0000 (60h): Continuous linear interpolation 1 (continuous operation with linear interpolation 1) 110 0001 (61h): Linear interpolation 1 110 0010 (62h): Continuous linear interpolation 2 (continuous operation with linear interpolation 2) 110 0011 (63h): Linear interpolation 2 110 0100 (64h): CW circular interpolation operation 110 0101 (65h): CCW circular interpolation operation. 110 1001 (69h): Linear interpolation 1 controlled by PA/PB input. 110 1100 (6Ch): CW circular interpolation operation controlled by PA/PB input. 110 1101 (6Dh): CCW circular interpolation operation controlled by PA/PB input.
7	Not defined	(Always set to 0.)

Bits	Bit name	Description
Optional setting items		
8	MSDE	1: Decelerates (deceleration stop) by turning ON the SD input.
9	MINP	1: Completes operation by turning ON the INP input.
10	MSMD	Specify an acceleration/deceleration type for high speed feed. (0: Linear accel/decel. 1: S-curve accel/decel.)
11	MCCE	1: Stop COUNTER1 (command position) This is used to move a mechanical part without changing the PLC control position
12	METM	Specify the operation complete timing. (0: End of cycle. 1: End of pulse.) When using the vibration reduction function, select "End of pulse."
13	MSDP	Specify the ramping-down point for high speed feed. (0: Automatic setting. 1: Manual setting.) Effective for positioning operations and linear interpolation feeding.
14	MPCS	1: While in automatic operation, control the number of pulses after the PCS input is turned ON. (Override 2 for the target position.)
15	MIPF	1: Make a constant, synthetic speed while performing interpolation feeding.
16 to 17	MSN0 to 1	When you want to control an operation block, specify a sequence number using 2 bits. By reading the main status (MSTSW), a sequence number currently being executed (SSC0 to 1) can be checked. Setting the sequence number does not affect the operation.
18 to 19	MSY0 to 1	After writing a start command, the LSI will start an axis synchronization operation based on other timing. 00: Start immediately. 01: Start by a $\overline{\text{STA}}$ input ($\overline{\text{CSTA}}$ or $\overline{\text{STA}}$) or command 06h, 2Ah. 10: Start with an internal synchronous start signal. 11: Start when a specified axis stops moving.
20 to 21	MAX0 to 1	Specify an axis to check for movement when MSY0 to 1 = 11. 01: Start when the X axis stops moving. (Only effective for Y axis) 10: Start when the Y axis stops moving. (Only effective for X axis)
22 to 23	Not defined	(Always set to 0.)
24	MSPE	1:Deceleration stop or immediate stop by $\overline{\text{CSTP}}$ input. This is used for a simultaneous stop with another axis when this other axis stops with an error.
25	MSPO	1: Outputs a $\overline{\text{CSTP}}$ (simultaneous stop) signal when stopping due to an error.
26	MADJ	Specify an FH correction function. (0: ON. 1: OFF.) When the ramping-down point, automatic setting and interpolation control of the S-curve accel/decel are selected, turn ON the FH correction function.
27	MPIE	1: Continue end point approaching operation after completion of a circular interpolation operation
28 to 31	Not defined	(Always set to 0.)

8-3-9. RIP (PRIP) registers

These registers are used to set the center position for circular interpolation or a master axis feed amount for linear interpolation 2.
PRIP is the 2nd pre-register for RIP.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
&	&	&	&																												

- When MOD (bits 0 to 6) of the RMD register are set as shown below, the register is enabled.
  110 0010 (62h): Continuous linear interpolation 2 (continuous operation with the linear interpolation 2).
  110 0011 (63h): Linear interpolation 2.
  110 0100 (64h): Circular interpolation in a CW direction.
  110 0101 (65h): Circular interpolation in a CCW direction.
- With Continuous linear interpolation 2 and Linear interpolation 2, specify the feed amount on the master axis using an incremental value.
- With circular interpolation, enter an circular center position using an absolute value.
- Setting range: -134,217,727 to +134,217,727

8-3-10. RUS (PRUS) registers

These registers are used to specify the S-curve range of the S-curve acceleration.
PRUS is the 2nd pre-register for RUS.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*															

The normal setting range is 1 to 32,767.
When 0 is entered, the value of (RFH - RFL)/2 will be calculated internally and applied.

8-3-11. RDS (PRDS) registers

These registers are used to specify the S-curve range of the S-curve deceleration.
PRDS is the 2nd pre-register for RDS.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*															

The normal setting range is 1 to 32,767.
When 0 is entered, the value of (RFH – RFL)/2 will be calculated internally and applied.

8-3-12. RFA register

This register is used to specify the low speed for backlash correction or slip correction.
This is also used as a reverse low speed for a zero return operation.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*																

Although the setting range is 1 to 65,535, the actual speed [pps] varies with the speed magnification rate setting in the RMG register.

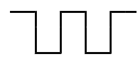
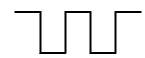
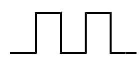
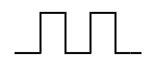
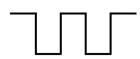
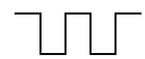
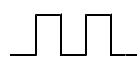
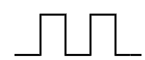
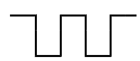
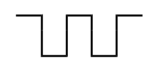
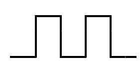
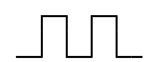
Note 1: Bits marked with an "*" asterisk will be ignored when written and are 0 when read.
Note 2: Bits marked with an "&" symbol will be ignored when written and will be the same value as the upper most bit among the no marked bits. (Sign extension)

8-3-13. RENV1 register

This register is used for Environment setting 1. This is mainly used to set the specifications for input/output terminals.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ERCL	EPW2	EPW1	EPW0	EROR	EROE	ALML	ALMM	ORGL	SDL	SDLT	SDW	ELM	PMD2	PMD1	PMD0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PDTC	PCSM	INTM	DTMF	DRF	FLTR	DRL	PCSL	LTCL	INPL	CLR1	CLR0	STPM	STAM	ETW1	ETW0

<table>
<tr><th>Bits</th><th>Bit name</th><th>Description</th></tr>
<tr><td>0 to 2</td><td>PMD0 to 2</td><td>Specify OUT output pulse details
<table>
<tr><td rowspan="2">PMD0 to 2</td><td colspan="2">When feeding in a positive direction</td><td colspan="2">When feeding in a negative direction</td></tr>
<tr><td>OUT output</td><td>DIR output</td><td>OUT output</td><td>DIR output</td></tr>
<tr><td>000</td><td></td><td>High</td><td></td><td>Low</td></tr>
<tr><td>001</td><td></td><td>High</td><td></td><td>Low</td></tr>
<tr><td>010</td><td></td><td>Low</td><td></td><td>High</td></tr>
<tr><td>011</td><td></td><td>Low</td><td></td><td>High</td></tr>
<tr><td>100</td><td></td><td>High</td><td>High</td><td></td></tr>
<tr><td>111</td><td></td><td>Low</td><td>Low</td><td></td></tr>
</table></td></tr>
<tr><td>3</td><td>ELM</td><td>Specify the process to occur when the EL input is turned ON. (0: Immediate stop. 1: Deceleration stop.) Note 1</td></tr>
<tr><td>4</td><td>SDM</td><td>Specify the process to occur when the SD input is turned ON. (0: Deceleration only. 1: Deceleration and stop.)</td></tr>
<tr><td>5</td><td>SDLT</td><td>Specify the latch function of the SD input. (0: OFF. 1: ON.)
Turns ON when the SD signal width is short.
When the SD input is OFF while starting, the latch signal is reset.
The latch signal is also reset when SDLT is 0.</td></tr>
<tr><td>6</td><td>SDL</td><td>Specify the SD signal input logic. (0: Negative logic. 1: Positive logic.)</td></tr>
<tr><td>7</td><td>ORGL</td><td>Specify the ORG signal input logic. (0: Negative logic. 1: Positive logic.)</td></tr>
<tr><td>8</td><td>ALMM</td><td>Specify the process to occur when the ALM input is turned ON. (0: Immediate stop. 1: Deceleration stop.)</td></tr>
<tr><td>9</td><td>ALML</td><td>Specify the ALM signal input logic. (0: Negative logic. 1: Positive logic.)</td></tr>
<tr><td>10</td><td>EROE</td><td>1: Automatically outputs an ERC signal when the axis is stopped immediately by a +EL, -EL, ALM, or $\overline{\text{CEMG}}$ input signal. However, the ERC signal is not output when a deceleration stop occurs on the axis. When the EL signal is specified for a normal stop, by setting MOD = "010X000" (feed to the EL position) in the RMD register, the ERC signal is output if an immediate stop occurs.</td></tr>
<tr><td>11</td><td>EROR</td><td>1: Automatically output the ERC signal when the axis completes a zero return.</td></tr>
<tr><td>12 to 14</td><td>EPW0 to 2</td><td>Specify the pulse width of the ERC output signal.
000: 12 µs 001: 102 µs 010: 409 µs 011:1.6 ms
100: 13 ms 101: 52 ms 110: 104ms 111: Level output</td></tr>
</table>

Bit	Bit name	Description
15	ERCL	Specify the ERC signal output logic. (0: Negative logic. 1: Positive logic.)
16 to 17	ETW0 to 1	Specify the ERC signal OFF timer time. 00: 0 µs 10: 1.6 ms, 01: 12 µs 11: 104 ms
18	STAM	Specify the $\overline{\text{STA}}$ signal input type. (0: Level trigger. 1: Edge trigger.)
19	STPM	Specify a stop method using $\overline{\text{CSTP}}$ input. (0: Immediate stop. 1: Deceleration stop.)
20 to 21	CLR0 to 1	Specify a CLR input. 00: Clear on the falling edge 10: Clear on a LOW. 01: Clear on the rising edge 11: Clear on a HIGH.
22	INPL	Specify the INP signal input logic. (0: Negative logic. 1: Positive logic.)
23	LTCL	Specify the operation edge for the LTC signal. (0: Falling. 1: Rising)
24	PCSL	Specify the PCS signal input logic. (0: Negative logic. 1: Positive logic.)
25	DRL	Specify the +DR, -DR signal input logic. (0: Negative logic. 1: Positive logic.)
26	FLTR	1: Apply a filter to the +EL, -EL, SD, ORG, ALM, or INP inputs. When a filter is applied, signals pulses shorter than 4 µs are ignored.
27	DRF	1: Apply a filter on the +DR, -DR, or PE inputs. When a filter is applied, signals pulses shorter than 32 ms are ignored.
28	DTMF	1: Turn OFF the direction change timer (0.2 ms) function.
29	INTM	1: Mask an INT output. (Changes the interrupt circuit.)
30	PCSM	1: Only allow the PCS input on the local axis $\overline{\text{CSTA}}$ signal. (To provide compatibility with the PCL6045.)
31	PDTC	1: Keep the pulse width at a 50% duty cycle.

Note: When a deceleration stop (ELM = 1) has been specified to occur when the EL input turns ON, the axis will start the deceleration when the EL input is turned ON. Therefore, the axis will stop by passing over the EL position. In this case, be careful to avoid collisions of mechanical systems.

8-3-14. RENV2 register

This is a register for the Environment 2 settings. Specify the function of the general-purpose port, EA/EB input, and PA/PB input.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
P7M1	P7M0	P6M1	P6M0	P5M1	P5M0	P4M1	P4M0	P3M1	P3M0	P2M1	P2M1	P1M1	P1M0	P0M1	P0M0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	0	PDIR	PIM1	PIM0	EZL	EDIR	EIM1	EIM0	PINF	EINF	P1L	P0L

Bits	Bit name	Description
0 to 1	P0M0 to 1	Specify the operation of the P0/FUP terminals 00: General-purpose input 01: General-purpose output 10: Output the FUP (acceleration) signal. 11: General-purpose one shot signal output (T = 26 ms) Note: 1 & 2.
2 to 3	P1M0 to 1	Specify the operation of the P1/FDW terminals 00: General-purpose input 01: General-purpose output 10: Output the DFW (deceleration) signal. 11: General-purpose one shot signal output (T = 26 ms) Note: 1 & 2.
4 to 5	P2M0 to 1	Specify the operation of the P2/MVC terminal. 00: General-purpose input 01: General-purpose output 10: Output the MVC (low speed feeding) signal with negative logic. 11: Output the MVC (low speed feeding) signal with positive logic.
6 to 7	P3M0 to 1	Specify the operation of the P3/CP1 (+SL) terminals. 00: General-purpose input 01: General-purpose output 10: Output the CP1 (satisfied the Comparator 1 conditions) signal with negative logic. 11: Output the CP1 (satisfied the Comparator 1 conditions) signal with positive logic.
8 to 9	P4M0 to 1	Specify the operation of the P4/CP2 (-SL) terminals. 00: General-purpose input 01: General-purpose output 10: Output the CP2 (satisfied the Comparator 2 conditions) signal with negative logic. 11: Output the CP2 (satisfied the Comparator 2 conditions) signal with positive logic.
10 to 11	P5M0 to 1	Specify the operation of the P5/CP3 terminals. 00: General-purpose input 01: General-purpose output 10: Output the CP3 (satisfied the Comparator 3 conditions) signal with negative logic. 11: Output the CP3 (satisfied the Comparator 3 conditions) signal with positive logic.
12 to 13	P6M0 to 1	Specify the operation of the P6/CP4/ID terminals. 00: General-purpose input 01: General-purpose output 10: Output the CP4 (satisfied the Comparator 4 conditions) signal with negative logic. 11: Output the CP4 (satisfied the Comparator 4 conditions) signal with positive logic.
14 to 15	P7M0 to 1	Specify the operation of the P7/CP5 terminals. 00: General-purpose input 01: General-purpose output 10: Output the CP5 (satisfied the Comparator 5 conditions) signal with negative logic. 11: Output the CP5 (satisfied the Comparator 5 conditions) signal with positive logic.

Bits	Bit name	Description
16	P0L	Specify the output logic when the P0 terminal is used for FUP or as a one shot. (0: Negative logic. 1: Positive logic.)
17	P1L	Specify the output logic when the P1 terminal is used for FDW or as a one shot. (0: Negative logic. 1: Positive logic.)
18	EINF	1: Apply a noise filter to EA/EB input. Note 3.
19	PINF	1: Apply a noise filter to PA/PB input. Note 3.
20 to 21	EIM0 to 1	Specify the EA/EB input operation. 00: Multiply a 90˚ phase difference by 1 (Count up when the EA input phase is ahead.) 01: Multiply a 90˚ phase difference by 2 (Count up when the EA input phase is ahead.) 10: Multiply a 90˚ phase difference by 4 (Count up when EA input phase is ahead.) 11: Count up when the EA signal rises, count down when the EB signal falls.
22	EDIR	1: Reverse the counting direction of the EA/EB inputs.
23	EZL	Specify EZ signal input logic. (0: Falling edge. 1: Rising edge.)
24 to 25	PIM0 to 1	Specify the PA/PB input operation. 00: Multiply a 90˚ phase difference by 1 (Count up when the PA input phase is ahead.) 01: Multiply a 90˚ phase difference by 2 (Count up when the PA input phase is ahead.) 10: Multiply a 90˚ phase difference by 4 (Count up when PA input phase is ahead.) 11: Count up when the EA signal rises, count down when the PB signal falls.
26	PDIR	1: Reverse the counting direction of the PA/PB inputs.
27 to 31	Not defined	(Always set to 0.)

Note 1: For details about outputting a general-purpose one shot signal, see 7-2 "General-purpose output bit control commands."

Note 2: Dedicated output terminals have been added for the FUP, FDW, and MVC signals. However, they can also be output on the P0 to P2 terminals (the same as the PCL6045).

Note 3: Signals lasting less than 3 reference clock cycles will be ignored.

8-3-15. RENV3 register

This is a register for the Environment 3 settings. Zero return methods and counter operation specifications are the main function of this register.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	BSYC	CI41	CI40	CI31	CI30	CI21	CI20	EZD3	EZD2	EZD1	EZD0	ORM3	ORM2	ORM1	ORM0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CU4H	CU3H	CU2H	0	CU4B	CU3B	CU2B	CU1B	CU4R	CU3R	CU2R	CU1R	CU4C	CU3C	CU2C	CU1C

Bits	Bit name	Description
0 to 3	ORM0 to 3	Specify a zero return method. 0000: Zero return operation 0 - Stops immediately (deceleration stop when feeding at high speed) by changing the ORG input from OFF to ON. - Counter reset timing: When the ORG input is turned ON. 0001:Zero return operation 1 - Stops immediately (deceleration stop when feeding at high speed) by changing the ORG input from OFF to ON, and feeds in the opposite direction at RFA low speed until ORG input is turned OFF. Then, feeds in the original direction at RFA speed. While doing so, it will stop immediately when the ORG input is turned ON again. - COUNTER reset timing: When ORG input is turned ON from OFF. 0010: Zero return operation 2 - When feeding at low speed, movement on the axis stops immediately by counting the EZ signal after the ORG input is turned ON. When feeding at high speed, movement on the axis decelerates when the ORG input is turned ON and stops immediately by counting the EZ counts. - COUNTER reset timing: When counting the EZ signal. 0011: Zero return operation 3 - When feeding at low speed, movement on the axis stops immediately by counting the EZ signal after the ORG input is turned ON. When feeding at high speed, the axis will decelerate and stop by counting the EZ signal after the ORG input is turned ON. - COUNTER reset timing: When counting the EZ signal. 0100: Zero return operation 4 - Stops immediately (deceleration stop when feeding at high speed) by turning the ORG input ON, and feeds in the reverse direction at RFA low speed. Stops immediately by counting the EZ signal. - COUNTER reset timing: When counting the EZ signal. 0101:Zero return operation 5 - Stop immediately (deceleration stop when feeding at high speed) and reverse direction when the ORG input is turned ON. Then, stop immediately when counting the EZ signal. - COUNTER reset timing: When counting the EZ signal. 0110: Zero return operation 6 - Stop immediately (deceleration stop when ELM = 1) by turning ON the EL input, and reverse at RFA low speed. Then stop immediately by turning OFF the EL input. - COUNTER reset timing: When EL input is OFF. 0111: Zero return operation 7 - Stop immediately (deceleration stop when ELM = 1) by turning ON the EL input, and reverse direction at RFA low speed. Then stop immediately by counting the EL signal. - COUNTER reset timing: When stopped by counting the EL input. 1000: Zero return operation 8 - Stop immediately (deceleration stop when ELM = 1) and reverse direction by turning ON the EL signal. Then stop immediately (deceleration stop when feeding at high speed) when counting the EZ signal. - COUNTER reset timing: When counting the EZ signal. 1001: Zero return operation 9 - After executing a Zero return operation 0, move back to the zero position (operate until COUNTER2 = 0). 1010: Zero return operation 10 - After executing a Zero return operation 3, move back to the zero position (operate until COUNTER2 = 0). 1011: Zero return operation 11 - After executing a Zero return operation 5, move back to the zero position (operate until COUNTER2 = 0). 1100: Zero return operation 12 - After executing a Zero return operation 8, move back to the zero position (operate until COUNTER2 = 0).

Bits	Bit name	Description
4 to 7	EZD0 to 3	Specify the EZ count up value that is used for zero return operations. 0000 (1st count) to 1111 (16th count)
8 to 9	CI20 to 21	Select the input count source for COUNTER2 (mechanical position). 00: EA/EB input 01: Output pulse 10: PA/PB input
10 to 11	CI30 to 31	Select the input count source for COUNTER3 (deflection counter) 00: Output pulse and EA/EB input (deflection counter) 01: Output pulse and PA/PB input (deflection counter) 10: EA/EB input and PA/PB input (deflection counter)
12 to 13	CI40 to 41	Select the input count source for COUNTER4 (general-purpose) 00: Output pulse 01: EA/EB input 10: PA/PB input 11: Divide the CLK count by 2
14	BSYC	1: Operate COUNTER4 only while operating ($\overline{BSY}$ is low).
15	Not defined	(Always set to 0.)
16	CU1C	1: Reset COUNTER1 (command position) when the CLR input turns ON.
17	CU2C	1: Reset COUNTER2 (mechanical position) when the CLR input turns ON.
18	CU3C	1: Reset COUNTER3 (deflection counter) when the CLR input turns ON.
19	CU4C	1: Reset COUNTER4 (general-purpose) when the CLR input turns ON.
20	CU1R	1: Reset COUNTER1 (command position) when the zero return is complete.
21	CU2R	1: Reset COUNTER2 (mechanical position) when the zero return is complete.
22	CU3R	1: Reset COUNTER3 (deflection counter) when the zero return is complete.
23	CU4R	1: Reset COUNTER4 (general-purpose) when the zero return is complete.
24	CU1B	1: Operate COUNTER1 (command position) while in backlash/slip correction mode.
25	CU2B	1: Operate COUNTER2 (mechanical position) while in backlash/slip correction mode.
26	CU3B	1: Operate COUNTER3 (deflection counter) while in backlash/slip correction mode.
27	CU4B	1: Operate COUNTER4 (general-purpose) while in backlash/slip correction mode.
28	Not defined	(Always set to 0.)
29	CU2H	1: Stop the counting operation on COUNTER2 (mechanical position). Note 1.
30	CU3H	1: Stop the counting operation on COUNTER3 (deflection counter).
31	CU4H	1: Stop counting operation of COUNTER4 (general-purpose).

Note 1: To stop the counting on COUNTER1 (command position), change MCCE (bit 11) in the RMD register.

8-3-16. RENV4 register

This register is used for Environment 4 settings. Set up comparators 1 to 4.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	C2D1	C2D0	C2S2	C2S1	C2S0	C2C1	C2C0	0	C1D1	C1D0	C1S2	C1S1	C1S0	C1C1	C1C0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
C4D1	C4D0	C4S3	C4S2	C4S1	C4S0	C4C1	C4C0	0	C3D1	C3D0	C3S2	C3S1	C3S0	C3C1	C3S0

Bit	Bit name	Description
0 to 1	C1C0 to 1	Select a comparison counter for comparator 1. Note 1 00: COUNTER1 (command position) 01: COUNTER2 (mechanical position) 10: COUNTER3 (deflection counter) 11: COUNTER4 (general-purpose)
2 to 4	C1S0 to 2	Select a comparison method for comparator 1. Note 2 001: RCMP1 data = comparison counter (regardless of counting direction) 010: RCMP1 data = comparison counter (while counting up) 011: RCMP1 data = comparison counter (while counting down) 100: RCMP1 data > Comparison counter data 101: RCMP1 data < Comparison counter data 110: Use as positive end software limit (RCMP1< COUNTER1) Others: Treats that the comparison conditions are not satisfied.
5 to 6	C1D0 to 1	Select a process to execute when the Comparator 1 conditions are met. 00: None (use as an $\overline{INT}$, terminal output, or internal synchronous start) 01: Immediate stop. 10: Deceleration stop. 11: Change operation data to pre-register data (change speed).
7	Not defined	(Always set to 0.)
8 to 9	C2C0 to 1	Select a comparison counter for Comparator 2. Note 1. 00: COUNTER1 (command position) 01: COUNTER2 (mechanical position) 10: COUNTER3 (deflection counter) 11: COUNTER4 (general purpose)
10 to 12	C2S0 to 2	Select a comparison method for Comparator 2. Note 2. 001: RCMP2 data = Comparison counter (regardless of counting direction) 010: RCMP2 data = Comparison counter (while counting up) 011: RCMP2 data = Comparison counter (while counting down) 100: RCMP2 data > Comparison counter data 101: RCMP2 data < Comparison counter data 110: Use as positive end software limit (RCMP2>COUNTER1]) Others: Treats that the comparison conditions do not meet.
13 to 14	C2D0 to 1	Select a process to execute when the Comparator 2 conditions are met. 00: None (use as an $\overline{INT}$, terminal output, or internal synchronous start) 01: Immediate stop. 10: Deceleration stop. 11: Change operation data to pre-register data (change speed).
15	Not defined	(Always set to 0.)

Bit	Bit name	Description (RENV4 continued)
16 to 17	C3C0 to 1	Select a comparison counter for Comparator 3. Note 1 00: COUNTER1 (command position) 01: COUNTER2 (mechanical position) 10: COUNTER3 (deflection counter) 11: COUNTER4 (general-purpose)
18 to 20	C3S0 to 2	Select a comparison method for comparator 3. Note 2 001: RCMP3 data = comparison counter (regardless of counting direction) 010: RCMP3 data = comparison counter (while counting up) 011: RCMP3 data = comparison counter (while counting down) 100: RCMP3 data > Comparison counter data 101: RCMP3 data < Comparison counter data 110: Prohibited setting Others: Treats that the comparison conditions do not meet.
21 to 22	C3D0 to 1	Select a process to execute when the Comparator 3 conditions are met. 00: None (use as an $\overline{\text{INT}}$, terminal output, or internal synchronous start) 01: Immediate stop. 10: Deceleration stop. 11: Change operation data to pre-register data (change speed).
23	Not defined	(Always set to 0.)
24 to 25	C4C0 to 1	Select a comparison counter for Comparator 4. Note 1. 00: COUNTER1 (command position) 01: COUNTER2 (mechanical position) 10: COUNTER3 (deflection counter) 11: COUNTER4 (general purpose)
26 to 29	C4S0 to 3	Select a comparison method for Comparator 4. Note 3. 0001: RCMP4 data = Comparison counter (regardless of counting direction) 0010: RCMP4 data = Comparison counter (while counting up) 0011: RCMP4 data = Comparison counter (while counting down) 0100: RCMP4 data > Comparison counter data 0101: RCMP4 data < Comparison counter data 0111: Treats that the comparison conditions do not meet. 1000: Use as IDX (synchronous) signal output (regardless of counting direction) 1001: Use as IDX (synchronous) signal output (while counting up) 1010: Use as IDX (synchronous) signal output (while counting down) Others: Treats that the comparison conditions do not meet.
30 to 31	C4D0 to 1	Select a process to execute when the Comparator 4 conditions are met. 00: None (use as an $\overline{\text{INT}}$, terminal output, or internal synchronous start) 01: Immediate stop. 10: Deceleration stop. 11: Change operation data to pre-register data (change speed).

Note 1: When COUNTER3 (deflection counter) is selected as the comparison counter, the LSI compares the counted absolute value and the comparator data. (Absolute value range: 0 to 32,767.)

Note 2: When you specify C1S0 to 2 = 110 (positive software limit) or C2S0 to 2 = 110 (negative software limit), select COUNTER1 (specified position) as the comparison counter.
When a software limit is set, movement on the axis stops regardless of the setting on C1D0 to 1 or on C2D0 to 1. (When a deceleration stop is selected, movement on the axis will stop after decelerating when it is being fed at high speed.)

Note 3: When C4S0 to 3 is set to 1000 to 1010 (synchronous signal output), select COUNTER4 (general-purpose) for the comparison counter. The other counters cannot be selected.
To set the comparator, select a positive value.

8-3-17. RENV5 register

This is a register for the Environment 5 settings. Settings for Comparator 5 are its main use.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LT0F	LTFD	LTM1	LTM0	0	IDL2	IDL1	IDL0	C5D1	C5D0	C5S2	C5S1	C5S0	C5C2	C5C1	C5C0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	0	0	0	0	0	0	SYI1	SYI0	SYO3	SYO2	SYO1	SYO0

Bit	Bit name	Description
0 to 2	C5C0 to 2	Select a comparison counter for comparator 5. 000: COUNTER1 (command position) 001: COUNTER2 (mechanical position) 010: COUNTER3 (deflection counter) 011: COUNTER4 (general-purpose) 100: Positioning counter 101: Current speed data
3 to 5	C5S0 to 2	Select a comparison method for comparator 5. 001: RCMP5 data = Comparison counter (regardless of counting direction) 010: RCMP5 data = Comparison counter (while counting up) 011: RCMP5 data = Comparison counter (while counting down) 100: RCMP5 data > Comparison counter 101: RCMP5 data < Comparison counter Others: Treats that the comparison conditions do not meet.
6 to 7	C5D0 to 1	Select a process to execute when the Comparator 5 conditions are met. 00: None (use as an INT, terminal output, or internal synchronous start) 01: Immediate stop. 10: Deceleration stop. 11: Change operation data to pre-register data (change speed).
8 to 10	IDL0 to 2	Enter the number of idling pulses. (0 to 7 pulses)
11	Not defined	(Always set to 0.)
12 to 13	LTM0 to 1	Specify the latch timing for a counter (COUNTER1 to 4). 00: When the LTC input turns ON. 01: On an ORG input 10: When the Comparator 4 conditions are met. 11: When the Comparator 5 conditions are met.
14	LTFD	1: Latch the current speed in place of COUNTER3.
15	LTOF	1: Stop the latch by timing of a hardware operation. (Only used by software.)
16 to 19	SYO0 to 3	Select the output timing of the internal synchronous signal. 0001: When the Comparator 1 conditions are met. 0010: When the Comparator 2 conditions are met. 0011: When the Comparator 3 conditions are met. 0100: When the Comparator 4 conditions are met. 0101: When the Comparator 5 conditions are met. 1000: When starting acceleration. 1001: When ending acceleration. 1010: When starting deceleration. 1011: When ending deceleration. Others: Internal synchronous signal output is OFF.
20 to 21	SYI0 to 1	Select an input source when starting with an internal synchronous signal. 00: Internal synchronous signal output from the X axis. 01: Internal synchronous signal output from the Y axis.
22 to 31	Not defined	(Always set to 0.)

8-3-18. RENV6 register

This is a register for the Environment 6 settings. It is primarily used to set feed amount correction data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	ADJ1	ADJ0	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Bit name	Description
0 to 11	BR0 to 11	Enter a backlash correction amount or a slip correction amount.
12 to 13	ADJ0 to 1	Select a feed amount correction method. 00: Turn OFF the correction function. 01: Backlash correction 10: Slip correction
14 to 31	Not defined	(Always set to 0.)

8-3-19. RENV7 register

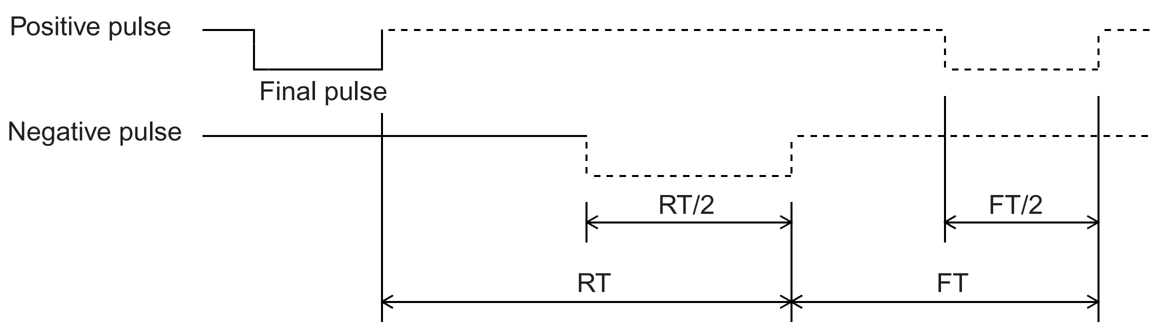
This is a register for the Environment 7 settings. It is primarily used to enter the time for the vibration reduction function. If both RT and FT data are other than zero, the vibration reduction function is turned ON.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RT15	RT14	RT13	RT12	RT11	RT10	RT9	RT8	RT7	RT6	RT5	RT4	RT3	RT2	RT1	RT0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FT15	FT14	FT13	FT12	FT11	FT10	FT9	FT8	FT7	FT6	FT5	FT4	FT3	FT2	FT1	FT0

Bit	Bit name	Description
0 to 15	RT0 to 15	Enter the RT time shown in the figure below. The units are 32 ticks of the reference clock (approx. 1.6 µs). Amount of time = 1.6 x the setting (µs) [When the reference clock = 19.6608 Mhz]
16 to 31	FT0 to 15	Enter the FT time shown in the figure below. The units are 32 ticks of the reference clock (approx. 1.6 µs).

The dotted lines in the figure below are pulses added by the vibration reduction function.



Amount of time [RT, FT] = 1.6 x the setting (µs) [When the reference clock = 19.6608 Mhz]

8-3-20. RCUN1 register

This is a register used for COUNTER1 (command position counter).
This is a counter used exclusively for command pulses.
Setting rage: -134,217,728 to +134,217,727.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
&	&	&	&																												

8-3-21. RCUN2 register

This is a register used for COUNTER2 (mechanical position counter).
It can count three types of pulses: Command pulses, encoder signals (EA/EB input), pulsar inputs (PA/PB input).
Setting range: -134,217,728 to +134,217,727.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
&	&	&	&																												

8-3-22. RCUN3 register

This is a register used for COUNTER3 (deflection counter).
It can count three types of deflections: Between command pulses and encoder signals, between command pulses and pulsar signals, and between encoder signals and pulsar signals.
Setting range: -32,768 to +32,767.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
&	&	&	&	&	&	&	&	&	&	&	&	&	&	&	&																

8-3-23.RCUN4 register

This is a register used for COUNTER4 (general-purpose counter).
It can count four types of signals: Command pulses, encoder signals (EA/EB input), pulsar signals (PA/PB input), and 1/2 ticks of the reference clock.
Setting range: -134,217,728 to +134,217,727.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
&	&	&	&																												

For details about the counters, see section 11-10, "Counters."

Note 1: Bits marked with an "*" asterisk will be ignored when written and are 0 when read.
Note 2: Bits marked with an "&" symbol will be ignored when written and will be the same value as the upper most bit among bits having no marks when read. (Sign extension)

8-3-24. RCMP1 register

Specify the comparison data for Comparator 1.
Setting range: -134,217,728 to +134,217,727.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
&	&	&	&																												

8-3-25. RCMP2 register

Specify the comparison data for Comparator 2.
Setting range: -134,217,728 to +134,217,727.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
&	&	&	&																												

8-3-26. RCMP3 register

Specify the comparison data for Comparator 3.
Setting range: -134,217,728 to +134,217,727.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
&	&	&	&																												

8-3-27. RCMP4 register

Specify the comparison data for Comparator 4.
Setting range: -134,217,728 to +134,217,727.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
&	&	&	&																												

8-3-28. RCMP5(PRCP5) register

Specify the comparison data for Comparator 5.
PRCP5 is the 2nd pre-register for RCMP5.
Setting range: -134,217,728 to +134,217,727.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
&	&	&	&																												

For details about the comparators, see section 11-11, "Comparator."

Note 1: Bits marked with an "*" asterisk will be ignored when written and are 0 when read.
Note 2: Bits marked with an "&" symbol will be ignored when written and will be the same value as the upper most bit among bits having no marks when read. (Sign extension)

8-3-29. RIRQ register

Enables event interruption cause.

Bits set to 1 that will enable an event interrupt for that event.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IROL	IRLT	IRCL	IRC5	IRC4	IRC3	IRC2	IRC1	IRDE	IRDS	IRUE	IRUS	IRND	IRNM	IRN	IREN

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	0	0	0	0	0	0	0	0	0	IRSA	IRDR	IRSD

Bit	Bit name	Description
0	IREN	Stopping normally.
1	IRN	Starting the next operation continuously.
2	IRNM	Writing to the 2nd pre-register.
3	IRND	Writing to the 2nd pre-register for Comparator 5.
4	IRUS	Starting acceleration.
5	IRUE	When ending acceleration.
6	IRDS	When starting deceleration.
7	IRDE	When ending deceleration.
8	IRC1	When Comparator 1 conditions are met.
9	IRC2	When Comparator 2 conditions are met.
10	IRC3	When Comparator 3 conditions are met.
11	IRC4	When Comparator 4 conditions are met.
12	IRC5	When Comparator 5 conditions are met.
13	IRCL	When resetting the count value with a CLR signal input.
14	IRLT	When latching the count value with an LTC signal input.
15	IROL	When latching the count value with an ORG signal input.
16	IRSD	When the SD input is ON.
17	IRDR	When the ±DR input is changed.
18	IRSA	When the $\overline{\text{CSTA}}$ input is ON.
19 to 31	Not defined	(Always set to 0.)

8-3-30. RLTC1 register

Latched data for COUNTER1 (command position). (Read only.)
The contents of COUNTER1 are copied when triggered by the LTC, an ORG input, or an LTCH command.
Data range: -134,217,728 to +134,217,727.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
&	&	&	&																												

8-3-31. RLTC2 register

Latched data for COUNTER2 (mechanical position). (Read only.)
The contents of COUNTER2 are copied when triggered by the LTC, an ORG input, or an LTCH command.
Data range: -134,217,728 to +134,217,727.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
&	&	&	&																												

8-3-32. RLTC3 register

Latched data for COUNTER3 (deflection counter) or current speed. (Read only.)
The contents of COUNTER3 or the current speed are copied when triggered by the LTC, an ORG input, or an LTCH command. When the LTFD in the RENV5 register is 0, the register latches the COUNTER3 data. When the LTFD is 1, the register latches the current speed. When the LTFD is 1 and movement on the axis is stopped, the latched data will be 0.
Data range when LTFD is 0: -32,768 to +32,767.
Data range when LTDF is 1: 0 to 65535.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
$	$	$	$	$	$	$	$	$	$	$	$	$	$	$	$																

Bits marked with a "$" will be the same as bit 15 when LTFD (bit 14) in the RENV5 register is 0 (sign extension), and they will be 0 when the LTFD is 1.

8-3-33. RLTC4 register

Latched data for COUNTER4 (general-purpose). (Read only.)
The contents of COUNTER4 are copied when triggered by the LTC, an ORG input, or an LTCH command.
Data range: -134,217,728 to +134,217,727.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
&	&	&	&																												

For details about the counter data latch, see section 11-10, " Counter."

Note 1: Bits marked with an "*" asterisk will be ignored when written and are 0 when read.
Note 2: Bits marked with an "&" symbol will be ignored when written and will be the same value as the upper most bit among bits having no marks when read. (Sign extension)

8-3-34. RSTS register

The extension status can be checked. (Read only.)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SDIN	SLTC	SCLR	SDRM	SDRP	SEZ	SERC	SPCS	SEMG	SSTP	SSTA	SDIR	CND3	CND2	CND1	CND0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	0	0	0	0	0	0	0	0	0	0	SEST	SINP

Bit	Bit name	Description
0 to 3	CND0 to 3	Reports the operation status. 0000: Under stopped condition 0001: Waiting for DR input 0010: Waiting for $\overline{\text{STA}}$ input 0011: Waiting for an internal synchronous signal 0100: Waiting for another axis to stop. 0101: Waiting for a completion of ERC timer 0110: Waiting for a completion of direction change timer 0111: Correcting backlash 1000: Waiting for PA/PB input 1001: Feeding at FA low speed. 1010: Feeding at FL low speed. 1011: Accelerating 1100: Feeding at FH low speed. 1101: Decelerating 1110: Waiting for INP input. 1111: Others (controlling start)
4	SDIR	Operation direction (0: Positive direction. 1: Negative direction.)
5	SSTA	Becomes 1 when the $\overline{\text{CSTA}}$ input signal is turned ON.
6	SSTP	Becomes 1 when the $\overline{\text{CSTP}}$ input signal is turned ON.
7	SEMG	Becomes 1 when the $\overline{\text{CEMG}}$ input signal is turned ON.
8	SPCS	Becomes 1 when the PCS input signal is turned ON.
9	SERC	Becomes 1 when the ERC input signal is turned ON.
10	SEZ	Becomes 1 when the EZ input signal is turned ON.
11	SDRP	Becomes 1 when the +DR input signal is turned ON.
12	SDRM	Becomes 1 when the -DR input signal is turned ON.
13	SCLR	Becomes 1 when the CLR input signal is turned ON.
14	SLTC	Becomes 1 when the LTC input signal is turned ON.
15	SDIN	Becomes 1 when the SD input signal is turned ON. (Status of SD input terminal.)
16	SINP	Becomes 1 when the INP input signal is turned ON.
17	SEST	Becomes 1 when the STA input signal is turned ON.
18 to 31	Not defined	(Always set to 0.)

8-3-35. REST register

Used to check the error interrupt cause. (Read only.)
The corresponding bit will be "1" when that item has caused an error interrupt.
This register is reset when read.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ESAO	ESPO	ESIP	ESDT	0	ESSD	ESEM	ESSP	ESAL	ESML	ESPL	ESC5	ESC4	ESC3	ESC2	ESC1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	0	0	0	0	0	0	0	0	0	0	ESPE	ESEE

Bit	Bit name	Description
0	ESC1	Stopped when Comparator 1 conditions were met. (+SL)
1	ESC2	Stopped when Comparator 2 conditions were met. (-SL)
2	ESC3	Stopped when Comparator 3 conditions were met.
3	ESC4	Stopped when Comparator 4 conditions were met.
4	ESC5	Stopped when Comparator 5 conditions were met.
5	ESPL	Stopped by the +EL input being turned ON.
6	ESML	Stopped by the -EL input being turned ON.
7	ESAL	Stopped by the ALM input being turned ON.
8	ESSP	Stopped Stopped by the $\overline{\text{CSTP}}$ input being turned ON.
9	ESEM	Stopped by the $\overline{\text{CEMG}}$ input being turned ON.
10	ESSD	Decelerated and stopped by the SD input being turned ON.
11	Not defined	(Always set to 0.)
12	ESDT	Stopped by an operation data error.
13	ESIP	Simultaneous stop with another axis due to an error stop on the other axis during interpolation.
14	ESPO	An overflow occurred in the PA/PB input buffer counter.
15	ESAO	An out of range count occurred in the positioning counter during interpolation.
16	ESEE	An EA/EB input error occurred.
17	ESPE	A PA/PB input error occurred.
18 to 31	Not defined	(Always set to 0.)

8-3-36. RIST register

This register is used to check the cause of event interruption. (Read only.)
When an event interrupt occurs, the bit corresponding to the cause will be set to 1.
This register is reset when read.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ISOL	ISLT	ISCL	ISC5	ISC4	ISC3	ISC2	ISC1	ISDE	ISDS	ISUE	ISUS	ISND	ISNM	ISN	ISEN

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	0	0	0	0	0	0	0	0	ISSA	ISMD	ISPD	ISSD

Bit	Bit name	Description
0	ISEN	Stopped automatically.
1	ISN	The next operation starts continuously.
2	ISNM	Available to write operation to the 2nd pre-register.
3	ISND	Available to write operation to the 2nd pre-register for Comparator 5.
4	ISUS	Starting acceleration.
5	ISUE	Ending acceleration.
6	ISDS	Starting deceleration.
7	ISDE	Ending deceleration.
8	ISC1	The comparator 1 conditions were met.
9	ISC2	The comparator 2 conditions were met.
10	ISC3	The comparator 3 conditions were met.
11	ISC4	The comparator 4 conditions were met.
12	ISC5	The comparator 5 conditions were met..
13	ISCL	The count value was reset by a CLR signal input.
14	ISLT	The count value was latched by an LTC input.
15	ISOL	The count value was latched by an ORG input.
16	ISSD	The SD input turned ON.
17	ISPD	The +DR input changed.
18	ISMD	The –DR input changed.
19	ISSA	The $\overline{\text{CSTA}}$ input turned ON.
20 to 31	Not defined	(Always set to 0.)

8-3-37. RPLS register

This register is used to check the value of the positioning counter (number of pulses left for feeding). (Read only.) At the start, this value will be the absolute value in the RMV register. Each pulse that is output will decrease this value by one.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0																												

8-3-38. RSPD register

This register is used to check the EZ count value and the current speed. (Read only.)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AS15	AS14	AS13	AS12	AS11	AS10	AS9	AS8	AS7	AS6	AS5	AS4	AS3	AS2	AS1	AS0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	0	0	0	0	0	IDC2	IDC1	IDC0	ECZ3	ECZ2	ECZ1	ECZ0

Bit	Bit name	Description
0 to 15	AS0 to 15	Read the current speed as a step value (same units as for RFL and RFH). When stopped the value is 0.
16 to 19	ECZ0 to 3	Read the count value of EZ input that is used for a zero return.
20 to 22	IDC0 to 2	Read the idling count value.
23 to 31	Not defined	(Always set to 0.)

8-3-39. RSDC register

This register is used to check the automatically calculated ramping-down point value for the positioning operation. (Read only.)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0																								

8-3-40. RCI (PRCI) registers

These registers are used to set circular interpolation stepping number.
PRCI is the 2nd pre-register for the RCI.
To decelerate during a circular interpolation, enter the number of steps required for the circular interpolation.
Entering a number other than 0 can decelerate the speed by using an automatic ramping-down point.
Setting range: 0 to 2,147,483,648.
This register is used exclusively for circular interpolation operations. It can only read and write on the X axis.
For details about the number of steps, see section 9-8-7, "Operation during interpolation."

