4-Axes Motor Control IC with Interpolation Function

MCX314As User's Manual

2006-03-07 Ver. 1.6

NOVA electronics

Update history

3/7/2006 Ver. 1.6

P115~117 (the following items in the table)

	Wavelength → Width
	Reservation Time → Hold Time
	Established Time → Setup Time

1/6/2006 Ver. 1.5

P70 line 22	the start → the end
P70 line 24	the end → the start

5/20/2005 Ver.1.4

P67 line 41	nWR2~3 → nWR1~3
P73 line 42	in the table in 4.7 Output Register: WR4, the value “ZOUT0” for D12 → “UOUT0”
P80 line 14	/finish point: -8,388,608 ~ +8,388,607 → /finish point: −2,147,483,646 ~ +2,147,483,646
P97 line 42	nWR2~5 → nWR1~5

3/14/2005 Ver.1.3

P62	replaced diagram

11/17/2004 Ver. 1.2

P15 line 30	tolerance → jitter
P36 line 17	② → (2)
P36 line 40	as the pulse count (P) → as the output pulse numbers (P)
P38 line 53	Interruption of automatic home search → Suspension of automatic home search
P54 line 6	During the power resetting, → When resetting,
P58 line 14	nEXPP → nEXPM
P58 line 19-22	(Corrected a paragraph.)
P64 line 56	HKMT+ → HLMT+
P65 line 6	HKMT- → HLMT-
P70 line 31-32	(Added a paragraph, “Each axis is with…”.)
P82 line 33	Acceleration/Deceleration and jerk is → Acceleration/Deceleration is
P91 line 11-12	(Corrected a paragraph.)
P93 line 13	within this period of time → after this period of time
P93 line 23	real position → logical position
P94 line 8	real position → logical position
P94 line 22	real position → logical position
P94 line 36	real position → logical position
P117 line 11	WRN→nOUT7~0 → WRN↑→nOUT7~0
P124 line 27-34	(Added descriptions of “multiple…” to the end of each line.)
P124 line 31-32	(Added descriptions about “initial speed…”.)

Introduction

Before using the MCX314As, please read this manual thoroughly to ensure correct usage within the scope of the specification such as the signal voltage, signal timing, and operation parameter values.

In general, semiconductor products sometimes malfunction or fail to function. When incorporating this IC in a system, make sure that a safe system is designed to avoid any injuries or property damage caused by malfunctioning of this IC.

This IC is designed for application in general electronic devices (industrial automation devices, industrial robots, measuring instruments, computers, office equipment, household electrical goods, and so on). This IC is not intended for the use in high-performance high-reliability equipment whose failure or malfunctioning may directly cause death or injuries (atomic energy control equipment, aerospace equipment, transportation equipment, medical equipment, and various safety devices) and the operation for such use is not guaranteed. The customer shall be responsible for the use of this IC in any such high-performance and high-reliability equipment.

Installation of this IC

This IC is provided in the form of a lead-free package. The installation conditions are different from those of the conventional lead-soldered IC. See Chapter 16 for the installation conditions of this IC.

Notes on S-curve acceleration/deceleration driving

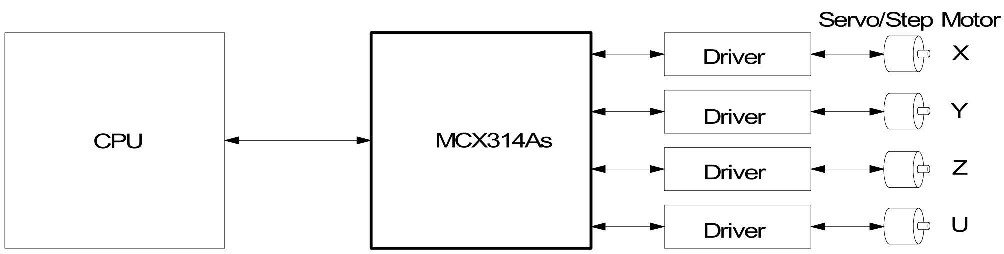
This IC is equipped with a function that performs decelerating stop for a fixed pulse drive with S-curve deceleration of the symmetrical acceleration/deceleration. However, when the initial speed is set to an extremely low speed (10 or less), slight premature termination or creep may occur. Before using a S-curve deceleration drive, make sure that your system allows premature termination or creep.