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
&																															

8-3-41. RIPS register

This register is used to check the interpolation setting status and the operation status. (Read only.)
This register is shared by the X and Y axes and is the same when reading from either of axes.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	IPFy	IPFx	0	0	IPSy	IPSx	0	0	IPEy	IPEx	0	0	IPLy	IPLx

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	0	0	0	0	SED1	SED0	SDM1	SDM0	IPCC	IPCW	IPE	IPL

Bit	Bit name	Description
0	IPLx	1: X axis is in linear interpolation 1 mode.
1	IPLy	1: Y axis is in linear interpolation 1 mode.
2 to 3	Not defined	(Always 0.)
4	IPEx	1: X axis is in linear interpolation 2 mode.
5	IPEy	1: Y axis is in linear interpolation 2 mode.
6 to 7	Not defined	(Always set to 0.)
8	IPSx	1: X axis is in circular interpolation mode.
9	IPSy	1: Y axis is in circular interpolation mode.
10 to 11	Not defined	(Always set to 0.)
12	IPFx	1: X axis is specified for constant synthetic speed.
13	IPFy	1: Y axis is specified for constant synthetic speed.
14 to 15	Not defined	(Always set to 0.)
16	IPL	1: Executing linear interpolation 1.
17	IPE	1: Executing linear interpolation 2.
18	IPCW	1: Executing a CW directional circular interpolation.
19	IPCC	1: Executing a CCW directional circular interpolation.
20 to 21	SDM0 to 1	Current phase of a circular interpolation (00: 1st phase, 01: 2nd phase, 10: 3rd phase, 11: 4th phase)
22 to 23	SED0 to 1	Final phase in a circular interpolation (00: 1st phase, 01: 2nd phase, 10: 3rd phase, 11: 4th phase)
24 to 31	Not defined	(Always set to 0.)

9. Operation Mode

Specify the basic operation mode using the MOD area (bits 0 to 6) in the RMD (operation mode) register.

9-1. Continuous operation mode using command control

This is a mode of continuous operation. A start command is written and operation continues until a stop command is written.

MOD: 00h Continuous operation in the positive direction
08h Continuous operation in the negative direction

Stop by turning ON the EL signal corresponding to the direction of operation.
When operation direction is positive, +EL can be used. When operation direction is negative, -EL is used.
In order to start operation in the reverse direction after stopping the motion by turning ON the EL signal, a new start command must be written.

9-2. Positioning operation mode

The following five operation types are available for positioning operations.

- ♦ Positioning operation (specify a target position with an incremental value) <MOD: 41h>
- ♦ Return the command position to zero <MOD: 44h>
- ♦ Return the mechanical position to zero <MOD: 45h>
- ♦ One pulse operation <Positive direction MOD: 46h, Negative direction MOD: 4Eh>
- ♦ Timer operation <MOD: 47h>

9-2-1. Positioning operation (specify a target position using an incremental value)

This is a positioning mode used by placing a value in the RMV (target position) register. (MOD: 41h)
The feed direction is determined by the sign set in the RMV register.
When the value of the positioning counter drops to 0, movement on the axis stops. When you set the RMV register value to zero to start a positioning operation, the LSI will stop outputting pulses immediately.

9-2-2. Return the command position to zero

This mode is used to keep feeding until the value in COUNTER1 (command position) becomes zero. (MOD: 44h)
The number of output pulses and feed direction are set automatically by internal calculations based on the COUNTER1 value when starting.

9-2-3. Return the mechanical position to zero

This mode is used to continue operations until the value in COUNTER2 (mechanical position) becomes zero. (MOD: 45h)
The number of output pulses and feed direction are set automatically by internal calculations based on the COUNTER2 value when starting.

9-2-4. One pulse operation

This mode outputs a single pulse.

MOD: 46h One pulse operation in the positive direction
4Eh One pulse operation in the negative direction

This operation is identical to a positioning operation (incremental target positioning) that writes a "1" (or "-1") to the RMV register. However, with this operation, you do need not to write a "1" or "-1" to the RMV register.

9-2-5. Timer operation

This mode allows the internal operation time to be used as a timer. (MOD: 47h)
The internal effect of this operation is identical to the positioning operation. However, the LSI does not output any pulses (they are masked).
Therefore, the internal operation time using the low speed start command will be a product of the frequency of the output pulses and the RMV register setting. (Ex.: When the frequency is 1000 pps and the RMS register is set to 120 pulses, the internal operation time will be 120 ms.)
Write a positive number (1 to 134,217,727) into the RMV register.
The ±EL input signal, SD input signal, and software limits are ignored. (These are always treated as OFF.)
The ALM input signal $\overline{\text{CSTP}}$ input signal, and $\overline{\text{CEMG}}$ input signals are effective.
The backlash/slip correction, vibration restriction function, and when changing direction, this timer function is disabled.
The LSI stops counting from COUNTER1 (command position).
Regardless of the MINP setting (bit 9) in the RMD (operation mode) register, an operation complete delay controlled by the INP signal will not occur.
In order to eliminate deviations in the internal operation time, set the METM (bit 12) in the RMD register to zero and use the cycle completion timing of the output pulse as the operation complete timing.

9-3. Pulsar (PA/PB) input mode

This mode is used to allow operations from a pulsar input.
In order to enable pulsar input, bring the $\overline{PE}$ terminal LOW.
It is also possible to apply a filter on the $\overline{PE}$ input.
After writing a start command, when a pulsar signal is input, the LSI will outputs pulses to the OUT terminal.
Use an FL low speed start (STAFL: 50h) or an FH low speed start (STAFH: 51h).
Two methods are available for inputting pulsar signals through the PA/PB input terminal by setting the RENV2 (environmental setting 2) register as follows.

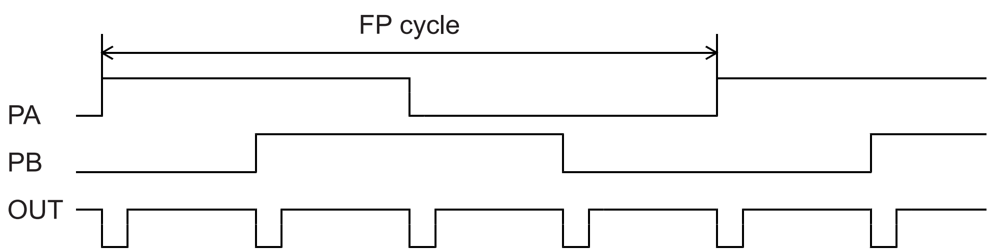
- ♦ Supply a 90˚ phase difference signal (1x, 2x, or 4x).
- ♦ Supply either positive or negative pulses.

There are four types of pulsar input mode operation, as shown in the table below.
The feed direction in continuous operation can be changed by setting of the RENV2 register without modifying wiring to the PA/PB terminals.

MOD	Operation type	Operation direction
01h	Continuous operation using pulsar inputs.	Determined by input whether it is PA or PB.
51h	Positioning operation using pulsar inputs.	Determined by the sign in the RMV register.
54h	Command position zero reset operation using pulsar inputs	Automatic setting using the RCUN1 value when starting.
55h	Mechanical position zero reset operation using pulsar inputs.	Automatic setting using the RCUN2 value when starting.

The relationship of the output pulse timing to the PA/PB inputs is shown below.

[Ex.: 90˚ phase difference 4x for PA/PB input]



When the pulsar input mode is selected, movement on the axis will stop by turning ON the EL signal corresponding to the current feed direction. However, it will not stop if the same signal for the opposite direction is received.
An error interrupt ($\overline{INT}$output) will not be produced.
To end this operation mode, write an immediate stop command (49h).

The maximum input frequency for pulsar signals is restricted by the FL speed when an FL low speed start is used, and by the FH speed when an FH low speed start is used. The LSI outputs $\overline{INT}$ signal as errors when both the PA and PB inputs change simultaneously, or when the input frequency is exceeded, or if the input/output buffer counter (deflection adjustment 16-bit counter for pulsar input and output pulse) overflows. This can be monitored by the REST (error interrupt factor) register.

<The relationship between the FH (FL) speed [pps] and the pulsar input frequency FP [pps]>

PA/PB input method	Use range
2 pulse input	FP < FH(FL)
90˚ phase difference 1x	FP < FH(FL)
90˚ phase difference 2x	FP < FH(FL) / 2
90˚ phase difference 4x	FP < FH(FL) / 4

Setting the PA/PB input <Set PIM0 to 1 (bits 24 to 25) in RENV2 > 00: 90˚ phase difference 1x 10: 90˚ phase difference 4x 01: 90˚ phase difference 2x 11: 2 pulse, up and down input	[RENV2] (WRITE) 31 24 0 0 0 0 0 - n n
Setting the PA/PB input counting direction <Set PDIR (bit 26) in RENV2> 0: Count up when the PA phase leads or on the PA rising edge. 1: Count up when the PB phase leads or on the PB falling edge.	[RENV2] (WRITE) 31 24 0 0 0 0 0 n - -
Setting ±DR□$\overline{PE}$ input filter <Set DRF (bit 27) in RENV1> 1: Apply a filter on the ±DR input or the $\overline{PE}$ input. When a filter is applied, pulse signals shorter than 32 ms are ignored.	[RENV1] (WRITE) 31 24 - - - - n - - -
Setting a PA/PB input noise filter <Set PINF (bit 19) in RENV2> 1: Apply a noise filter on PA/PB. When a filter is applied, signals lasting less than 3 reference clock cycles will be ignored.	[RENV2] (WRITE) 23 16 - - - - n - - -
Reading operation status <CND (bit 0 to 3) in RSTS> 1000: Waiting for a PA/PB input.	[RSTS] (READ) 7 0 - - - - n n n n
Reading out the PA/PB input error. <ESPE (bit 17) in REST> ESPE (bit 17) = 1: A PA/PB input error occurred	[REST] (READ) 23 16 0 0 0 0 0 0 n -
Reading the PA/PB input buffer counter status <ESPO (bit 14) in REST > ESPO(bit 14)=1: An overflow occurred	[REST] (READ) 15 8 - n - - - - - -

* In the descriptions in the right hand column, "n" refers to the bit position. "0" refers to bit positions where it is prohibited to write any value except zero and the bit will always be zero when read.

9-3-1. Continuous operation using a pulsar input (MOD: 01h)

This mode allows continuous operation using a pulsar input.
When PA/PB signals are input after writing a start command, the LSI will output pulses to the OUT terminal.
The feed direction depends on PA/PB signal input method and the value set in PDIR.

PA/PB input method	PDIR	Feed direction	PA/PB input
90˚ phase difference signal (1x, 2x, and 4x)	0	Positive direction	When the PA phase leads the PB phase.
		Negative direction	When the PB phase leads the PA phase.
	1	Positive direction	When the PB phase leads the PA phase.
		Negative direction	When the PA phase leads the PB phase.
2 pulse input of positive and negative pulses	0	Positive direction	PA input rising edge.
		Negative direction	PB input rising edge.
	1	Positive direction	PB input rising edge.
		Negative direction	PA input rising edge.

9-3-2. Positioning operations using a pulsar input (MOD: 51h)

This mode allows positioning using a pulsar input.
The feed direction is determined by the sign in the PMV (target position) register.
When starting, the RMV register value is loaded into the positioning counter. When PA/PB signals are input, the LSI outputs pulses and the positioning counter counts down. (Pulses are output in the operation direction set in the RMV, regardless of the pulsar signal's forward or backward rotation)
When the value in the positioning counter reaches zero, movement on the axis will stop. Set the RMV register value to zero and start the positioning operation. The LSI will stop movement on the axis immediately, without outputting any command pulses.

9-3-3. Command position zero return operation using a pulsar input (MOD: 54h)

This mode is used to feed the axis using a pulsar input until the value in COUNTER1 (command position) becomes zero. The number of pulses output and the feed direction are set automatically by internal calculation, using the COUNTER1 value when starting.

Set the COUNTER1 value to zero and start the positioning operation, the LSI will stop movement on the axis immediately, without outputting any command pulses.

9-3-4. Mechanical position zero return operation using a pulsar input (MOD: 55h)

This mode is used to feed the axis using pulsar input until the value in COUNTER2 (mechanical position) becomes zero. The number of pulses output and the feed direction are set automatically by internal calculation, using the COUNTER2 value when starting.

Set the COUNTER2 value to zero and start the positioning operation. The LSI will stop movement on the axis immediately, without outputting any command pulses.

9-4. External switch (±DR) operation mode

This mode allows operations with inputs from an external switch.
To enable inputs from an external switch, bring the $\overline{PE}$ terminal LOW.
After writing a start command, when a +DR/-DR signal is input, the LSI will output pulses to the OUT terminal.
The following two external switch operation modes are available.

- ♦ Continuous operation using an external switch (MOD: 02h)
- ♦ Positioning operations using an external switch (MOD: 56h)

Set the RENVI (environment 1) register to specify the output logic of the ±DR input signal. The $\overline{INT}$ signal can be set to send an output when ±DR input is changed.
The RSTS (extension status) register can be used to check the operating status and monitor the ±DR input.
It is also possible to apply a filter to the ±DR or $\overline{PE}$ inputs.

Set the input logic of the +DR/-DR signals <Set DRL (bit 25) in RENV1 > 0: Negative logic 1: Positive logic	[RENV1] (WRITE) 31 24 - - - - - - n -
Applying a ±DR or $\overline{PE}$ input filter <Set DRF (bit 27) in RENV1> 1: Apply a filter to ±DR input or $\overline{PE}$ inputs When a filter is applied, pulses shorter than 32 ms will be ignored.	[RENV1] (WRITE) 31 24 - - - - n - - -
Setting an event interrupt cause <Set IRDR (bit 17) in RIRQ> 1: Output the $\overline{INT}$ signal when ±DR signal changed input.	[RIRQ] (WRITE) 23 16 0 0 0 0 0 - n -
Reading the event interrupt cause <ISPD (bit 17) and ISMD (bit 18) in RIST> ISPD(bit 17) = 1: When the +DR signal input changes. ISMD(bit 18) = 1: When the -DR signal input changes.	[RIST] (READ) 23 16 0 0 0 0 - n n -
Read operation status < CND (bits 0 to 3) in RSTS> 0001: Waiting for a DR input	[RSTS] (READ) 7 0 - - - - n n n n
Reading the ±DR signal <SDRP (bit 11) and SDRM (bit 12) in RSTS> SDRP = 0: +DR signal is OFF SDRP = 1: +DR signal is ON SDRM = 0: -DR signal is OFF SDRM = 1: -DR signal is ON	[RSTS] (READ) 15 8 - - - n n - - -

9-4-1. Continuous operation using an external switch (MOD: 02h)

This mode is used to operate an axis only when the DR switch is ON.
After writing a start command, turn the +DR signal ON to feed the axis in the positive direction, turn the –DR signal ON to feed the axis in the negative direction, using a specified speed pattern.
By turning ON an EL signal for the feed direction, movement on the axis will stop. However, the axis can be fed in the reverse direction.
An error interrupt ($\overline{INT}$ output) will not occur.
To end this operation mode, write an immediate stop command (49h).
If the axis is being fed with high speed commands (52h, 53h), movement on the axis will decelerate and stop when the DR input turns OFF. If the DR input for reverse direction turns ON while decelerating, movement on the axis will decelerate and stop. Then it will start in the opposite direction.

[Setting example]

1) Bring the $\overline{PE}$ input LOW.
2) Specify RFL, RFH, RUR, RDR, and RMG (speed setting).
3) Enter "0000010" for MOD (bits 0 to 6) in the RMD (operation mode) register
4) Write a start command (50h to 53h).

CND (bits 0 to 3) of the RSTS (extension status) register will wait for "0001: DR input."
In this condition, turn ON the +DR or -DR input terminal. The axis will move in the specified direction using the specified speed pattern as long as the terminal is kept ON.

9-4-2. Positioning operation using an external switch (MOD: 56h)

This mode is used for positioning based on the DR input rising timing.
When started, the data in the RMV register is loaded into the positioning counter. When the DR input is turned ON, the LSI will output pulses and the positioning counter will start counting down pulses. Even if the DR input is turned

OFF while feeding, the feeding operation will not be affected. When the positioning counter value becomes zero, the axis will stop operation. Make the REV register zero, and start the positioning operation. The axis will stop immediately, without outputting any command pulses.
Turn ON the +DR signal to feed in the positive direction. Turn ON the –DR signal to feed in the negative direction. By turning ON the EL signal corresponding to the feed direction, the axis will stop operation and issue an error interrupt ($\overline{\text{INT}}$ output).

9-5. Zero position operation mode

The following three zero position operation modes are available.

- Zero return operation <Positive direction MOD: 10h, Negative direction MOD: 18h>
- Leaving zero position operation <Positive direction MOD: 12h, Negative direction MOD: 1Ah>
- Zero position search operation<Positive direction MOD: 15h, Negative direction MOD: 1Dh>

Depending on the operation method, the zero position operation uses the ORG, EZ, or ±EL inputs.
Specify the input logic of the ORG input signal in the RENV1 (environment 1) register. This register's terminal status can be monitored with an SSTSW (sub status) command.
Specify the input logic of the EZ input signal in the RENV2 (environment 2) register. Specify the number for EZ to count up to for a zero return complete condition in the RENV3 (environment 3) register. This register's terminal status can be monitored by reading the RSTS (extension status) register.
Specify the logic for the ±EL input signal using the ELL input terminals. Specify the operation to execute when the signal turns ON (immediate stop/deceleration stop) in the RENV1 register. This register's terminal status can be monitored with an SSTSW (sub status) command.
An input filter can be applied to the ORG input signal and ±EL input signal by setting the RENV1 register.

Set the ORG signal input logic <Set ORGL (bit 7) in RENV1 > 0: Negative logic 1: Positive logic	[RENV1] (WRITE) 7 0 n - - - - - - -
Read the ORG signal <SORG (bit14) in SSTSW> 0: Turn OFF the ORG signal 1: Turn ON the ORG signal	[SSTSW] (READ) 15 8 - n - - - - - -
Set the EZ signal input logic <Set EZL (bit 23) in RENV2> 0: Falling edge 1: Rising edge	[RENV2] (WRITE) 23 16 n - - - - - - -
Set the EZ count <Set EZD0 to 3 (bits 4 to 7) in RENV3> Specify the number for EZ to count up to that will indicate a zero return completion. Enter the value (the count minus 1) in EZD0 to 3. Setting range: 0 to 15.	[RENV3] (WRITE) 7 0 n n n n - - - -
Read the EZ signal < SEZ (bit 10) in RSTS> 0: Turn OFF the EZ signal 1: Turn ON the EZ signal	[RSTS] (READ) 15 8 - - - - - n - -
Set the ±EL signal input logic <ELL input terminal> L: Positive logic input H: Negative logic input	
Specify a method for stopping when the ±EL signal turns ON <Set ELM (bit 3) in RENV1 > 0: Immediate stop when the ±EL signal turns ON. 1: Deceleration stop when the ±EL signal turns ON.	[RENV1] (WRITE) 7 0 - - - - n - - -
Read the ±EL signal <SPEL (bit 12), SMEL (bit 13) in SSTSW> SPEL = 0: Turn OFF + EL signal SPEL = 1: Turn ON + EL signal SMEL = 0: Turn OFF - EL signal SMEL = 1: Turn ON - EL signal	[SSTSW] (READ) 15 8 - - n n - - - -
Applying an input filter to the ±EL and ORG inputs <Set FLTR (bit 26) in RENV1> 1: Apply a filter to the ±EL and ORG inputs. By applying a filter, pulses shorter than 4 μs will be ignored.	[RENV1] (WRITE) 27 20 - n - - - - - -

9-5-1. Zero return operation

After writing a start command, the axis will continue feeding until the conditions for a zero return complete are satisfied.

MOD: 10h Positive direction zero return operation
18h Negative direction zero return operation

When a zero return is complete, the LSI will reset the counter and output an ERC (deflection counter clear) signal. The RENV3 register is used to set the basic zero return method. That is, whether or not to reset the counter when the zero return is complete. Specify whether or not to output the ERC signal in the RENV1 register.

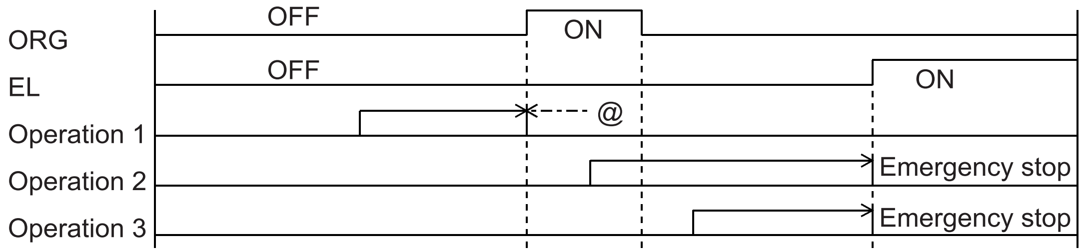
For details about the ERC signal, see 11-6-2, "ERC signal."

Set the zero return method <Set ORM0 to 3 (bits 0 to 3) in RENV3> 0000: Zero return operation 0 - Stop immediately (deceleration stop when feeding at high speed) when the ORG signal turns ON - COUNTER reset timing: When the ORG input signal turns ON. 0001: Zero return operation 1 - Stop immediately (deceleration stop when feeding at high speed) when the ORG signal turns ON. Next, feed in the reverse direction at RFA low speed until the ORG signal turns OFF. Then, the axis moves back in the original direction at RFA speed and stops immediately when ORG turns ON again. - COUNTER reset timing: When the ORG input signal turns ON. 0010: Zero return operation 2 - When feeding at low speed, after the ORG signal turns ON, movement on the axis stops immediately when the EZ counter finishes counting up. When feeding at high speed, after the ORG signal turns ON, the axis decelerates and stops immediately when the EZ counter finishes counting up. - COUNTER reset timing: When the EZ counter finishes counting up. 0011: Zero return operation 3 - When feeding at low speed, after the ORG signal turns ON, movement on the axis stops immediately when the EZ counter finishes counting up. When feeding at high speed, after the ORG signal turns ON, the axis decelerates and stops immediately when the EZ counter finishes counting up. - COUNTER reset timing: When the EZ counter finishes counting up. 0100: Zero return operation 4 - Movement on the axis stops immediately (decelerate and stop when feeding at high speed) when the ORG input is turned ON. Next, the direction of movement is reversed at RFA low speed. Then, it stops immediately when the EZ counter finishes counting up. - COUNTER reset timing: When the EZ counter finishes counting up. 0101: Zero return operation 5 - Movement on the axis stops immediately and is reversed (decelerates and stops when feeding at high speed) when the ORG input is turned ON. Then, all movement stops immediately (decelerates and stops when feeding at high speed) when the EZ counter finishes counting up. - COUNTER reset timing: When the EZ counter finishes counting up. 0110: Zero return operation 6 - Movement on the axis stops immediately (decelerates and stops when ELM is 1) when the EL signal turns ON, and it reverses at RFA low speed. Then, all movement stops immediately when the EL signal is turned OFF.	[RENV3] (WRITE) 7 0 - - - - n n n n

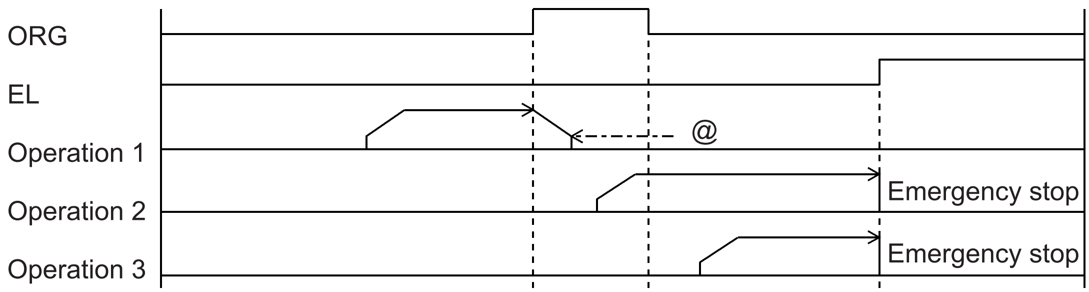
- COUNTER reset timing: When the EL signal is turned OFF. 0111: Zero return operation 7 - Movement on the axis stops immediately (decelerates and stops when ELM is 1) when the EL signal turns ON, and reverses at RFA low speed. Then, all movement stops immediately when the EZ counter finishes counting up. - COUNTER reset timing: When the EZ counter finishes counting up. 1000: Zero return operation 8 Movement on the axis stops immediately (decelerates and stops when ELM is 1) when the EL signal turns ON, and reverses. Then it stops immediately (decelerates and stops when feed at high speed) when the EZ counter finishes counting up. - COUNTER reset timing: When the EZ counter finishes counting up. 1001: Zero return operation 9 - After the process in zero return operation 0 has executed, it returns to zero (operates until COUNTER2 = 0). 1010: Zero return operation 10 - After the process in zero return operation 3 has executed, it returns to zero (operates until COUNTER2 = 0). 1011: Zero return operation 11 - After the process in zero return operation 5 has executed, it returns to zero (operates until COUNTER2 = 0). 1100: Zero return operation 12 - After the process in zero return operation 8 has executed, it returns to zero (operates until COUNTER2 = 0).	
Settings after a zero return complete <Set CU1R to 4R (bits 20 to 23) in RENV3> CU1R (bit 20) =1: Reset COUNTER1 (command position) CU2R (bit 21) =1: Reset COUNTER2 (mechanical position) CU3R (bit 22) =1: Reset COUNTER3 (deflection counter) CU4R (bit 23) =1: Reset COUNTER4 (general-purpose)	[RENV3] (WRITE) 23 16 n n n n - - - -
Setting the ERC signal for automatic output <Set EROR (bit 11) in RENV1> 0: Does not output an ERC signal when a zero return is complete. 1: Automatically outputs an ERC signal when a zero return is complete.	[RENV1] (WRITE) 15 8 - - - - n - - -

9-5-1-1. Zero return operation 0 (ORM = 0000)

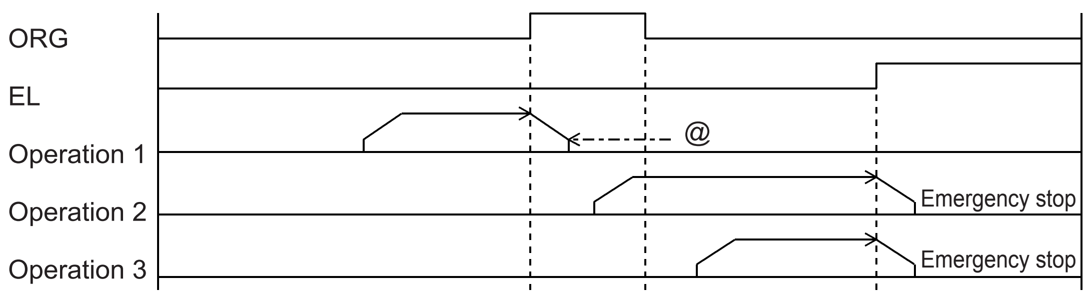
□ Low speed operation <Sensor: EL (ELM = 0), ORG>



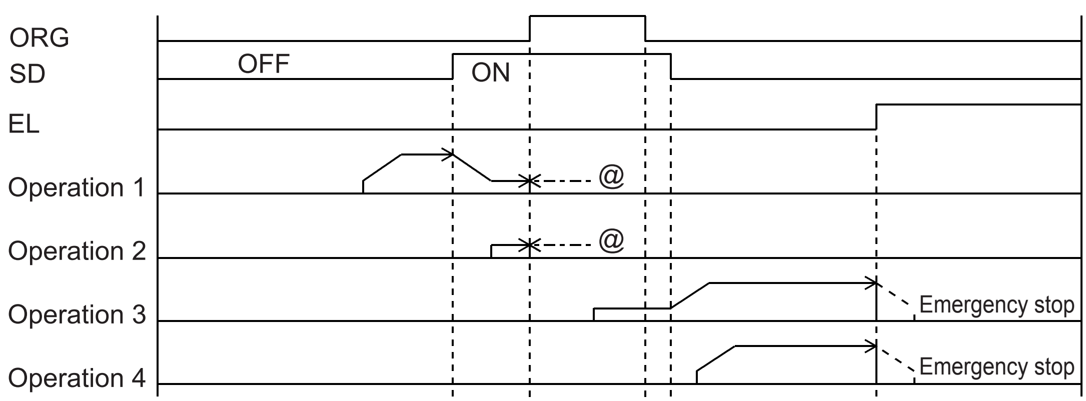
■ High speed operation <Sensor: EL (ELM = 0), ORG>
Even if the axis stops normally, it may not be at the zero position. However, COUNTER2 (mechanical position) provides a reliable value.



■ High speed operation <Sensor: EL (ELM = 1), ORG>
Even if the axis stops normally, it may not be at the zero position. However, COUNTER2 (mechanical position) provides a reliable value.



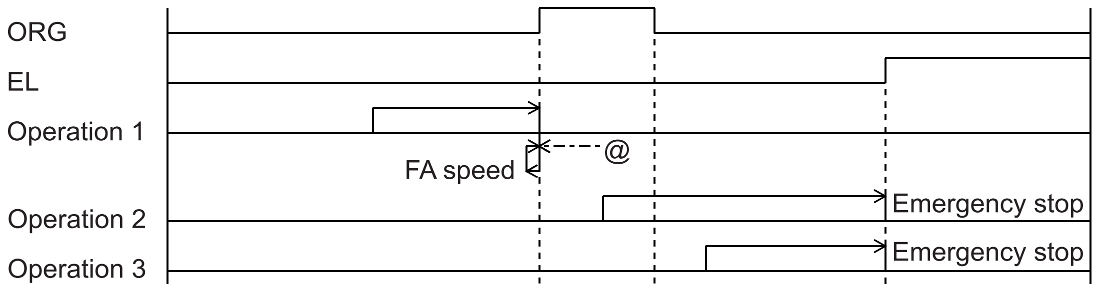
■ High speed operation <Sensor: EL (ELM = 1), SD (SDM = 0, SDLT = 0), ORG>



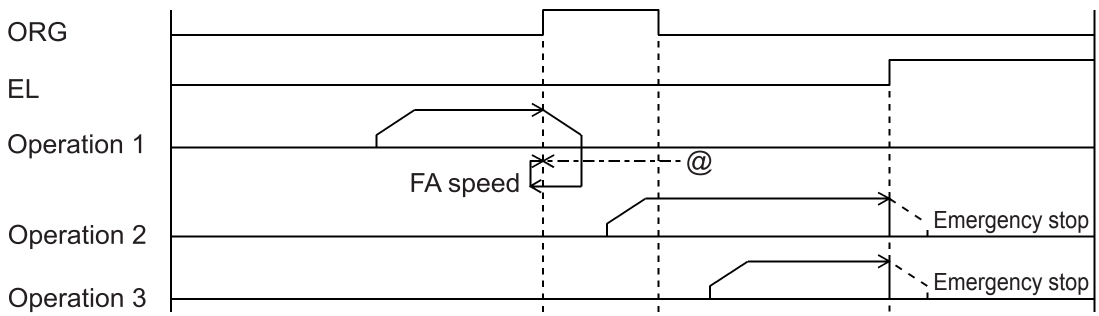
Note: Positions marked with "@" reflect the ERC signal output timing when "Automatically output an ERC signal" is selected for the zero stopping position.
Deceleration stops (dotted lines) using the EL signal are timed when ELM is set to 1 in the RENV1 register.

9-5-1-2. Zero return operation 1 (ORM=0001)

□ Low speed operation <Sensor: EL (ELM = 0), ORG>

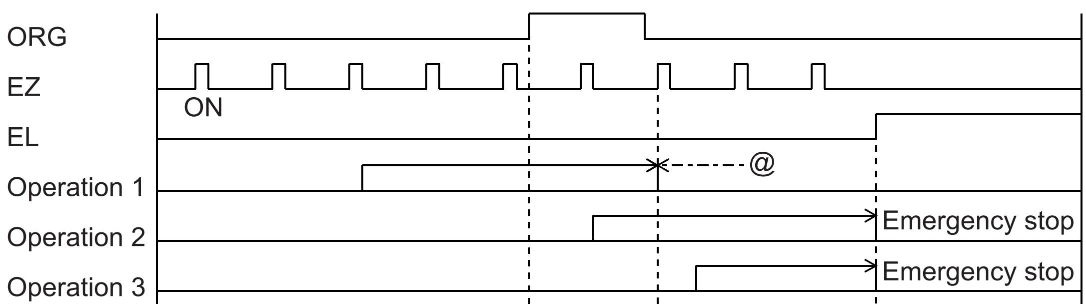


■ High speed operation <Sensor: EL, ORG>

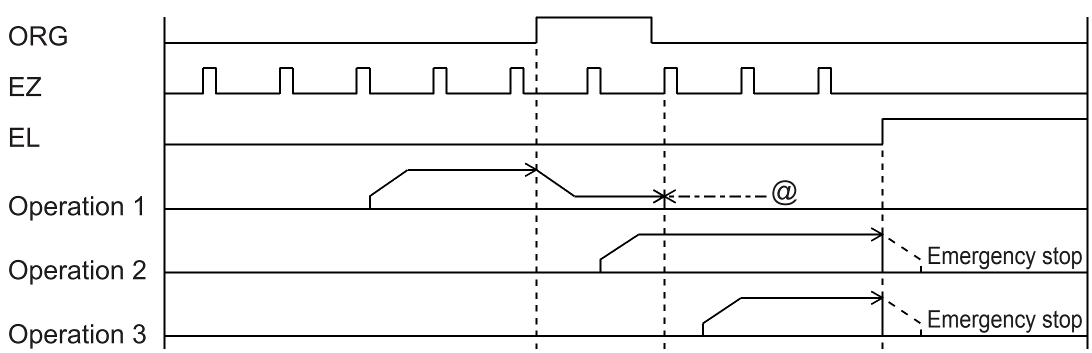


9-5-1-3. Zero return operation 2 (ORM = 0010)

□ Low speed operation <Sensor: EL (ELM = 0), ORG, EZ (EZD = 0001)>



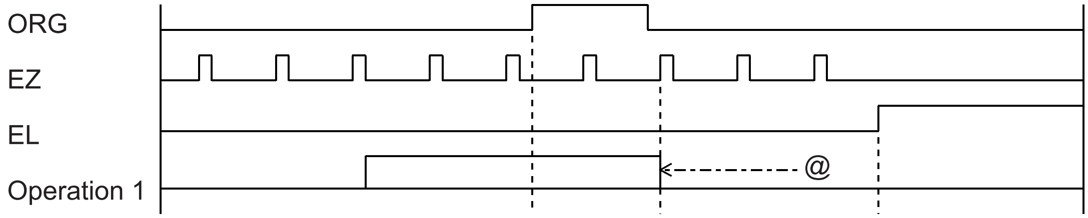
■ High speed operation <Sensor: EL, ORG, EZ (EZD = 0001)>



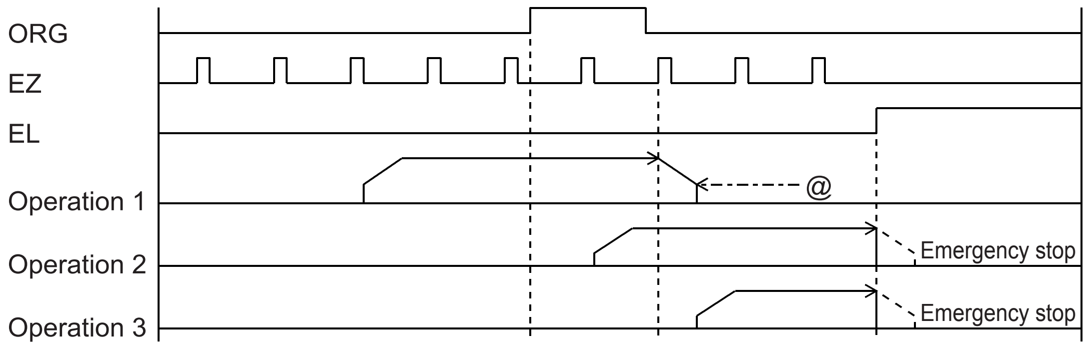
Note: Positions marked with "@" reflect ERC signal output timing when "Automatically output an ERC signal" is selected for the zero stopping position.
Deceleration stops (dotted lines) using the EL signal are timed when ELM is set to 1 in the RENV1 register.

9-5-1-4. Zero return operation 3 (ORM = 0011)

□ Low speed operation <Sensor: EL, ORG, EZ (EZD = 0001)>

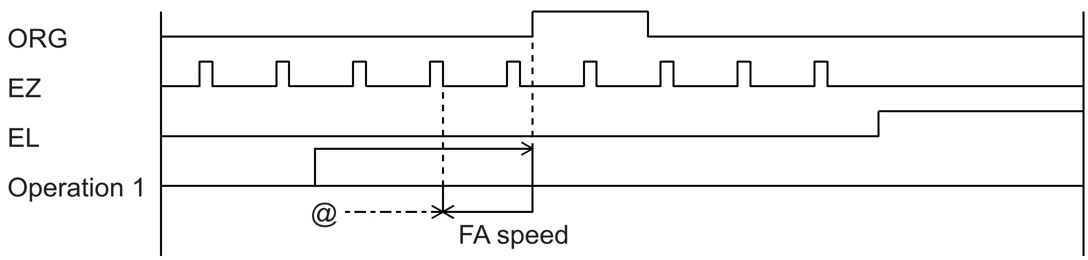


■ High speed operation <Sensor: EL,ORG, EZ (EZD = 0001)>

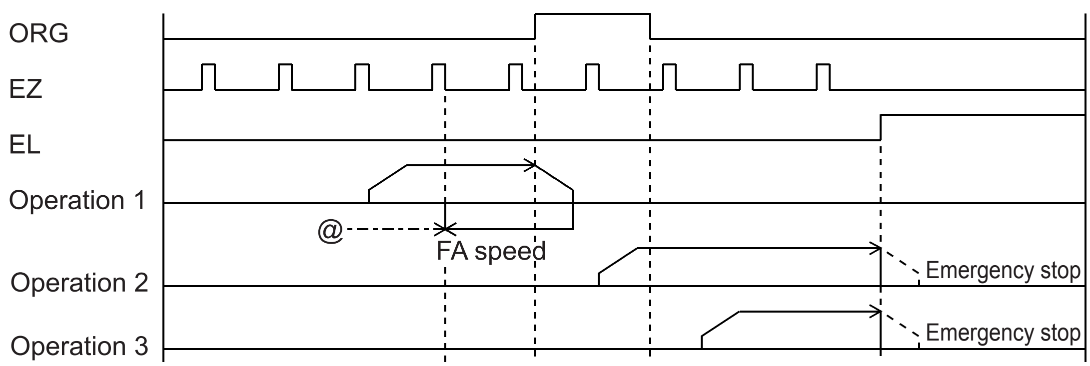


9-5-1-5. Zero return operation 4 (ORM = 0100)

□ Low speed operation <Sensor: EL, ORG, EZ (EZD = 0001)>



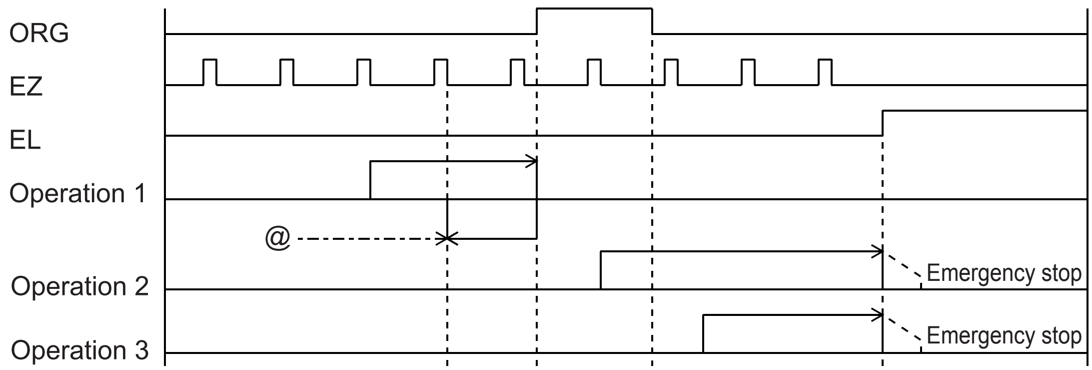
■ High speed operation <Sensor: EL, ORG, EZ (EZD = 0001)>



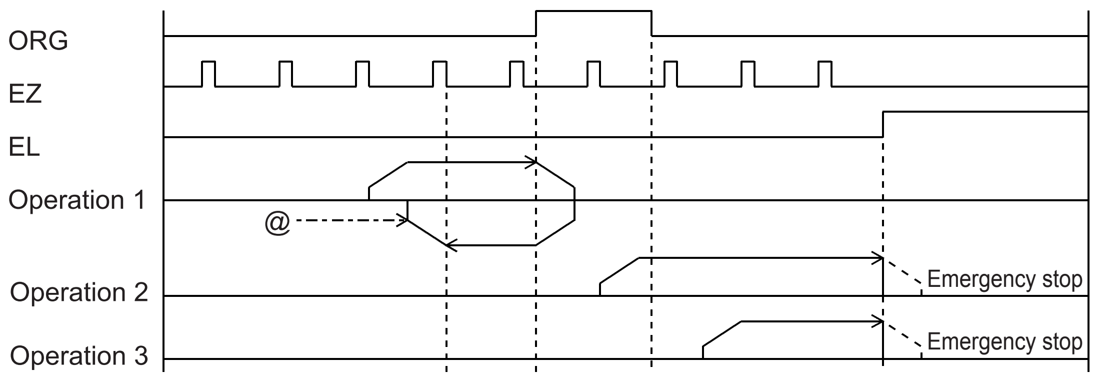
Note: Positions marked with "@" reflect the ERC signal output timing when "Automatically output an ERC signal" is selected for the zero stopping position.
Deceleration stops (dotted lines) using the EL signal are timed when ELM is set to 1 in the RENV1 register.

9-5-1-6. Zero return operation 5 (ORM = 0101)

□ Low speed operation <Sensor: EL, ORG, EZ (EZD = 0001)>

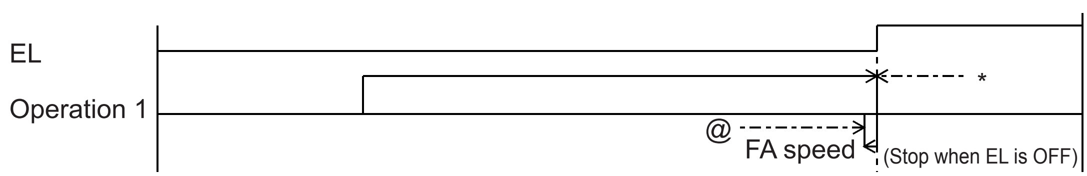


■ High speed operation <Sensor: EL, ORG, EZ (EZD = 0001)>

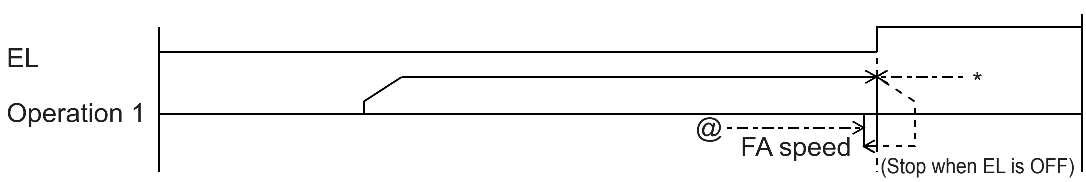


9-5-1-7. Zero return operation 6 (ORM = 0110)

□ Low speed operation <Sensor: EL>



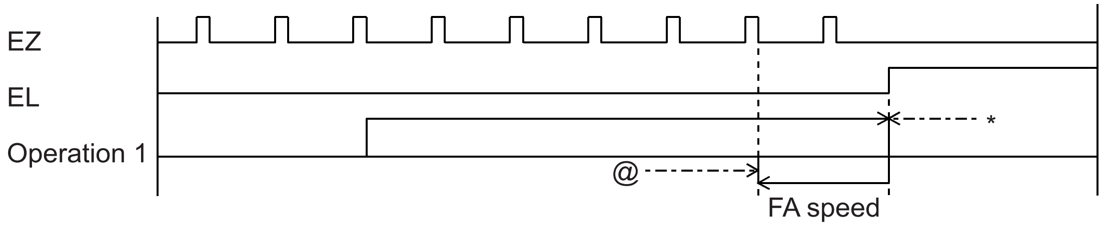
■ High speed operation <Sensor: EL>



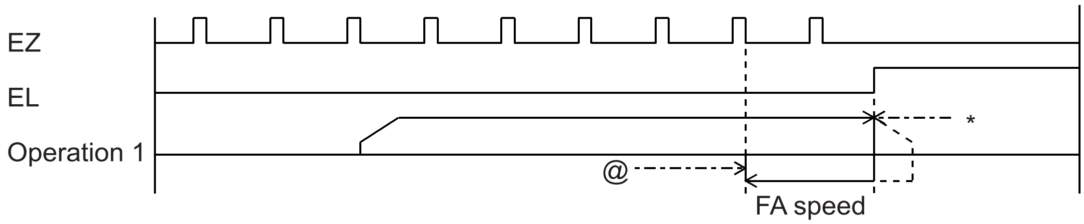
Note: Positions marked with "@" reflect the ERC signal output timing when "Automatically output an ERC signal" is selected for the zero stopping position.
Also, when EROE (bit 10) is 1 in the RENV1 register and ELM (bit 3) is 0, the LSI will output an ERC signal at positions marked with an asterisk (*).
Deceleration stops (dotted lines) using the EL signal are timed when ELM is set to 1 in the RENV1 register.

9-5-1-8. Zero return operation 7 (ORM = 0111)

□ Low speed operation <Sensor: EL, EZ (EZD = 0001)>

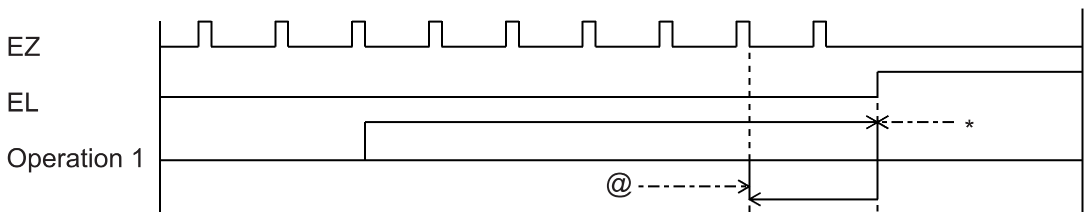


■ High speed operation <Sensor: EL, EZ (EZD = 0001)>

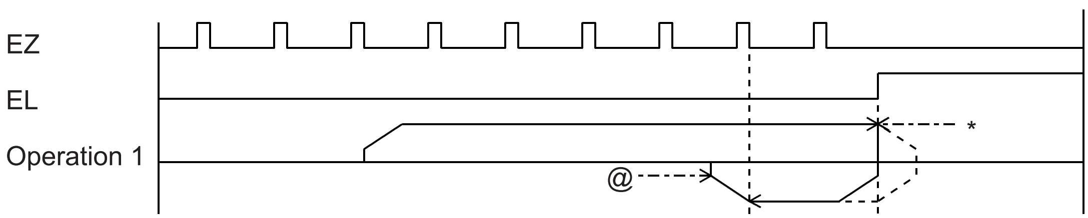


9-5-1-9. Zero return operation 8 (ORM=1000)

□ Low speed operation <Sensor: EL, EZ (EZD = 0001)>

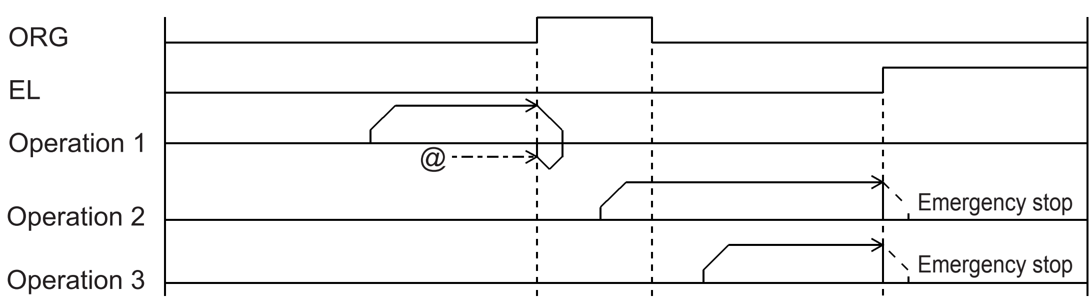


■ High speed operation <Sensor: EL, EZ (EZD = 0001)>



9-5-1-10. Zero return operation 9 (ORM = 1001)

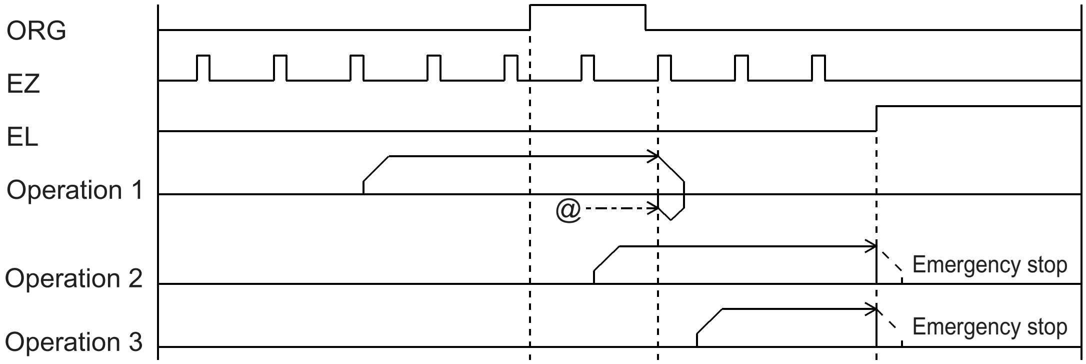
■ High speed operation <Sensor: EL, ORG)>



Note: Positions marked with "@" reflect the ERC signal output timing when "Automatically output an ERC signal" is selected for the zero stopping position.
Also, when EROE (bit 10) is 1 in the RENV1 register and ELM (bit 3) is 0, the LSI will output an ERC signal at positions marked with an asterisk (*).
Deceleration stops (dotted lines) using the EL signal are timed when ELM is set to 1 in the RENV1 register.

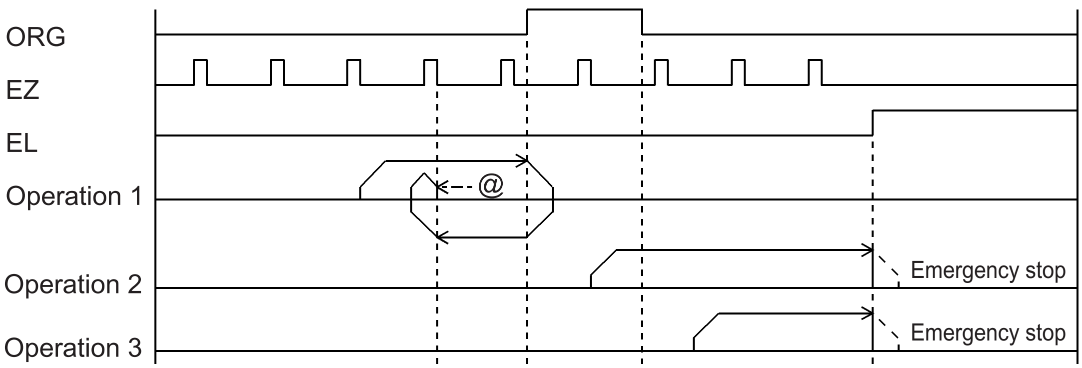
9-5-1-11. Zero return operation 10 (ORM = 1010)

■ High speed operation <Sensor: EL, ORG, EZ (EZD = 0001)>



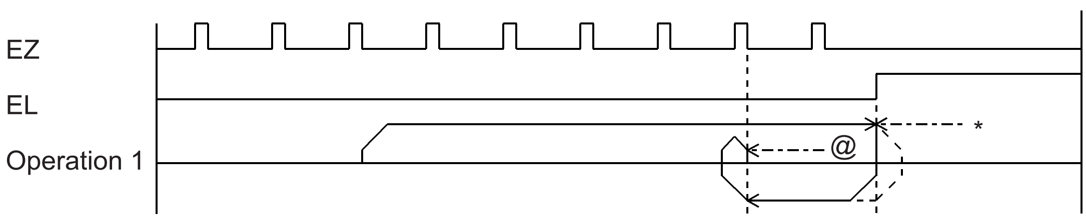
9-5-1-12. Zero return operation 11 (ORM = 1011)

■ High speed operation <Sensor: EL, ORG, EZ (EZD = 0001)>



9-5-1-13. Zero return operation 12 (ORM = 1100)

■ High speed operation <Sensor: EL, EZ (EZD = 0001)>



Note: Positions marked with "@" reflect the ERC signal output timing when "Automatically output an ERC signal" is selected for the zero stopping position.
Also, when EROE (bit 10) is 1 in the RENV1 register and ELM (bit 3) is 0, the LSI will output an ERC signal at positions marked with an asterisk (*).
Deceleration stops (dotted lines) using the EL signal are timed when ELM is set to 1 in the RENV1 register.

9-5-2. Leaving the zero position operations

After writing a start command, the axis will leave the zero position (when the ORG input turns ON).

<u>Make sure to use the "Low speed start command" when leaving the zero position.</u>

When you write a start command while the ORG input is OFF, the LSI will stop the movement on the axis as a normal stop, without outputting pulses.

If the axis is started at low speed while the ORG input is ON, when the ORG input turns OFF the LSI will stop movement on the axis immediately, after outputting one pulse. (Normal stop)

MOD: 12h Leave the zero position in the positive direction

1Ah Leave the zero position in the negative direction

9-5-3. Zero search operation

This mode is used to add functions to a zero return operation. It consists of the following possibilities.

1) A "Zero return operation" is made in the opposite direction to the one specified.
2) A "Leaving the zero position using positioning operations" is executed in the opposite direction to the one specified.
3) A "Zero return operation" is executed in the specified direction.

Operation 1: If the ORG input is turned ON after starting, movement on the axis will stop normally.

Operation 2: If the ORG input is already turned ON when starting, the axis will leave the zero position using positioning operations, and then begin a "zero return operation."

Operation 3: If movement on the axis is stopped by an EL signal while operating in the specified direction, the axis will execute a "zero return operation (ORM = 0000)" and a "leaving the zero position by positioning" in the opposite direction. Then it will execute a "zero return operation" in the specified direction.

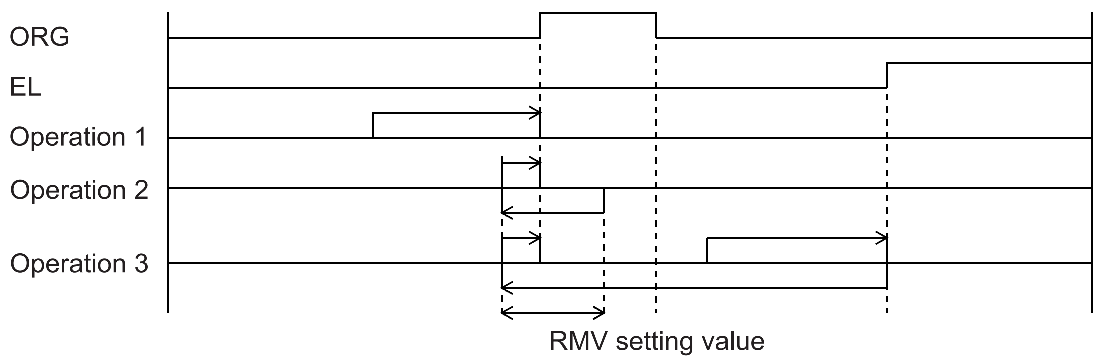
When "leaving the zero position by positioning," the axis will repeat the positioning operation for the number of pulses specified in the RMV (target position) register, until the zero position has been left. Enter a positive number (1 to 134,217,727) in the RMV register.

MOD: 15h Zero search operation in the positive direction

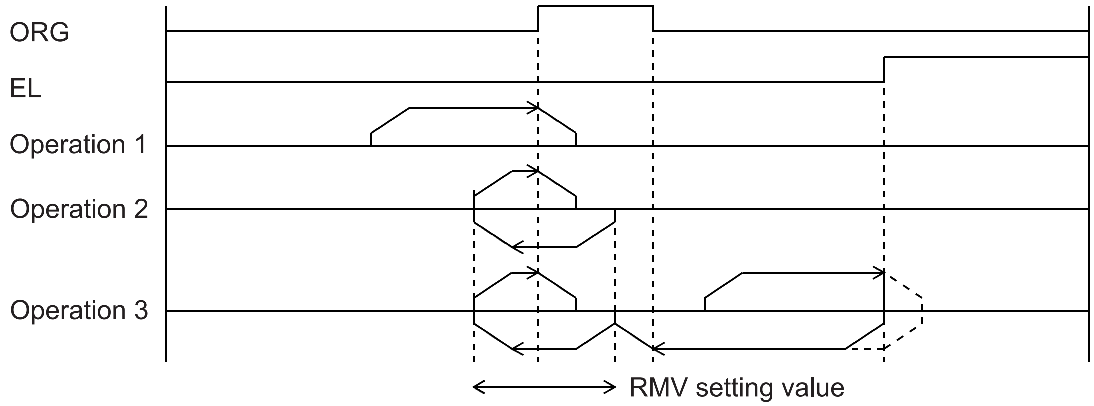
1Dh Zero search operation in the negative direction

9-5-3-1. Zero return operation 0 (ORM=0000)

□ Low speed operation <Sensor: EL, ORG>



■ High speed operation <Sensor: EL, ORG>
Even if the axis stops normally, it may not be at the zero position. However, COUNTER2 (mechanical position) provides a reliable value.



9-6. EL or SL operation mode

The following three modes of EL or SL (soft limit) operation are available.

- ♦ Move to an EL or SL position <Positive direction MOD:20h, Negative direction MOD:28h>
- ♦ Leave an EL or SL position <Positive direction MOD:22h, Negative direction MOD:2Ah>

To specify the ±EL input signal, set the input logic using the ELL input terminal. Select the operation type (immediate stop / deceleration stop) when the input from that terminal is ON in the RENV1 (Environment setting 1) register. The status of the terminal can be monitored using the SSTSW (sub status) register.
For details about setting the SL (software limit), see section 11-11-2, "Software limit function."

Select the ±EL signal input logic <ELL input terminal> L: Positive logic input H: Negative logic input	
Select the stop method to use when the ±EL signal is turned ON <Set ELM (bit 3) in RENV1> 0: Stop immediately when the ±EL signal turns ON. 1: Decelerates and stops when the ±EL signal turns ON.	[RENV1] (WRITE) 7 0 - - - - n - - -
Reading the ±EL signal <SPEL (bit 12), SMEL (bit 13) in SSTSW> SPEL=0: Turn OFF +EL signal SPEL=1: Turn ON +EL signal SMEL=0: Turn OFF -EL signal SMEL=1: Turn ON -EL signal	[SSTSW] (READ) 15 8 - - n n - - - -
Setting the ±EL input filter <Set the FLTR (bit 26) in RENV1 > 1: Apply a filter to the ±EL input. After applying a filter, signals shorter than 4 μs will be ignored.	[RENV1] (WRITE) 27 20 - n - - - - - -

9-6-1. Feed until reaching an EL or SL position

This mode is used to continue feeding until the EL or SL (soft limit) signal is turned ON and then the operation stops normally.

When a start command is written on the position where the EL or SL signal is turned ON, the LSI will not output pulses and it will stop the axis normally. When a start command is written to the axis while the EL and SL signals are OFF, the axis will stop when the EL or SL signal is turned ON. (Normal stop.)

MOD: 20h Feed until reaching the +EL or +SL position.
28h Feed until reaching the -EL or -SL position.

9-6-2. Leaving an EL or SL position

This mode is used to continue feeding until the EL or SL (software limit) signal is turned OFF.

When a start command is written on the position where the EL and SL signals are turned OFF, the LSI will not output pulses and it will stop the axis normally.

When a start command is written to the axis while the EL or SL signal is ON, the axis will stop when the EL signal (or SL signal) is turned OFF.

MOD: 22h Leave from a -EL or -SL position
2Ah Leave from a + EL or +SL position

9-7. EZ count operation mode

This mode is used to count EZ signal of the number (EZD set value +1) written into the RENV3 register.

MOD: 24h Feed until the EZ count is complete in positive direction.
2Ch Feed until the EZ count is complete in negative direction.

After a start command is written, the axis stops immediately (or decelerates and stops when feeding at high speed) after the EZ count equals the number stored in the register.

The EZ count can be set from 1 to 16.

Use the low speed start command (50h, 51h) for this operation. When the high speed start command is used, the axis will start decelerating and stop when the EZ signal turns ON, so that the motion on the axis overruns the EZ position.

Specify logical input for the EZ signal in the RENV2 (environment setting 2) register, and the EZ number to count to in the RENV3 (environment setting 3) register. The terminal status can be monitored by reading the RSTS (extension status) register.

Setting the input logic of the EZ signal <Set EZL (bit 23) in RENV2> 0: Falling edge 1: Rising edge	[RENV2] (WRITE) 23 16 n - - - - - - -
Setting the EZ count number <Set EZD0 to 3 (bits 4 to 7) in RENV3> Specify the EZ count number after a zero return complete condition. Enter a value (the number to count to minus 1) in EZD 0 to 3. Setting range: 0 to 15.	[RENV3] (WRITE) 7 0 n n n n - - - -
Reading the EZ signal < SEZ (bit 10) in RSTS> 0: Turn OFF the EZ signal 1: Turn ON the EZ signal	[RSTS] (READ) 15 8 - - - - - n - -

9-8. Interpolation operations

9-8-1.Interpolation operations

In addition to single operations for each axis, this LSI can control the following interpolation operations:

- ♦ Two-axis continuous linear interpolation 1 (MOD: 60h)
- ♦ Two-axis linear interpolation 1 (MOD: 61h)
- ♦ 1 to 2 axes continuous linear interpolation 1 (MOD: 62h)
- ♦ 1 to 2 axes linear interpolation 2 (MOD: 63h)
- ♦ Two-axis CW/CCW circular interpolation (MOD: 64h/65h)
- ♦ Linear interpolation 1, using PA/PB inputs (MOD: 69h)
- ♦ Circular interpolation, using PA/PB inputs (MOD: 6Ch/6Dh)

Continuous linear interpolation is the same as the linear interpolation used to feed multiple axes at specified rates, and to start and stop feeding using commands such as the continuous mode commands.
Linear interpolation 1 is used to interpolate between the X and Y axes.
Linear interpolation 2 is used to control three axes or more using more than one LSI, and to control feeding using linear interpolation.
The interpolation settings and operation status can be monitored by reading the RIPS (interpolation status) register.
The RIPS register is shared by the X and Y axes. Reading from any axis will return the identical information.
Write start and stop commands to both axes by setting SELx and SELy in COMB1.

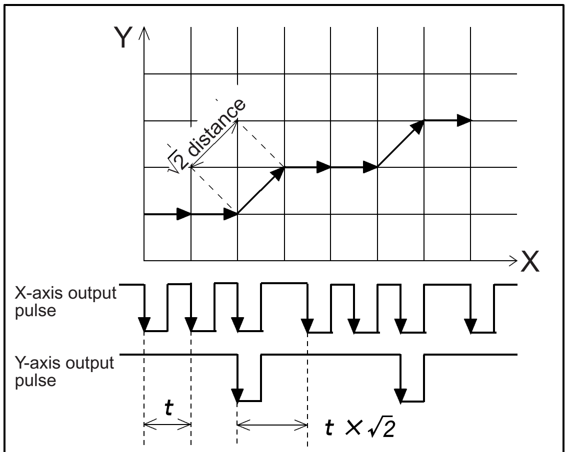
In Linear interpolation 1 (MOD = 69h), CW circular interpolation (MOD = 6Ch), and CCW circular interpolation (MOD = 6Dh) using a PA/PB input, the LSI provides interpolated operations that the PA/PB input synchronizes with an external signal. In this case, the feed direction of the PA/PB inputs are ignored, and direction of operation specified in the operation mode and the feed amount setting will be used.

9-8-2. Interpolation control axis

In Circular interpolation and Linear interpolation 1, specify the speed for one axis only. This axis is referred to as the interpolation control axis. The X axis is usually used as the interpolation control axis.
SRUN, SEND, and SERR in MSTSW (main status) will change identically for the axes being interpolated.
In linear interpolation 2, either axis can be the interpolation control axis.
RSPD (speed monitor) is only effective for the X axis. However, when Linear interpolation 2 is used, the value read out will be the speed of the master axis.

9-8-3. Constant synthetic speed control

Constant synthetic speed control is a function used to maintain a constant synthetic speed on the axes during Linear interpolation1 and Circular interpolation operations. Linear interpolation 2 cannot use the constant synthetic speed control function.
To enable this function, set MIPF (bit 15) in RMD (operation mode register) to 1 for the desired axis, to enable constant synthetic speed control.
If MIPF on both the X and Y axes is set to 1 and both axes are supplied pulses simultaneously, the LSI will multiply the interval until next pulse output by the square root of 2.
When the constant synthetic speed control function is enabled (MIPF is set to 1), the synthetic speed used for interpolation will be the operation speed (RFH) or initial speed (RFL) on the X axis.
The PCL6025 (this LSI) can even provide constant synthetic speed control for circular interpolations using acceleration and deceleration.



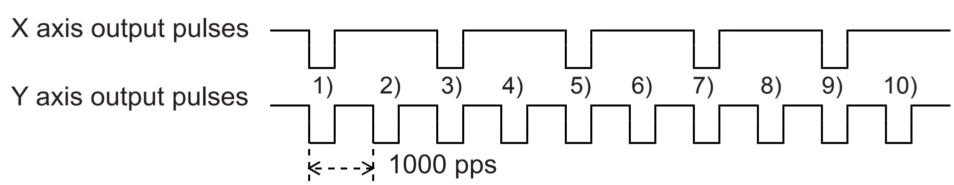
[An example of 2 axes interpolation] Figure 1

9-8-4. Linear interpolation 1

Linear interpolation 1 is used to allow a single LSI to provide interpolation operations between the X and Y axes. (MOD: 61h)
If only one axis is specified and operation is started, an error (ESDT: Stop due to operation data error) will occur.
After specifying the operation speed for the X axis, specify the use of constant synchronous speed control in the RMD register for both the X and Y axes, and specify the end point in the RMV register.
The direction of operation is determined by the sign of the value in the RMV register.
Automatically, the axis with the maximum feed amount (maximum absolute value in the RMV register) will be considered the master axis. The other axis will be the slave axis.
When a start command is written, the LSI will output pulses to the master axis and the slave axis will be supplied a smaller number of pulses than the master axis. Write a start command by setting either the SELx or SELy bits in COMB1 to 1. Either axis can be used to write a start command.
In Continuous linear interpolation 1 (MOD: 60h) pulses are output with the same timing as in Linear interpolation 1. However, the LSI will not stop the operation until a stop command is written.

[A setting example]
Set as shown in the table below and then write a start command (0351h). The LSI will output pulses using the timing shown in the figure below. Even if the vacant columns are set, the operation will not be affected.

Setting	X axis	Y axis
MOD	61h	61h
MIPF	0 (OFF)	0 (OFF)
RMV value	5	10
Operation speed	1000 pps	
Interpolation control axis	O	
Master axis / slave axis	Slave axis	Master axis



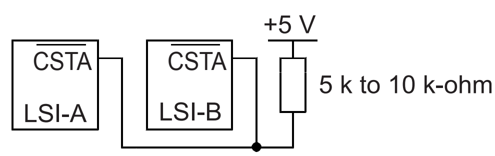
9-8-5. Linear interpolation 2

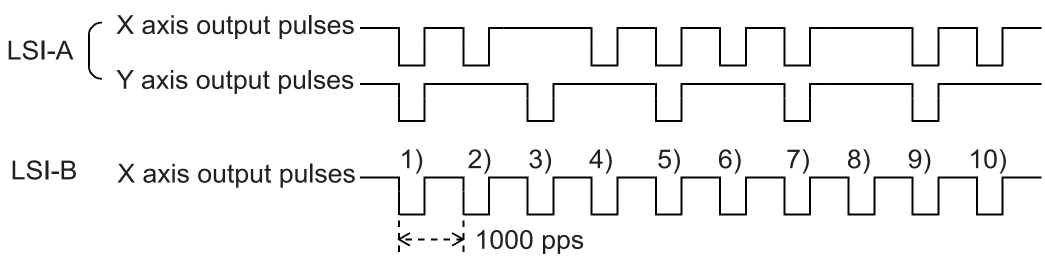
Linear interpolation 2 is used for linear interpolations between 3 or more axes and uses more than one LSI for control. (MOD: 63h)
In order to execute a linear interpolation using multiple LSIs, you must use a simultaneous start signal ($\overline{CSTA}$ signal).
For details about the $\overline{CSTA}$ signal, see section 11-7, "External start, simultaneous start."
The axis with the maximum amount to be fed is referred to as the master axis during the interpolation and the other axes are slave axes.
Enter the RMV register setting for the master axis in the RIP registers of each axis (including the master axis).
In the RMV registers of the slave axes, enter end point of each axis.
Specify the speed data (RFL, RFH, RUR, RDR, RMG, RDP, RUS, and RDS) for the slave axes to be the same as for the master axis.
The feed direction is determined by the sign of the value in the RMV register.
After writing "01" into MSY (bits 18 and 19) in the RMD (operation mode) register of the axes, write a start command and set the axes to wait for the $\overline{STA}$ signal input. By entering a $\overline{CSTA}$ signal, all of the axes on all of the LSIs will start at the same time.
The master axis provides pulses constantly. The slave axes provide some of the pulses fed to the master axis, but some are omitted.
Continuous interpolation 2 (MOD: 62h) outputs pulses using the same timing as Linear interpolation 2. However, Continuous linear interpolation 2 will not stop the operation until a stop command is written.
Continuous interpolation 2 cannot set an automatic ramping-down point.

[Setting example]
1) Connect the $\overline{CSTA}$ signals between LSI-A and LSI-B.
2) Set up the LSIs as shown below. (Set the RMD to start with an $\overline{STA}$ input.)
3) Write start commands (LSI-A: 0351h, LSI-B: 0151h).
4) Write a $\overline{CSTA}$ signal input command (06h) to the X axis on LSI-A.

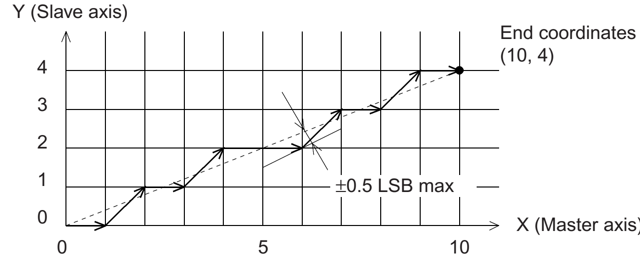
After completing steps 1) to 4) above, the LSIs will output pulses using the timing shown in the figure below.

Setting	LSI-A		LSI-B
	X axis	Y axis	X axis
RMD	00040063h	00040063h	00040063h
RMV value	8	5	10
RIP value	10	10	10
Operation speed	1000 pps	1000 pps	1000 pps
Master axis / slave axis	Slave axis	Slave axis	Master axis





[Precision of linear interpolation]
As shown in the figure on the right, linear interpolation executes an interpolation from the current coordinates to the end coordinates.
The positional precision of a specified line during linear interpolation will be ±0.5 LSB throughout the interpolation range.

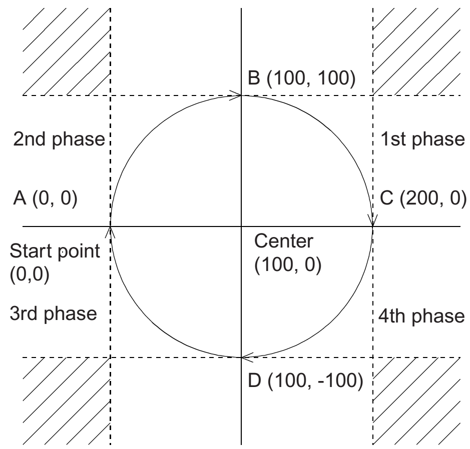


9-8-6. Circular interpolation

This function provides CW circular interpolation (MOD: 64h) and CCW circular interpolation (MOD: 65h) between the X and Y axes.
If only one axis is specified for circular interpolation and a start command is written, a data setting error will occur.
Circular interpolation takes the current position as the starting point (coordinate 0, 0) regardless of the values in the counters (COUNTER1 to 4).
After specifying the speed for each axis being interpolated, specify whether or not to apply constant synthetic speed control (MIPF in the RMD register), the end points (the RMV register value), and the center point (the RIP register value). If the end point is 0 (the starting point), both axes will draw a simple circle.
The synthetic speed used in the circular interpolation will be the speed set for the axes being interpolated (FH/FL) if the constant synthetic speed control is ON (MIPF = 1) for both axes.
Write a start command after setting SELx and SELy in COMB1 to 1. Either axis can be used to write a start command.

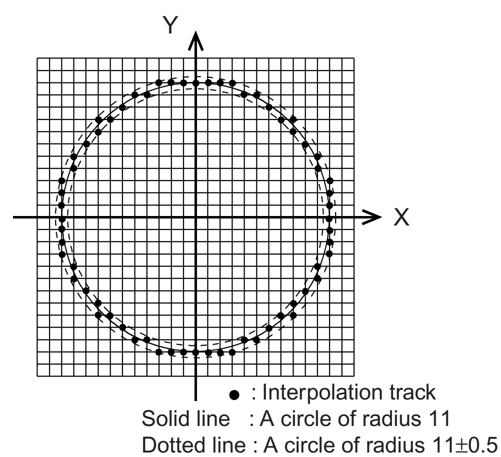
[Setting example]
As shown in the table below, specify the MOD, MIPF, RMV, RIP and operation speed for each axis and write a start command (ex. 0351h) that will be used by both axes. The axes will move as shown on the right.

Set value \ Step No.	A		B		C		D	
	X axis	Y axis	X axis	Y axis	X axis	Y axis	X axis	Y axis
MOD	64h (CW circular interpolation)							
MIPF	1 (turn ON constant synthetic speed control)							
RMV value	0	0	100	100	200	0	100	-100
RIP value	100	0	100	0	100	0	100	0
Operation result	Simple circle		90° arc		180° arc		270° arc	



The circular interpolation terminates when either of the two axes reaches the end point in the last phase. The position at which the interpolation ends will not be the target point. After the circular interpolation has terminated, the circular interpolation function executes a "Linear interpolation 1" (MOD = 61h) using the "Remaining pulses start command," and the axes will move from the final circular interpolation position to the target coordinate position (the final drawing point in the operation).
By setting MPIE to 1 in the RMD register, after the circular interpolation has finished the axes will automatically start moving until they use up the number of left over pulses, using the Linear interpolation 1 function. (Automatically draw end point function.)
If the end point of the circular interpolation is set within the shaded areas, the axes will not stop moving (perpetual circular motion).

[Circular interpolation precision]
The circular interpolation function draws an circular from the current position to the end coordinate moving CW or CCW.
The positional deviation from the specified curve is ±0.5 LSB.
The figure on the right is an example of how to draw a simple circle with a radius of 11 units.



9-8-7. Operation during interpolation

- Acceleration/deceleration operations

Acceleration and deceleration (linear and S-curve) can be used with Linear interpolation 1 and circular interpolation operations.
In order to set a ramping-down point automatically, give RUR and RDR (accel/decel rate) and RUS and RDS (S-curve range at accel/decel) the same values.
When the synthetic constant speed control is applied, there will be a difference between the specified speed and actual operating speed, which means that the ramping-down point cannot be calculated manually. Therefore, the ramping-down point must be set for automatic calculation.
In order to use accel/decel for a circular interpolation operation, the total number of steps in the circular interpolation (incrementing only the X axis, only the Y axis only, or both the X and Y axes) must be put into RCI (circular interpolation number of steps register).
For a circle of 360°, a rough calculation of the value to put in the RCI will be as follows:

$$RCI \equiv \sqrt{RIPx^2 + RIPy^2} \times \sqrt{2} \times 4$$ (RIP = Circular interpolation center position register = radius)

When an circular has an angle "n", a rough calculation can be obtained by RCI x (n / 360)

Due to limitations of circular interpolation, there will be some deviation from the calculated value (such as when one axis reaches its end point, the interpolation operation will terminate). If a precise number of pulses is required for RCI, then you must actually operate the axes and count the number of steps. The LSI counts up "1" when either of the two axes output a pulse. It also counts up "1" when the both axes output a pulse. Therefore, the actual number of steps will be ORed result of the total number of X and Y axes pulse that are output.

- Error stop

If any of the axes stops with an error, all of the axes being interpolated will stop (SERR = 1). By reading the REST (error stop cause) register, you can determine which axis actually stopped with an error.

- SD input

When SD input is enabled (MSDE (bit 8) in the RMD register is set to 1), and if the SD input turns ON on either of the axes, both axes will decelerate or decelerate and stop.
<u>Note: In Linear interpolation 2, the axes on other LSIs will not decelerate.</u>

- Idling control

If any axis is in idling range, none of the axes being interpolated will accelerate.
<u>Note: In Linear interpolation 2, axes controlled by other LSIs will be accelerated, even when using the idling control.</u>

- Correction function

When phases are changed during circular interpolation, backlash correction and slip correction control cannot be used.

- Continuous interpolation

You can execute continuous interpolation using the pre-register.
Continuous interpolation is used to continue a linear interpolation or a circular interpolation.
An example of the settings for continuous interpolation using the pre-register is shown in section 11-14-1, "Start triggered by a stop on another axis."

- Automatically drawn end point operation

In circular interpolation, when any of the axes reaches the final coordinate in the last phase, the circular interpolation is terminated. The end coordinate will not usually match the final (target) coordinate, except when drawing a simple circle. An operation to move from the final coordinate to the target coordinate is referred to as a "drawing the end point operation."
In order to automatically draw the end point, set MPIE (bit 27) in PRMD for both axes to 1. This setting will cause the axes to move to the target coordinate after completing the circular interpolation. (Automatically drawn end point function)
The automatically drawn end point function starts with the remaining number of pulses to perform a linear interpolation (MOD = 61h) after completing the circular interpolation.

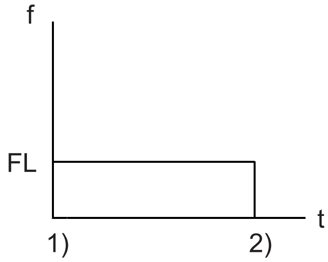
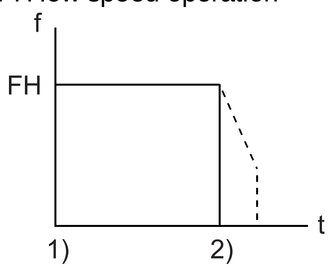
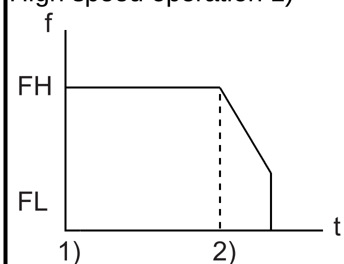
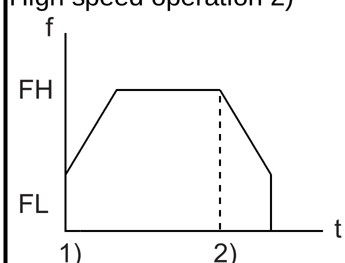
- Operating speed when the synthetic constant speed control is applied
  The synthetic constant speed control is a control used to keep the speed of both axes in a simultaneous operation the same as the single operation speed of the master axis. This does not maintain a constant linear track (or circular track) speed to the target position.
  With synthetic constant speed control, the track will pass over the coordinate cross point that is expressed by the number of pulses equal to the motor resolution, as shown in Figure 1 in section 9-8-3, "Synthetic constant speed control." Therefore, the track of the actual motion will cover a greater distance than a straight line to the target position. As a result, the feed time (feed distance / set speed) to the target when using synthetic constant speed control will be up to 8% longer when compared with the straight line feed time (when the synthetic angle for X and Y is 26.5°).
  In order to feed to the target position in a straight line using linear interpolation, you may not use the synthetic constant speed control. In this case, calculate the synthetic speed for the X and Y axes and enter that speed.

  <Set the speed for a synthetic linear interpolation speed when the synthetic constant speed control is OFF>

$$\text{Set speed} = \frac{\text{Synthetic speed x Amount of master axis feed}}{\sqrt{(\text{Amount of master axis feed})^2 + (\text{Amount of slave axis feed})^2}}$$

10. Speed patterns

10-1. Speed patterns

Speed pattern	Continuous mode	Positioning operation mode
FL low speed operation 	1) Write an FL low speed start command (50h). 2) Stop feeding by writing an immediate stop (49h) or deceleration stop (4Ah) command.	1) Write an FL low speed start command (50h). 2) Stop feeding when the positioning counter reaches zero, or by writing an immediate stop (49h) or deceleration stop (4Ah) command.
FH low speed operation 	1) Write an FH low speed start command (51h). 2) Stop feeding by writing an immediate stop command (49h). * If a deceleration stop command (4Ah) is written, the axis will decelerate and stop.	1) Write an FH low speed start command (51h). 2) Stop feeding when the positioning counter reaches zero, or by writing an immediate stop (49h) command. * If a deceleration stop command (4Ah) is written, the axis will decelerate and stop.
High speed operation 1) 	1) Write high speed start command 1 (52h). 2) Start deceleration by writing a deceleration stop command (4Ah). * If an immediate stop command (49h) is written, the axis will stop immediately.	1) Write high speed start command 1 (52h). 2) Start deceleration when a ramping-down point is reached or by writing a deceleration stop command (4Ah). * When positioning with a high speed start command 1 (52h), <u>the ramping-down point is fixed to the manual setting,</u> regardless of the setting for MSDP (bit 13) in the RMD. If the ramping-down point setting (RDP) is zero, the axis will stop immediately. * If an immediate stop command (49h) is written, the axis will stop immediately.
High speed operation 2) 	1) Write high speed command 2 (53h). 2) Start deceleration by writing a deceleration stop command (4Ah). * If an immediate stop command (49h) is written, the axis will stop immediately.	1) Write high speed start command 2 (53h). 2) Start deceleration when a ramping-down point is reached or by writing a deceleration stop command (4Ah). * If the ramping-down point is set to manual (MSDP = 1 in the RMD), and the ramping-down value (RDP) is zero, the axis will stop immediately. * When an immediate stop command (49h) is written, the axis will stop immediately.

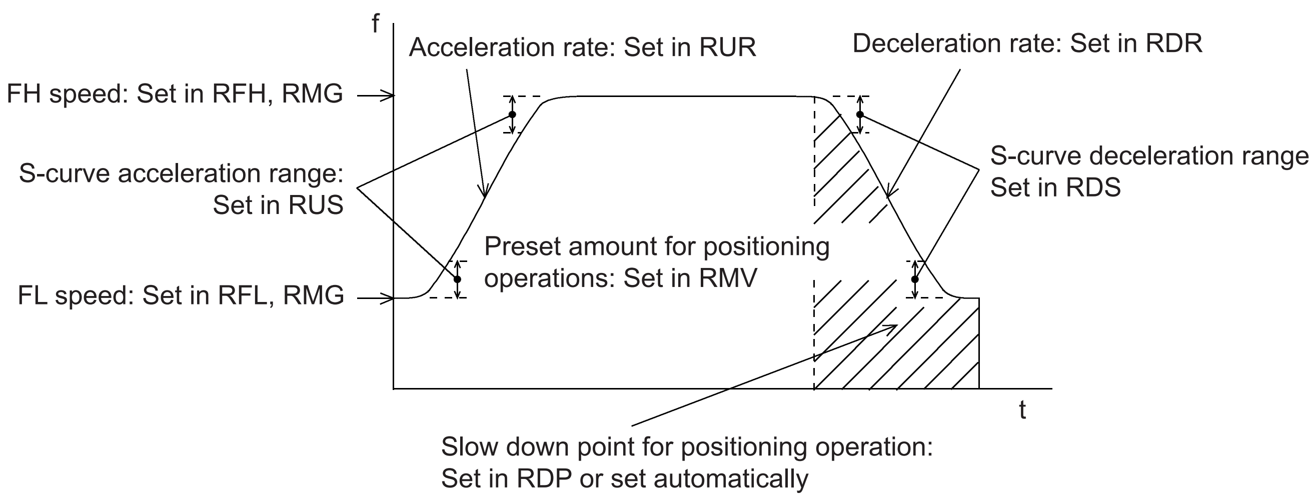
10-2. Speed pattern settings

Specify the speed pattern using the registers (pre-registers) shown in the table below.
If the next register setting is the same as the current value, there is no need to write to the register again.

Register	Description	Bit length setting range	Setting range	Pre-register
RMV	Positioning amount	28	-134,217,728 to 134,217,727 (8000000h) (7FFFFFFh)	PRMV
RFL	Initial speed	16	1 to 65,535 (0FFFFh)	PRFL
RFH	Operation speed	16	1 to 65,535 (0FFFFh)	PRFH
RUR	Acceleration rate	16	1 to 65,535 (0FFFFh)	PRUR
RDR	Deceleration rate Note 1	16	0 to 65,535 (0FFFFh)	PRDR
RMG	Speed magnification rate	12	2 to 4,095 (0FFFh)	PRMG
RDP	Ramping-down point	24	0 to 16,777,215 (0FFFFFFh)	PRDP
RUS	S-curve acceleration range	15	0 to 32,767 (7FFFh)	PRUS
RDS	S-curve deceleration range	15	0 to 32,767 (7FFFh)	PRDS

Note 1: If RDR is set to zero, the deceleration rate will be the value set in the RUR.

[Relative position of each register setting for acceleration and deceleration factors]



- RFL: FL speed setting register (16-bit)
  Specify the speed for FL low speed operations and the start speed for high speed operations (acceleration/deceleration operations) in the range of 1 to 65,535 (0FFFFh).
  The speed will be calculated from the value in RMG.

$$\text{FL speed [pps]} = \text{RFL} \times \frac{\text{Reference clock frequency [Hz]}}{(\text{RMG} + 1) \times 65536}$$

- RFH: FH speed setting register (16-bit)
  Specify the speed for FH low speed operations and the start speed for high speed operations (acceleration/deceleration operations) in the range of 1 to 65,535 (0FFFFh).
  When used for high speed operations (acceleration/deceleration operations), specify a value larger than RFL.
  The speed will be calculated from the value placed in RMG.

$$\text{FH speed [pps]} = \text{RFH} \times \frac{\text{Reference clock frequency [Hz]}}{(\text{RMG} + 1) \times 65536}$$

- RUR: Acceleration rate setting register (16-bit)
  Specify the acceleration characteristic for high speed operations (acceleration/deceleration operations), in the range of 1 to 65,535 (0FFFFh)
  Relationship between the value entered and the acceleration time will be as follows:

  1) Linear acceleration (MSMD = 0 in the RMD register)

$$\text{Acceleration time [s]} = \frac{(\text{RFH - RFL}) \times (\text{RUR} + 1) \times 4}{\text{Reference clock frequency [Hz]}}$$

  2) S-curve without a linear range (MSMD=1 in the RMD register and RUS register =0)

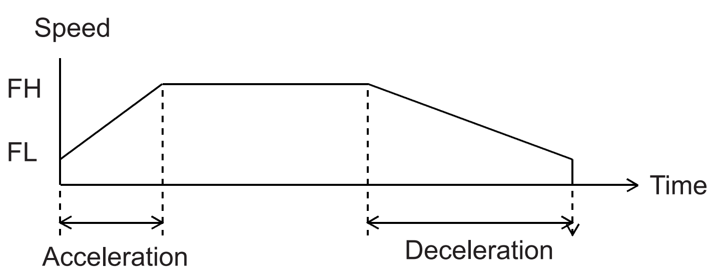
$$\text{Acceleration time [s]} = \frac{(\text{RFH - RFL}) \times (\text{RUR} + 1) \times 8}{\text{Reference clock frequency [Hz]}}$$

  3) S-curve with a linear range (MSMD=1 in the RMD register and RUS register >0)

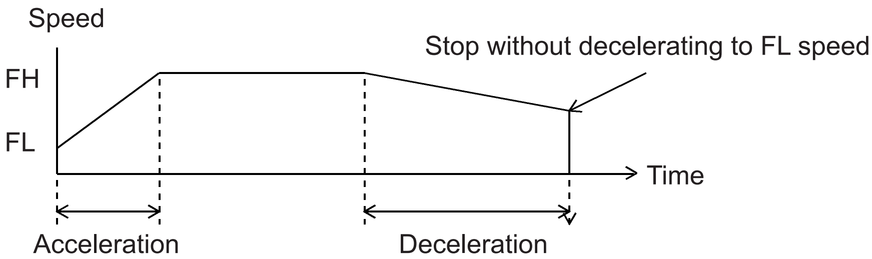
$$\text{Acceleration time [s]} = \frac{(\text{RFH-RFL} + 2 \times \text{RUS}) \times (\text{RUR} + 1) \times 4}{\text{Reference clock frequency [Hz]}}$$

- RDR: Deceleration rate setting register (16-bit)
  Normally, specify the deceleration characteristics for high speed operations (acceleration/deceleration operations) in the range of 1 to 65,535 (0FFFFh).
  Even if the ramping-down point is set to automatic (MSDP = 0 in the RMD register), the value placed in the RDR register will be used as the deceleration rate.
  However, when RDR = 0, the deceleration rate will be the value placed in the RUR.
  When the ramping-down point is set automatically, make sure that the (deceleration time) ≤ (acceleration time x 2) for independent axis operations, and that the (deceleration time) = (acceleration time) for interpolation operations.
  If the deceleration (deceleration time) > (acceleration time x 2) in independent axis operations, or if (deceleration time) > (acceleration time) in interpolation operations, the axis may not decrease the speed to the specified FL speed when stopping. In this case, use a manual ramping-down point (MSDP = 1 in the RMD register).

  < When (deceleration time) ≤ (acceleration time x 2) using an automatic ramping-down point >



<When (deceleration time) > (acceleration time x 2) using an automatic ramping-down point >



The relationship between the value entered and the deceleration time is as follows.