1. OUTLINE

MCX314As is a 4-axis motion control IC which can control 4 axes of either stepper motor or pulse type servo drivers for position, speed, and interpolation controls. All of the MCX314As' s function are controlled by specific registers. There are command registers, data registers, status registers and mode registers. This motion control IC has the following built-in functions:

■ Individual Control for 4 Axes

MCX314As controls motors through pulse string driving. The IC can control motors of four axes independently with a single chip. Each of the four axes has identical function capabilities, and is controlled by the same method of operation with constant speed, trapezoidal or S-curve driving.

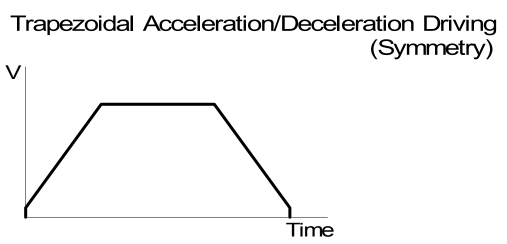


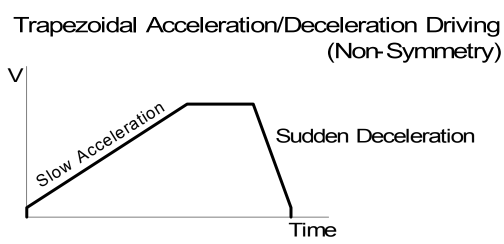
■ Speed Control

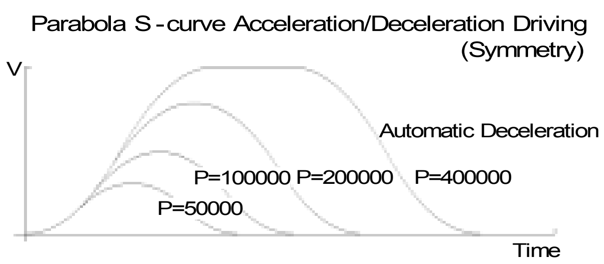
The speed range of the pulse output is from 1PPS to 4MPPS for constant speed, trapezoidal or S-curve acceleration/deceleration driving. Speed accuracy of the pulse output is less than ± 0.1% (at CLK=16MHz). The speed of driving pulse output can be freely changed during the driving.

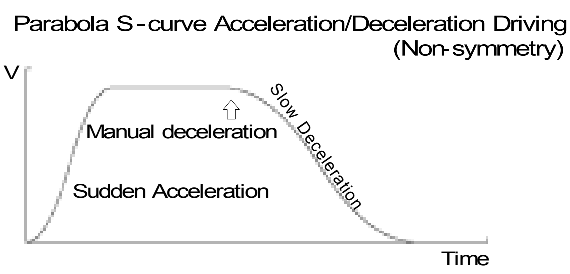
■ Acceleration/deceleration driving

The IC can control each axis for acceleration/deceleration of constant speed driving, trapezoidal acceleration/deceleration driving (symmetry/non-symmetry), and S-curve acceleration/deceleration. Automatic acceleration/deceleration of linear acceleration fixed speed pulse driving is available. Since a primary linear increase/decrease method is applied for S-curve acceleration/deceleration, the speed curve forms a secondary parabola acceleration/deceleration curve. In S-curve acceleration and deceleration fixed pulse driving, automatic deceleration is available for symmetrical S-curve only and triangle waveforms during S-curve acceleration/deceleration are prevented by a special method.



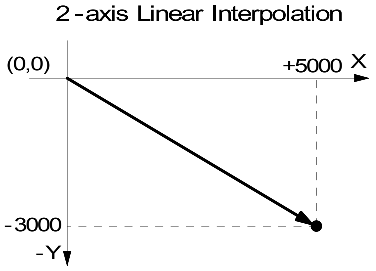


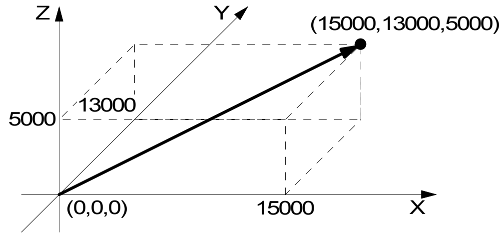




■ Linear Interpolation

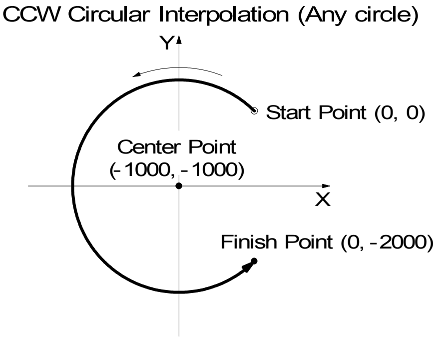
Any 2 or 3 axes can be selected to perform linear interpolation. The position boundary is between coordinates −2,147,483,646 and +2,147,483,646 (signed 32-bit format), and the positioning error is within ± 0.5 LSB (Least Significant Bit). The interpolation speed range is from 1 PPS to 4 MPPS.