1) Linear deceleration (MSMD = 0 in the RMD register)

$$\text{Deceleration time [s]} = \frac{(RFH - RFL) \times (RDR + 1) \times 4}{\text{Reference clock frequency [Hz]}}$$

2) S-curve deceleration without a linear range (MSMD=1 in the RMD register and RDS register = 0)

$$\text{Deceleration time [s]} = \frac{(RFH - RFL) \times (RDR + 1) \times 8}{\text{Reference clock frequency [Hz]}}$$

3) S-curve deceleration with a linear range (MSMD=1 in the RMD register and RDS register >0)

$$\text{Deceleration time [s]} = \frac{(RFH-RFL + 2 \times RDS) \times (RDR + 1) \times 4}{\text{Reference clock frequency [Hz]}}$$

- RMG: Magnification rate register (12-bit)
  Specify the relationship between the RFL and RFH settings and the speed, in the range of 2 to 4,095 (0FFFh).
  As the magnification rate is increased, the speed setting units will tend to be approximations. Normally set the magnification rate as low as possible.
  The relationship between the value entered and the magnification rate is as follows.

$$\text{Magnification rate} = \frac{\text{Reference clock frequency [Hz]}}{(RMG + 1) \times 65536}$$

[Magnification rate setting example, when the reference clock =19.6608 MHz] (Output speed unit: pps)

Setting	Magnification rate	Output speed range	Setting	Magnification rate	Output speed range
2999 (0BB7h)	0.1	0.1 to 6,553.5	59 (3Bh)	5	5 to 327,675
1499 (5DBh)	0.2	0.2 to 13,107.0	29 (1Dh)	10	10 to 655,350
599 (257h)	0.5	0.5 to 32,767.5	14 (0Eh)	20	20 to 1,310,700
299 (12Bh)	1	1 to 65,535	5 (5h)	50	50 to 3,276,750
149 (95h)	2	2 to 131,070	2 (2h)	100	100 to 6,553,500

- RDP: Ramping-down point register (24-bits)

Specify the value used to determine the deceleration start point for positioning operations that include acceleration and deceleration.
The meaning of the value specified in the RDP changes with the "ramping-down point setting method ", (MSDP) in the RMD register.

<When set to manual (MSDP=1 in the RMD register)>

Set the number of pulses at which to start deceleration, in the range of 0 to16,777,215 (0FFFFFFh).
The optimum value for the ramping-down point can be calculated as shown in the equation below.

1) Linear deceleration (MSMD=0 of the RMD register)

$$\text{Optimum value [Number of pulses]} = \frac{(RFH^2 - RFL^2) \times (RDR + 1)}{(RMG + 1) \times 32768}$$

However, the optimum value for a triangle start, without changing the value in the RFH register while turning OFF the FH correction function (MADJ = 1 in the RMD register) will be calculated as shown the equation below. (When using idling control, modify the value for RMV in the equation below by deducting the number of idling pulses from the value placed in the RMV register. The number of idling pulses will be "1 to 6" when IDL = 2 to 7 in RENV5.)

$$\text{Optimum value [Number of pulses]} = \frac{RMV \times (RDR + 1)}{RUR + RDR + 2}$$

2) S-curve deceleration without a linear range (MSMD=1 in the RMD register and the RDS register =0)

$$\text{Optimum value [Number of pulses]} = \frac{(RFH^2 - RFL^2) \times (RDR + 1) \times 2}{(RMG + 1) \times 32768}$$

3) S-curve deceleration with a linear range (MSMD=1 in the RMD register and the RDS register >0)

$$\text{Optimum value [Number of pulses]} = \frac{(RFH + RFL) \times (RFH - RFL + 2 \times RDS) \times (RDR + 1)}{(RMG + 1) \times 32768}$$

Start deceleration at the point when the (positioning counter value) ≤ (RDP set value).

<When set to automatic (MSDP = 0 in the RMD register)>

This is an offset value for the automatically set ramping-down point. Set in the range of -8,388,608 (800000h) to 8,388,607 (7FFFFFFh).
When the offset value is a positive number, the axis will start deceleration at an earlier stage and will feed at the FL speed after decelerating. When a negative number is entered, the deceleration start timing will be delayed. If the offset is not required, set to zero.
When the value for the ramping-down point is smaller than the optimum value, the speed when stopping will be faster than the FL speed. On the other hand, if it is larger than the optimum value, the axis will feed at FL low speed after decelerating.

- RUS: S-curve acceleration range register (15-bit)

  Specify the S-curve acceleration range for S-curve acceleration/deceleration operations in the range of 1 to 32,767 (7FFFh).

  The S-curve acceleration range S_{SU} will be calculated from the value placed in RMG.

  $$S_{SU} \text{ [pps]} = \text{RUS} \times \frac{\text{Reference clock frequency [Hz]}}{(\text{RMG} + 1) \times 65536}$$

  In other words, speeds between the FL speed and (FL speed + S_{SU}), and between (FH speed – S_{SU}) and the FH speed, will be S-curve acceleration operations. Intermediate speeds will use linear acceleration.

  However, if zero is specified, "(RFH-RFL)/2" will be used for internal calculations, and the operation will be an S-curve acceleration without a linear component.

- RDS: S-curve deceleration range setting register (15-bit)

  Specify the S-curve deceleration range for S-curve acceleration/deceleration operations in the range of 1 to 32,767 (7FFFh).

  The S-curve acceleration range S_{SD} will be calculated from the value placed in RMG.

  $$S_{SD} \text{ [pps]} = \text{RDS} \times \frac{\text{Reference clock frequency [Hz]}}{(\text{RMG} + 1) \times 65536}$$

  In other words, speeds between the FL speed and (FL speed + S_{SD}), and between (FH speed - S_{SD}) and the FH speed, will be S-curve deceleration operations. Intermediate speeds will use linear deceleration.

  However, if zero is specified, "(RFH-RFL)/2" will be used for internal calculations, and the operation will be an S-curve deceleration without a linear component.

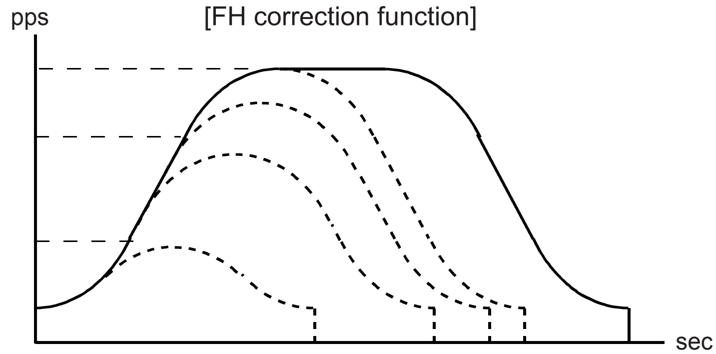
Register	Details	Register	Details
RMV	Positioning amount	RDR	Deceleration rate
RFL	Initial speed	RMG	Speed magnification rate
RFH	Operation speed	RUS	S-curve range when accelerating
RUR	Acceleration rate	RDS	S-curve range when decelerating

10-3. FH correction (eliminate triangle driving)

When the FH correction function is turned ON (MADJ = 0 in the RMD register), and when the feed amount is too small for a normal acceleration and deceleration operation, the LSI will automatically lower the FH speed to eliminate triangle driving.

However, if values in the RUR and RDR registers are set so that the (deceleration time) > (acceleration time x 2), do not use the FH correction function.

In order to eliminate triangle driving without using the FH correction function (MADJ = 1 in the RMD register), lower the FH speed before starting the acceleration/deceleration operation. (When using idling control, enter a value for RMV in the equation below after deducting the number of idling pulses. The number of idling pulses will be 1 to 6 when IDL = 2 to 7 in RENV5.)



Automatic correction of the maximum speed for changing the feed amount.

< To execute an FH correction manually>

1) Linear acceleration/deceleration speed (MSMD=0 in the RMD register)

When $RMV \leq \dfrac{(RFH^2 - RFL^2) \times (RUR + RDR + 2)}{(RMG+1) \times 32768}$

$$RFH \leq \sqrt{\frac{(RMG + 1) \times 32768 \times RMV}{RUR + RDR + 2} + RFL^2}$$

2) S-curve acceleration without linear acceleration (MSMD=1 in the RMD and RDS registers = 0)

When $RMV \leq \dfrac{(RFH^2 - RFL^2) \times (RUR + RDR + 2) \times 2}{(RMG + 1) \times 32768}$

$$RFH \leq \sqrt{\frac{(RMG + 1) \times 32768 \times RMV}{(RUR + RDR + 2) \times 2} + RFL^2}$$

3) S-curve acceleration/deceleration with linear acceleration/deceleration (MSMD = 1 in the RMD register and the RUS register > 0, RDS register > 0)

(3)-1. When RUS = RDS

(i) Set up a small linear acceleration range

$RMV \leq \dfrac{(RFH + RFL) \times (RFH - RFL + 2 \times RUS) \times (RUR + RDR + 2)}{(RMG + 1) \times 32768}$ and

When $RMV > \dfrac{(RUS+RFL) \times RUS \times (RUR + RDR + 2) \times 8}{(RMG + 1) \times 32768}$

$$RFH \leq - RSU + \sqrt{(RUS - RFL)^2 + \frac{(RMG + 1) \times 32768 \times RMV}{(RUR + RDR + 2)}}$$

(ii) Eliminate the linear acceleration/deceleration range

When $RMV \leq \dfrac{(RUS + RFL) \times RUS \times (RUR + RDR + 2) \times 8}{(RMG + 1) \times 32768}$

Change to S-curve acceleration/deceleration without a linear acceleration/deceleration range (RUS = 0, RDS = 0),

$$RFH \leq \sqrt{\frac{(RMG + 1) \times 32768 \times RMV}{(RUR + RDR + 2) \times 2} + RFL^2}$$

Register	Detail	Register	Detail
RMV	Positioning amount	RDR	Deceleration rate
RFL	Initial speed	RMG	Speed magnification rate
RFH	Operation speed	RUS	S-curve range when accelerating
RUR	Acceleration rate	RDS	S-curve range when decelerating

(3)-2. When RUS < RDS

(i) Set up a small linear acceleration/deceleration range

$$RMV \leq \frac{(RFH+RFL) \times \{(RFH-RFL) \times (RUR+RDR+2) + 2 \times RUS \times (RUR+1) + 2 \times RDS \times (RDR+1)\}}{(RMG+1) \times 32768} \text{ and}$$

$$\text{When } RMV > \frac{(RDS+RFL) \times \{RDS \times (RUR + 2 \times RDR + 3) + RUS \times (RUR + 1)\} \times 4}{(RMG + 1) \times 32768}$$

$$RFH \leq \frac{-A + \sqrt{A^2 + B}}{RUR + RDR + 2}$$

However, $A = RUS \times (RUR+1)+RDS \times (RDR+1)$
$B= \{(RMG + 1) \times 32768 \times RMV - 2 \times A \times RFL + (RUR + RDR + 2) \times RFL^2\} \times (RUR + RDR + 2)$

(ii) Eliminate the linear acceleration/deceleration range and set up a small linear acceleration section.

$$RMV \leq \frac{(RDS + RFL) \times \{RDS \times (RUR + 2 \times RDR + 3)\} + RUS \times (RUR +1)\} \times 4}{(RMG + 1) \times 32768} \text{ and}$$

$$\text{When } RMV > \frac{(RUS + RFL) \times RUS \times (RUR + RDR + 2) \times 8}{(RMG + 1) \times 32768}$$

Change to S-curve acceleration/deceleration without any linear acceleration/deceleration (RUS>0, RDS=0)

$$RFH \leq \frac{-A + \sqrt{A^2 + B}}{RUR+2 \times RDR + 3}$$

However, $A = RUS \times (RUR + 1)$,
$B= \{(RMG + 1) \times 32768 \times RMV - 2 \times A \times RFL + (RUR + 2 \times RDR + 3) \times RFL^2\} \times (RUR + 2 \times RDR + 3)$

(iii) Eliminate the linear acceleration/deceleration range

$$\text{When } RMV \leq \frac{(RUS + RFL) \times RUS \times (RUR + RDR + 2) \times 8}{(RMG + 1) \times 32768}$$

Change to S-curve acceleration/deceleration without any linear acceleration/deceleration (RUS=0, RDS=0),

$$RFH \leq \sqrt{\frac{(RMG + 1) \times 32768 \times RMV}{(RUR + RDR + 2) \times 2} + RFL^2}$$

Register	Detail	Register	Detail
RMV	Positioning amount	RDR	Deceleration rate
RFL	Initial speed	RMG	Speed magnification rate
RFH	Operation speed	RUS	S-curve range when accelerating
RUR	Acceleration rate	RDS	S-curve range when decelerating

(3)-3. When RUS>RDS

(i) Set up a small linear acceleration/deceleration range

$$\text{RMV} \leq \frac{(\text{RFH+RFL}) \times \{(\text{RFH - RFL}) \times (\text{RUR+RDR+2}) + 2 \times \text{RUS} \times (\text{RUR+1}) + 2 \times \text{RDS} \times (\text{RDR+1})\}}{(\text{RMG +1}) \times 32768} \text{ and}$$

$$\text{When RMV} > \frac{(\text{RUS + RFL}) \times \{\text{RUS} \times (2 \times \text{RUR + RDR + 3}) + \text{RDS} \times (\text{RDR + 1}) \times 4}{(\text{RMG + 1}) \times 32768}$$

$$\text{RFH} \leq \frac{-A + \sqrt{A^2 + B}}{\text{RUR + RDR + 2}}$$

However, $A = \text{RUS} \times (\text{RUR} + 1) + \text{RDS} \times (\text{RDR} + 1)$,
$B = \{(\text{RMG} + 1) \times 32768 \times \text{RMV} - 2 \times A \times \text{RFL} + (\text{RUR} + \text{RDR} + 2) \times \text{RFL}^2\} \times (\text{RUR} + \text{RDR} + 2)$

(ii) Eliminate the linear acceleration section and set up a small linear deceleration range.

$$\text{RMV} \leq \frac{(\text{RUS + RFL}) \times \{\text{RUS} \times (2 \times \text{RUR + RDR + 3}) + \text{RDS} \times (\text{RDR + 1})\} \times 4}{(\text{RMG + 1}) \times 32768} \text{ and}$$

$$\text{When RMV} > \frac{(\text{RDS + RFL}) \times \text{RDS} \times (\text{RUR + RDR + 2}) \times 8}{(\text{RMG + 1}) \times 32768}$$

Change to S-curve acceleration/deceleration without any linear acceleration (RUS = 0, RDS > 0)

$$\text{RFH} \leq \frac{-A + \sqrt{A^2 + B}}{2\text{x RUR + RDR + 3}}$$

However, $A = \text{RDS} (\text{RDR} + 1)$,
$B = \{(\text{RMG} + 1) \times 32768 \times \text{RMV} - 2 \times A \times \text{RFL} + (2 \times \text{RUR} + \text{RDR} + 3) \times \text{RFL}^2\} \times (2 \times \text{RUR} + \text{RDR} + 3)$

(iii) Eliminate the linear acceleration/deceleration range

$$\text{When RMV} \leq \frac{(\text{RDS + RFL}) \times \text{RDS} \times (\text{RUR + RDR + 2}) \times 8}{(\text{RMG + 1}) \times 32768}$$

Change to S-curve acceleration/deceleration without any linear acceleration/deceleration (RUS = 0, RDS = 0),

$$\text{RFH} \leq \sqrt{\frac{(\text{RMG + 1}) \times 32768 \times \text{RMV}}{(\text{RUR + RDR + 2}) \times 2} + \text{RFL}^2}$$

Register	Detail	Register	Detail
RMV	Positioning amount	RDR	Deceleration rate
RFL	Initial speed	RMG	Speed magnification rate
RFH	Operation speed	RUS	S-curve range when accelerating
RUR	Acceleration rate	RDS	S-curve range when decelerating

10-4. Example of setting up an acceleration/deceleration speed pattern

Ex. Reference clock = 19.6608 MHz

When the start speed =10 pps, the operation speed =100 kpps, and the accel/decel time = 300 ms,

1) Select the 2x mode for multiplier rate in order to get 100 kpps output
   RMG = 149 (95h)

2) Since the 2x mode is selected to get an operation speed 100 kpps,
   RFH = 50000 (C350h)

3) In order to set a start speed of 10 pps, the rate magnification is set to the 2x mode.
   RFL = 5 (0005h)

4) In order to make the acceleration/deceleration time 300 ms, set RUR = 28,494, from the equation for the acceleration time and the RUR value.

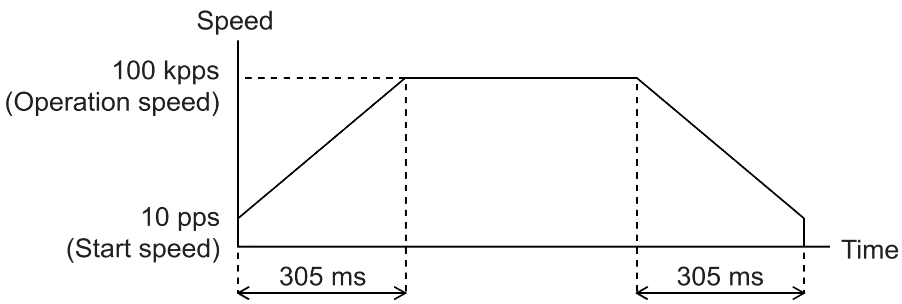
$$\text{Acceleration time [s]} = \frac{(\text{RFH} - \text{RFL}) \times (\text{RUR} + 1) \times 4}{\text{Reference clock frequency [Hz]}}$$

$$0.3 = \frac{(50000 - 5) \times (\text{RUR} + 1) \times 4}{19.6608 \times 10^{6}}$$

RUR = 28.494

However, since only integers can be entered for RUR, use 28 or 29. The actual acceleration/deceleration time will be 295 ms if RUR = 28, or 305 ms if RUR = 29.

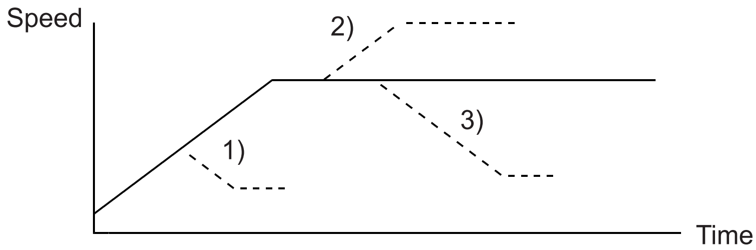
An example of the speed pattern when RUR = 29



10-5. Changing speed patterns while in operation

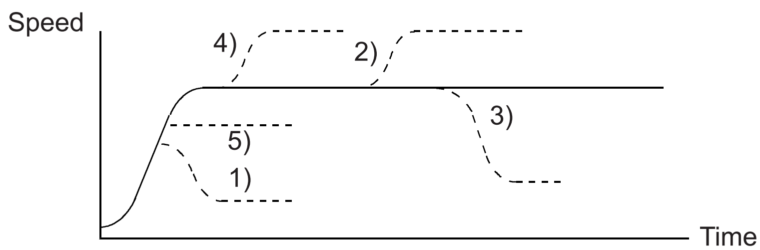
By changing the RFH, RUR, RDR, RUS, or RDS registers during operation, the speed and acceleration can be changed on the fly. However, if the ramping-down point was set to automatic (MSDP = 0 in the RDM register) for the positioning mode, do not change the values for RFL, RUR, RDR, RUS, or RDS. The automatic ramping-down point function will not work correctly.

An example of changing the speed pattern by changing the speed, during a linear acceleration/deceleration operation



1) Use a small RFH while accelerating or decelerating the axis until it reaches the correct speed.
2), 3) Change RFH after the acceleration/deceleration is complete. The axis will continue accelerating or decelerating until it reaches the new speed.

An example of changing the speed pattern by changing the speed during S-curve acceleration/deceleration operation



1) Use a small RFH and if (change speed) < speed before change) and the axis will accelerate/decelerate using an S-curve until it reaches the correct speed.
5) Use a small RFH and if (change speed) ≥ speed before change) and the axis will accelerate/decelerate without changing the S-curve's characteristic until it reaches the correct speed.
4) Use a large RFH while accelerating and the axis will accelerate to the original speed entered without changing the S-curve's characteristic. Then it will accelerate again until it reaches the newly set speed.
2), 3) If RFH is changed after the acceleration/deceleration is complete, the axis will accelerate/decelerate using an S-curve until it reaches the correct speed.

11. Description of the functions

11-1. Reset

After turning ON the power, make sure to reset the LSI before beginning to use it.
To reset the LSI, hold the $\overline{\text{RST}}$ terminal LOW while supplying at least 10 cycles of a reference clock signal.
After a reset, the various portions of the LSI will be configured as follows.

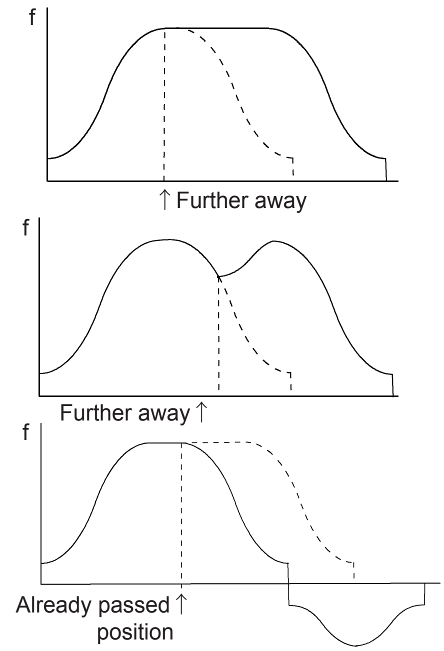
Item	Reset status (initial status) n = x, y
Internal registers	0
Control command buffer	0
Axis assignment buffer	0
Input/output buffer	0
$\overline{\text{INT}}$ terminal	HIGH
$\overline{\text{WRQ}}$ terminal	HIGH
$\overline{\text{IFB}}$ terminal	HIGH
D0 to D7 terminals	High Z (impedance)
D8 to D15 terminals	High Z (impedance)
P0n to P7n terminals	Input terminal
$\overline{\text{CSTA}}$ terminal	HIGH
$\overline{\text{CSTP}}$ terminal	HIGH
OUTn terminal	HIGH
DIRn terminal	HIGH
ERCn terminal	HIGH
$\overline{\text{BSYn}}$ terminal	HIGH
$\overline{\text{FUPn}}$ terminal	HIGH
$\overline{\text{FDWn}}$ terminal	HIGH
$\overline{\text{MVCn}}$ terminal	HIGH

11-2. Position override

This LSI can override (change) the target position freely during operation.
There are two methods for overriding the target position.

11-2-1. Target position override 1

By rewriting the target position data (RMV register value), the target position can be changed.
The starting position is used as a reference to change target position.



1) If the new target position is further away from the original target position during acceleration or low speed operation, the axis will maintain the operation using the same speed pattern and it will complete the positioning operation at the position specified in the new data (new RMV value).

2) If the new target position is further away from the original target position during deceleration, the axis will accelerate from the current position to FH speed and complete the positioning operation at the position specified in the new data (new RMV value).

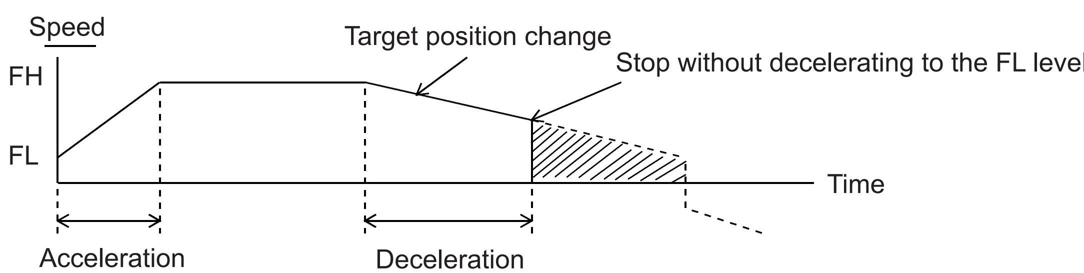
3) If the axis has already passed over the new target position, or the target position is changed to a position that is closer than the original position during deceleration, movement on the axis will decelerate and stop. Then, the movement will reverse and complete the positioning operation at the position specified in the new data (new RMV value).

The axis accelerates/decelerates only when starting in high speed. The target position data (RMV register value) can be rewritten any number of times until the positioning operation is complete.
The conditions necessary for being able to rewrite the target position are that the axis is being moved (SRUN = 1) and that the CND (bits 0 to 3) in the extension status register (RSTS) is set to any of the following 5 values.

1) CND=0111 (Correcting backlash)
2) CND=1010 (Feeding at FL low speed)
3) CND=1011 (Accelerating)
4) CND=1100 (Feeding at FH high speed)
5) CND=1101 (Decelerating)

The axis may stop moving to make a status check. Therefore, it is necessary to check whether the current position matches the rewritten position at the completion of the operation (SEDN=1).
While the LSI is waiting for an INP signal or waiting for another axis to stop, the LSI is stopping its internal operation. Therefore, the LSI cannot accept an attempt to rewrite the target position data.

Note: If the ramping-down point is set to automatic and the (deceleration time) > (acceleration time x 2), it may be the case that the axis cannot reduce the speed to the FL level, as shown below. In this case, if the target position is set closer than original position and the axis is decelerating, the axis will decelerate along the deceleration curve to the new override position, and then slow to the FL speed and finally stop. Then it will start moving to the new position.
Therefore, the axis will overrun the original target position during deceleration (shaded area).



To avoid creating an overrun condition, make sure that the deceleration time is less than two times the acceleration time, or if the deceleration time is more than double the acceleration time, make the ramping-down point a manual setting.

11-2-2. Target position override 2 (PCS signal)

By setting MPCS = 1 in the RMD (operation mode) register, the start counting performed by the positioning counter that is set by the RMV can be delayed until the PCS signal is input.
After writing a start command (start outputting command pulses), the axis can be positioned by the PCS input signal for the number of pulses set in the RMV register ON timing, or by writing a command.

A PCS input signal can change the input logic. The PCS terminal status can be monitored using the RSTS register (extension status).

Setting pulse control using the PCS input <Set MPCS (bit 14) in RMD> 1: Positioning for the number of pulses stored in the RMV, starting from the time at which the PCS input signal is turned ON.	[RMD] (WRITE) 15 8 - n - - - - - -
Setting the PCS input logic <Set PCSL (bit 24) in RENV1> 0: Negative logic 1: Positive logic	[RENV1] (WRITE) 31 24 - - - - - - - n
Reading the PCS signal < SPCS (bit 8) in MRSTS> 0: Turn OFF PCS 1: Turn ON PCS	[RSTS] (READ) 15 8 - - - - - - - n
PCS substitution input <STAON: Operation command> Perform processes that are identical to those performed by supplying a PCS signal.	[Operation command] 28h

11-3. Output pulse control

11-3-1. Output pulse mode

There are four types of common command pulse output modes and two types of 2-pulse modes.

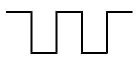
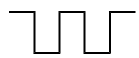
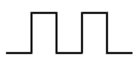
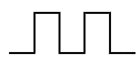
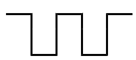
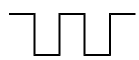
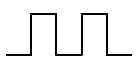
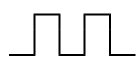
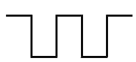
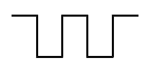
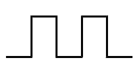
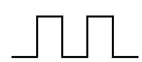
Common pulse mode: Outputs operation pulses from the OUT terminal and outputs the direction signal from the DIR terminal.

2-pulse mode: Outputs positive direction operation pulses from the OUT terminal, and outputs negative direction operation pulses from the DIR terminal.

The output mode for command pulses is set in PMD (bits 0 to 2) in RENV1 (environment setting 1).
If motor drivers using the common pulse mode need a lag time (since the direction signal changes, until receiving a command pulse), use a direction change timer.
When DTMP (bit 28) in the RENV1 (environment setting 1) is set to 0, the operation can be delayed for one direction change timer unit (0.2 ms), after changing the direction identification signal.

Setting the pulse output mode <Set PMD0 to 2 (bits 0 to 2) in RENV1>

[RENV1] (WRITE)
7 0
| - | - | - | - | - | n | n | n |

PMD0 to 2	When feeding in the positive direction		When feeding in the negative direction	
	OUT output	DIR output	OUT output	DIR output
000		High		Low
001		High		Low
010		Low		High
011		Low		High
100		High	High	
111		Low	Low	

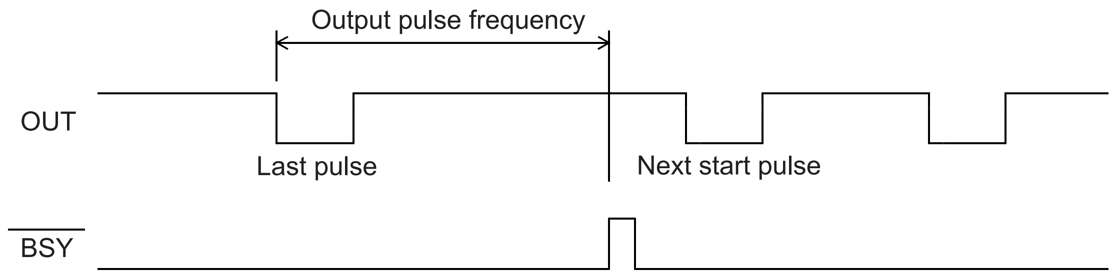
Setting the direction change timer (0.2 ms) function <Set DTMF (bit 28) in RENV1>
0: ON
1: OFF

[RENV1] (WRITE)
31 24
| - | - | - | n | - | - | - | - |

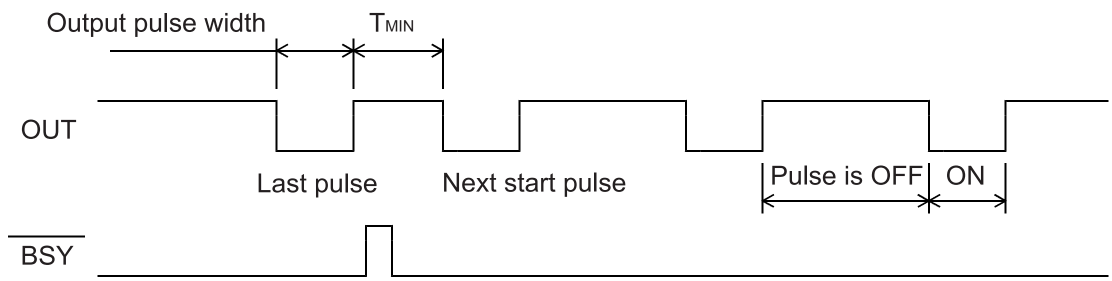
11-3-2. Control the output pulse width and operation complete timing

In order to increase the stopping speed, this LSI controls the output pulse width.
When the output pulse speed is slower than 1/8192 of reference clock (approx. 2.4 Kpps when CLK = 19.6608 MHz), the pulse width is constant and is 4096 cycles of the reference clock (approx. 200 µs when CLK = 19.6608 MHz). For faster pulse speeds than this, the duty cycle is kept constant (approx. 50%). By setting PDTC (bit 13) in the RENV1 register (environment setting 1), the output pulse width can be set to make a constant duty cycle (50%).
Also, when setting METM (operation completion timing setting) in the RMD register (operation mode), the operation complete timing can be changed.

1) When METM = 0 (the point at which the output frequency cycle is complete) in the RMD register



2) When METM = 1 (when the output pulse is OFF) in the RMD register



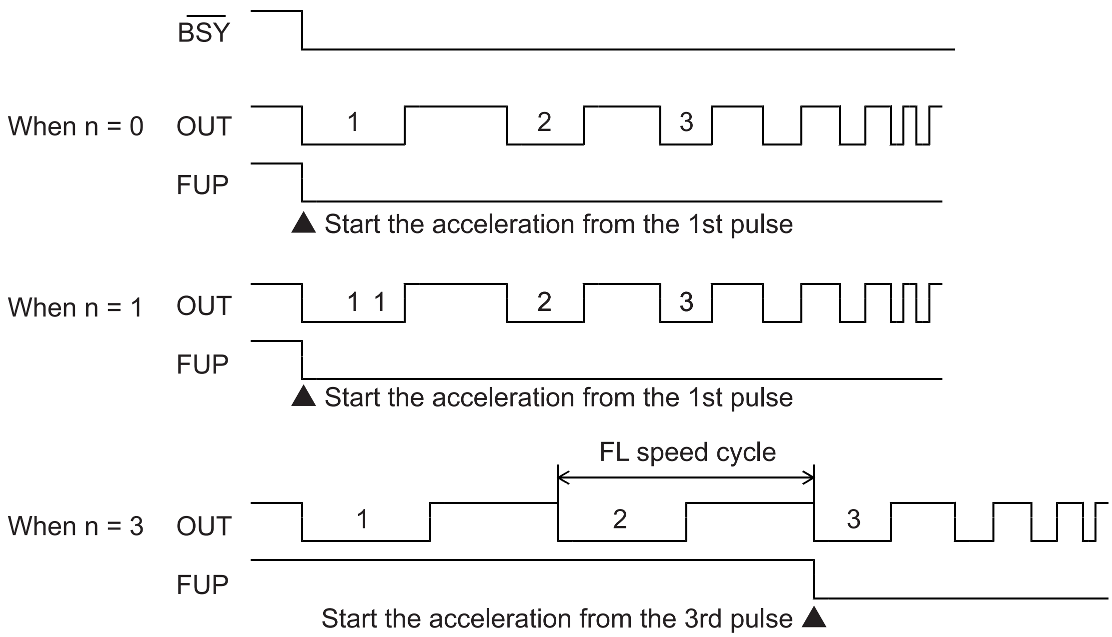
When set to "complete when the output pulse is OFF," the time interval "Min" from the last pulse until the next starting pulse output will be T_{MIN} = 14 x T_{CLK}. (T_{CLK}: Reference clock frequency)

Setting the operation complete timing <Set METM (bit 12) in RMD> 0: At the end of a cycle of a particular output frequency 1: Complete when the output pulse turns OFF.	[RMD] (WRITE) 15 8 - - - n - - - -
Setting the output pulse width <Set PDTC (bit 31) in RENV1> 0: Automatically change between a constant output pulse and a constant duty cycle (approx. 50%) in accord with variations in speed. 1: Keep the output pulse width at a constant duty cycle (approx. 50%).	[RENV1] (WRITE) 31 24 n - - - - - - -

11-4. Idling control

When starting an acceleration or a deceleration operation, it can be started after the output of a few pulses at FL speed (idling output). Set the number of pulses for idling in IDL of the RENV5 register (environment setting 5).
If you will not be using this function, enter a value "n" of 0 or 1. The LSI will start the acceleration at the same time it begins outputting pulses. Therefore, the start speed obtained from an initial 2-pulse frequency will be faster than the FL speed.
To use this function, enter a value "n" of 2 to 7. The LSI will start the acceleration by beginning its output on the "n"th pulse. Therefore, the start speed will be the FL speed and the FL speed can be set to start automatically at upper speed limit.
If this function is used with the positioning mode, the total feed amount will not change.

[Setting idling pulses and the acceleration start timing]



Set the number of idling pulses <Set IDL0 to 2 (bits 8 to 10) in RENV5> Specify the number of idling pulses, from 0 to 7. Start accelerating at FL speed after outputting the specified number of pulses.	[RENV5] (WRITE) 15 8 - - - - - n n n
Read the idling control counter value < IDC0 to 2 (bits 20 to 22) in RSPD> Read the idling control counter.	[RSPD] (READ) 23 16 0 n n n - - - -

11-5. Mechanical external input control

11-5-1. +EL, -EL signal

When an end limit signal (a +EL signal when feeding in the + direction) in the feed direction turns ON while operating, the axis will stop immediately or decelerate and stop. After stopping, even if the EL signal is turned OFF, the axis will remain stopped.
If the EL signal is ON when writing a start command, the axis cannot start moving in the direction of the particular EL signal that is ON.
By setting ELM in the RENV1 (environment setting 1) register, the stopping pattern for use when the EL signal is turned ON can be set to immediate stop or deceleration stop (high speed start only).
The minimum pulse width of the EL signal is 80 reference clock cycles (4 µs) when the input filter is ON. When the input filter is turned OFF, the minimum pulse width is two reference clock cycle (0.1 µs).
The EL signal can be monitored by reading SSTSW (sub status).
By reading the REST register, you can check for an error interrupt caused by the EL signal turning ON.
When in the timer mode, this signal is ignored. Even in this case, the EL signal can be monitored by reading SSTSW (sub status).
The input logic of the EL signal can be set for each axis using the ELL input terminal.

Set the input logic of the ±EL signal <ELL input terminal> L: Positive logic input H: Negative logic input	
Stop method to when the ±EL signal turns ON <Set ELM (bit 3) in RENV1> 0: Immediate stop by turning ON the ±EL signal 1: Deceleration stop by turning ON the ±EL signal	[RENV1] (WRITE) 7 0 - - - - n - - -
Reading the ±EL signal <SPEL (bit 12), SMEL (bit 13) in SSTSW> SPEL = 0:Turn OFF the +EL signal SPEL = 1: Turn ON the +EL signal SMEL = 0:Turn OFF the –EL signal SMEL = 1: Turn ON the –EL signal	[SSTSW] (READ) 15 8 - - n n - - - -
Reading the stop cause when the ±EL signal turns on <ESPL (bit 5), ESML (bit 6) in REST> ESPL = 1: Stop by turning ON the +EL signal ESML = 1: Stop by turning ON the –EL signal	[REST] (READ) 7 0 - n n - - - - -
Setting the ±EL input filter <Set FLTR (bit 26) in RENV1> 1: Apply a filter to the ±EL input Apply a filter and any signals shorter than 4 µs pulse width are ignored.	[RENV1] (WRITE) 27 20 - n - - - - - -

Note 1: Operation after turning ON the EL signal may be different for the zero return operation (9-5-1), the zero search operation (9-5-3), and the EL or SL operation mode (9-6). See the description of each operation mode.

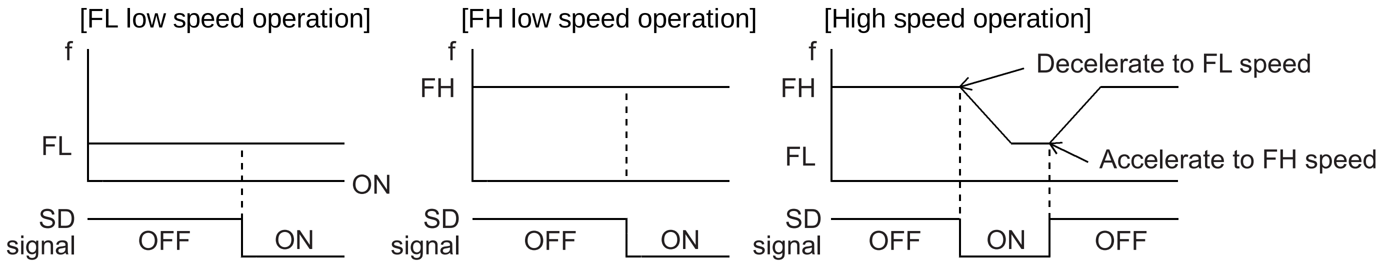
11-5-2. SD signal

If the SD signal input is disabled by setting MSDE in the RMD register (operation mode), the SD signal will be ignored.

If the SD signal is enabled and the SD signal is turned ON while in operation, the axis will: 1) decelerate, 2) latch and decelerate, 3) decelerate and stop, or 4) latch and perform a deceleration stop, according to the setting of SDM and SDLT in the RENV1 register (environment setting 1).

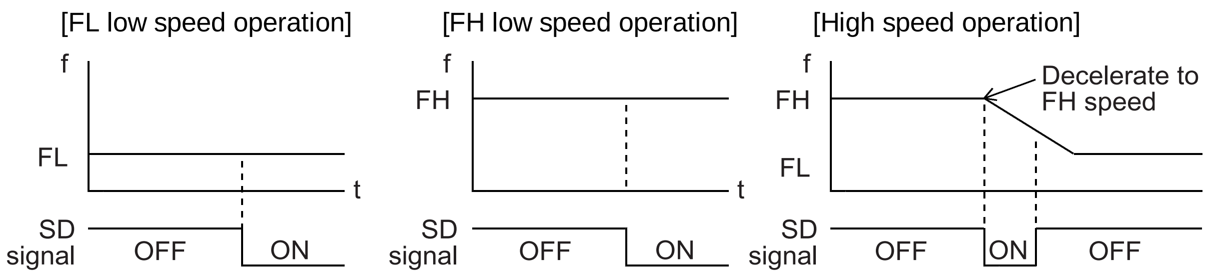
1) Deceleration < SDM (bit 4) = 0, SDLT (bit 5) = 0 in RENV1 register>

- While feeding at low speed, the SD signal is ignored. While in high speed operation the axis decelerates to the FL speed when the SD signal is turned ON. After decelerating, or while decelerating, if the SD signal turns OFF, the axis will accelerate to the FH speed.
- If the SD signal is turned ON when the high speed command is written, the axis will operate at FL speed. When the SD signal is turned OFF, the axis will accelerate to FH speed.



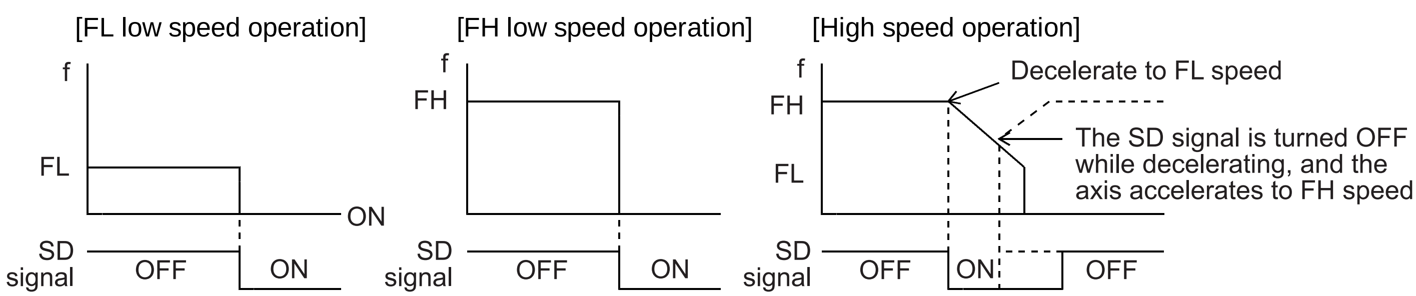
2) Latch, and decelerate <SDM (bit 4) = 0, SDLT (bit 5) = 1 in RENV1 register>

- While feeding at low speed, the SD signal is ignored. While in high speed operation, decelerates to FL speed by turning the SD signal ON. Even if the SD signal is turned OFF after decelerating or while decelerating, the axis will continue moving at FL speed and will not accelerate to FH speed.
- If the SD signal is turned ON while writing a high speed command, the axis will feed at FL speed. Even if the SD signal is turned OFF, the axis will not accelerate to FH speed.



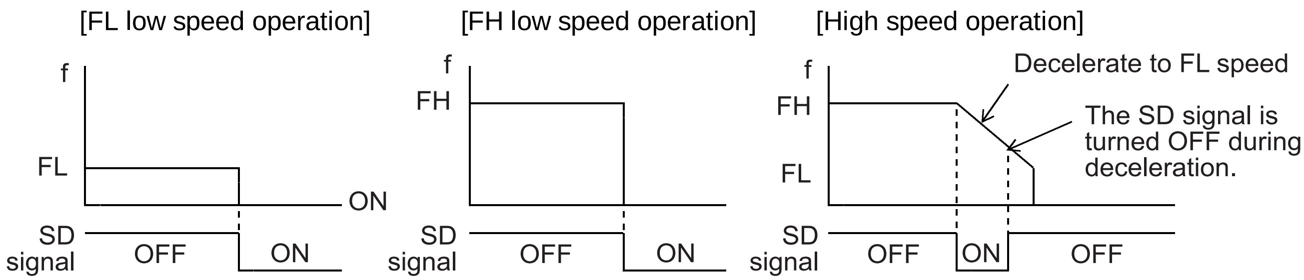
3) Deceleration stop <SDM (bit 4) = 1, SDLT (bit 5) = 0 in RENV1 register>

- If the SD signal is turned ON while in low speed operation, the axis will stop. While in high speed operation, the axis will decelerate to FL speed when the SD signal is turned ON, and then stop. If the SD signal is turned OFF during deceleration, the axis will accelerate to FH speed.
- If the SD signal is turned ON after writing a start command, the axis will complete its operation without another start.
- When stopped, the axis will outputs an $\overline{INT}$ signal.



4) Latched, deceleration stop <SDM (bit 4) = 1, SDLT (bit 5)=1 in RENV1>

- If the SD signal is turned ON while in low speed operation, the axis will stop. If the SD signal is turned ON while in high speed operation, the axis will decelerate to FL speed and then stop. Even if the SD signal is turned OFF during deceleration, the axis will not accelerate.
- If the SD signal is turned ON while writing a start command, the axis will not start moving and the operation will not be completed.
- While stopped, the LSI outputs an $\overline{\text{INT}}$ signal.



The input logic of the SD signal can be changed. If the latched input is set to accept input from the SD signal, and if the SD signal is OFF at the next start, the latch will be reset. The latch is also reset when the latch input is set to zero.

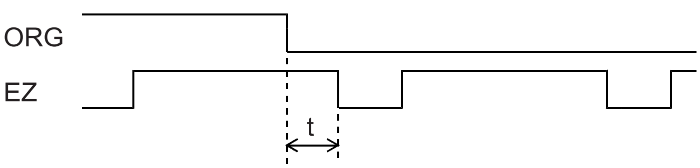
The minimum pulse width of the SD signal is 80 reference clock cycles (4.0 µs) when the input filter is ON. When the input filter is turned OFF, the minimum pulse width is two reference clock cycle (0.1 µs). (When CLK = 19.6608 MHz.)
The latch signal of the SD signal can be monitored by reading SSTSW (sub status). The SD signal terminal status can be monitored by reading RSTS (extension status). By reading the REST register, you can check for an error interrupt caused by the SD signal turning ON.

Enable/disable SD signal input <Set MSDE (bit 8) in RMD> 0: Enable SD signal input 1: Disable SD signal input	[RMD] (WRITE) 15 8 - - - - - - - n
Input logic of the SD signal <Set SDL(bit 6) in RENV1> 0: Negative logic 1: Positive logic	[RENV1] (WRITE) 7 0 - n - - - - - -
Set the operation pattern when the SD signal is turned ON <Set SDM (bit 4) in RENV1> 0: Decelerates on receiving the SD signal and feeds at FL low speed 1: Decelerates and stops on receiving the SD signal	[RENV1] (WRITE) 7 0 - - - n - - - -
Select the SD signal input type <Set SDLT (bit 5) in RENV1> 0: Level input 1: Latch input To release the latch, turn OFF the SD input when next start command is written or select Level input.	[RENV1] (WRITE) 7 0 - - n - - - - -
Reading the latch status of the SD signal <SSD (bit 15) in SSTSW> 0: The SD latch signal is OFF 1: The SD latch signal is ON	[SSTSW] (READ) 15 8 n - - - - - - -
Reading the SD signal < SDIN (bit 15) in the RSTS register> 0: The SD signal is OFF 1: The SD signal is ON	[RSTS] (READ) 15 8 n - - - - - - -
Reading the cause of an $\overline{\text{INT}}$ when stopped by the SD signal <ESSD (bit 10) in RESET> 1: Deceleration stop caused by the SD signal turning ON	[REST] (READ) 15 8 - - - - 0 n - -
Apply an input filter to SD <Set FLTR (bit 26) in RENV1> 1: Apply a filter to the SD input By applying a filter, signals with a pulse width of 4 µs or less will be ignored.	[RENV1] (WRITE) 27 20 - n - - - - - -

11-5-3. ORG, EZ signals

These signals are enabled in the zero return modes (zero return, leave zero position, and zero position search) and in the EZ count operation modes. Specify the operation mode and the operation direction using the RMD register (operation mode).
Since the ORG signal input is latched internally, there is no need to keep the external signal ON.
The ORG latch signal is reset when stopped.
The minimum pulse width of the ORG signal is 80 reference clock cycles (4 µs) when the input filter is ON. When the input filter is turned OFF, the minimum pulse width is two reference clock cycle (0.1 µs). (When CLK = 19.6608 MHz.)
The input logic of the ORG signal and EZ signal can be changed using the RENV1 register (environment setting 1).
The ORG terminal status can be monitored by reading SSTSW (sub status). The EZ terminal status can be monitored by reading the RSTS register (extension status).
For details about the zero return operation modes, see 9-5, "Zero position operation mode."

ORG signal and EZ signal timing



(i) When $t \geq 2 \times T_{CLK}$, counts.
(ii) When $T_{CLK} < t < 2 \times T_{CLK}$, counting is undetermined.
(iii) When $t \leq T_{CLK}$, do not count.
T_{CLK}: Reference clock frequency

Enabling the ORG and EZ signals <Set MOD (bits 0 to 6) in RMD> 001 0000: Zero return in the positive direction 001 0010: Leave zero position in the positive direction 001 0101: Zero position search in the positive direction 010 0100: EZ counting in the positive direction 001 1000: Zero return in the negative direction 001 1010: Leave zero position in the negative direction 001 1101: Zero position search in the negative direction 010 1100: EZ count operation in the negative direction	[RMD] (WRITE) 7 0 0 n n n n n n n
Set the zero return method <Set ORM0 to 3 (bits 0 to 3) in RENV3> See the RENV3 register description	[RENV3] (WRITE) 7 0 - - - - n n n n
Set the input logic for the ORG signal <Set ORGL (bit 7) in RENV1> 0: Negative logic 1: Positive logic	[RENV1] (WRITE) 7 0 n - - - - - - -
Read the ORG signal <SORG (bit 14) in SSTSW> 0: The ORG signal is OFF 1: The ORG signa is ON	[SSTSW] (READ) 15 8 - n - - - - - -
Apply an input filter to ORG <Set FLTR (bit 26) in RENV1> 1: Apply a filter to the ORG input By applying a filter, signals with a pulse width of 4 µs or less will be ignored.	[RENV1] (WRITE) 27 20 - n - - - - - -
Set the EZ count number <Set EZD0 to 3 (bits 4 to 7) in RENV3> Set the zero return completion condition and the EZ count number for counting. Specify the value (the number to count to – 1) in EZD0 to 3. The setting range is 0 to 15.	[RENV3] (WRITE) 7 0 n n n n - - - -
Specify the input logic of the EZ signal <Set EZL (bit 23) in RENV2> 0: Falling edge 1: Rising edge	[RENV2] (WRITE) 23 16 n - - - - - - -
Read the EZ signal <SEZ (bit 10) in RSTS> 0: The EZ signal is OFF 1: The EZ signal is ON	[RSTS] (READ) 15 8 - - - - - n - -

11-6. Servomotor I/F (Case in digital servo)

11-6-1. INP signal

The pulse strings input accepting servo driver systems have a deflection counter to count the difference between command pulse inputs and feedback pulse inputs. The driver controls to adjust the difference to zero. In other words, the effective function of servomotors is to delete command pulses and, even after the command pulses stop, the servomotor systems keep feeding until the count in the deflection counter reaches zero.

This LSI can receive a positioning complete signal (INP signal) from a servo driver in place of the pulse output complete timing, to determine when an operation is complete.
When the INP signal input is used to indicate the completion status of an operation, the $\overline{BSY}$ signal when an operation is complete, the main status (bits 0 to 5 of the MSTSW, stop condition), and the extension status (CND0 to 3, operation status) will also change when the INP signal is input.

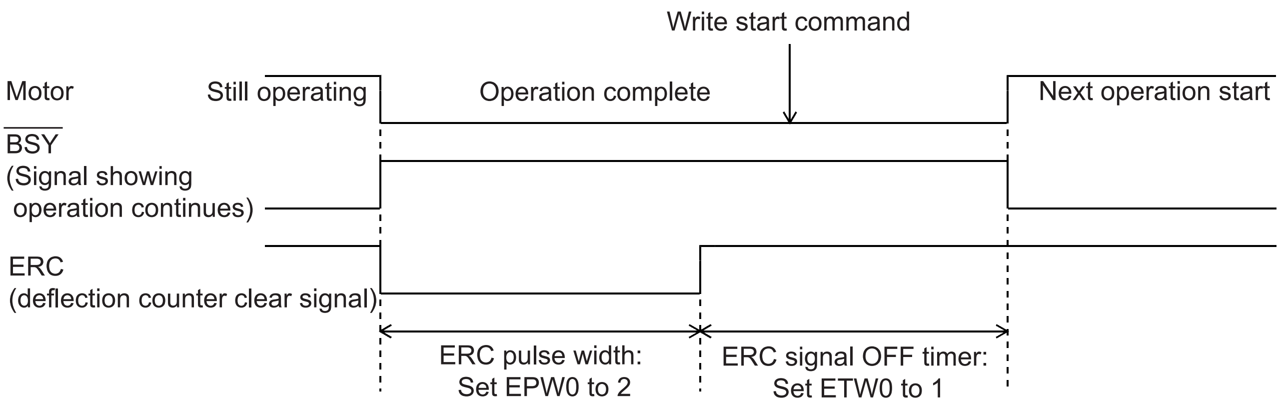
The input logic of the INP signal can be changed.
The minimum pulse width of the INP signal is 80 reference clock cycles (4 µs) when the input filter is ON. If the input filter is OFF, the minimum pulse width will be 2 reference clock cycles (0.1 µs). (When CLK = 19.6608 MHz)
The INP signal can be monitored by reading the RSTS register (extension status).

Set the operation complete delay using the INP signal <Set MINP (bit 9) in RMD> 0: No operation complete delay waiting for the INP signal. 1: Operation complete (status, $\overline{BSY}$) delay until the INP signal turns ON.	[RMD] (WRITE) 15 8 - - - - - - n -
Input logic of the INP signal <Set INPL (bit 22) in RENV1> 0: Negative logic 1: Positive logic	[RENV1] (WRITE) 23 16 - n - - - - - -
Reading the INP signal <SINP (bit 16) in RSTS> 0: The INP signal is OFF 1: The INP signal is ON	[RSTS] (READ) 23 16 0 0 0 0 0 0 0 n
Set the INP input filter <FLTR (bit 26) in RENV1> 1: Apply a filter to the INP input. By applying a filter, pulses less than 4 µs in width are ignored.	[RENV1] (WRITE) 27 20 - n - - - - - -

11-6-2. ERC signal

A servomotor delays the stop until the deflection counter in the driver reaches zero, even after command pulses have stopped being delivered. In order to stop the servomotor immediately, the deflection counter in the servo driver must be cleared.

This LSI can output a signal to clear the deflection counter in the servo driver. This signal is referred to as an "ERC signal." The ERC signal is output as one shot signal or a logic level signal. The output type can be selected by setting the RENV1 register (environment setting 1). If an interval is required for the servo driver to recover after turning OFF the ERC signal (HIGH) before it can receive new command pulses, the ERC signal OFF timer can be selected by setting the RENV1 register.



In order to output an ERC signal at the completion of a zero return operation, set EROR (bit 11) = 1 in the RENV1 register (environment setting 1) to make the ERC signal an automatic output. For details about ERC signal output timing, see the timing waveform in section 9-5-1, "Zero return operation."

In order to output an ERC signal for an immediate stop based on the EL signal, ALM signal, or $\overline{CEMG}$ signal input, or on the emergency stop command (05h), set EROE (bit 10) = 1 in the RENV1 register, and set automatic output for the ERC signal. (In the case of a deceleration stop, the ERC signal cannot be output, even when set for automatic output.)

The ERC signal can be output by writing an ERC output command (24h).

The output logic of the ERC signal can be changed by setting the RENV1 register. Read the RSTS (extension status) register to monitor the ERC signal.

Set automatic output for the ERC signal <Set EROE (bit 10) in RENV1> 0: Does not output an ERC signal when stopped by EL, ALM, or $\overline{CEMG}$ input. 1: Automatically outputs an ERC signal when stopped by EL, ALM, or $\overline{CEMG}$ input.	[RENV1] (WRITE) 15 8 - - - - - n - -
Set automatic output for the ERC signal <Set EROR (bit 11) in RENV1> 0: Does not output an ERC signal at the completion of a zero return operation. 1: Automatically outputs an ERC signal at the completion of a zero return operation.	[RENV1] (WRITE) 15 8 - - - - n - - -
Set the ERC signal output width <Set EPW0 to 2 (bits 12 to 14) in RENV1> 000: 12 µs 100: 13 ms 001: 102 µs 101: 52 ms 010: 408 µs 110: 104 ms 011: 1.6 ms 111: Logic level output	[RENV1] (WRITE) 15 8 - n n n - - - -
Select output logic for the ERC signal <Set ERCL (bit 15) in RENV1> 0: Negative logic 1: Positive logic	[RENV1] (WRITE) 15 8 n - - - - - - -
Specify the ERC signal OFF timer time <Set ETW0 to 1 (bits 16 to 17) in RENV1> 00: 0 µs 10: 1.6 ms 01: 12 µs 11: 104 ms	[RENV1] (WRITE) 23 16 - - - - - - n n
Read the ERC signal <SERCR (bit 9) in STS> 0: The ERC signal is OFF 1: The ERC signal is ON	[RSTS] (READ) 15 8 0 - - - - - n -

Emergency stop command <CMEMG: Operation command> Output an ERC signal	[Operation command] 05h
ERC signal output command <ERCOUT: Control command> Turn ON the ERC signal	[Control command] 24h
ERC signal output reset command <ERCRST: Control command> Turn OFF the ERC signal	[Control command] 25h

11-6-3. ALM signals

Input alarm (ALM) signal.
When the ALM signal turns ON while in operation, the axis will stop immediately or decelerates and stops. However, the axis only decelerates and stops on an ALM signal if it was started with a high speed start. If the axis was started at low speed, the axis will stop immediately when the ALM signal is input.
If the ALM signal is ON when a start command is written, the LSI will not output any pulses.
The minimum pulse width of the ALM signal is 80 reference clock cycles (4 µs) if the input filter is ON. If the input filter is OFF, the minimum pulse width is 2 reference clock cycles (0.1 µs). (When CLK = 19.6608 HMz.)
The input logic of the ALM signal can be changed. The signal status of the ALM signal can be monitored by reading SSTSW (sub status).

Stop method when the ALM signal is ON <Set ALMM (bit 8) in RENV1> 0: Stop immediately when the ALM signal is turned ON 1: Deceleration stop (high speed start only) when the ALM signal is turned ON	[RENV1] (WRITE) 15 8 - - - - - - - n
Input logic setting of the ALM signal <Set ALML (bit 9) in RENV1> 0: Negative logic 1: Positive logic	[RENV1] (WRITE) 15 8 - - - - - - n -
Read the ALM signal <SALM (bit 11) in SSTSW> 0: The ALM signal is OFF 1: The ALM signal is ON	[SSTSW] (READ) 15 8 - - - - n - - -
Reading the cause of a stop when the ALM signal is turned ON <ESAL (bit 7) in RESET> 1: Stop due to the ALM signal being turned ON	[REST] (READ) 7 0 n - - - - - - -
Set the ALM input filter <Set FLTR (bit 26) in RENV1> 1: Apply a filter to the ALM input When a filter is applied, pulse less than 4 µs pulse in width will be ignored.	[RENV1] (WRITE) 27 20 - n - - - - - -

11-7. External start / simultaneous start

11-7-1. $\overline{CSTA}$, $\overline{STA}$ signal

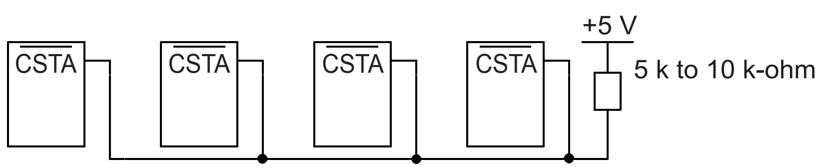
This LSI can start when triggered by an external signal on the $\overline{CSTA}$ and $\overline{STA}$ terminals. Set MSY (bits 18 and 19) = 01 in the RDM register (operation mode) and the LSI will start feeding when the $\overline{CSTA}$ or $\overline{STA}$ goes LOW.
The bidirectional $\overline{CSTA}$ terminal can be used for a simultaneous start from multiple LSIs. The $\overline{STA}$ terminal (input) can be used for starting one axis from an external signal. $\overline{CSTA}$ stands for "Common $\overline{STA}$ (start)."
When you want to control multiple axes using more than one LSI, connect the $\overline{CSTA}$ terminal on each LSI and set the axes to "waiting for $\overline{STA}$ input", to start them all at the same time. In this example a start signal can be output through the $\overline{CSTA}$ terminal. The combined use of the $\overline{CSTA}$ and $\overline{STA}$ terminals is also possible.
The input logic on the $\overline{CSTA}$ and $\overline{STA}$ terminals cannot be changed.
By setting the RIRQ register (event interrupt cause), the $\overline{INT}$ signal can be output together with a simultaneous start (when the $\overline{STA}$ input is ON). By reading the RIST register, the cause of an event interrupt can be checked.
The operation status (waiting for $\overline{STA}$ input), and status of the $\overline{STA}$ terminal (OR of the $\overline{CSTA}$ and $\overline{STA}$ signals) can be monitored by reading the RSTS register (extension status).

<How to make a simultaneous start>
Set MSY0 to 1 (bits 18 and 19) in the RMD register for the axes you want to start. Write a start command and put the LSI in the "waiting for $\overline{STA}$ input "status. Then, start the axes simultaneously by either of the methods described below.

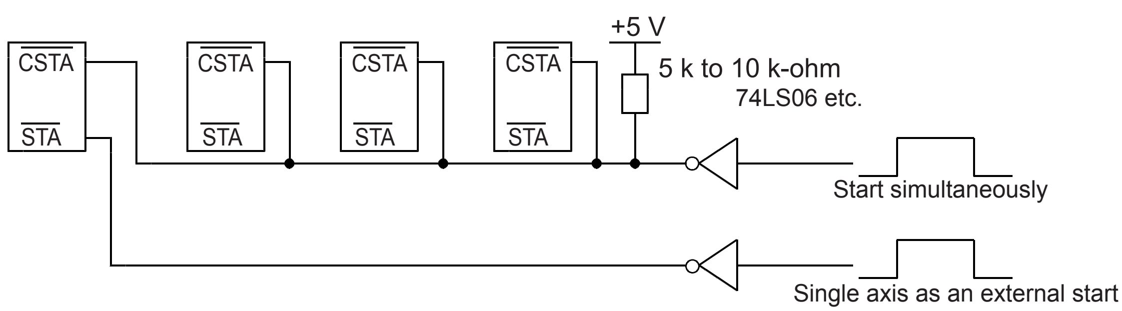
1) By writing a simultaneous start command, the LSI will output a one shot signal of 8 reference clock cycles (approx. 0.4 µs when CLK = 19.6608 MHz) from the $\overline{CSTA}$ terminal.
2) Input hardware signal from outside.
   Supply a hardware signal by driving the terminal with open collector output (74LS06 or equivalent).

$\overline{CSTA}$/$\overline{STA}$ signals can be supplied as level trigger or edge trigger inputs. However, when level trigger input is selected, if $\overline{CSTA}$ = L or $\overline{STA}$ = L and a start command is written, the axis will start immediately.
After connecting the $\overline{CSTA}$ terminals on each LSI, each axis can still be started independently using start commands.
To release the "waiting for $\overline{STA}$ input" condition, write an immediate stop command (49h).

1) To start axes controlled by different LSIs simultaneously, connect the LSIs as follows.



2) To start simultaneously from an external circuit, or use a single axis as an external start, connect the LSIs as follows.



Supply a one shot input signal with a pulse width of at least 4 reference clock cycles (approx. 0.2 µs when CLK = 19.6608).

Setting the Waiting for $\overline{STA}$ input <MSY0 to 1 (bits 18 and 19) in RMD> 01: Start by inputting a $\overline{CSTA}$ signal	[RMD] (WRITE) 23 16 - - - - n n - -
Specify the input specification for the $\overline{STA}$ signal <Set STAM (bit 18) in RENV1> 0: Level trigger input for the $\overline{CSTA}$ or $\overline{STA}$ signal 1: Edge trigger input for the $\overline{CSTA}$ or $\overline{STA}$ signal	[RENV1] (WRITE) 23 16 - - - - - n - -
Read the $\overline{STA}$ signal <SSTA (bit 5) in RSTS> 0: The $\overline{CSTA}$ or $\overline{STA}$ signal is OFF 1: The $\overline{CSTA}$ or $\overline{STA}$ signal is ON	[RSTS] (READ) 7 0 - - n - - - - -
Read the operation status <CND (bits 0 to 3) in RSTS> 0010: Waiting for $\overline{STA}$ input	[RSTS] (READ) 7 0 - - - - n n n n
Set an event interrupt cause <Set IRSA (bit 18) in RIRQ> 1: Output an $\overline{INT}$ signal when the $\overline{STA}$ input is ON.	[RIRQ] (WRITE) 23 16 - - - - - n - -
Reading the event interrupt cause <ISSA (bit 19) in RIST> 1: When the $\overline{STA}$ signal is ON.	[RIST] (READ) 23 16 0 0 0 0 n - - -
Simultaneous start command <CMSTA: Operation command> Output a one shot pulse 8 reference clock cycles long from the $\overline{CSTA}$ terminal.	[Operation command] 06h
Local axis only, simultaneous start command <SPSTA: Operation command> Used the same way as when an $\overline{STA}$ signal is supplied, for a local axis only.	[Operation command]] 2Ah

11-7-2. PCS signal

The PCS input is a terminal originally used for the target position override 2 function. By setting PCSM (bit 30) = 1 in the RENV1 register (environment setting 1), and MSY (bits 18 and 19) = 01 in the RMD register (operation mode), the PCS input cannot be used as a $\overline{STA}$ signal for individual axes.
The input logic of the PCS input signal can be changed. The terminal status can be monitored by reading the RSTS register (extension status).
This function is provided in order to make the chip compatible with the PCL6045. The PCL6025 uses $\overline{STA}$ terminals that are available for each axis.

Specify the function of the PCS signal <Set PCSM (bit 30) in RENV1> 1: Make PCS input effective on only the local axis.	[RENV1] (WRITE) 31 24 - n - - - - - -
Set the Waiting for $\overline{CSTA}$ input <Set MSY0 to 1 (bits 18 and 19) in RMD> 01: Start on a $\overline{CSTA}$ input.	[RMD] (WRITE) 23 16 - - - - n n - -
Set the input logic of the PCS signal <Set PCSL (bit 24) in RENV1> 0: Negative logic 1: Positive logic	[RENV1] (WRITE) 31 24 - - - - - - - n
Read the PCS signal <SPCS (bit 8) in RSTS> 0: The PCS signal is OFF 1: The PCS signal is ON	[RSTS] (READ) 15 8 - - - - - - - n

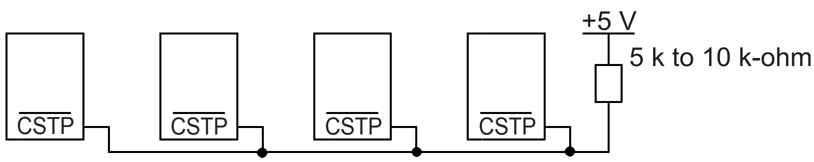
11-8. External stop / simultaneous stop

This LSI can execute an immediate stop or a deceleration stop triggered by an external signal using the $\overline{\text{CSTP}}$ terminal. Set MSPE (bit 24) = 1 in the RMD register (operation mode) to enable a stop from a $\overline{\text{CSTP}}$ input. The axis will stop immediately or decelerate and stop when the $\overline{\text{CSTP}}$ terminal is LOW. However, a deceleration stop is only used for a high speed start. When the axis is started at low speed, the signal on the $\overline{\text{CSTP}}$ terminal will cause an immediate stop.
The input logic of the $\overline{\text{CSTP}}$ terminal cannot be changed.
When multiple LSIs are used to control multiple axes, connect all of the $\overline{\text{CSTP}}$ terminals from each LSI and input the same signal so that the axes which are set to stop on a $\overline{\text{CSTP}}$ input can be stopped simultaneously. In this case, a stop signal can also be output from the $\overline{\text{CSTP}}$ terminal.
When an axis stops because the $\overline{\text{CSTP}}$ signal is turned ON, an $\overline{\text{INT}}$ signal can be output. By reading the REST register, you can determine the cause of an error interrupt. You can monitor $\overline{\text{CSTP}}$ terminal status by reading the RSTS register (extension status).

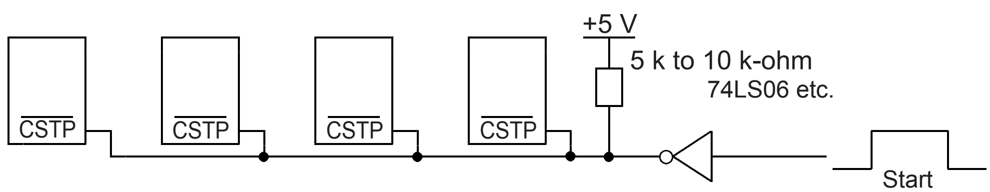
<How to make a simultaneous stop>
Set MSPE (bit 24) = 1 in the RMD register for each of the axes that you want to stop simultaneously. Then start these axes.
Stop these axes using either of the following two methods.
1) By writing a simultaneous stop command, the $\overline{\text{CSTP}}$ terminal will output a one shot signal 8 reference clock cycles in length (approx. 0.4 µs when CLK = 19.6608 MHz).
2) Supply an external hardware signal
Supply a hardware signal using an open collector output (74LS06 or equivalent).

Even when the $\overline{\text{CSTP}}$ terminals on LSIs are connected together, each axis can still be stopped independently by using the stop command.

1) Connect the terminals as follows for a simultaneous stop among different LSIs.



2) To stop simultaneously using an external circuit, connect as follows.



Supply a one shot signal 4 reference clock cycles or more in length (approx. 0.2 µs when CLK = 19.6608 MHz).

Setting to enable $\overline{\text{CSTP}}$ input <Set MSPB (bit 24) in RMD> 1. Enable a stop from the $\overline{\text{CSTP}}$ input. (Immediate stop, deceleration stop)	[RMD] (WRITE) 31 24 0 0 0 0 0 - - n
Specify the stop method to use when the $\overline{\text{CSTP}}$ signal is turned ON. <Set STPM (bit 19) in RENV1> 0: Immediate stop when the $\overline{\text{CSTP}}$ signal is turned ON. 1: Deceleration stop when the $\overline{\text{CSTP}}$ signal is turned ON.	[RENV1] (WRITE) 23 16 - - - - n - - -
Read the $\overline{\text{CSTP}}$ signal <SSTP (bit 6) in RSTS> 0: The $\overline{\text{CSTP}}$ signal is OFF 1: The $\overline{\text{CSTP}}$ signal is ON	[RSTS] (READ) 7 0 - n - - - - - -

Read the cause of an error input < ESSP (bit 8) in REST> 1. When stopped because the $\overline{CSTP}$ signal turned ON.	REST] (READ) 16 8 - - - - - - - n
Simultaneous stop command <CMSTP: Operation command> Outputs a one shot pulse of 8 reference clock cycles in length from the $\overline{CSTP}$ terminal.	[Operation command] 07h

11-9. Emergency stop

This LSI has a $\overline{CEMG}$ input terminal for use as an emergency stop signal.
While in operation, if the $\overline{CEMG}$ input goes LOW or if you write an emergency stop command, both the X and Y axes will stop immediately. While the $\overline{CEMG}$ input remains LOW, neither axis can be operated.
The logical input of the $\overline{CEMG}$ terminal cannot be changed.

When the axes are stopped because the $\overline{CEMG}$ input was turned ON, the LSI will output an $\overline{INT}$ signal. By reading the REST register, the cause of the error interruption can be determined.
The status of the $\overline{CEMG}$ terminal can be monitored by reading the REST register (extension status).

Read the $\overline{CEMG}$ signal <SEMG (bit 7) in RSTS> 0: The $\overline{CEMG}$ signal is OFF 1: The $\overline{CEMG}$ signal is ON	[RSTS] (READ) 7 0 n - - - - - - -
Read the cause of an error interrupt <ESEM (bit 9) in REST> 1. Stopped when the $\overline{CEMG}$ signal was turned ON.	[REST] (READ) 16 8 - - - - - - n -
Emergency stop command <CMEMG: Operation command> The operation is the same as when a $\overline{CEMG}$ signal is input.	[Operation command] 05h

11-10. Counter

11-10-1. Counter type and input method

In addition to the positioning counter, this LSI contains four other counters. These counters offer the following functions.

- ♦ Control command position and mechanical position
- ♦ Detect a stepper motor that is "out of step" using COUNTER3 (deflection counter) and a comparator.
- ♦ Output a synchronous signal using COUNTER4 (general-purpose) and a comparator.

The positioning counter is loaded with an absolute value for the RMV register (target position) with each start command, regardless of the operation mode selected. It decrements the value with each pulse that is output. However, if MPCS (bit 14) of the RMD register (operation mode) is set to 1 and a position override 2 is executed, the counter does will not decrement until the PCS input turned ON.

Input to COUNTER1 is exclusively for output pulses. However COUNTERS2 to 4 can be selected as follows by setting the RENV3 register (environment setting 3).

	COUNTER1	COUNTER2	COUNTER3	COUNTER4
Counter name	Command position	Mechanical position	Deflection	General-purpose
Counter type	Up/down counter	Up/down counter	Deflection counter	Up/down counter
Number of bits	28	28	16	28
Output pulse	Possible	Possible	Possible	Possible
Encoder (EA/EB) input	Not possible	Possible	Possible	Possible
Pulsar (PA/PB) input	Not possible	Possible	Possible	Possible
1/2 of reference clock	Not possible	Not possible	Not possible	Not possible

Specify COUNTER2 (mechanical position) input <CI20 to 21 (bit 8 & 9) in RENV3> 00: EA/EB input 01: Output pulses 10: PA/PB input	[RENV3] (WRITE) 15 8 - - - - - - n n
Set COUNTER3 (deflection) input <CI30 to 31 (bit 10 & 11) in RENV3> 00: Measure the deflection between output pulses and EA/EB input 01: Measure the deflection between output pulses and PA/PB input 10: Measure the deflection between EA/EB input and PA/PB input	[RENV3] (WRITE) 15 8 - - - - n n - -
Set COUNTER4 (general-purpose) input <CI40 to 41 (bit 12 & 13) in RENV3> 00: Output pulses 01: EA/EB input 10: PA/PB input 11: Reference clock (CLK) / 2.	[RENV3] (WRITE) 15 8 - - n n - - - -

The EA/EB and PA/PB input terminal, that are used as inputs for the counter, can be set for one of two signal input types by setting the RENV2 (environment setting 2) register.

1) Signal input method: Input 90° phase difference signals (1x, 2x, 4x)
   Counter direction: Count up when the EA input phase is leading. Count down when the EB input phase is leading.
2) Signal input method: Input 2 sets of positive and negative pulses.
   Counter direction: Count up on the rising edge of the EA input. Count down on the falling edge of the EB input.

The counter direction or EA/EB and PA/PB input signals can be reversed.

The LSI can be set to sense an error when both the EA and EB inputs, or both the PA and PB inputs change simultaneously, and this error can be detected using the REST (error interrupt cause) register.

Setting the EA/EB input <Set EIM0 to 1 (bit 20 & 21) in RENV2> 00: 90˚ phase difference, 1x 01: 90˚ phase difference, 2x 10: 90˚ phase difference, 4x 11: 2 sets of up or down input pulses	[RENV2] (WRITE) 23 16 - - n n 0 0 - -
Specify the EA/EB input count direction <Set to EDIR (bit 22) in RENV2> 0: Count up when the EA phase is leading. Or, count up on the rising edge of EA. 1: Count up when the EB phase is leading. Or, count up on the rising edge of EB.	[RENV2] (WRITE) 23 8 - n - - 0 0 - -
Specify the PA/PB input <Set to PIM0 to 1 (bit 24 & 25) in RENV2> 00: 90˚ phase difference, 1x 01: 90˚ phase difference, 2x 10: 90˚ phase difference, 4x 11: 2 sets of up or down input pulses	[RENV2] (WRITE) 31 24 0 0 0 0 0 - n n
Specify the PA/PB input count direction <Set to PDIR (bit 26) in RENV2> 0: Count up when the PA phase is leading. Or, count up on the rising edge of PA. 1: Count up when the PB phase is leading. Or, count up on the rising edge of PB.	[RENV2] (WRITE) 31 24 0 0 0 0 0 n - -
Reading EA/EB, PA/PB input error <ESEE (bit 16), ESPE (bit 17) in the REST> ESEE (bit 16) = 1: An EA/EB input error occurred. ESPE (bit 17) = 1: An PA/PB input error occurred.	[REST] (READ) 23 16 0 0 0 0 0 0 n n

11-10-2. Counter reset

All the counters can be reset using any of the following three methods.

1) When the CLR input signal turns ON (set in RENV3).
2) When a zero return is executed (set in RENV3).
3) When a command is written.

The CLR input timing can be set in RENV1 (environment setting 1). An $\overline{INT}$ signal can be output when a CLR input is the cause of an event interrupt.

Action when the CLR signal turns ON <Set CU1C to 4C (bit 16 to 19) in the RENV3> CU1C (bit 16) =1: Reset COUNTER1 (command position). CU2C (bit 17) =1: Reset COUNTER2 (mechanical position). CU3C (bit 18) =1: Reset COUNTER3 (deflection). CU4C (bit 19) =1: Reset COUNTER4 (general-purpose).	[RENV3] (WRITE) 23 16 - - - - n n n n
Action when a zero return is complete <Set CU1R to 4R (bit 20 to 23) in RENV3> CU1R (bit 20) =1: Reset COUNTER1 (command position). CU2R (bit 21) =1: Reset COUNTER2 (mechanical position) CU3R (bit 22) =1: Reset COUNTER3 (deflection) CU4R (bit 23) =1: Reset COUNTER4 (general-purpose)	[RENV3] (WRITE) 23 16 n n n n - - - -
Action for the CLR signal <Set CLR0 and 1 (bit 20 & 21) in RENV1> 00: Clear on the falling edge 01: Clear on the rising edge 10: Clear on a LOW level 11: Clears on a HIGH level	[RENV1] (WRITE) 23 16 - - n n - - - -
Reading the CLR signal <SCLR (bit 13) in RSTS> 0: The CLR signal is OFF 1: The CLR signal is ON	[RSTS] (READ) 15 8 - - n - - - - -
Set event interrupt cause <Set IRCL (bit 13) in RIRQ> 1: Output an $\overline{INT}$ signal when resetting the counter value by turning the CLR signal ON.	[RIRQ] (WRITE) 15 8 - - n - - - - -
Read the event interrupt cause <ISCL (bit 13) in RIST> 1: When you want to reset the counter value by turning ON the CLR signal.	[RIST] (READ) 15 8 - - n - - - - -

Counter reset command <CUN1R to CUN4R: Control command>	[Control command]
Set COUNTER1 (command position) to zero.	20h
Set COUNTER2 (mechanical position) to zero.	21h
Set COUNTER3 (deflection) to zero.	22h
Set COUNTER4 (general-purpose) to zero.	23h

11-10-3. Latch the counter and count condition

All the counters can latch their counts using any of the following methods. The setting is made in RENV5 (environment setting 5) register.

1) Turn ON the LTC signal.
2) Turn ON the ORG signal.
3) When the conditions for Comparator 4 are satisfied.
4) When the conditions for Comparator 5 are satisfied.
5) When a command is written.

The current speed can also be latched instead of COUNTER3 (deflection). Items 1) to 4) above can also be latched by hardware timing.

The LTC input timing can be set by in RENV1 (environment setting 1). An $\overline{INT}$ signal can be output when a counter value is latched by turning ON the LTC signal or the ORG signal. This allows you to identify the cause of an event interrupt.

Specify the latch method for a counter (1 to 4) <Set LTM0 to 1 (bit 12 to 13) in RENV5> 00: Turn ON the LTC signal. 01: Turn ON the ORG signal. 10: When the conditions for Comparator 4 are satisfied. 11: When the conditions for Comparator 5 are satisfied	[RENV5] (WRITE) 15 8 - - n n - - - -
Specify the latch method for the current speed <Set LTFD (bit 14) in RENV5> 1: Latch the current speed instead of COUNTER 3 (deflection).	[RENV5] (WRITE) 15 8 - n - - - - - -
Specify latching using hardware <Set LTOF (bit 15) in RENV5> 1: Do not latch 1) to 4) above with hardware timing.	[RENV5] (WRITE) 15 8 n - - - - - - -
Specify the LTC signal mode <Set LTCL (bit 23) in RENV1> 0: Latch on the falling edge. 1: Latch on the rising edge.	[RENV1] (WRITE) 23 16 n - - - - - - -
Set an event interrupt cause <Set IRLT (bit 14) and IROT (bit 15) in RIRQ> IRLT = 1: Output an $\overline{INT}$ signal when the counter value is latched by the LTC signal being turned ON. IROT = 1: Output an $\overline{INT}$ signal when the counter value is latched by the ORG signal being turned ON.	[RIRQ] (WRITE) 15 8 n n - - - - - -
Read the event interrupt cause <ISLT (bit 14), ISOL (bit 15) in RIST> ISLT = 1: Latch the counter value when the LTC signal turns ON. ISOL = 1: Latch the counter value when the ORG signal turns ON.	[RIST] (READ) 15 8 n n - - - - - -
Read the LTC signal <SLTC (bit 14) in RSTS> 0: The LTC signal is OFF 1: The LTC signal is ON	[RSTS] (READ) 15 8 - n - - - - - -
Counter latch command <LTCH: Control command> Latch the contents of the counters (COUNTER1 to 4).	[Control command] 29h

COUNTER1 (command position) stops when the RMD (operation mode) register is set to stop the counter while in timer mode operation.
COUNTER2 (mechanical position), COUNTER3 (deflection), and COUNTER4 (general-purpose) stop when the RENV3 (environment setting 3) register is set to stop.
By setting the RENV3 register, you can stop counting pulses while performing a backlash or slip correction.
COUNTER4 (general-purpose) can be set to count only during operation (BSY = low) using the RENV3 register.
By specifying 1/2 of the CLK (reference clock) signal, the time after the start can be controlled.

Stopping COUNTER1 (command) <Set MCCE (bit 11) in RMD> 1. Stop COUNTER1 (command position).	[RMD] (WRITE) 15 8 - - - - n - - -
Specify the counting operation for COUNTERS 2 to 4 <Set CU2H to 4H (bits 29 to 31) in RENV3> CU2H (bit 29) = 1: Stop COUNTER2 (mechanical position) CU3H (bit 30) = 1: Stop COUNTER3 (deflection) CU4H (bit 31) = 1: Stop COUNTER4 (general-purpose)	[RENV3] (WRITE) 31 24 n n n 0 - - - -
Setting the counters for backlash or slip correction <Set CU1B to 4B (bits 24 to 27) in RENV3> CU1B (bit 16) = 1: Enable COUNTER1 (command position) CU2B (bit 17) = 1: Enable COUNTER2 (mechanical position) CU3B (bit 18) = 1: Enable COUNTER3 (deflection) CU4B (bit 19) = 1: Enable COUNTER4 (general-purpose)	[RENV3] (WRITE) 31 24 - - - 0 n n n n
Specify the counting conditions for COUNTER4 <Set BSYC (bit 14) in RENV3> 1. Enable COUNTER4 (general-purpose) only while operating (BSY = L).	[RENV3] (WRITE) 15 8 - n - - - - - -

11-11. Comparator

11-11-1. Comparator types and functions

This LSI has 5 circuits/axes using 28-bit comparators. Comparators 1 to 4 can be used as comparison counters and can be assigned as COUNTERS 1 to 4. Comparator 5 can be assigned as COUNTER 1 to 4, a positioning counter, or to track the current speed. There are 9 methods of comparison and 4 types of processing that can occur when the comparator conditions are satisfied.

Specify the comparator conditions in the RENV4 (environment 4) and RENV5 (environment 5) registers. By using these comparators, you can perform the following.

- ♦ Use comparators for INT outputs, external output of comparison data, and for internal synchronous starts
- ♦ Immediate stop and deceleration stop operations.
- ♦ Change operation data to pre-register data (used to change speed while operating).
- ♦ Set up a software limit function.
- ♦ Detect out of step stepper motors using COUNTER3 (deflection) and a comparator.
- ♦ Output a synchronous signal (IDX) using COUNTER4 (general-purpose) and a comparator.

Comparator 5 is equipped with a pre-register. It also can output an $\overline{INT}$ signal when the comparator's conditions are satisfied.

[Comparison data] Each comparator can select the data for comparison from the items in the following table.

Comparison data	Comparator 1		Comparator 2		Comparator 3		Comparator 4		Comparator 5	
		C1C0 to 1		C2C0 to 1		C3C0 to 1		C4C0 to 1		C5C0 to 2
COUNTER1 (command position)	O	"00"	O	"00"	O	"00"	O	"00"	O	"000"
COUNTER2 (mechanical position)	O	"01"	O	"01"	O	"01"	O	"01"	O	"001"
COUNTER3 (deflection)	O	"10"	O	"10"	O	"10"	O	"10"	O	"010"
COUNTER4 (general-purpose)	O	"11"	O	"11"	O	"11"	O	"11"	O	"011"
Positioning counter									O	"100"
Current speed									O	"101"
Pre-register	None		None		None		None		Yes	
Major application	+SL		-SL				IDX output			

- O: Comparison possible. Blank: Comparison not possible.
- +SL, -SL are used for software limits.
- If COUNTER3 (deflection) is selected as the comparison counter, the LSI will compare the absolute value of the counter with the comparator data. (Absolute value range: 0 to 32,767)
- The bit assignments of the comparison data settings are as follows:
  C1C0 to 1 (RENV4 bits 0 & 1), C2C0 to 1 (RENV4 bits 8 & 9), C3C0 to 1 (RENV4 bits 16 & 17), C4C0 to 1 (RENV4 bits 24 & 25), C5C0 to 2 (RENV5 bits 0 to 2)

[Comparison method] Each comparator can be assigned a comparison method from the table below.

Comparison method	Comparator 1		Comparator 2		Comparator 3		Comparaotr 4		Comparator 5	
		C1S0 to 2		C2S0 to 2		C3S0 to 2		C4S0 to 3		C5S0 to 2
Comparator = Comparison counter (regardless of count direction)	O	"001"	O	"001"	O	"001"	O	"0001"	O	"001"
Comparator = Comparison counter (Count up only)	O	"010"	O	"010"	O	"010"	O	"0010"	O	"010"
Comparator = Comparison counter (count down only)	O	"011"	O	"011"	O	"011"	O	"0011"	O	"011"
Comparator > Comparison counter	O	"100"	O	"100"	O	"100"	O	"0100"	O	"100"
Comparator < Comparison counter	O	"101"	O	"101"	O	"101"	O	"0101"	O	"101"
Use for software limits	O	"110"	O	"110"		Prohibited				
IDX (synchronous signal) output (regardless of counting direction)							O	"1000"		
IDX (synchronous signal) output (count up only)							O	"1001"		
IDX (synchronous signal) output (count down only)							O	"1010"		

- O: Comparison possible. Blank: Comparison not possible.
- When used for software limits, Comparator 1 is a positive direction limit and the comparison method is comparator < comparison counter. Comparator 2 is the negative limit value and the comparison method is comparator > comparison counter. Select COUNTER1 (command position) for the comparison counter.
- Comparator 3 must not have C3S0 to 2 set to a value of 110. Setting any of the values may result in failing to satisfy the comparison conditions.
- When C4S0 to 3 = 1000 to 1010 for Comparator 4 <IDX (synchronous signal) output>, select COUNTER4 (general-purpose) for use as the comparison counter. Other counters cannot be used for this function. Enter a positive value for the comparator setting.
- The bit assignments for various comparison methods are as follows:
  C1S0 to 2 (RENV4 bits 2 to 4), C2S0 to 2 (RENV4 bits 10 to 12), C3S0 to 1 (RENV4 bits 18 to 20), C4S0 to 3 (RENV4 bits 26 to 29), C5S0 to 2(RENV5 bits 3 to 5)

[Processing method when comparator conditions are satisfied] The processing method that is used when the conditions are satisfied can be selected from the table below.

Processing method when the conditions are met	Comparator 1	Comparator 2	Comaprator 3	Comparator 4	Comparator 5
	C1D0 to 1	C2D0 to 1	C3D0 to 1	C4D0 to 1	C5D0 to 1
Do nothing	"00"	"00"	"00"	"00"	"00"
Immediate stop operation	"01"	"01"	"01"	"01"	"01"
Deceleration stop operation	"10"	"10"	"10"	"10"	"10"
Change operation data to pre-register data	"11"	"11"	"11"	"11"	"11"

- "Do nothing " is mainly used for INT output, external output of comparison result, or internal synchronous starts.
- The "Change operation data to pre-register data" is mainly used to change the current operation speed.
- The bit assignments to select a processing method are as follows:
  C1D0 to 1 (RENV4 bits 5 & 6), C2D0 to 1 (RENV4 bits 13 & 14), C3D0 to 1 (RENV4 bits 21 & 22), C4D0 to 1 (RENV4 bits 30 & 31), C5D0 to 1 (RENV5 bits 6 & 7)

[How to set the INT output, external output of comparison results, and internal synchronous starting]

Set an event interrupt cause <Set IRC1 to 5 (bit 8 to 12) in RIRQ> IRC1 (bit 8) = 1: Output $\overline{INT}$ signal when the Comparator 1 conditions are satisfied. IRC2 (bit 9) = 1: Output $\overline{INT}$ signal when the Comparator 2 conditions are satisfied. IRC3 (bit 10) = 1: Output $\overline{INT}$ signal when the Comparator 3 conditions are satisfied. IRC4 (bit 11)= 1: Output $\overline{INT}$ signal when the Comparator 4 conditions are satisfied. IRC5 (bit 12)= 1: Output $\overline{INT}$ signal when the Comparator 5 conditions are satisfied.	[RIRQ] (WRITE) 15 8 - - - n n n n n
Read the event interrupt cause <ISC1 to 5 (bit 8 to 12) in RIST> IRC1 (bit 8) = 1: When the Comparator 1 conditions are satisfied. IRC2 (bit 9) = 1: When the Comparator 2 conditions are satisfied. IRC3 (bit 10) = 1: When the Comparator 3 conditions are satisfied. IRC4 (bit 11) = 1: When the Comparator 4 conditions are satisfied. IRC5 (bit 12) = 1: When the Comparator 5 conditions are satisfied.	[RIST] (READ) 15 8 - - - n n n n n
Read the comparator condition status <SCP1 to 5 (bits 8 to 12) in MSTSW> SCP1 (bit 8) = 1: When the Comparator 1 conditions are satisfied. SCP2 (bit 9) = 1: When the Comparator 2 conditions are satisfied. SCP3 (bit 10) = 1: When the Comparator 3 conditions are satisfied. SCP4 (bit 11) = 1: When the Comparator 4 conditions are satisfied. SCP5 (bit 12) = 1: When the Comparator 5 conditions are satisfied.	[MSTSW] (READ) 15 8 - - - n n n n n
Specify the P3/CP1 (+SL) terminal specifications <P3M0 to 1 (bits 6 & 7) in RENV2> 00: General-purpose input 01: General-purpose output 10: Output a CP1 (Comparator 1 conditions satisfied) signal using negative logic. 11: Output a CP1 (Comparator 1 conditions satisfied) signal using positive logic.	[RENV2] (WRITE) 7 0 n n - - - - - -
Specify the P4/CP2 (-SL) terminal specifications <P4M0 to 1 (bits 8 & 9) in RENV2> 00: General-purpose input 01: General-purpose output 10: Output CP2 (Comparator 2 conditions satisfied) signal using negative logic. 11: Output CP2 (Comparator 2 conditions satisfied) signal using positive logic.	[RENV2] (WRITE) 15 8 - - - - - - n n
Specify the P5/CP3 terminal specifications <Set P5M0 to 1 (bits 10 & 11) in RENV2> 00: General-purpose input 01: General-purpose output 10: Output CP3 (Comparator 3 conditions satisfied) signal using negative logic. 11: Output CP3 (Comparator 3 conditions satisfied) signal using positive logic.	[RENV2] (WRITE) 15 8 - - - - n n - -
Specify the P6/CP4 terminal specifications <Set P6M0 to 1 (bits 12 & 13) in RENV2> 00: General-purpose input 01: General-purpose output 10: Output CP4 (Comparator 4 conditions satisfied) signal using negative logic. 11: Output CP4 (Comparator 4 conditions satisfied) signal using positive logic.	[RENV2] (WRITE) 15 8 - - n n - - - -
Specify the P7/CP5 terminal specifications <Set P7M0 to 1 (bits 14 & 15) in RENV2> 00: General-purpose input 01: General-purpose output 10: Output CP5 (Comparator 5 conditions satisfied) signal using negative logic. 11: Output CP5 (Comparator 5 conditions satisfied) signal using positive logic.	[RENV2] (WRITE) 15 8 n n - - - - - -
Specify the output timing for an internal synchronous signal <Set SYO1 to 3 (bits 16 to 19) in RENV5> 0001: When the Comparator 1 conditions are satisfied. 0010: When the Comparator 2 conditions are satisfied. 0011: When the Comparator 3 conditions are satisfied. 0100: When the Comparator 4 conditions are satisfied. 0101: When the Comparator 5 conditions are satisfied. 1000: When the acceleration starts. 1001: When the acceleration is complete. 1010: When the deceleration starts 1011: When the deceleration is complete. Others: Turn OFF internal synchronous output signal	[RENV5] (WRITE) 23 16 - - - - n n n n

11-11-2. Software limit function

A software limit function can be set up using comparators 1 and 2.
Select COUNTER1 (command position) as a comparison counter for comparators 1 and 2.
Use Comparator 1 for a positive direction limit and Comparator 2 for a negative direction limit to stop the axis based on the results of the comparator and the operation direction.
When the software limit function is used the following process can be executed.

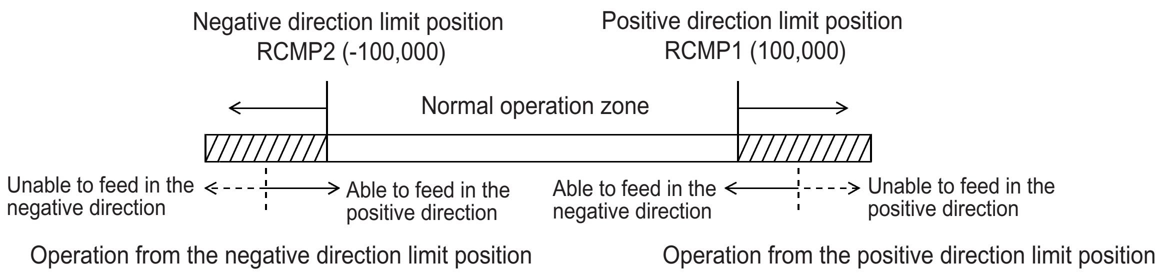
1) Stop pulse output immediately
2) Decelerate and then stop pulse output

If a software limit is ON while writing a start command, the axis will not start to move in the direction in which the software limit is enabled. However, it can start in the opposite direction.

[Setting example]

RENV4=00007070h: Use Comparator 1 as positive direction software limit. Use Comparator 2 as negative direction software limit.
Set to stop immediately when the software limit is reached.

RCMP1= 100,000: Positive direction limit value



RCMP2= -100,000: Negative direction limit value

If the software limit is enabled, movement on the axis will stop unconditionally, regardless of the setting for "processing method when the Comparator 1 and 2 conditions are satisfied" in RENV4. This is the same as the response to the EL signal.
If a "deceleration stop" is selected as the processing method when the conditions are satisfied, the axis will decelerate and stop only when started at high speed. If any other processing method is selected, the axis will stop immediately.

11-11-3. Out of step stepper motor detection function

If the deflection counter value controlled by the motor command pulses and the feed back pulses from an encoder on a stepper motor exceeds the maximum deflection value, the LSI will declare that the stepper motor is out of step. The LSI monitors stepper motor operation using COUNTER3 (the deflection counter) and a comparator.

The process which takes place after an out of step condition is detected can be selected from the table. [Processing method to use when the comparator conditions are satisfied].

For this function, use an encoder with the same resolution as the stepper motor.

COUNTER3 (deflection) can be cleared by writing a set command to the deflection counter.

There are two methods for inputting a feedback signal: Input 90˚ phase difference signals (1x, 2x, 4x) on the EA/EB terminals, input two sets of positive and negative pulses.

If both EA and EB signals change at the same time, the LSI will treat this as an error and output an $\overline{\text{INT}}$ signal.

[Setting example]

RENV4 = 00360000h: Satisfy the conditions of Comparator 3 < COUNTER3 (deflection)
Stop immediately when the conditions are satisfied.
RCMP3 = 32: The maximum deflection value is "32" pulses.
RIRQ = 00000400h: Output an $\overline{\text{INT}}$ signal when the conditions for Comparator 3 are satisfied.

Specify the EA/EB input <Set EIM0 to 1 (bits 20 & 21) in RENV2> 00: 90˚ phase difference, 1x 01: 90˚ phase difference, 2x 10: 90˚ phase difference, 4x 11: Two sets of up and down input pulses	[RENV2] (WRITE) 23 16 - - n n 0 0 - -
Specify the EA/EB input count direction <Set EDIR (bit 22) in RENV2> 0: When the EA phase is leading, or count up on the EA rising edge. 1: When the EB phase is leading, or count up on the EB rising edge	[RENV2] (WRITE) 23 8 - n - - 0 0 - -
Read the EA/EB input error <ESEE (bit 16) in REST> 1: An EA/EB input error has occurred.	[REST] (READ) 23 16 0 0 0 0 0 0 - n
Counter reset command <CUN3R: Control command> Clear COUNTER3 (deflection) to zero.	[Control command] 22h

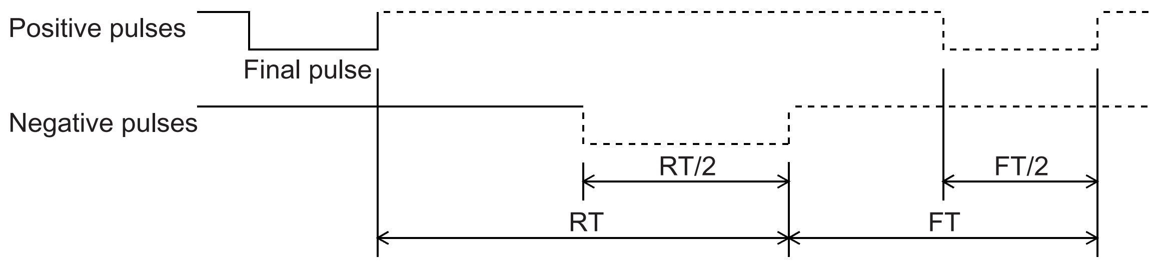
11-12. Backlash correction and slip correction

This LSI has backlash and slip correction functions. These functions output the number of command pulses specified for the correction value in the speed setting in the RFA (correction speed) register.
The backlash correction is performed each time the direction of operation changes. The slip correction function is performed before a command, regardless of the feed direction. The correction amount and method is specified in the RENV6 (environment setting 6) register.
The operation of the counter (COUNTER 1 to 4) can be set using the RENV3 (environment setting 3) register.

Enter the correction value <BR0 to 11 (bits 0 to 11) in RENV6> Backlash or slip correction amount (0 to 4,095).	[RENV6] (WRITE) 15 8 - - - - n n n n 7 0 n n n n n n n n
Set the correction method <ADJ0 to 1 (bits 12 &13) in RENV6> 00: Turn the correction function OFF 01: Backlash correction 10: Slip correction	[RENV6] (WRITE) 15 8 - - n n - - - -
Action for backlash/slip correction <CU1B to 4B (bit 24 to 27) in RENV3> CU1B (bit 16) = 1: Enable COUNTER1 (command position) CU2B (bit 17) = 1: Enable COUNTER2 (mechanical position) CU3B (bit 18) = 1: Enable COUNTER3 (deflection) CU4B (bit 19) = 1: Enable COUNTER4 (general-purpose)	[RENV3] (WRITE) 31 24 - - - 0 n n n n

11-13. Vibration restriction function

This LSI has a function to restrict vibration when stopping by adding one pulse of reverse operation and one pulse of forward operation shortly after completing a command pulse operation.
Specify the output timing for additional pulses in the RENV7 (environment setting 7) register.
When both the reverse timing (RT) and the forward timing (FT) are non zero, the vibration restriction function is enabled.
The dotted lines below are pulses added by the vibration restriction function. (An example in the positive direction)



Specify the reverse operation timing <Set RT0 to 15 (bits 0 to 15) in RENV7> RT range: 0 to 65,535 The units are 32x the reference clock frequency (approx. 1.6 µs when CLK = 19,6608 MHz) Settable range: 0 to approx. 0.1 sec.	[RENV7] (WRITE) 15 8 n n n n n n n n 7 0 n n n n n n n n
Specify the forward operation timing <Set FT0 to 15 (bits 16 to 31) in RENV7> FT range: 0 to 65,535 The units are 32x the reference clock frequency (approx. 1.6 µs when CLK = 19,6608 MHz) Settable range: 0 to approx. 0.1 sec.	[RENV7] (WRITE) 31 24 n n n n n n n n 23 16 n n n n n n n n

11-14. Synchronous starting

This LSI can perform the following operation by setting the RMD (operation mode) register in advance.

- ♦ Start triggered by another axis stopping.
- ♦ Start triggered by an internal synchronous signal from another axis.

The internal synchronous signal output is available with 9 types of timing. They can be selected by setting the RENV5 (environment setting 5) register. By setting the RIRQ (event interrupt cause) register, an $\overline{INT}$ signal can be output at the same time the internal synchronous signal is output. You can determine the cause of event interrupt by reading the RIST register. The operation status can be checked by reading the RSTS (extension status) register.

Specify the synchronous starting method <Set MSY0 to 1 (bits 18 & 19) in RMD> 10: Start with an internal synchronous signal. 11: Start triggered by another axis stopping.	[RMD] (WRITE) 23 16 - - - - n n - -
Specify an axis (setting example) <Set MAX0 to 1 (bits 20 to 21) in RMD) 01: Start when the X axis stops. 10: Start when the Y axis stops.	[RMD] (WRITE) 23 16 - - n n - - - -
Specify the internal synchronous signal output timing <Set SYO1 to 3 (bits 16 to 19) in RENV5> 0001: When the Comparator 1 conditions are satisfied. 0010: When the Comparator 2 conditions are satisfied. 0011: When the Comparator 3 conditions are satisfied. 0100: When the Comparator 4 conditions are satisfied. 0101: When the Comparator 5 conditions are satisfied. 1000: When the acceleration is started. 1001: When the acceleration is complete. 1010: When the deceleration is started. 1011: When the acceleration is complete Others: Internal synchronous output signal is OFF.	[RENV5] (WRITE) 23 16 - - - - n n n n
Specify the input for the internal synchronous signal <Set SYI0 to 1 (bits 20 & 21) in RENV5> 00: Use an internal synchronous signal output by the X axis. 01: Use an internal synchronous signal output by the Y axis.	[RENV5] (WRITE) 23 16 - - n n - - - -
Read the operation status <CND (bits 0 to 3) in RSTS> 0011: Wait for an internal synchronous signal. 0100: Wait for another axis to stop.	[RSTS] (READ) 7 0 - - - - n n n n
Select the event interrupt ($\overline{INT}$ output) cause <Set bit 4 to 12 of RIRQ> IRUS (bit 4) = 1: When the acceleration is started. IRUE (bit 5) = 1: When the acceleration is complete. IRUS (bit 6) = 1: When the acceleration is started. IRUS (bit 7) = 1: When the deceleration is complete. IRC1 (bit 8) = 1: When the Comparator 1 conditions are satisfied. IRC2 (bit 9) = 1: When the Comparator 2 conditions are satisfied. IRC3 (bit 10) = 1: When the Comparator 3 conditions are satisfied. IRC4 (bit 11) = 1: When the Comparator 4 conditions are satisfied. IRC5 (bit 12) = 1: When the Comparator 5 conditions are satisfied.	[RIRQ] (WRITE) 7 0 n n n n - - - - 15 8 - - - n n n n n
Read the event interrupt ($\overline{INT}$ output) cause <Bit 4 to 12 of RIST> IRUS (bit 4) = 1: When the acceleration is started. IRUS (bit 5) = 1: When the acceleration is complete. IRUS (bit 6) = 1: When the deceleration is started. IRUS (bit 7) = 1: When the deceleration is complete. IRC1 (bit 8) = 1: When the Comparator 1 conditions are satisfied. IRC2 (bit 9) = 1: When the Comparator 2 conditions are satisfied. IRC3 (bit 10) = 1: When the Comparator 3 conditions are satisfied. IRC4 (bit 11) = 1: When the Comparator 4 conditions are satisfied. IRC5 (bit 12) = 1: When the Comparator 5 conditions are satisfied.	[RIST] (READ) 7 0 n n n n - - - - 15 8 - - - n n n n n

11-14-1. Start triggered by another axis stopping

If "Y axis stop" is set as the start condition for the X axis, the X axis will start once the Y axis has operated and then stopped.
Example 1 shows the settings when a "Y axis stop" is set as the start condition for the X axis.

[Example 1]
After setting steps 1) to 3), start and stop the Y axis and then the X axis will start.
1) Set MSY0 to 1 (bits 18 & 19) in RMD for the X axis to "11." (Start triggered by another axis stopping)
2) Set MAX0 to 1 (bit 20s to 21) in RMD for the X axis to "10." (When the Y axis stops)
3) Write a start command for the X axis.

In order to use "Another axis stops" as a start condition, the axis specifying this condition (X axis) must be ready to start its process and then it can wait for the other axis to stop. At this point the other axis (the Y axis) can be started and stopped. Therefore, this function cannot use a stop of itself as a starting condition.
For example, if the X and Y axes are performing circular interpolation, and if "Both X and Y axes stop" is set as a start condition in the pre-register for the next operation, when X an Y are "waiting for both axes to stop" (so that they can start the linear interpolation at the end of the circular interpolation), since they are already stopped the change "from operation to stop" will not occur while they are waiting. Therefore the X and Y axes will never start the linear interpolation.

In the case of a continuous interpolation operation, put the next operation into the pre-register without setting any stop condition, so that the axes will continue their operation. The settings for these steps is shown in Example 2. In order to understand this clearly, the example only describes sections that are related to the operations. The settings for speed and acceleration are omitted. The setting for waiting for $\overline{\text{CSTA}}$ input (RMD = 0004_0064h) on the first line is used to start the operation after setting all the operation conditions.

[Setting example 2]
How to set up a continuous interpolation (X-Y axis circular interpolation followed by an X-Y axis linear interpolation)

Step	Register	X axis	Y axis	Description
1	PRMV	10000	10000	X and Y axes perform an circular interpolation operation of a 90° curve with a radius of 10000
	PRIP	10000	0	
	PRMD	0004_0064h	0004_0064h	X and Y axes are waiting for $\overline{\text{CSTA}}$
	Start command: Write 0351h (FH low speed start)			X and Y axes start command
2	PRMV	10000	5000	X and Y axes perform a linear interpolation with an end point (1000, 5000)
	PRMD	0000_0061h	0000_0061h	
	Start command: Write 0351h (FH low speed start)			X and Y axes start command

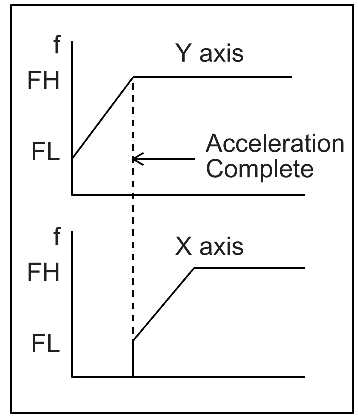
After the settings above are complete, turn ON the $\overline{\text{CSTA}}$ input. The LSI will execute a continuous operation in the order shown below.
1. The X and Y axes perform a CW circular interpolation operation of a 90° curve with a radius of 10000.
2. The X and Y axes perform a linear interpolation (10000, 5000)

11-14-2. Starting from an internal synchronous signal

There are 9 types of internal synchronous signal output timing. They can be selected by setting the RENV5 register.

The monitor signal for the internal synchronous signal can be output externally.

Example 1 below shows how to use the end of an acceleration for the internal synchronous signal.



[Setting example 1]

After completing steps 1) to 3) below, write a start command to the X and Y axes, the X axis will start when the Y axis completes its acceleration.

1) Set MSY0 to 1 (bits 18 &19) in the X axis RMD to 10. (Start with an internal synchronous signal)
2) Set SYI0 to 1 (bits 20 & 21) in the X axis to 01. (Use an internal synchronous signal from the Y axis.)
3) Set SYO0 to 3 (bits 16 to 19) in the Y axis RENV5 to 1001. (Output an internal synchronous signal when the acceleration is complete)

Example 2 shows how to start another axis using the satisfaction of the comparator conditions to generate an internal synchronous signal.

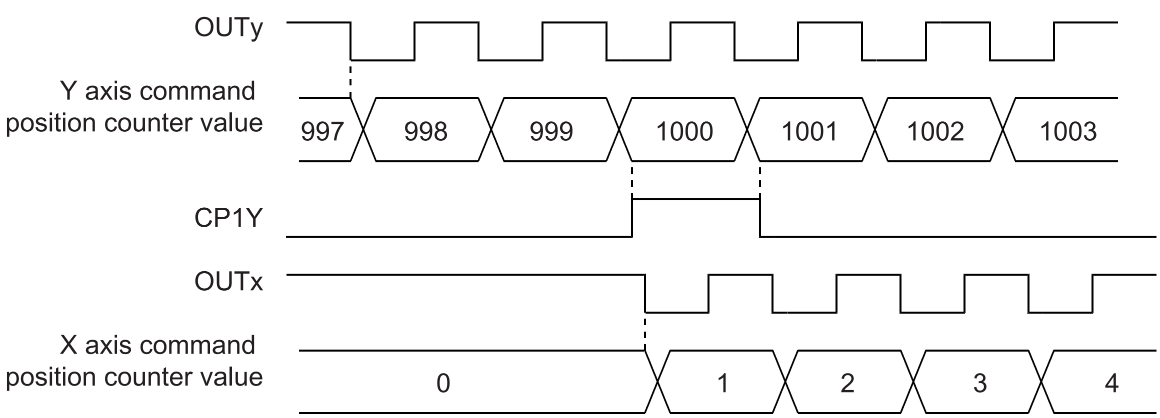
Be careful, since comparator conditions satisfied by timing and the timing of the start of another axis may be different according to the comparison method used by the comparators.

[Example 2]

Use COUNTER1 (command position) and Comparator 1 to start the X axis when the Y axis = 1000.

1) Set MSY0 to 1 (bits 18 & 19) in the Y axis RMD to 10. (Start from an internal synchronous signal)
2) Set SYI0 to 1 (bits 20 & 21) in the X axis RENV5 to 01. (Use an internal synchronous signal from the Y axis)
3) Set SYO0 to 3 (bits 16 to 19) in the Y axis RENV5 to 0001. (Output an internal synchronous signal when the Comparator 1 conditions are satisfied)
4) Set C1C0 to 1 (bits 0 & 1) in the Y axis RENV4 to 00. (Comparator 1 comparison counter is COUNTER1)
5) Set C1S0 to 2 (bits 2 to 4) in the Y axis RENV4 to 001. (Comparison method: Comparator 1 = Comparison counter)
6) Set C1D0 to 1 (bits 5 & 6) in the Y axis RENV4 to 00. (Do nothing when the Comparator 1 condition are satisfied)
7) Set the RCMP1 value of the Y axis to 1000. (Comparison counter value of Comparator 1 is 1000.)
8) Write start commands for the X and Y axes.

The timing chart below shows the period after the Comparator 1 conditions are established and the X axis starts.



Note: In the example above, even if the Y feed amount is set to 2000 and the X feed amount is set to 1000, the X axis will be 1 when the Y axis position equals 1000. Therefore, the operation complete position will be one pulse off for both the X and Y axes. In order to make the operation complete timing the same, set the RCMP1 value to 1001 or set the comparison conditions to "Comparator 1 < comparison counter."

Specify the use of the P0/FUP terminal <Set P0M0 to 1 (bits 0 & 1) in RENV2> 10: Output an FUP (accelerating) signal	[RENV2] (WRITE) 7 0 - - - - - - n n
Specify the use of the P1/FDW terminal <Set P1M0 to 1 (bits 2 & 3) in RENV2> 10: Output an FDW (decelerating) signal	[RENV2] (WRITE) 7 0 - - - - n n - -
Select the output logic for P0 (one shot) / FUP <Set P0L (bit 16) in RENV2> 0: Negative logic 1: Positive logic	[RENV2] (WRITE) 23 16 - - - - 0 0 - n
Select the output logic for P1 (one shot) / FDW <Set P1L (bit 17) in RENV2> 0: Negative logic 1: Positive logic	[RENV2] (WRITE) 23 16 - - - - 0 0 n -
Specify the use of the P3/CP1 (+SL) terminal <Set P3M0 to 1 (bits 6 & 7) in RENV2> 10: Output CP1 (Comparator 1 conditions are satisfied) using negative logic. 11: Output CP1 (Comparator 1 conditions are satisfied) using positive logic.	[RENV2] (WRITE) 7 0 n n - - - - - -
Specify the use of the P4/CP2 (-SL) terminal <Set P4M0 to 1 (bits 8 & 9) in RENV2> 10: Output CP2 (Comparator 2 conditions are satisfied) using negative logic. 11: Output CP2 (Comparator 2 conditions are satisfied) using positive logic.	[RENV2] (WRITE) 15 8 - - - - - - n n
Specify the use of the P5/CP3 terminal <Set P5M0 to 1 (bits 10 & 11) in RENV2> 10: Output CP3 (Comparator 3 conditions are satisfied) using negative logic. 11: Output CP3 (Comparator 3 conditions are satisfied) using positive logic.	[RENV2] (WRITE) 15 8 - - - - n n - -
Specify the use of the P6/CP4 terminal <Set P6M0 to 1 (bits 12 & 13) in RENV2> 10: Output CP4 (Comparator 4 conditions are satisfied) using negative logic. 11: Output CP4 (Comparator 4 conditions are satisfied) using positive logic.	[RENV2] (WRITE) 15 8 - - n n - - - -
Specify the use of the P7/CP5 terminal <Set P7M0 to 1 (bits 14 & 15) in RENV2> 10: Output CP5 (Comparator 5 conditions are satisfied) using negative logic. 11: Output CP5 (Comparator 5 conditions are satisfied) using positive logic.	[RENV2] (WRITE) 15 8 n n - - - - - -

11-15. Output an interrupt signal

This LSI can output an interrupt signal ($\overline{INT}$ signal) : There are 17 types of errors and 19 types of events that can cause an $\overline{INT}$ signal to be output . All of the error causes will always output an $\overline{INT}$ signal. Each of the event causes can be set in the RIRQ register to output an $\overline{INT}$ signal or not.
The LSI continues to output the $\overline{INT}$ signal until all of the causes on the related axes have been cleared.
An error interrupt is cleared when a "REST (error cause) register read out command" is written. An event cause is cleared when an "RIST (event cause) register read command" is written.
To identify the axis on which an interrupt occurred, read the REST and RIST registers for all of the axes being operated. The interrupt flag will also be cleared by reading these registers. However, if the RIRQ register is not enabled for a particular cause of an event, no event interrupt will occur. In this case, you don't need to read the RIST registers.
The interrupt status can be checked by reading the MSTSW (main status).

The $\overline{INT}$ signal output can be masked by setting the RENV1 (environment setting 1) register.
If the $\overline{INT}$ output is masked (INTM = 1 in RENV1), and when the interrupt conditions are satisfied, the status will change. However, the $\overline{INT}$ signal will not go LOW, but will remain HIGH.
While the interrupt conditions are satisfied and if the output mask is turned OFF (INTM = 0 in RENV1), the $\overline{INT}$ signal will go LOW.
If you will not be using the $\overline{INT}$ terminal, leave the terminal open.
If you use multiple LSIs, you are not allowed to make wired OR connections between $\overline{INT}$ terminals.

Read the interrupt status <SERR (bit 4), SINT (bit 5) in MSTSW> SERR = 1: Becomes 1 when an error interrupt occurs. Becomes 0 by reading REST. SINT = 1: Becomes 1 when an event interrupt occurs. Becomes 0 by reading RIST.	[MSTSW] (READ) 7 … 0 - - n n - 0 - -
Set the interrupt mask <INTM (bit 29) in RENV1> 1: Mask INT output.	[RENV1] (WRITE) 31 … 23 - - n - - - - -
Read the cause of the error interrupt <RREST: Read out command> Copy the data in the RESET register (error interrupt cause) to BUF.	[Read command] F2h
Read the event interrupt cause <RRIST: Read out command> Copy the data in the RIST register (event interrupt cause) to BUF.	[Read command] F3h
Set the event interrupt cause <WRIRQ: Write command> Write the BUF data to the RIRQ register (event interrupt cause).	[Write command] ACh

[Error interrupt causes] < The cause of an interrupt makes the corresponding bit "1">

Error interrupt cause	Cause (REST)	
	Bit	Bit name
Stopped by Comparator 1 conditions being satisfied (+SL)	0	ESC1
Stopped by Comparator 2 conditions being satisfied (-SL)	1	ESC2
Stopped by Comparator 3 conditions being satisfied	2	ESC3
Stopped by Comparator 4 conditions being satisfied	3	ESC4
Stopped by Comparator 5 conditions being satisfied	4	ESC5
Stopped by turning ON the +EL input	5	ESPL
Stopped by turning ON the -EL input	6	ESML
Stopped by turning ON the ALM input	7	ESAL
Stopped by turning ON the $\overline{\text{CSTP}}$ input	8	ESSP
Stopped by turning ON the $\overline{\text{CEMG}}$ input	9	ESEM
Deceleration stopped by turning ON the SD input	10	ESSD
(Always 0)	11	Not defined
Stopped by an operation data error.	12	ESDT
Simultaneously stopped with another axis due to an error stop on the other axis during an interpolation operation	13	ESIP
An overflow of PA/PB input buffer counter occurred	14	ESPO
An over range count occurred while positioning in an interpolation operation	15	ESAO
An EA/EB input error occurred.	16	ESEE
An PA/PB input error occurred.	17	ESPE

[Event interrupt causes] < The corresponding interrupt bit is set to 1 and then an interrupt occurred>

Event interrupt cause	Set cause (RIRQ)		Cause (RIST)	
	Bit	Bit name	Bit	Bit name
Automatic stop	0	IREN	0	ISEN
The next operation starts continuously	1	IRNX	1	ISNX
When it is possible to write an operation to the 2nd pre-register	2	IRNM	2	ISNM
When it is possible to write to the 2nd pre-register for Comparator 5	3	IRND	3	ISND
When acceleration starts	4	IRUS	4	ISUS
When acceleration ends	5	IRUE	5	ISUE
When deceleration starts	6	IRDS	6	ISDS
When deceleration ends	7	IRDE	7	ISDE
When the Comparator 1 conditions are satisfied	8	IRC1	8	ISC1
When the Comparator 2 conditions are satisfied	9	IRC2	9	ISC2
When the Comparator 3 conditions are satisfied	10	IRC3	10	ISC3
When the Comparator 4 conditions are satisfied	11	IRC4	11	ISC4
When the Comparator 5 conditions are satisfied	12	IRC5	12	ISC5
When the counter value is reset by a CLR signal input	13	IRCL	13	ISCL
When the counter value is latched by an LTC input	14	IRLT	14	ISLT
When the counter value is latched by an ORG input	15	IROL	15	ISOL
When the SD input is turned ON	16	IRSD	16	ISSD
When the +DR input changes	17	IRDR	17	ISPD
When the -DR input changes			18	ISMD
When the $\overline{\text{CSTA}}$ input is turned ON	18	IRSA	19	ISSA

12. Electrical Characteristics

12-1. Absolute maximum ratings

Item	Symbol	Rating	Unit
Power supply voltage	$V_{dd}5$	-0.3 to +7.0	V
	$V_{dd}3$	-0.3 to +5.0	
Input voltage	V_{IN}	-0.3 to $V_{dd}5$ +0.5	V
Input current	I_{IN}	±30	mA
Storage temperature	Tstg	-60 to +150	˚C

12-2. Recommended operating conditions

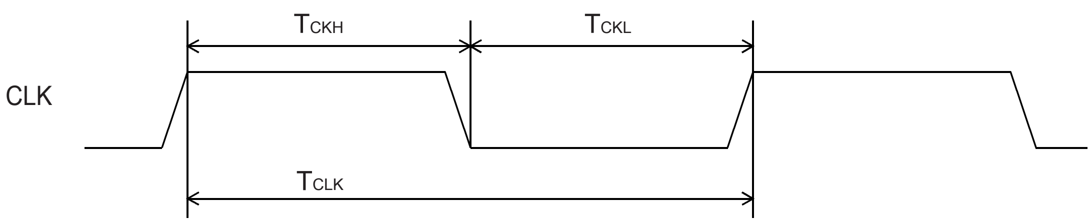
Item	Symbol	Rating	Unit
Power supply voltage	$V_{dd}5$	4.5 to 5.5	V
	$V_{dd}3$	2.97 to 3.63	
Ambient temperature	T J	-40 to +85	˚C

12-3. DC characteristics

Item	Symbol	Condition	Min.	Max.	Unit
Static current consumption	$I_{dd}5$	CLK = 0 Hz, No load		45	μA
	$I_{dd}3$			90	
Current consumption	$I_{dd}5$	CLK = 20 Hz, Output frequency = 6.666667 Mhz, No load		3	mA
	$I_{dd}3$			79	mA
Output leakage current	I_{OZ}		-1	1	μA
Input capacitance				10	pF
LOW input current	I_{IL}		-1		μA
HIGH input current	I_{IH}			1	μA
LOW input current	V_{IL}	Inputs and input/output terminals, except CLK.		0.6	V
		CLK terminal		0.8	V
HIGH input current	V_{IH}	Inputs and input/output terminals, except CLK.	2.4		V
		CLK terminal	4.0		V
LOW output voltage	V_{OL}	I_{OL} = 8 mA		0.4	V
HIGH output voltage	V_{OH}	I_{OH} = -8 mA	$V_{dd}5$-0.4		V
LOW output current	I_{OL}	V_{OL} = 0.4 V		8	mA
HIGH output current	I_{OH}	V_{OH} = 2.4 V	-8		mA
Internal pull up resistance	R_{UP}		30	144	K-ohm

12-4. AC characteristics 1) (reference clock).

Item	Symbol	Condition	Min.	Max.	Unit
Reference clock frequency	f_{CLK}			20	MHz
Reference clock cycle	T_{CLK}		50		ns
Reference clock HIGH width	T_{CKH}		20		ns
Reference clock LOW width	T_{CKL}		20		ns



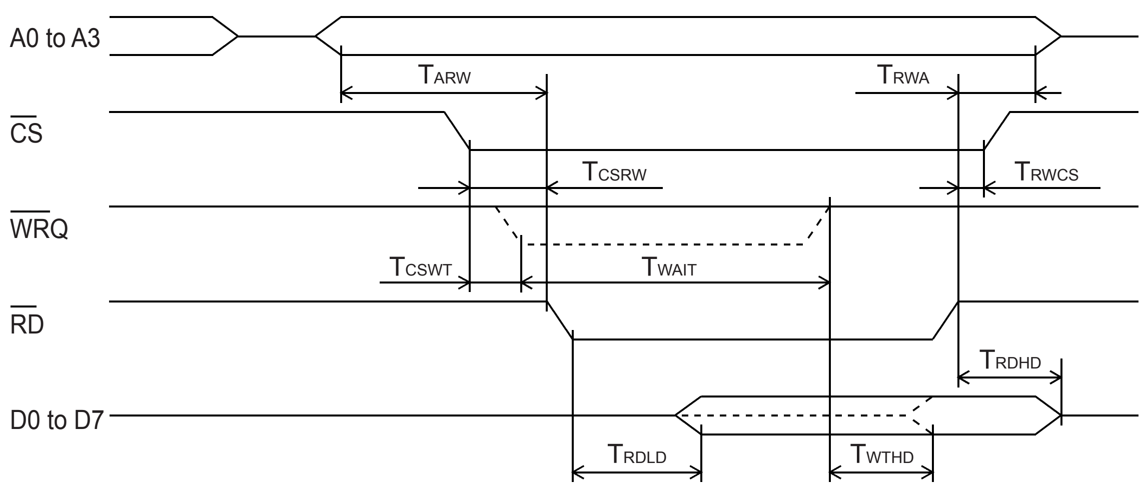
12-5. AC characteristics 2) (CPU I/F)

12-5-1. CPU-I/F 1) (IF1 = H, IF0 = H) Z80

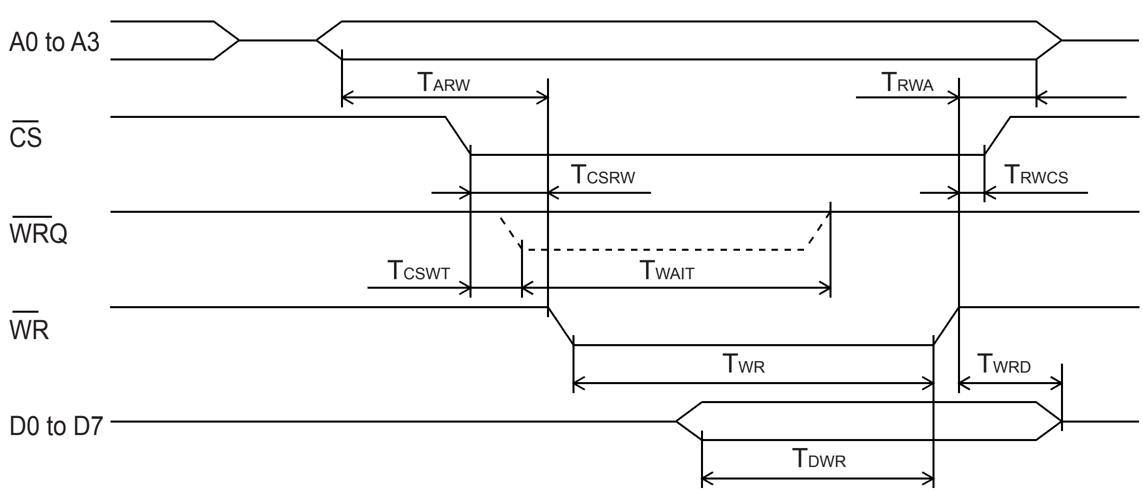
Item	Symbol	Condition	Min.	Max.	Unit
Address setup time for $\overline{RD}$, $\overline{WR}$ ↓	T_{ARW}		17		ns
Address hold time for $\overline{RD}$, $\overline{WR}$ ↑	T_{RWA}		0		ns
$\overline{CS}$ setup time for $\overline{RD}$, $\overline{WR}$ ↓	T_{CSRW}		5		ns
$\overline{CS}$ hold time for $\overline{RD}$, $\overline{WR}$ ↑	T_{RWCS}		0		ns
$\overline{WRQ}$ ON delay time for $\overline{CS}$ ↓	T_{CSWT}	C_L = 40pF		11	ns
$\overline{WRQ}$ signal LOW time	T_{WAIT}			$4T_{CLK}$ +18	ns
Data output delay time for $\overline{RD}$ ↓	T_{RDLD}	C_L = 40pF		15	ns
Data output delay time for $\overline{WRQ}$ ↑	T_{WTHD}	C_L = 40pF		9	ns
Data float delay time for $\overline{RD}$ ↑	T_{RDHD}	C_L = 40pF		10	ns
$\overline{WR}$ signal width	T_{WR}	Note 1	18		ns
Data setup time for $\overline{WR}$ ↑	T_{DWR}		20		ns
Data hold time for $\overline{WR}$ ↑	T_{WRD}		0		ns

Note 1: When a $\overline{WRQ}$ signal is output, the duration will be the interval between $\overline{WRQ}$ = H and $\overline{WR}$ = H.

<Read cycle>



<Write cycle>

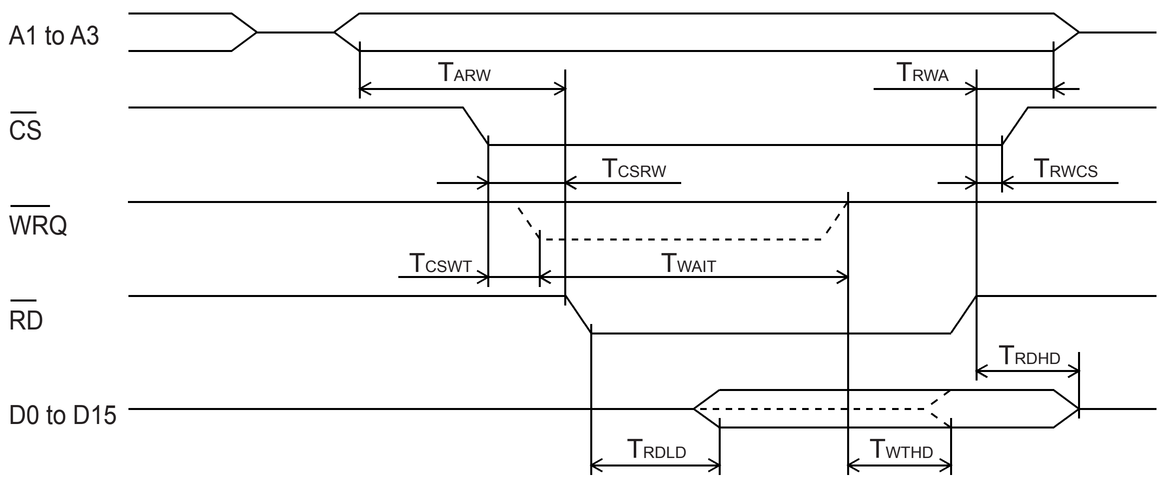


12-5-2. CPU-I/F 2) (IF1 = H, IF0 = L) 8086

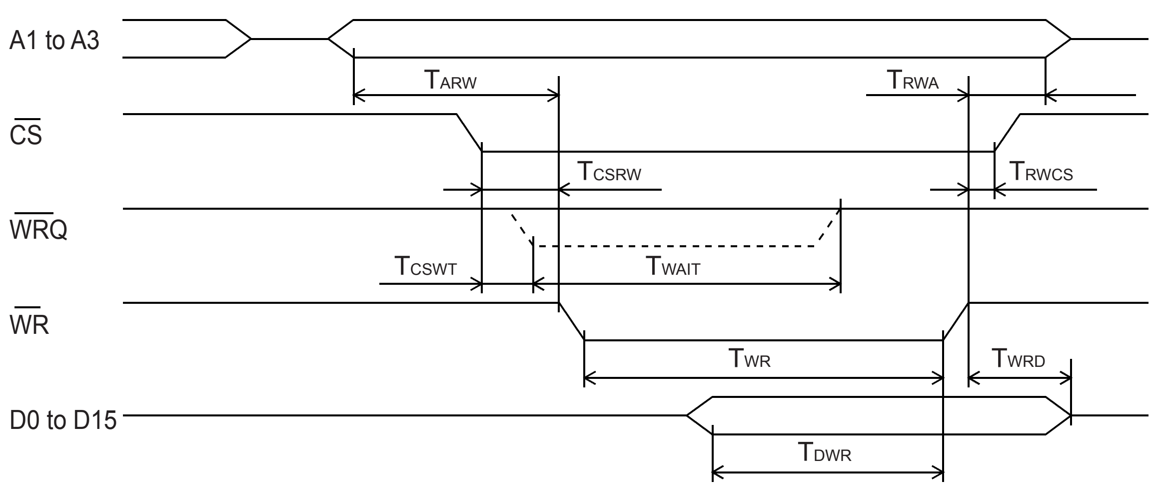
Item	Symbol	Condition	Min.	Max.	Unit
Address setup time for $\overline{RD}$, $\overline{WR}$ ↓	T_{ARW}		17		ns
Address hold time for $\overline{RD}$, $\overline{WR}$ ↑	T_{RWA}		0		ns
$\overline{CS}$ setup time for $\overline{RD}$, $\overline{WR}$ ↓	T_{CSRW}		5		ns
$\overline{CS}$ hold time for $\overline{RD}$, $\overline{WR}$ ↑	T_{RWCS}		0		ns
$\overline{WRQ}$ ON delay time for $\overline{CS}$ ↓	T_{CSWT}	C_L = 40pF		11	ns
$\overline{WRQ}$ signal LOW time	T_{WAIT}			$4T_{CLK}$ +18	ns
Data output delay time for $\overline{RD}$ ↓	T_{RDLD}	C_L = 40pF		27	ns
Data output delay time for $\overline{WRQ}$ ↑	T_{WTHD}	C_L = 40pF		21	ns
Data float delay time for $\overline{RD}$ ↑	T_{RDHD}	C_L = 40pF		23	ns
$\overline{WR}$ signal width	T_{WR}	Note 1	18		ns
Data setup time for $\overline{WR}$ ↓	T_{DWR}		20		ns
Data hold time for $\overline{WR}$ ↓	T_{WRD}		0		ns

Note 1: When a $\overline{WRQ}$ signal is output, the duration will be the interval between $\overline{WRQ}$ = H and $\overline{WR}$ = H.

<Read cycle>



<Write cycle>

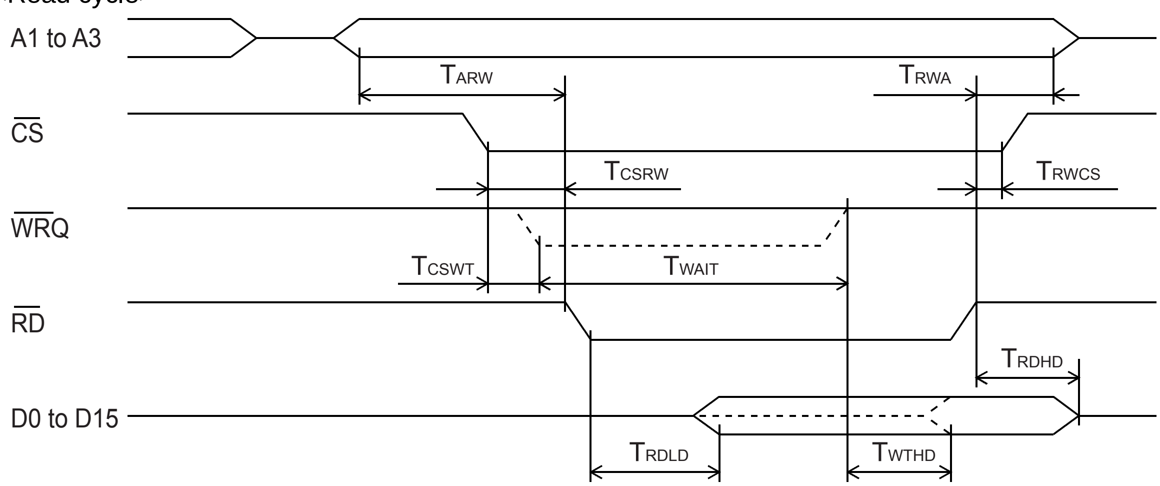


12-5-3. CPU-I/F 3) (IF1 = L, IF0 = L) H8

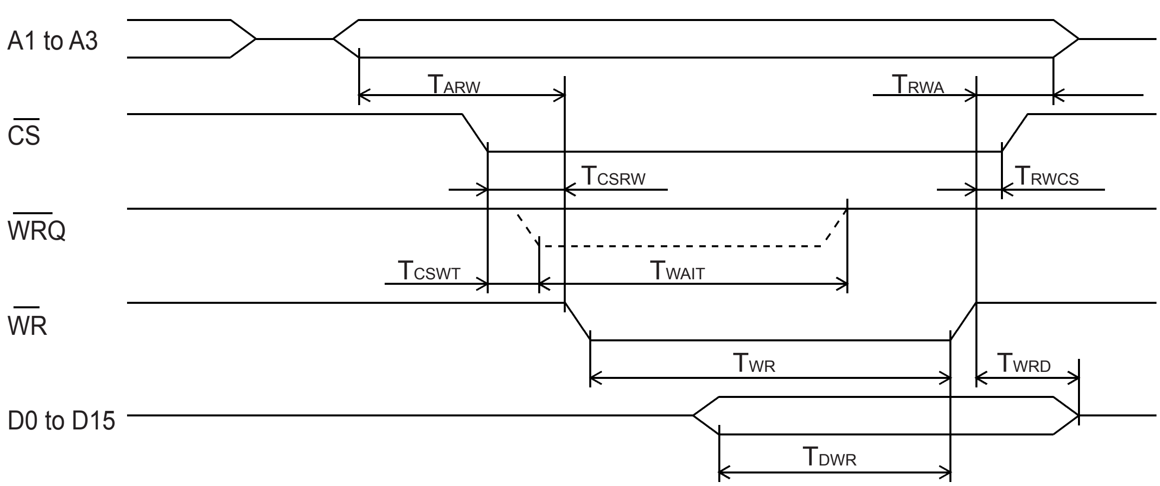
Item	Symbol	Condition	Min.	Max.	Unit
Address setup time for $\overline{RD}$, $\overline{WR}$ ↓	T_{ARW}		17		ns
Address hold time for $\overline{RD}$, $\overline{WR}$ ↑	T_{RWA}		0		ns
$\overline{CS}$ setup time for $\overline{RD}$, $\overline{WR}$ ↓	T_{CSRW}		5		ns
$\overline{CS}$ hold time for $\overline{RD}$, $\overline{WR}$ ↑	T_{RWCS}		0		ns
$\overline{WRQ}$ ON delay time for $\overline{CS}$ ↓	T_{CSWT}	C_L = 40pF		11	ns
$\overline{WRQ}$ signal LOW time	T_{WAIT}			$4T_{CLK}$ +18	ns
Data output delay time for $\overline{RD}$ ↓	T_{RDLD}	C_L = 40pF		27	ns
Data output delay time for $\overline{WRQ}$ ↑	T_{WTHD}	C_L = 40pF		21	ns
Data float delay time for $\overline{RD}$ ↑	T_{RDHD}	C_L = 40pF		23	ns
$\overline{WR}$ signal width	T_{WR}	Note 1	18		ns
Data setup time for $\overline{WR}$ ↑	T_{DWR}		20		ns
Data hold time for $\overline{WR}$ ↑	T_{WRD}		0		ns

Note 1: When a $\overline{WRQ}$ signal is output, the duration will be the interval between $\overline{WRQ}$ = H and $\overline{WR}$ = H.

<Read cycle>



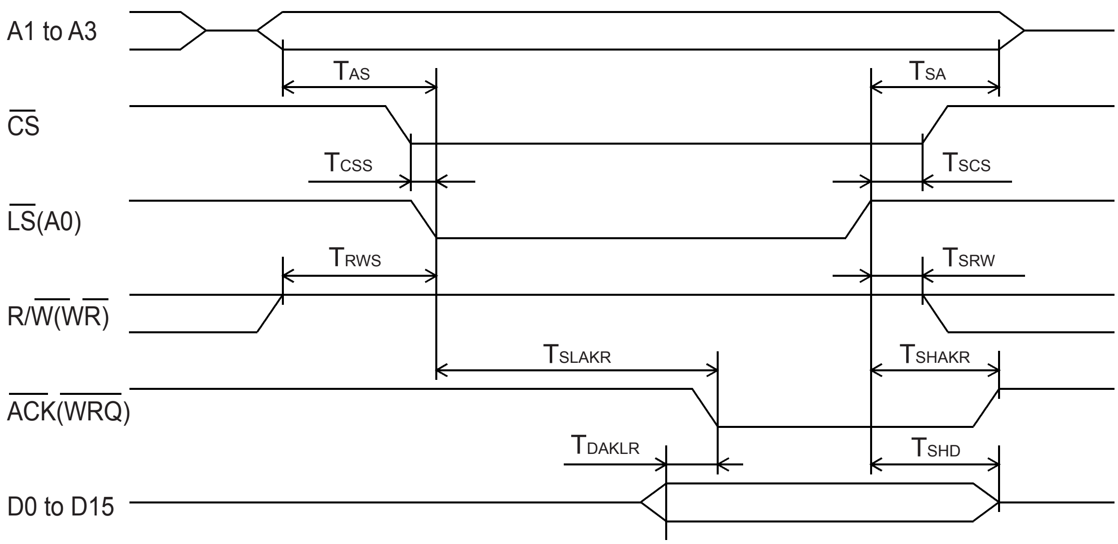
<Write cycle>



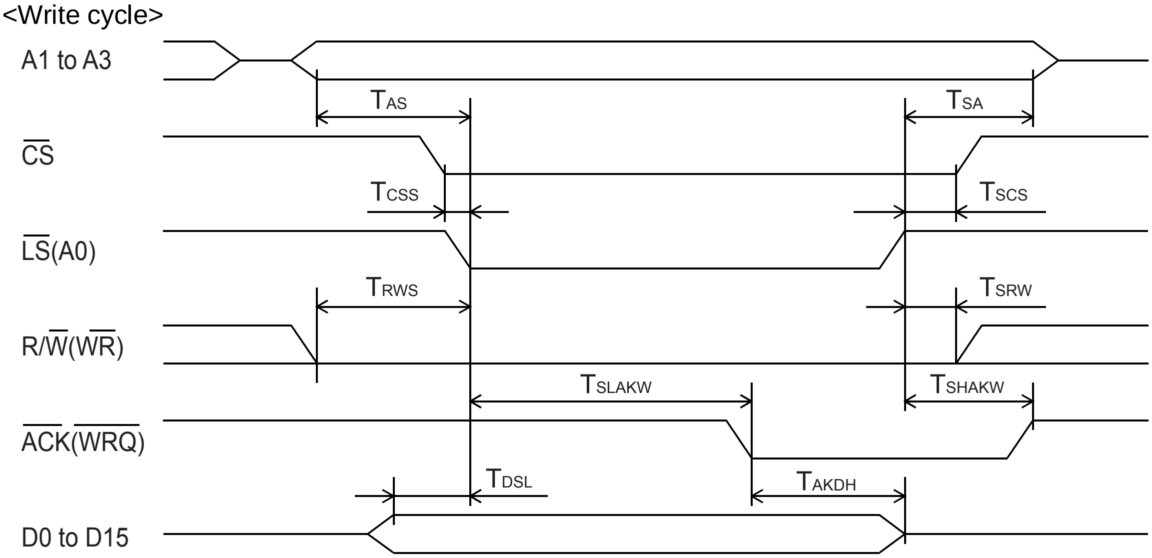
12-5-4. CPU-I/F 4) (IF1 = L, IF0 = L) 68000

Item	Symbol	Condition	Min.	Max.	Unit
Address setup time for $\overline{LS}$ ↓	T_{AS}		17		ns
Address hold time for $\overline{LS}$ ↑	T_{SA}		0		ns
CS setup time for $\overline{LS}$ ↓	T_{CSS}		7		ns
CS hold time for $\overline{LS}$ ↑	T_{SCS}		0		ns
R/$\overline{W}$ setup time for $\overline{LS}$ ↓	T_{RWS}		3		ns
R/$\overline{W}$ hold time for $\overline{LS}$ ↑	T_{SRW}		15		ns
$\overline{ACK}$ ON delay time for $\overline{LS}$ ↓	T_{SLAKR}	C_L = 40pF	$1T_{CLK}$	$4T_{CLK}$+18	ns
	T_{SLAKW}	C_L = 40pF	$1T_{CLK}$	$4T_{CLK}$+18	ns
$\overline{ACK}$ OFF delay time for $\overline{LS}$ ↑	T_{SHAKR}	C_L = 40pF		16	ns
	T_{SHAKW}	C_L = 40pF		16	ns
Data output advance time for $\overline{ACK}$ ↓	T_{DAKLR}	C_L = 40pF	$1T_{CLK}$		ns
Data float delay time for $\overline{LS}$ ↑	T_{SHD}	C_L = 40pF		24	ns
Data setup time for $\overline{LS}$ ↑	T_{DSL}		20		ns
Data hold time for $\overline{ACK}$ ↓	T_{AKDH}		0		ns

<Read cycle>



<Write cycle>

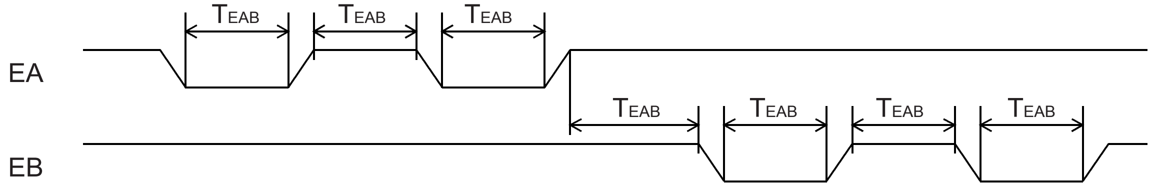


12-6. Operation timing

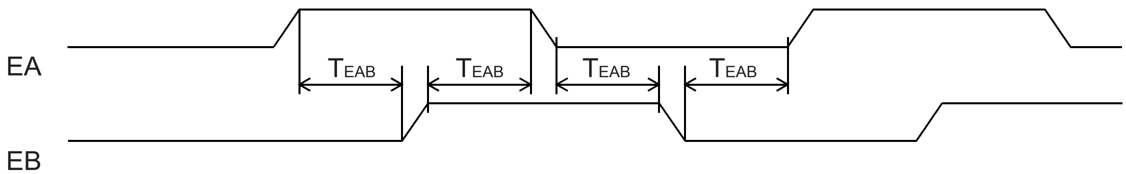
Item	Symbol	Condition	Min.	Max.	Unit
$\overline{\text{RST}}$ input signal width		Note.1	$10T_{CLK}$		ns
CLR input signal width			$2T_{CLK}$		ns
EA, EB input signal width	T_{EAB}		$2T_{CLK}$		ns
EZ input signal width			$2T_{CLK}$		ns
PA, PB input signal width	T_{PAB}		$2T_{CLK}$		ns
ALM input signal width		Note. 2	$2T_{CLK}$		ns
INP input signal width		Note. 2	$2T_{CLK}$		ns
ERC output signal width		RENV1 bit 12 to 14 = 000	$254T_{CLK}$	$255T_{CLK}$	ns
		RENV1 bit 12 to 14 = 001	$254 \times 8T_{CLK}$	$255 \times 8T_{CLK}$	
		RENV1 bit 12 to 14 = 010	$254 \times 32T_{CLK}$	$255 \times 32T_{CLK}$	
		RENV1 bit 12 to 14 = 011	$254 \times 128T_{CLK}$	$255 \times 128T_{CLK}$	
		RENV1 bit 12 to 14 = 100	$254 \times 1024T_{CLK}$	$255 \times 1024T_{CLK}$	
		RENV1 bit 12 to 14 = 101	$254 \times 4096T_{CLK}$	$255 \times 4096T_{CLK}$	
		RENV1 bit 12 to 14 = 110	$254 \times 8192T_{CLK}$	$255 \times 8192T_{CLK}$	
		RENV1 bit 12 to 14 = 111	LEVEL output		
+EL, -EL input signal width		Note. 2	$2T_{CLK}$		ns
SD input signal width		Note. 2	$2T_{CLK}$		ns
ORG input signal width		Note. 2	$2T_{CLK}$		ns
+DR, -DR input signal width		Note. 3	$2T_{CLK}$		ns
$\overline{\text{PE}}$ input signal width		Note. 3	$2T_{CLK}$		ns
PCS input signal width			$2T_{CLK}$		ns
LTC input signal width			$2T_{CLK}$		ns
$\overline{\text{CSTA}}$ Output signal width			$8T_{CLK}$		ns
$\overline{\text{CSTA}}$ Input signal width			$5T_{CLK}$		ns
$\overline{\text{CSTP}}$ Output signal width			$8T_{CLK}$		ns
$\overline{\text{CSTP}}$ Input signal width			$5T_{CLK}$		ns
$\overline{\text{BSY}}$ signal ON delay time	T_{CMDBSY}			$5T_{CLK}$	ns
	T_{STABSY}			$7T_{CLK}$	ns
Start delay time	T_{CMDPLS}			$15T_{CLK}$	ns
	T_{STAPLS}			$17T_{CLK}$	ns

Note 1: The actual CLK input signal is 10 cycles longer while the $\overline{\text{RST}}$ terminal is LOW.
Note 2: If the input filter is ON < FLTR (bit 26) = 1 in RENV1 >, the minimum time will be $80T_{CLK}$.
Note 3: If the input filter is ON < DRF (bit 27) = 1 in RENV1 >, the minimum time will be $655,360T_{CLK}$.
Note 4: The signals above are common to both the X and Y axes.

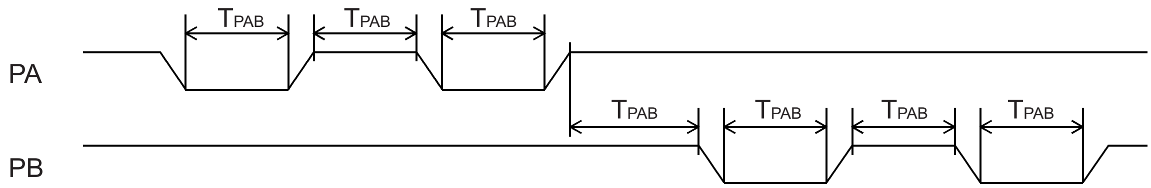
1) When the EA, EB inputs are in the 2-pulse mode



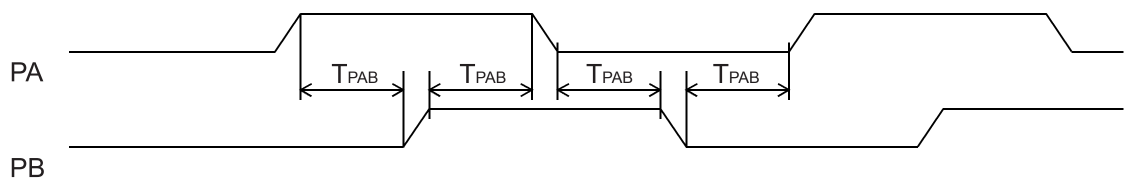
2) When the EA, EB inputs are in the 90˚ phase-difference mode



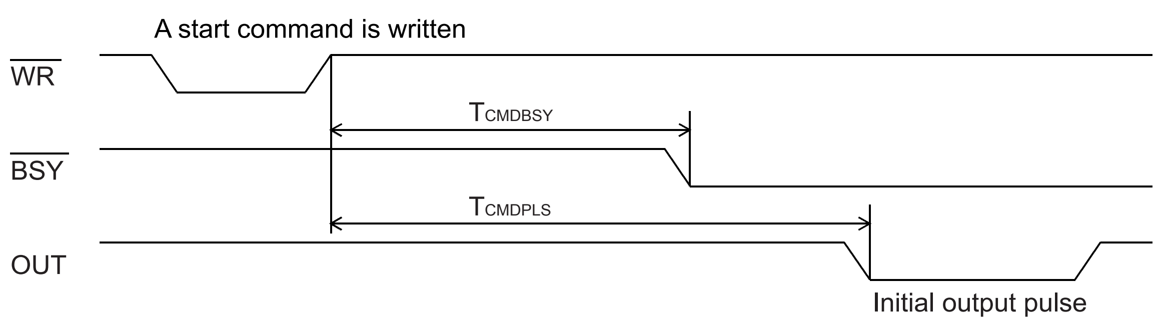
3) When the PA, PB inputs are in the 2-pulse mode



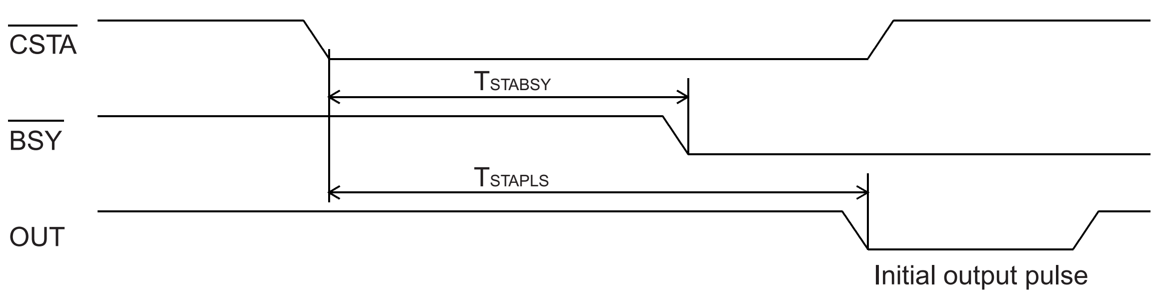
4) When the PA, PB inputs are in the 90˚ phase-difference mode



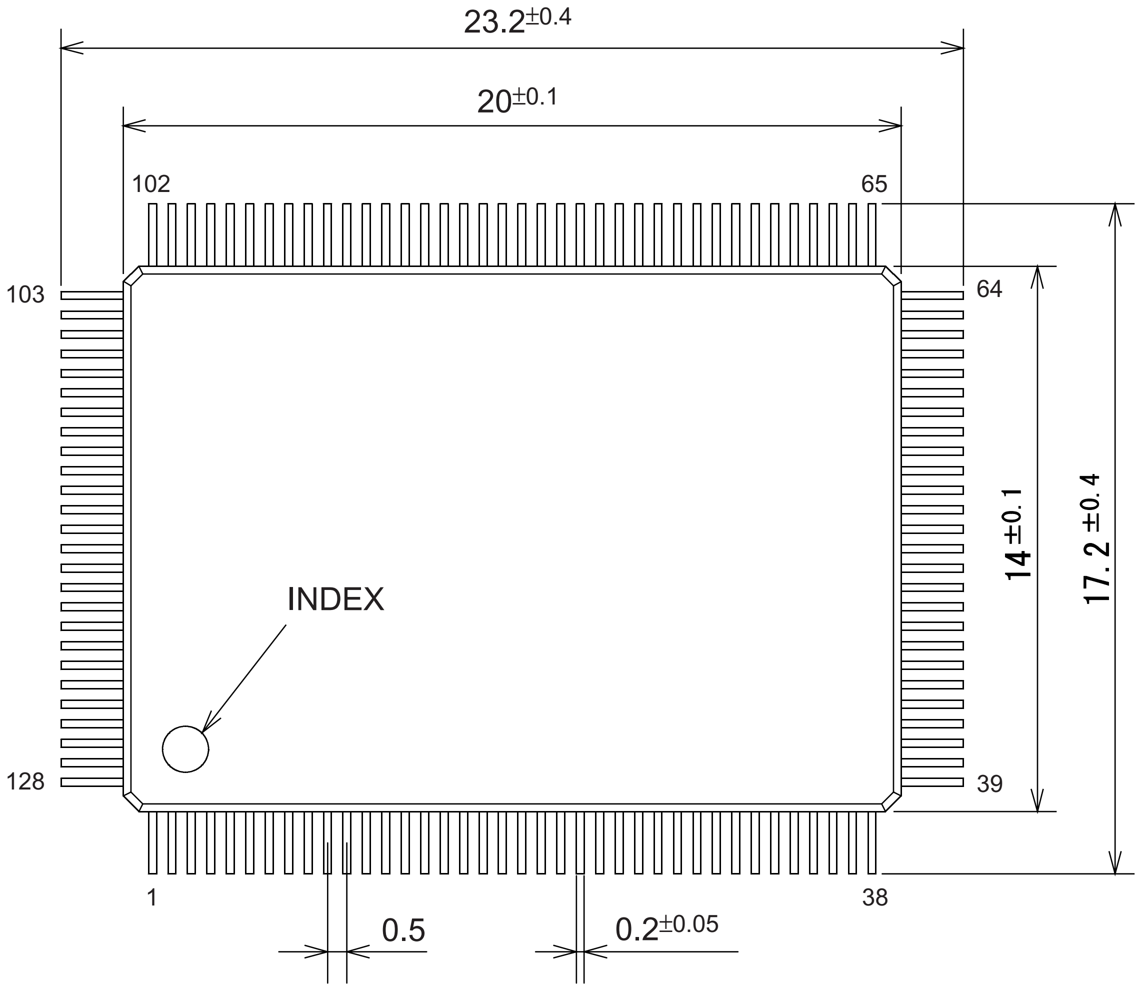
5) Timing for the command mode (when I/M = H, and B/$\overline{W}$ = H)

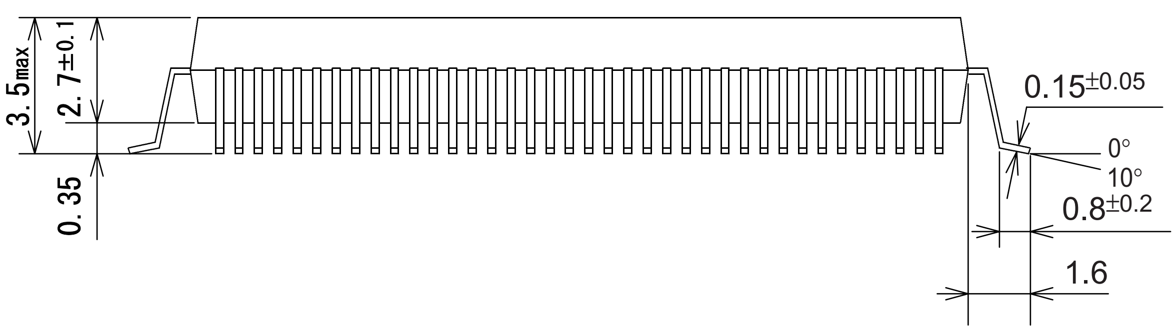


6) Simultaneous start timing



13. External Dimensions





Appendix: List of various items

Appendix 1: List of commands

<Operation commands>

COMB0	Symbol	Description	COMB0	Symbol	Description
05h	CMEMG	Emergency stop	50h	STAFL	FL low speed start
06h	CMSTA	$\overline{CSTA}$ output (simultaneous start)	51h	STAFH	FH low speed start
07h	CMSTP	$\overline{CSTP}$ output (simultaneous stop)	52h	STAD	High speed start 1 (FH low speed -> Deceleration stop)
40h	FCHGL	Immediate change to FL low speed	53h	STAUD	High speed start 2 (acceleration -> FH low speed -> deceleration)
41h	FCHGH	Immediate change to FH low speed	54h	CNTFL	FL low speed start for remaining number of pulses
42h	FSCHL	Decelerate to FL speed	55h	CNTFH	FH low speed start for remaining number of pulses
43h	FSCHH	Accelerate to FH speed	56h	CNTD	High speed start 1 for remaining number of pulses
49h	STOP	Immediate stop	57h	CNTUD	High speed start 2 for remaining number of pulses
4Ah	SDSTP	Deceleration stop			

< General-purpose port control commands>

COMB0	Symbol	Description	COMB0	Symbol	Description
10h	P0RST	Set the P0 terminal LOW	18h	P0SET	Set the P0 terminal HIGH
11h	P1RST	Set the P1 terminal LOW	19h	P1SET	Set the P1 terminal HIGH
12h	P2RST	Set the P2 terminal LOW	1Ah	P2SET	Set the P2 terminal HIGH
13h	P3RST	Set the P3 terminal LOW	1Bh	P3SET	Set the P3 terminal HIGH
14h	P4RST	Set the P4 terminal LOW	1Ch	P4SET	Set the P4 terminal HIGH
15h	P5RST	Set the P5 terminal LOW	1Dh	P5SET	Set the P5 terminal HIGH
16h	P6RST	Set the P6 terminal LOW	1Eh	P6SET	Set the P6 terminal HIGH
17h	P7RST	Set the P7 terminal LOW	1Fh	P7SET	Set the P7 terminal HIGH

<Control commands>

COMB0	Symbol	Description	COMB0	Symbol	Description
00h	NOP	(Invalid command)	26h	PRICAN	Clear the operation pre-register
04h	SRST	Software reset	27h	PCPCAN	Clear the RCMP5 pre-register
20h	CUN1R	Reset COUNTER1 (command position)	28h	STAON	Substitute PCS input
21h	CUN2R	Reset COUNTER2 (mechanical position)	29h	LTCH	Substitute LTC input
22h	CUN3R	Reset COUNTER3 (deflection counter)	2Ah	SPSTA	Uses the same process as the $\overline{CSTA}$ input, but for this axis
23h	CUN4R	Reset COUNTER4 (general-purpose)	2Bh	PRISHF	Shift the operation pre-register data
24h	ERCOUT	Output an ERC signal	2Ch	PCPSHF	Shift the RCMP5 pre-register
25h	ERCRS	Reset the ERC signal			

<Register control commands>

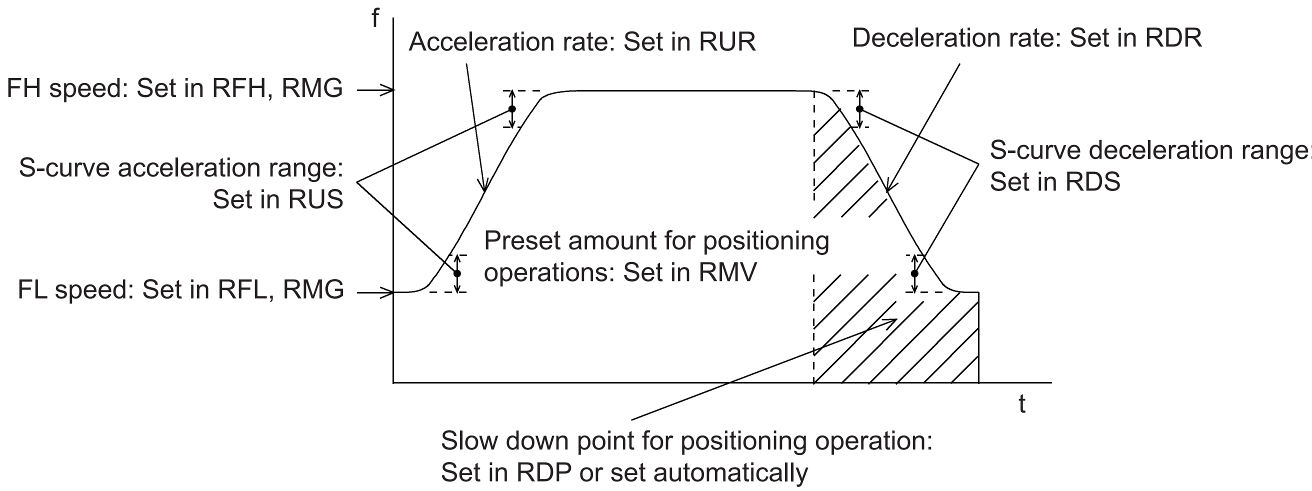
No.	Register	Description	Read command		Write command		2nd pre-register	Read command		Write command	
			COMB0	Symbol	COMB0	Symbol		COMB0	Symbol	COMB0	Symbol
1	RMV	Number of feed pulses / target position	D0h	RRMV	90h	WRMV	PRMV	C0h	RPRMV	80h	WPRMV
2	RFL	Initial speed	D1h	RRFL	91h	WRFL	PRFL	C1h	RPRFL	81h	WPRFL
3	RFH	Operation speed	D2h	RRFH	92h	WRFH	PRFH	C2h	RPRFH	82h	WPRFH
4	RUR	Acceleration rate	D3h	RRUR	93h	WRUR	PRUR	C3h	RPRUR	83h	WPRUR
5	RDR	Deceleration rate	D4h	RRDR	94h	WRDR	PRDR	C4h	RPRDR	84h	WPRDR
6	RMG	Speed magnification rate	D5h	RRMG	95h	WRMG	PRMG	C5h	RPRMG	85h	WPRMG
7	RDP	Ramping-down point	D6h	RRDP	96h	WRDP	PRDP	C6h	RPRDP	86h	WPRDP
8	RMD	Operation mode	D7h	RRMD	97h	WRMD	PRMD	C7h	RPRMD	87h	WPRMD
9	RIP	Circular interpolation center	D8h	RRIP	98h	WRIP	PRIP	C8h	RPRIP	88h	WPRIP
10	RUS	S-curve range while accelerating	D9h	RRUS	99h	WRUS	PRUS	C9h	RPRUS	89h	WPRUS
11	RDS	S-curve range while decelerating	DAh	RRDS	9Ah	WRDS	PRDS	CAh	RPRDS	8Ah	WPRDS
12	RFA	Feed speed to correct feed distance	DBh	RRFA	9Bh	WRFA					
13	RENV1	Environment setting 1	DCh	RRENV1	9Ch	WRENV1					
14	RENV2	Environment setting 2	DDh	RRENV2	9Dh	WRENV2					
15	RENV3	Environment setting 3	DEh	RRENV3	9Eh	WRENV3					
16	RENV4	Environment setting 4	DFh	RRENV4	9Fh	WRENV4					
17	RENV5	Environment setting 5	E0h	RRENV5	A0h	WRENV5					
18	RENV6	Environment setting 6	E1h	RRENV6	A1h	WRENV6					
19	RENV7	Environment setting 7	E2h	RRENV7	A2h	WRENV7					
20	RCUN1	COUNTER1 (command position)	E3h	RRCUN1	A3h	WRCUN1					
21	RCUN2	COUNTER2 (mechanical position)	E4h	RRCUN2	A4h	WRCUN2					
22	RCUN3	COUNTER3 (deflection counter)	E5h	RRCUN3	A5h	WRCUN3					
23	RCUN4	COUNTER4 (general-purpose)	E6h	RRCUN4	A6h	WRCUN4					
24	RCMP1	Comparator 1 data	E7h	RRCMP1	A7h	WRCMP1					
25	RCMP2	Comparator 2 data	E8h	RRCMP2	A8h	WRCMP2					
26	RCMP3	Comparator 3 data	E9h	RRCMP3	A9h	WRCMP3					
27	RCMP4	Comparator 4 data	EAh	RRCMP4	AAh	WRCMP4					
28	RCMP5	Comparator 5 data	EBh	RRCMP5	ABh	WRCMP5	PRCP5	CBh	RPRCP5	8Bh	WPRCP5
29	RIRQ	Enable various event interrupts (INTs)	ECh	RRIRQ	ACh	WRIRQ					
30	RLTC1	COUNTER1 latch data	EDh	RRLTC1							
31	RLTC2	COUNTER2 latch data	EEh	RRLTC2							
32	RLTC3	COUNTER3 latch data	EFh	RRLTC3							
33	RLTC4	COUNTER4 latch data	F0h	RRLTC4							
34	RSTS	Extension status	F1h	RRSTS							
35	REST	Error INT status	F2h	RREST							
36	RIST	Event INT status	F3h	RRIST							
37	RPLS	Positioning counter	F4h	RRPLS							
38	RSPD	EZ counter, speed monitor	F5h	RRSPD							
39	RSDC	Ramping-down point	F6h	RRSDC							
40	RCI	Number of steps for circular interpolation	FCh	RRCI	BCh	WRCI	PRCI	CCh	RPRCI	8Ch	WPRCI
41	RIPS	Interpolation status	FFh	RRIPS							

Appendix 2: Setting speed pattern

Register	Description	Bit length setting range	Setting range	Pre-register
RMV	Positioning amount	28	-134,217,728 to 134,217,727 (8000000h) (7FFFFFFh)	PRMV
RFL	Initial speed (FL speed)	16	1 to 65,535 (0FFFFh)	PRFL
RFH	Operation speed (FH speed)	16	1 to 65,535 (0FFFFh)	PRFH
RUR	Acceleration rate	16	1 to 65,535 (0FFFFh)	PRUR
RDR	Deceleration rate Note 1	16	0 to 65,535 (0FFFFh)	PRDR
RMG	Speed magnification rate	12	2 to 4,095 (0FFFh)	PRMG
RDP	Ramping-down point	24	0 to 16,777,215 (0FFFFFFh)	PRDP
RUS	S-curve acceleration range	15	0 to 32,767 (7FFFh)	PRUS
RDS	S-curve deceleration range	15	0 to 32,767 (7FFFh)	PRDS

Note 1: If RDR is set to zero, the deceleration rate will be the value set in the RUR.

[Relative position of each register setting for acceleration and deceleration factors]



- ♦ RFL: FL speed setting register (16-bit)
  Specify the speed for FL low speed operations and the start speed for high speed operations (acceleration/deceleration operations) in the range of 1 to 65,535 (0FFFFh).
  The speed will be calculated from the value in RMG.

$$\text{FL speed [pps]} = \text{RFL} \times \frac{\text{Reference clock frequency [Hz]}}{(\text{RMG} + 1) \times 65536}$$

- ♦ RFH: FH speed setting register (16-bit)
  Specify the speed for FH low speed operations and the start speed for high speed operations (acceleration/deceleration operations) in the range of 1 to 65,535 (0FFFFh).
  When used for high speed operations (acceleration/deceleration operations), specify a value larger than RFL.
  The speed will be calculated from the value placed in RMG.

$$\text{FH speed [pps]} = \text{RFH x } \frac{\text{Reference clock frequency [Hz]}}{(\text{RMG} + 1) \text{ x } 65536}$$

♦ RUR: Acceleration rate setting register (16-bit)
Specify the acceleration characteristic for high speed operations (acceleration/deceleration operations), in the range of 1 to 65,535 (0FFFFh)
Relationship between the value entered and the acceleration time will be as follows:

1) Linear acceleration (MSMD = 0 in the RMD register)

$$\text{Acceleration time [s]} = \frac{(RFH - RFL) \times (RUR + 1) \times 4}{\text{Reference clock frequency [Hz]}}$$

2) S-curve without a linear range (MSMD=1 in the RMD register and RUS register =0)

$$\text{Acceleration time [s]} = \frac{(RFH - RFL) \times (RUR + 1) \times 8}{\text{Reference clock frequency [Hz]}}$$

3) S-curve with a linear range (MSMD=1 in the RMD register and RUS register >0)

$$\text{Acceleration time [s]} = \frac{(RFH-RFL + 2 \times RUS) \times (RUR + 1) \times 4}{\text{Reference clock frequency [Hz]}}$$

♦ RDR: Deceleration rate setting register (16-bit)
Normally, specify the deceleration characteristics for high speed operations (acceleration/deceleration operations) in the range of 1 to 65,535 (0FFFFh).
Even if the ramping-down point is set to automatic (MSDP = 0 in the RMD register), the value placed in the RDR register will be used as the deceleration rate.
However, when RDR = 0, the deceleration rate will be the value placed in the RUR.
When the ramping-down point is set automatically, make sure that the (deceleration time) ≤ (acceleration time x 2) for independent axis operations, and that the (deceleration time) = (acceleration time) for interpolation operations.
If the deceleration (deceleration time) > (acceleration time x 2) in independent axis operations, or if (deceleration time) > (acceleration time) in interpolation operations, the axis may not decrease the speed to the specified FL speed when stopping. In this case, use a manual ramping-down point (MSDP = 1 in the RMD register).
The relationship between the value entered and the deceleration time is as follows.

1) Linear deceleration (MSMD = 0 in the RMD register)

$$\text{Deceleration time [s]} = \frac{(RFH - RFL) \times (RDR + 1) \times 4}{\text{Reference clock frequency [Hz]}}$$

2) S-curve deceleration without a linear range (MSMD=1 in the RMD register and RDS register = 0)

$$\text{Deceleration time [s]} = \frac{(RFH - RFL) \times (RDR + 1) \times 8}{\text{Reference clock frequency [Hz]}}$$

3) S-curve deceleration with a linear range (MSMD=1 in the RMD register and RDS register > 0)

$$\text{Deceleration time [s]} = \frac{(RFH-RFL + 2 \times RDS) \times (RDR + 1) \times 4}{\text{Reference clock frequency [Hz]}}$$

♦ RMG: Magnification rate register (12-bit)
Specify the relationship between the RFL and RFH settings and the speed, in the range of 2 to 4,095 (0FFFh).
As the magnification rate is increased, the speed setting units will tend to be approximations. Normally set the magnification rate as low as possible.
The relationship between the value entered and the magnification rate is as follows.

$$\text{Magnification rate} = \frac{\text{Reference clock frequency [Hz]}}{(RMG + 1) \times 65536}$$

[Magnification rate setting example, when the reference clock =19.6608 MHz] (Output speed unit: pps)

Setting	Magnification rate	Output speed range	Setting	Magnification rate	Output speed range
2999 (0BB7h)	0.1	0.1 to 6,553.5	59 (3Bh)	5	5 to 327,675
1499 (5DBh)	0.2	0.2 to 13,107.0	29 (1Dh)	10	10 to 655,350
599 (257h)	0.5	0.5 to 32,767.5	14 (0Eh)	20	20 to 1,310,700
299 (12Bh)	1	1 to 65,535	5 (5h)	50	50 to 3,276,750
149 (95h)	2	2 to 131,070	2 (2h)	100	100 to 6,553,500

- RDP: Ramping-down point register (24-bits)

Specify the value used to determine the deceleration start point for positioning operations that include acceleration and deceleration

The meaning of the value specified in the RDP changes with the "ramping-down point setting method ", (MSD0) in the RMD register.

<When set to manual (MSDP = 1 in the RMD register)>

The number of pulses at which to start deceleration, set in the range of 0 to16,777,215 (0FFFFFFh).

The optimum value for the ramping-down point can be calculated as shown in the equation below.

1) Linear deceleration (MSMD=0 of the RMD register)

$$\text{Optimum value [Number of pulses]} = \frac{(RFH^2 - RFL^2) \times (RDR + 1)}{(RMG + 1) \times 32768}$$

However, the optimum value for a triangle start, without changing the value in the RFH register while turning OFF the FH correction function (MADJ = 1 in the RMD register) will be calculated as shown the next equation below.

(When using idling control, modify the value for RMV in the equation below by deducting the number of idling pulses from the value placed in the RMV register. The number of idling pulses will be "1 to 62 when IDL = 2 to 7 in RNVI5.)

$$\text{Optimum value [Number of pulses]} = \frac{RMV \times (RDR + 1)}{RUR + RDR + 2}$$

2) S-curve deceleration without a linear range (MSMD=1 in the RMD register and the RDS register =0)

$$\text{Optimum value [Number of pulses]} = \frac{(RFH^2 - RFL^2) \times (RDR + 1) \times 2}{(RMG + 1) \times 32768}$$

3) S-curve deceleration with a linear range (MSMD=1 in the RMD register and the RDS register >0)

$$\text{Optimum value [Number of pulses]} = \frac{(RFH + RFL) \times (RFH - RFL + 2 \times RDS) \times (RDR + 1)}{(RMG + 1) \times 32768}$$

Start deceleration at the point when the (positioning counter value) ≤ (RDP set value).

<When set to automatic (MSDP = 0 in the RMD register)>

This is an offset value for the automatically set ramping-down point. Set in the range of -8,388,608 (800000h) to 8,388,607(7FFFFFFh).

When the offset value is a positive number, the axis will start deceleration at an earlier stage and will feed at the FL speed after decelerating. When a negative number is entered, the deceleration start timing will be delayed. If the offset is not required, set to zero.

When the value for the ramping-down point is smaller than the optimum value, the speed when stopping will be faster than the FL speed. On the other hand, if it is larger than the optimum value, the axis will feed at FL low speed after decelerating.

- RUS: S-curve acceleration range register (15-bit)

Specify the S-curve acceleration range for S-curve acceleration/deceleration operations in the range of 1 to 32,767 (7FFFh).

The S-curve acceleration range S_{SU} will be calculated from the value placed in RMG.

$$S_{SU} \text{ [pps]} = RU \times \frac{\text{Reference clock frequency [Hz]}}{(RMG + 1) \times 65536}$$

In other words, speeds between the FL speed and (FL speed + S_{SU}), and between (FH speed – Ssu) and the FH speed, will be S-curve acceleration operations. Intermediate speeds will use linear acceleration.

However, if zero is specified, "(RFH-RFL)/2" will be used for internal calculations, and the operation will be an S-curve acceleration without a linear component.

- RDS: S-curve deceleration range setting register (15-bit)

Specify the S-curve deceleration range for S-curve acceleration/deceleration operations in the range of 1 to 32,767 (7FFFh).

The S-curve acceleration range S_{SU} will be calculated from the value placed in RMG.

$$S_{SD} \text{ [pps]} = RDS \times \frac{\text{Reference clock frequency [Hz]}}{(RMG + 1) \times 65536}$$

In other words, speeds between the FL speed (FL speed + S_{SD}), and between (FH speed - S_{SD}) and the FH speed, will be S-curve deceleration operations. Intermediate speeds will use linear deceleration.

However, if zero is specified, "(RFH-RFL)/2" will be used for internal calculations, and the operation will be an S-curve deceleration without a linear component.

Appendix 3: Label list

Label	Type	Position	Description
A0	Terminal name		Address bus 0 (LSB)
A1	Terminal name		Address bus 1
A2	Terminal name		Address bus 2
A3	Terminal name		Address bus 3
ADJ0 to 1	Register bit	RENV6 12-13	Select the feed amount correction method
ALML	Register bit	RENV1 9	Set the input logic for the ALM signal (0: Negative, 1: Positive)
ALMM	Register bit	RENV1 8	Select the process to use when the ALM input is ON (0: Immediate stop, 1: Deceleration stop)
ALMx	Terminal name		X axis driver alarm signal (to stop the axis)
ALMy	Terminal name		Y axis driver alarm signal (to stop the axis)
AS0 to 15	Register bit	RSPD 0-15	Monitor current speed
BR0 to 11	Register bit	RENV6 0-11	Specify a backlash correction or slip correction amount.
BSYC	Register bit	RENV3 14	Increment/decrement COUNTER4 only while in operation (BSY = L)
$\overline{BSYx}$	Terminal name		Operation monitor output for the X axis
$\overline{BSYy}$	Terminal name		Operation monitor output for the Y axis
BUFB0	Byte map name	4 for Z80	Write/read the input/output buffer (bits 0 to 7).
BUFB1	Byte map name	5 for Z80	Write/read the input/output buffer (bits 8 to 15)
BUFB2	Byte map name	6 for Z80	Write/read the input/output buffer (bits 16 to 23)
BUFB3	Byte map name	7 for Z80	Write/read the input/output buffer (bits 24 to 31)
BUFW0	Word map name	4 for 80886	Write/read the input/output buffer (bits 0 to 15)
BUFW1	Word map name	6 for 8086	Write/read the input/output buffer (bits 16 to 31)
C1C0 to 1	Register bit	RENV4 0-1	Select a comparison counter for comparator1
C1D0 to 1	Register bit	RENV4 5-6	Select a process to execute when the comparator1 conditions are met
C1S0 to 2	Register bit	RENV4 2-4	Select a comparison method for comparator1
C2C0 to 1	Register bit	RENV4 8-9	Select a comparison counter for comparator2
C2D0 to 1	Register bit	RENV4 13-14	Select a process to execute when the comparator2 conditions are met
C2S0 to 2	Register bit	RENV4 10-12	Select a comparison method for comparator2
C3C0 to 1	Register bit	RENV4 16-17	Select a comparison counter for comparator3
C3D0 to 1	Register bit	RENV4 21-22	Select a process to execute when the comparator3 conditions are met
C3S0 to 2	Register bit	RENV4 18-20	Select a comparison method for comparator3
C4C0 to 1	Register bit	RENV4 24-25	Select a comparison counter for comparator4
C4D0 to 1	Register bit	RENV4 30-31	Select a process to execute when the comparator4 conditions are met
C4S0 to 3	Register bit	RENV4 26-29	Select a comparison method for comparator4
C5C0 to 2	Register bit	RENV5 0-2	Select a comparison counter for comparator5
C5D0 to 1	Register bit	RENV5 6-7	Select a process to execute when the comparator5 conditions are met
C5S0 to 2	Register bit	RENV5 3-5	Select a comparison method for comparator5
$\overline{CEMG}$	Terminal name		Emergency stop signal
CI20 to 21	Register bit	RENV3 8-9	Specify the input count COUNTER2 (mechanical position)
CI30 to 31	Register bit	RENV3 10-11	Specify the input count COUNTER3 (deflection counter)
CI40 to 41	Register bit	RENV3 12-13	Specify the input count COUNTER4 (general-purpose)
CLK	Terminal name		Reference clock (19.6608 MHz as standard)
CLR0 to 1	Register bit	RENV1 20-21	Select the CLR input mode
CLRx	Terminal name		Clear the counter input for the X axis
CLRy	Terminal name		Clear the counter input for the Y axis
CMEMG	Command	05h	Emergency stop
CMSTA	Command	06h	Output a $\overline{CSTA}$ (simultaneous start) signal
CMSTP	Command	07h	Output a $\overline{CSTP}$ (simultaneous stop) signal
CND0 to 3	Register bit	RSTS 0-3	Operation status monitor
CNTD	Command	56h	Remaining high speed start pulses (FH low speed -> Deceleration stop)
CNTFH	Command	55h	Remaining pulses FH low speed start pulses
CNTFL	Command	54h	Remaining pulses FL low speed start pulses
CNTUD	Command	57h	Remaining high speed start pulses (accelerate -> FH low speed -> deceleration stop)
COMB0	Byte map name	0 when Z80	Write control command
COMB1	Byte map name	1 when Z80	Axis selection
COMW	Word map name	0when 8086	Assign an axis, or write a control command
COUNTER1	Circuit name		28-bit counter for command position control
COUNTER2	Circuit name		28-bit counter for mechanical position control
COUNTER3	Circuit name		16-bit counter for the deflection counter
COUNTER4	Circuit name		28-bit counter for the general-purpose counter
$\overline{CS}$	Terminal name		Chip select signal
$\overline{CSTA}$	Terminal name		Simultaneous start signal
$\overline{CSTP}$	Terminal name		Simultaneous stop signal
CU1B	Register bit	RENV3 24	Operate COUNTER2 (mechanical position) with backlash/slip correction
CU1C	Register bit	RENV3 16	Reset COUNTER1 (command position) by turning ON the CLR input.
CU1R	Register bit	RENV3 20	Reset COUNTER1 (command position) when the zero return is complete
CU2B	Register bit	RENV3 25	Operate COUNTER2 (mechanical position) with backlash/slip correction
CU2C	Register bit	RENV3 17	Reset COUNTER2 (mechanical position) by turning ON the CLR input
CU2H	Register bit	RENV3 29	Stop the count on COUNTER2 (mechanical position)
CU2R	Register bit	RENV3 21	Reset COUNTER2 (mechanical position) when the zero return is complete
CU3B	Register bit	RENV3 26	Operate COUNTER3 (deflection) with backlash/slip correction

Label	Type	Position	Description
CU3C	Register bit	RENV3 18	Reset the COUNTER3 (deflection) by turning ON the CLR input.
CU3H	Register bit	RENV3 30	Stop the count on COUNTER3 (deflection)
CU3R	Register bit	RENV3 22	Reset COUNTER3 (deflection) when the zero return is complete
CU4B	Register bit	RENV3 27	Operate COUNTER4 (general-purpose) backlash/slip correction
CU4C	Register bit	RENV3 19	Reset COUNTER4 (general-purpose) by turning ON the CLR input
CU4H	Register bit	RENV3 31	Stop the count on COUNTER4 (general-purpose)
CU4R	Register bit	RENV3 23	Reset COUNTER4 (general-purpose) when the zero position operation is complete
CUN1R	Command	20h	Reset COUNTER1 (command position)
CUN2R	Command	21h	Reset COUNTER2 (mechanical position)
CUN3R	Command	22h	Reset COUNTER3 (deflection counter)
CUN4R	Command	23h	Reset COUNTER4 (general purpose)
D0	Terminal name		Data bus 0 (LSB)
D1	Terminal name		Data bus 1
D10	Terminal name		Data bus 10
D11	Terminal name		Data bus 11
D12	Terminal name		Data bus 12
D13	Terminal name		Data bus 13
D14	Terminal name		Data bus 14
D15	Terminal name		Data bus 15 (MSB)
D2	Terminal name		Data bus 2
D3	Terminal name		Data bus 3
D4	Terminal name		Data bus 4
D5	Terminal name		Data bus 5
D6	Terminal name		Data bus 6
D7	Terminal name		Data bus 7
D8	Terminal name		Data bus 8
D9	Terminal name		Data bus 9
DIRx	Terminal name		Motor drive direction signal for the X axis
DIRy	Terminal name		Motor drive direction signal for the Y axis
DRF	Register bit	RENV1 27	Apply a filter to +DR, -DR signal input
DRL	Register bit	RENV1 25	Select +DR, -DR signal input logic (0: Negative logic, 1: Positive logic)
+DRx	Terminal name		Manual (+) input for the X axis
-DRx	Terminal name		Manual (-) input for the X axis
+DRy	Terminal name		Manual (+) input for the Y axis
-DRy	Terminal name		Manual (-) input for the Y axis
DTMF	Register bit	RENV1 28	Turn OFF the direction change timer (0.2 ms)
EAx	Terminal name		Encoder A phase signal for the X axis
EAy	Terminal name		Encoder A phase signal for the Y axis
EBx	Terminal name		Encoder B phase signal for the X axis.
EBy	Terminal name		Encoder B phase signal for the Y axis
ECZ0 to 3	Register bit	RSPD 16-19	Read the count value of the EZ input to monitor the zero return
EDIR	Register bit	RENV2 22	Reverse the EA, EB input count direction
EIM0 to 1	Register bit	RENV2 20-21	Specify the EA, EB input parameters
EINF	Register bit	RENV2 18	Apply a noise filter to the EA/EB input
EIP	Register bit	REST 13	Simultaneous stop when the other axis stops during an interpolation operation
ELLx	Terminal name		Set the input logic of the end limit signal for the X axis
ELLy	Terminal name		Select the input logic of the end limit signal for the Y axis
ELM	Register bit	RENV1 3	Select the process to execute when the EL input is ON (0: Immediate stop, 1: Deceleration stop)
+ELx	Terminal name		(+) end limit signal for the X axis
-ELx	Terminal name		(-) end limit signal for the X axis
+ELy	Terminal name		(+) end limit signal for the Y axis
-ELy	Terminal name		(-) end limit signal for the Y axis.
EPW0 to 2	Register bit	RENV1 12-14	Specify the ERC output signal pulse width
ERCL	Register bit	RENV1 15	Set the output logic of the ERC signal (0: Negative logic, 1: Positive logic)
ERCOUT	Command	24h	Output an ERC signal
ERCRST	Command	25h	Reset the output when the ERC signal is set to level output
ERCx	Terminal name		Driver deflection clear output for the X axis
ERCy	Terminal name		Driver deflection clear output for the Y axis
EROE	Register bit	RENV1 10	Automatic output of the ERC signal
EROR	Register bit	RENV1 11	Auto output an ERC signal when the zero return is complete
ESAL	Register bit	REST 7	Equals 1 when stopped by the ALM input turning ON
ESAO	Register bit	REST 15	Equals 1 when the positioning counter exceeds the count range
ESC1	Register bit	REST 0	Stopped when the comparator1 conditions (+SL) are met
ESC2	Register bit	REST 1	Stopped when the comparator2 conditions (-SL) are met
ESC3	Register bit	REST 2	Stopped when the comaprator3 conditions (detect out-of-step) are met
ESC4	Register bit	REST 3	Stopped when the comparator4 conditions are met.
ESC5	Register bit	REST 4	Stopped when the comparator5 conditions are met
ESDT	Register bit	REST 12	Stopped by an operation data error
ESEE	Register bit	REST 16	An EA/EB input error occurred
ESEM	Register bit	REST 9	Stopped by turning ON the $\overline{\text{CEMG}}$ input

Label	Type	Position	Description
ESML	Register bit	REST 6	Stopped because the –EL input turned ON
ESOL	Register bit	REST 11	Release the feed amount limitation for a zero return or for leaving from the zero position
ESPE	Register bit	REST 17	A PA/PB input error occurred
ESPL	Register bit	REST 5	Stopped because the + EL input turned ON
ESPO	Register bit	REST 14	The PA/PB input buffer counter overflowed
ESSD	Register bit	REST 10	Deceleration stop caused by the SD input. turning ON
ESSP	Register bit	REST 8	Stopped because the $\overline{\text{CSTP}}$ input turned ON
EZL	Register bit	RENV2 23	Set the input logic for the EZ signal (0: Falling, 1: Rising)
EZx	Terminal name		X axis encoder Z phase signal
EZy	Terminal name		Y axis encoder Z phase signal
ETW0 to 1	Register bits	RENV1 16-17	Specify the ERC signal OFF timer time
EZD0 to 3	Register bits	RENV3 4-7	Enter an EZ count value for a zero return
FCHGH	Command	41h	Change immediately to FH speed
FCHGL	Command	40h	Change immediately to FL speed
$\overline{\text{FDWx}}$	Terminal name		Output deceleration operation monitor for the X axis.
$\overline{\text{FDWy}}$	Terminal name		Output deceleration operation monitor for the Y axis
FLTR	Register bit	RENV1 26	Apply input filter
FSCHH	Command	43h	Accelerate to FH speed
FSCHL	Command	42h	Accelerate to FL speed
FT0 to 15	Register bits	RENV7 16-31	Enter an FT time for the vibration reduction function
$\overline{\text{FUPx}}$	Terminal name		Acceleration operation monitor output for the X axis
$\overline{\text{FUPy}}$	Terminal name		Acceleration operation monitor output for the Y axis
IDC0 to 2	Register bits	RSPD 20-22	Monitor the idling count (0 to 7 pulses)
IDL0 to 2	Register bits	RENV5 8-10	Enter the number of idling pulse (0 to 7 pulses)
IF0	Terminal name		CPU-I/F mode selection 0
IF1	Terminal name		CPU-I/F mode selection 1
$\overline{\text{IFB}}$	Terminal name		CPU-I/F busy
INPL	Register bit	RENV1 22	Set the input logic for the INPU signal (0: Negative logic, 1: Positive logic)
INPx	Terminal name		In position input for the X axis
INPy	Terminal name		In position input for the Y axis
$\overline{\text{INT}}$	Terminal name		Interrupt request signal
INTM	Register bit	RENV1 29	Mask the INT output terminal
IOP0 to 7	Sub-status bits	SSTSW 0-7	Read the P0 to P7 terminal status.
IOPB	Byte map name	"2 " when using a Z80	Read the general I/O port
IPCC	Register bit	RIPS 19	Executing a CCW circular interpolation
IPCW	Register bit	RIPS 18	Executing a CW circular interpolation
IPE	Register bit	RIPS 17	Executing a linear interpolation by entering master axis feed amount
IPEx	Register bit	RIPS 4	X axis linear interpolation mode from a specified master axis feed amount
IPEy	Register bit	RIPS 5	Y axis linear interpolation mode from a specified master axis feed amount
IPFx	Register bit	RIPS 12	Specify a synthetic constant speed for the X axis
IPFy	Register bit	RIPS 13	Specify synthetic constant speed for the Y axis
IPL	Register bit	RIPS 16	Executing a normal linear interpolation
IPLx	Register bit	RIPS 0	X axis is in normal linear interpolation mode
IPLy	Register bit	RIPS 1	Y axis is in normal linear interpolation mode
IPSx	Register bit	RIPS 8	X axis is in circular interpolation mode
IPSy	Register bit	RIPS 9	Y axis is in circular interpolation mode
IRC1	Register bit	RIRQ 8	Enable an INT when the comparator1 conditions are met
IRC2	Register bit	RIRQ 9	Enable an INT when the comparator2 conditions are met
IRC3	Register bit	RIRQ 10	Enable an INT when the comparator3 conditions are met
IRC4	Register bit	RIRQ 11	Enable an INT when the comparator4 conditions are met
IRC5	Register bit	RIRQ 12	Enable an INT when the comparator5 conditions are met
IRCL	Register bit	RIRQ 13	Enable an INT when the count value is reset by a CLR input
IRDE	Register bit	RIRQ 7	Enable an INT when the deceleration is finished
IRDR	Register bit	RIRQ 17	Enable an INT when the ±DR input changes
IRDS	Register bit	RIRQ 6	Enable an INT when the deceleration starts
IREN	Register bit	RIRQ 0	Enable an INT when there is a normal stop
IRLT	Register bit	RIRQ 14	Enable an INT when the count value is latched by an LTC input
IRND	Register bit	RIRQ 3	Enable an INT when writing to the 2nd pre-register for comparator5 is enabled
IRNM	Register bit	RIRQ 2	Enable an INT when writing to 2nd pre-register for operation is enabled
IRNX	Register bit	RIRQ 1	Enable an INT by continuing to the next operation
IROL	Register bit	RIRQ 15	Enable an INT when the count value is latched by an ORG input
IRSA	Register bit	RIRQ 18	Enable an INT by turning ON the $\overline{\text{CSTA}}$ input
IRSD	Register bit	RIRQ 16	Enable an INT by turning ON the SD input
IRUE	Register bit	RIRQ 5	Enable an INT when the acceleration is finished
IRUS	Register bit	RIRQ 4	Enable an INT when acceleration starts
ISC1	Register bit	RIST 8	Comparator 1 conditioned status
ISC2	Register bit	RIST 9	Comparator 2 conditioned status
ISC3	Register bit	RIST 10	Comparator 3 conditioned status
ISC4	Register bit	RIST 11	Comparator 4 conditioned status
ISC5	Register bit	RIST 12	Comparator 5 conditioned status
ISCL	Register bit	RIST 13	Reset the count value when a CLR signal is input

Label	Type	Position	Description
ISDE	Register bit	RIST 7	Equals 1 when deceleration is finished
ISDS	Register bit	RIST 6	Equals 1 when deceleration starts
ISEN	Register bit	RIST 0	Equals 1 when stopped automatically
ISLT	Register bit	RIST 14	Equals 1 when the count value is latched by an LTC input
ISMD	Register bit	RIST 18	Equals 1 when a –DR input signal is input.
ISND	Register bit	RIST 3	Enable writing to the 2nd pre-register for comparator5
ISNM	Register bit	RIST 2	Enable writing to the 2nd pre-register for operations
ISNX	Register bit	RIST 1	Set to 1 when you want the next operation to run continuously
ISOL	Register bit	RIST 15	Latched count value from the ORG input
ISPD	Register bit	RIST 17	Equals 1 when the +DR input is ON
ISSA	Register bit	RIST 19	Equals 1 when the CSTA input is ON
ISSD	Register bit	RIST 16	Equals 1 when the SD input is ON
ISUE	Register bit	RIST 5	Equals 1 when the acceleration is finished
ISUS	Register bit	RIST 4	Equals 1 when to start acceleration
LTCH	Command	29h	Substitute the LTC input (for counting or latching)
LTCL	Register bit	RENV1 23	Select the trigger edge for the LTC signal (0: Falling edge, 1: Rising edge)
LTCx	Terminal name		Latch the input for the X axis
LTCy	Terminal name		Latch the input for the Y axis
LTFD	Register bit	RENV5 14	Latch the current speed data in place of COUNTER3
LTM0 to 1	Register bits	RENV5 12-13	Specify the latch timing of COUNTERS 1 to 4
LTOF	Register bit	RENV5 15	Stop the latch using hardware timing
MADJ	Register bit	RMD 26	Disable the FH correction function
MAX0 to 3	Register bits	RMD 20-21	Specify the axis used to control stopping for a simultaneous start
MCCE	Register bit	RMD 11	Stop the operation of COUNTER1 (command position)
METM	Register bit	RMD 12	Select the operation completion timing (0: Stop at the end of a cycle, 1: Stop on a pulse)
MINP	Register bit	RMD 9	The operation is complete when the INP input turns ON
MIPF	Register bit	RMD 15	Enable a synthetic constant speed during an interpolation operation
MPIE	Register bit	RMD 27	Execute an end point draw operation at end of a circular interpolation operation
MOD	Register bits	RMD 0-6	Operation mode selection
MPCI	Register bit	RMD 14	Start control positioning using a PCI input
MSDE	Register bit	RMD 8	Decelerate (decelerate and stop) when the SD input turns ON
MSDP	Register bit	RMD 13	Specify the ramping-down point manually
MSMD	Register bit	RMD 10	S-curve acceleration/deceleration (linear accel/decel when 0)
MSN0 to 1	Register bits	RMD 16-17	Sequence number used to control the operation block
MSPE	Register bit	RMD 24	Enable $\overline{\text{CSTP}}$ input
MSPO	Register bit	RMD 25	Output a $\overline{\text{CSTP}}$ (simultaneous stop) signal when stopped by an error
MSTSB0	Byte map name	0 when using a Z80	Read the main status bits (bits 0 to 7)
MSTSB1	Byte map name	1 when using a Z80	Read the main status bits (bits 8 to 15)
MSTSW	Word map name	0 when using an 8086	Read the main status bits(bits 0 to 15)
MSY0 to 1	Register bits	RMD 18-19	Specify the simultaneous start conditions
$\overline{\text{MVCx}}$	Terminal name		Monitor the output while feeding the X-axis at low speed
$\overline{\text{MVCy}}$	Terminal name		Monitor the output while feeding the Y-axis at low speed
NOP	Command	00h	(Invalid command)
ORGL	Register bit	RENV1 7	Select the input logic for the ORG signal (0: Negative logic, 1: Positive logic)
ORGx	Terminal name		Zero position signal for the X axis
ORGy	Terminal name		Zero position signal for the Y axis
ORM0 to 3	Register bits	RENV3 0-3	Select the zero return method
OTP0 to 7	General-purpose port name	OTPW 0-7	General-purpose port
OTPB	Byte map name	2 when using a Z80	Change the status of the general-purpose output port (only output the specified bits)
OTPW	Word map name	2 when using an 8086	Change status of the general-purpose output port (only output the specified bits)
OUTx	Terminal name		Motor driving pulse signal for the X axis
OUTy	Terminal name		Motor driving pulse signal for the Y axis
P+Register name			Refers to a particular pre-register
P0L	Register bit	RENV2 16	Select output logic of P0 terminal (0: Negative logic, 1: Positive logic)
P0x/FUPx	Terminal name		General-purpose port 0 for the X axis / Monitor output during acceleration
P0y/FUPy	Terminal name		General-purpose port 0 for the Y axis / Monitor output during acceleration
P1x/FDWx	Terminal name		General-purpose port 1 for the X axis / Monitor output during deceleration
P1y/FDWy	Terminal name		General-purpose port 1 for the Y axis / Monitor output during deceleration
P2x/MVCx	Terminal name		General-purpose port 2 for the X axis / Feeding at low speed
P2y/MVCy	Terminal name		General-purpose port 2 for the Y axis / Feeding at low speed
P3x/CP1x(+SLx)	Terminal name		General-purpose port 3 for the X axis / Comparator1 output (+ software limit)
P3y/CP1y(+SLy)	Terminal name		General-purpose port 3 for the Y axis / Comparator1 output (+ software limit)
P4x/CP2x(-SLx)	Terminal name		General-purpose port 4 for the X axis / Comparator2 output (- software limit)
P4y/CP2y(-SLy)	Terminal name		General-purpose port 4 for the Y axis / Comparator2 output (- software limit)
P5x/CP3x	Terminal name		General-purpose port 5 for the X axis / Comparator3 output
P5y/CP3y	Terminal name		General-purpose port 5 for the Y axis / Comparator3 output
P6x/CP4x/IDXx	Terminal name		General-purpose port 6 for the X axis / Comparator4 output (simultaneous output)

Label	Type	Position	Description
P6y/CP4y/IDXy	Terminal name		General-purpose port 6 for the Y axis / Comparator4 output (simultaneous output)
P7x/CP5x	Terminal name		General-purpose port 7 for the X axis / Comparator5 output
P7y/CP5y	Terminal name		General-purpose port 7 for the Y axis / Comparator5 output
P0M0 to 1	Register bits	RENV2 0-1	Specify the P0/FUP terminal details
P0RST	Command	10h	Set the general-purpose output port terminal P0 LOW
P0SET	Command	18h	Set the general-purpose output port terminal P0 HIGH
P1L	Register bit	RENV2 17	Set the P1 terminal output logic (0: Negative logic, 1: Positive logic)
P1M0 to 1	Register bits	RENV2 2-3	Specify the P1/FDW terminal details
P1RST	Command	11h	Set the general-purpose output port terminal P1 LOW
P1SET	Command	19h	Set the general-purpose output port terminal P1 HIGH
P2M0 to 1	Register bits	RENV2 4-5	Specify the P2/MVC terminal details
P2RST	Command	12h	Set the general-purpose output port terminal P2 LOW
P2SET	Command	1Ah	Set the general-purpose output port terminal P2 HIGH
P3M0 to 1	Register bits	RENV2 6-7	Specify the P3/CP1 (+SL) terminal details
P3RST	Command	13h	Set the general-purpose output port terminal P3 LOW
P3SET	Command	1Bh	Set the general-purpose output port terminal P3 HIGH
P4M0 to 1	Register bits	RENV2 8-9	Specify the P4/CP2 (-SL) terminal details
P4RST	Command	14h	Set the general-purpose output port terminal P4 LOW
P4SET	Command	1Ch	Set the general-purpose output port terminal P4 HIGH
P5M0 to 1	Register bits	RENV2 10-11	Specify the P5/CP3 terminal details
P5RST	Command	15h	Set the general-purpose output port terminal P5 LOW
P5SET	Command	1Dh	Set the general-purpose output port terminal P5 HIGH
P6M0 to 1	Register bits	RENV2 12-13	Specify the P6/CP4/IDX terminal details
P6RST	Command	16h	Set the general-purpose output port terminal P6 LOW
P6SET	Command	1Eh	Set the general-purpose output port terminal P6 HIGH
P7M0 to 1	Register bits	RENV2 14-15	Specify the P7/CP5 terminal details
P7RST	Command	17h	Set the general-purpose output port terminal P7 LOW
P7SET	Command	1Fh	Set the general-purpose output port terminal P7 HIGH
PAx	Terminal name		Manual pulsar phase A input for the X axis
PAy	Terminal name		Manual pulsar phase A input for the Y axis
PBx	Terminal name		Manual pulsar phase B input for the X axis
PBy	Terminal name		Manual pulsar phase B input for the Y axis
PCPCAN	Command	27h	Clear the pre-register (PRCP5) for PCMP5
PCSL	Register bit	RENV1 24	Set the input logic for the PCSn signal (0: Negative logic, 1: Positive logic)
PCSx	Terminal name		Start positioning control for the X axis
PCSy	Terminal name		Start positioning control for the Y axis
PDIR	Register bit	RENV2 26	Reverse the counting direction of the PA and PB inputs
$\overline{PEx}$	Terminal name		Enable the PA, PB, +DR, and –DR inputs for the X axis
$\overline{PEy}$	Terminal name		Enable the PA, PB, +DR, and –DR inputs for the Y axis
PIM0 to 1	Register bits	RENV2 24-25	Specify the PA and PB input details
PINF	Register bit	RENV2 19	Apply a noise filter to the PA/PB inputs
PMD0 to 2	Register bits	RENV1 0-2	Specify the output pulse details
PRICAN	Command	26h	Clear the operation pre-register
RCMP1	Register name		Comparison data for comparator1
RCMP2	Register name		Comparison data for comparator2
RCMP3	Register name		Comparison data for comparator3
RCMP4	Register name		Comparison data for comparator4
RCMP5	Register name		Comparison data for comparator5
RCUN1	Register name		COUNTER1 (command position)
RCUN2	Register name		COUNTER2 (mechanical position)
RCUN3	Register name		COUNTER3 (deflection counter)
RCUN4	Register name		COUNTER4 (general-purpose counter)
$\overline{RD}$	Terminal name		Read signal
RDP	Register name		Ramping-down point
RDR	Register name		Deceleration rate
RDS	Register name		S-curve range of deceleration
RENV1	Register name		Environment setting register 1 (Specify the input/output terminals)
RENV2	Register name		Environment setting register 2 (Specify the details for the general-purpose port)
RENV3	Register name		Environment setting register 3 (Specify the details for a zero return or counter)
RENV4	Register name		Environment setting register 4 (Specify the details for comparators 1 to 4))
RENV5	Register name		Environment setting register 5 (Specify the detail for comparator 5)
RENV6	Register name		Environment setting register 6 (Specify the feed amount correction)
RENV7	Register name		Environment setting register 7 (Specify the vibration reduction function details)
REST	Register name		Error INT status
RFA	Register name		Speed for feeding the feed correction amount
RFH	Register name		Operation speed
RFL	Register name		Initial speed
RIP	Register name		Center position of a circular interpolation / Master axis feed amount when executing a linear interpolation using multiple LSI chips
RIPS	Command	FFh	Copy the RIPS register data to BUF
RCI	Register name		Contains the number of steps for a circular interpolation
RIRQ	Register name		Enable various event interrupts

Label	Type	Position	Description
RIST	Register name		Event INT status
RLTC1	Register name		COUNTER1 (command position) latch data
RLTC2	Register name		COUNTER2 (mechanical position) latch data
RLTC3	Register name		COUNTER3 (deflection counter) latch data
RLTC4	Register name		COUNTER4 (general-purpose) latch data
RMD	Register name		Operation mode
RMG	Register name		Speed magnification rate
RMV	Register name		Feed amount or target position
RPLS	Register name		Number of pulses remaining to be fed
RPRCP5	Command	CBh	Copy PRCP5 data to BUF
RPRCI	Command	CCh	Copy PRCI data to BUF
RPRDP	Command	C6h	Copy PRDP data to BUF
RPRDR	Command	C4h	Copy PRDR data to BUF
RPRDS	Command	CAh	Copy PRDS data to BUF
RPRFH	Command	C2h	Copy PRFH data to BUF
RPRFL	Command	C1h	Copy PRFL data to BUF
RPRIP	Command	C8h	Copy PRIP data to BUF
RPRMD	Command	C7h	Copy PRMD data to BUF
RPRMG	Command	C5h	Copy PRMG data to BUF
RPRMV	Command	C0h	Copy PRMV data to BUF
RPRUR	Command	C3h	Copy PRUR data to BUF
RPRUS	Command	C9h	Copy PRUS data to BUF
RRCI	Command	FCh	Copy RCI data to BUF
RRCIC	Command	FDh	Copy RCIC data to BUF
RRCMP1	Command	E7h	Copy RCMP1 data to BUF
RRCMP2	Command	E8h	Copy RCMP2 data to BUF
RRCMP3	Command	E9h	Copy RCMP3 data to BUF
RRCMP4	Command	EAh	Copy RCMP4 data to BUF
RRCMP5	Command	EBh	Copy RCMP5 data to BUF
RRCUN1	Command	E3h	Copy RCUN1 data to BUF
RRCUN2	Command	E4h	Copy RCUN2 data to BUF
RRCUN3	Command	E5h	Copy RCUN3 data to BUF
RRCUN4	Command	E6h	Copy RCUN4 data to BUF
RRDP	Command	D6h	Copy RDP data to BUF
RRDR	Command	D4h	Copy RDR data to BUF
RRDS	Command	DAh	Copy RDS data to BUF
RRENV1	Command	DCh	Copy RENV1 data to BUF
RRENV2	Command	DDh	Copy RENV2 data to BUF
RRENV3	Command	DEh	Copy RENV3 data to BUF
RRENV4	Command	DFh	Copy RENV4 data to BUF
RRENV5	Command	E0h	Copy RENV5 data to BUF
RRENV6	Command	E1h	Copy RENV6 data to BUF
RRENV7	Command	E2h	Copy RENV7 data to BUF
RREST	Command	F2h	Copy REST data to BUF
RRFA	Command	DBh	Copy RFA data to BUF
RRFH	Command	D2h	Copy RFH data to BUF
RRFL	Command	D1h	Copy RFL data to BUF
RRIP	Command	D8h	Copy RIP data to BUF
RRIRQ	Command	ECh	Copy RIRQ data to BUF
RRIST	Command	F3h	Copy RIST data to BUF
RRLTC1	Command	EDh	Copy RLTC1 data to BUF
RRLTC2	Command	EEh	Copy RLTC2 data to BUF
RRLTC3	Command	EFh	Copy RLTC3 data to BUF
RRLTC4	Command	F0h	Copy RLTC4 data to BUF
RRMD	Command	D7h	Cop RMD data to BUF
RRMG	Command	D5h	Copy RMG data to BUF
RRMV	Command	D0h	Copy RMV data to BUF
RRPLS	Command	F4h	Copy RPLS data to BUF
RRSDC	Command	F6h	Copy RSDC data to BUF
RRSPD	Command	F5h	Copy RSPD data to BUF
RRSTS	Command	F1h	Copy RSTS data to BUF
RRUR	Command	D3h	Copy RUR data to BUF
RRUS	Command	D9h	Copy RUS data to BUF
RSDC	Register name		Automatically calculated value for the ramping-down point
RSPD	Register name		EZ count / Monitor current speed
$\overline{RST}$	Terminal name		Reset signal
RSTS	Register name		Extension status
RT0 to 15	Register bits	RENV7 0 to 15	Enter the RT time for the vibration reduction function
RUR	Register name		Acceleration rate
RUS	Register name		S-curve range during acceleration
SALM	Sub-status bit	SSTSW 11	Equals 1 when the ALM input is ON

Label	Type	Position	Description
SCLR	Register bit	RSTS 13	Equals 1 when the CLR input signal is ON
SCP1	Main status bit	MSTSW 8	Equals 1 when the CMP1 comparison conditions are met
SCP2	Main status bit	MSTSW 9	Equals 1 when the CMP2 comparison conditions are met
SCP3	Main status bit	MSTSW 10	Equals 1 when the CMP3 comparison conditions are met
SCP4	Main status bit	MSTSW 11	Equals 1 when the CMP4 comparison conditions are met
SCP5	Main status bit	MSTSW 12	Equals 1 when the CMP5 comparison conditions are met
SDIN	Register bit	RSTS 15	Equals 1 when the SD input signal is ON
SDIR	Register bit	RSTS 4	Set the operation direction (0: Plus direction, 1: Minus direction)
SDL	Register bit	RENV1 6	Set the input logic of the SD signal (0: Negative logic, 1: Positive logic)
SDLT	Register bit	RENV1 5	Specify the latch function for the SD input (0: ON, 1: OFF)
SDM	Register bit	RENV1 4	Select the process to execute when the SD input is ON (0: Deceleration only, 1: Decelerate and stop)
SDM0 to 1	Register bits	RIPS 20-21	Current phase of a circular interpolation
SDRM	Register bit	RSTS 12	Equals 1 when the -DR input signal is ON
SDRP	Register bit	RSTS 11	Equals 1 when the +DR input signal is ON
SDSTP	Command	4Ah	Deceleration stop
SDx	Terminal name		Ramping-down signal for the X axis
SDy	Terminal name		Ramping-down signal for the Y axis
SED0 to 1	Register bits	RIPS 22-23	Final phase of a circular interpolation
SELx	Command bit name	COMW 8	Select the X axis
SELy	Command bit name	COMW 9	Select the Y axis
SEMG	Register bit	RSTS 7	Equals 1 when the $\overline{CEMG}$ input signal is ON
SEND	Main status bit	MSTSW 2	Equals 0 when started, becomes 1 when stopped automatically
SERC	Register bit	RSTS 9	Equals 1 when the ERC output signal is ON
SERR	Main status bit	MSTSW 3	Equals 1 when an error interrupt occurs
SEZ	Register bit	RSTS 10	Equals 1 when the EZ input signal is ON
SFC	Sub-status bit	SSTSW 10	Equals 1 when feeding at low speed
SFD	Sub-status bit	SSTSW 9	Equals 1 when decelerating
SFU	Sub-status bit	SSTSW 8	Equals 1 when accelerating
SINP	Register bit	RSTS 16	Equals 1 when the INP input signal is ON
SINT	Main status bit	MSTSW 4	Equals 1 when an event interrupt occurs
SLTC	Register bit	RSTS 14	Equals 1 when the LTC input signal is ON
SMEL	Sub-status bit	SSTSW 13	Equals 1 when the –EL input is ON
SMOV	Main status bit	MSTSW 1	Equals 1 when the motor is operating
SORG	Sub-status bit	SSTSW 14	Equals 1 when the ORG input is ON
SPCS	Register bit	RSTS 8	Equals 1 when the PCS input signal is ON
SPDF	Main status bit	MSTSW 15	Equals 1 when the pre-register for comparator 5 is full
SPEL	Sub-status bit	SSTSW 12	Equals 1 when the +EL input is ON
SPRF	Main status bit	MSTSW 14	Equals 1 when the next-operation pre-register is full
SPSTA	Command	2Ah	Executes the same process as the $\overline{CSTA}$ input, but only for this axis
SRCH	Register bit	RENV3 2	Zero position search
SRST	Command	04h	Software reset
SRUN	Main status bit	MSTSW 0	Equals 1 while starting
SSC0 to 1	Main status bits	MSTSW 7-6	Sequence code
SSCM	Main status bit	MSTSW 0	Equals 1 when a start command has already been written
SSD	Sub-status bit	SSTSW 15	Equals 1 when the SD input is ON (latched signal)
SSTA	Register bit	RSTS 5	Equals 1 when the $\overline{CSTA}$ input signal is ON
SSTP	Register bit	RSTS 6	Equals 1 when the $\overline{CSTP}$ input signal is ON
SSTSB	Byte map name	3 when using a Z80	Used to read the sub status
SSTSW	Word map name	2 when using an 8086	Used to read the sub status, general input/output port
$\overline{STAx}$	Terminal name		X axis external start signal
$\overline{STAy}$	Terminal name		Y axis external start signal
STAD	Command	52h	High speed start 1 (FH low speed -> deceleration stop)
STAFH	Command	51h	Start using FH low speed
STAFL	Command	50h	Start using FL low speed
STAM	Register bit	RENV1 18	Set the input trigger type for the $\overline{CSTA}$ signal (0: Level trigger, 1: Edge trigger)
STAON	Command	28h	Substitute for a PCs input
STAUD	Command	53h	High speed start 2 (acceleration -> FH low speed -> deceleration stop)
STOP	Command	49h	Immediate stop
STPM	Register bit	RENV1 19	Set the $\overline{CSTP}$ stop method (0: Immediate stop, 1: Deceleration stop)
SYI0 to 1	Register bits	RENV5 20-21	Select the axis used to input an internal synchronous signal
SYO0 to 3	Register bits	RENV5 16-19	Set the output timing of the internal synchronous signal
WPRCI	Command	8Ch	Write BUF data into PRCI
WPRCP5	Command	8Bh	Write BUF data into PRCP5
WPRDP	Command	86h	Write BUF data into PRDP
WPRDR	Command	84h	Write BUF data into PRDR
WPRDS	Command	8Ah	Write BUF data into PRDS
WPRFH	Command	82h	Write BUF data into PRFH
WPRFL	Command	81h	Write BUF data into PRFL
WPRIP	Command	88h	Write BUF data into PRIP
WPRMD	Command	87h	Write BUF data into PRMD
WPRMG	Command	85h	Write BUF data into PRMG

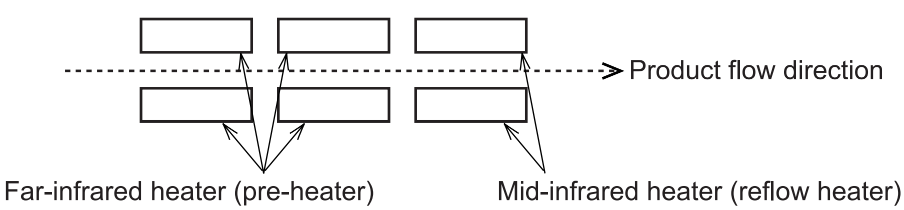
Label	Type	Position	Description
WPRMV	Command	80h	Write BUF data into PRMV
WPRUR	Command	83h	Write BUF data into PRUR
WPRUS	Command	89h	Write BUF data into PRUS
$\overline{WR}$	Terminal name		Write signal
WRCI	Command	BCh	Write BUF data into the RCI register
WRCMP1	Command	A7h	Write BUF data into the RCMP1 register
WRCMP2	Command	A8h	Write BUF data into the RCMP2 register
WRCMP3	Command	A9h	Write BUF data into the RCMP3 register
WRCMP4	Command	AAh	Write BUF data into the RCMP4 register
WRCMP5	Command	ABh	Write BUF data into the RCMP5 register
WRCUN1	Command	A3h	Write BUF data into the RCUN1 register
WRCUN2	Command	A4h	Write BUF data into the RCUN2 register
WRCUN3	Command	A5h	Write BUF data into the RCUN3 register
WRCUN4	Command	A6h	Write BUF data into the RCUN4 register
WRDP	Command	96h	Write BUF data into the RDP register
WRDR	Command	94h	Write BUF data into the RDR register
WRDS	Command	9Ah	Write BUF data into the RDS register
WRENV1	Command	9Ch	Write BUF data into the RENV1 register
WRENV2	Command	9Dh	Write BUF data into the RENV2 register
WRENV3	Command	9Eh	Write BUF data into the RENV3 register
WRENV4	Command	9Fh	Write BUF data into the RENV4 register
WRENV5	Command	A0h	Write BUF data into the RENV5 register
WRENV6	Command	A1h	Write BUF data into the RENV6 register
WRENV7	Command	A2h	Write BUF data into the RENV7 register
WRFA	Command	9Bh	Write BUF data into the RFA register
WRFH	Command	92h	Write BUF data into the RFH register
WRFL	Command	91h	Write BUF data into the RFL register
WRIP	Command	98h	Write BUF data into the RIP register
WRIRQ	Command	ACh	Write BUF data into the RIRQ register
WRMD	Command	97h	Write BUF data into the RMD register
WRMG	Command	95h	Write BUF data into the RMG register
WRMV	Command	90h	Write BUF data into the RMV register
$\overline{WRQ}$	Terminal name		Wait request signal
WRUR	Command	93h	Write BUF data into the RUR register
WRUS	Command	99h	Write BUF data into the RUS register

[Handling Precautions]

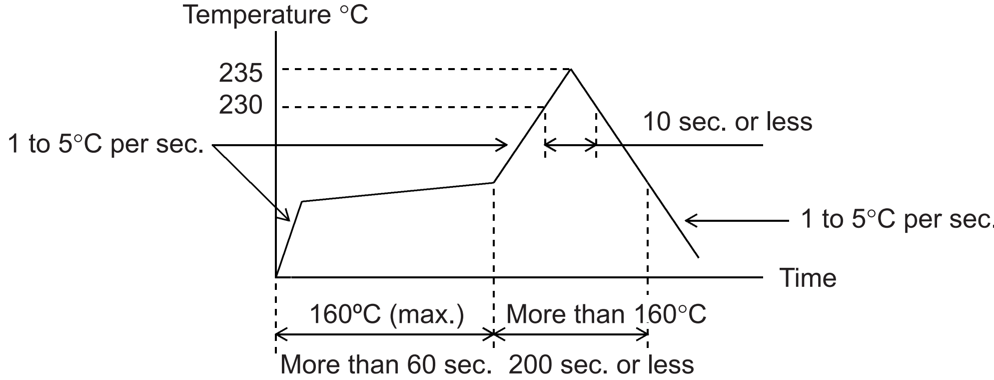
1. Design precautions
   1) Never exceed the absolute maximum ratings, even for a very short time.
   2) Take precautions against the influence of heat in the environment, and keep the temperature around the LSI as cool as possible.
   3) Please note that ignoring the following may result in latching up and may cause overheating and smoke.
      - Do not apply a voltage greater than +5 V to the input/output terminals and do not pull them below GND.
      - Make sure you consider the input timing when power is applied.
      - Be careful not to introduce external noise into the LSI.
      - Hold the unused input terminals to +5 V or GND level.
      - Do not short-circuit the outputs.
      - Protect the LSI from inductive pulses caused by electrical sources that generate large voltage surges, and take appropriate precautions against static electricity.
   4) Provide external circuit protection components so that overvoltages caused by noise, voltage surges, or static electricity are not fed to the LSI.

2. Precautions for transporting and storing LSIs
   1) Always handle LSIs carefully and keep them in their packages. Throwing or dropping LSIs may damage them.
   2) Do not store LSIs in a location exposed to water droplets or direct sunlight.
   3) Do not store the LSI in a location where corrosive gases are present, or in excessively dusty environments.
   4) Store the LSIs in an anti-static storage container, and make sure that no physical load is placed on the LSIs.

3. Precautions for installation
   1) In order to prevent damage caused by static electricity, pay attention to the following.
      - Make sure to ground all equipment, tools, and jigs that are present at the work site.
      - Ground the work desk surface using a conductive mat or similar apparatus (with an appropriate resistance factor). However, do not allow work on a metal surface, which can cause a rapid change in the electrical charge on the LSI (if the charged LSI touches the surface directly) due to extremely low resistance.
      - When picking up an LSI using a vacuum device, provide anti-static protection using a conductive rubber pick up tip. Anything which contacts the leads should have as high a resistance as possible.
      - When using a pincer that may make contact with the LSI terminals, use an anti-static model. Do not use a metal pincer, if possible.
      - Store unused LSIs in a PC board storage box that is protected against static electricity, and make sure there is adequate clearance between the LSIs. Never directly stack them on each other, as it may cause friction that can develop an electrical charge.
   2) Operators must wear wrist straps which are grounded through approximately 1M-ohm of resistance.
   3) Use low voltage soldering devices and make sure the tips are grounded.
   4) Do not store or use LSIs, or a container filled with LSIs, near high-voltage electrical fields, such those produced by a CRT.
   5) To preheat LSIs for soldering, we recommend keeping them at a high temperature in a completely dry environment, i.e. 125℃ for 24 hours. The LSI must not be exposed to heat more than 2 times.
   6) When using an infrared reflow system to apply solder, we recommend the use of a far-infrared pre-heater and mid-infrared reflow devices, in order to ease the thermal stress on the LSIs.

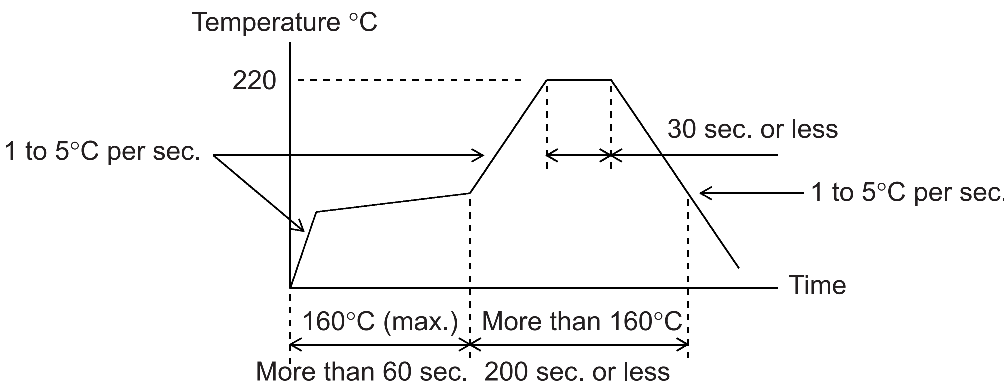


The maximum heat they can be exposed to is 235°C for 10 seconds or less. The increase in temperature must be at a rate of 1 to 5°C per second. At this temperature, the solder can be reflowed a maximum of 2 times.



[Recommended temperature pattern for infrared reflow system]

7) When using hot air for solder reflow, the restrictions are the same as for infrared reflow equipment.
8) When using vapor phase solder, the LSI can be exposed to a maximum of 220°C for 30 seconds or less. The increase in temperature must be at a rate of 1 to 5°C per second. At this temperature, the solder can be reflowed a maximum of 2 times.



[Recommended temperature pattern for reflowing vapor phase solder]

4. Other precautions
1) When the LSI will be used in poor environments (high humidity, corrosive gases, or excessive amounts of dust), we recommend applying a moisture prevention coating.
2) The package resin is made of fire-retardant material; however, it can burn. When baked or burned, it may generate gases or fire. Do not use it near ignition sources or flammable objects.
3) This LSI is designed for use in commercial apparatus (office machines, communication equipment, measuring equipment, and household appliances). If you use it in any device that may require high quality and reliability, or where faults or malfunctions may directly affect human survival or injure humans, such as in nuclear power control devices, aviation devices or spacecraft, traffic signals, fire control, or various types of safety devices, we will not be liable for any problem that occurs, even if it was directly caused by the LSI. Customers must provide their own safety measures to ensure appropriate performance in all circumstances.

NPM **NIPPON PULSE MOTOR CO., LTD.**
Tokyo Office: No. 16-13, 2-chome, Hongo, Bunkyo-ku, Tokyo 113-0033, Japan
Phone: 81-3-3813-8841 Fax: 81-3-3813-2940
E-mail: int-l@npm.co.jp http://www.pulsemotor.com
U.S. Branch Office: 1047 Norwood St., Suite B, Radford, VA 24141, U.S.A.
Phone: 1-540-633-1677 Fax: 1-540-6331674
E-mail: info@pulsemotor-usa.com http://www.pulsemotor-usa.com

MNAL. No. PCL-6025-1 1B-A-5110-0.2 (5110) ims