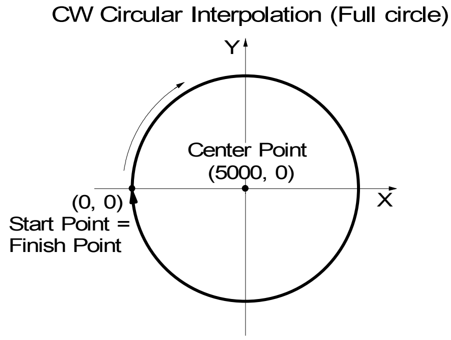




■ Circular Interpolation

Any 2 axes can be selected to perform circular interpolation. The position boundary is between coordinates −2,147,483,646 and +2,147,483,646 (signed 32-bit format), and the positioning error is within ± 1.0 LSB. The interpolation speed range is from 1 PPS to 4 MPPS.



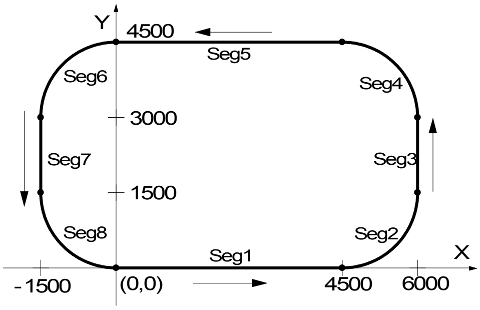


■ Bit Pattern Interpolation

This interpolation driving receives, for each axis in 16-bit units, interpolation data that was converted to bit patterns through the operation by the upper-level CPU and outputs interpolation pulses consecutively at the specified drive speed. This function enables drawing of various loci created by the upper-level CPU.

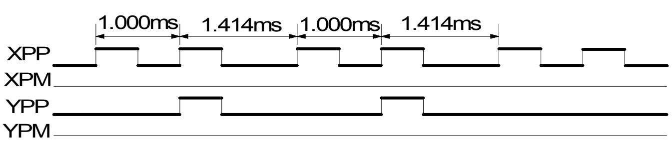
■ Continuous Interpolation

Different interpolation methods can be used continuously, linear interpolation → circular interpolation → linear interpolation …. The maximum drive speed of performing continuous interpolation is 2 MHz.



■ Constant Vector Speed Control

This function performs a constant vector speed. During the interpolation driving, MCX314As can set a 1.414 times pulse cycle for 2-axis simultaneous pulse output, and a 1.732-time pulse cycle for 3-axis simultaneous pulse output.



Example of Pulse Output of 2-Axis Interpolation Constant Vector Speed
(Vector speed: 1000pps)

■ Position Control

Each axis has a 32-bit logic position counter and a 32-bits real position counter. The logic position counter counts the number of output pulse, and the real position counter counts the feedback number of pulse from the external encoder or linear scale.

■ Compare Register and Software Limit

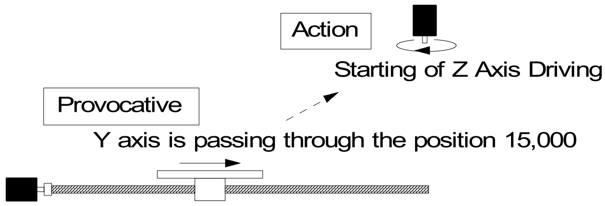
Each axis has two 32-bit compare registers for logical position counter and real position counter. The comparison result can be read from the status registers. The comparison result can be notified by an interrupt signal. These registers can be also functioned as software limits.

■ Automatic home search

This IC is equipped with a function that automatically executes a home search sequence without CPU intervention. The sequence comprises high-speed near home search → low-speed home search → encoder Z-phase search → offset drive. This function reduces the CPU load.

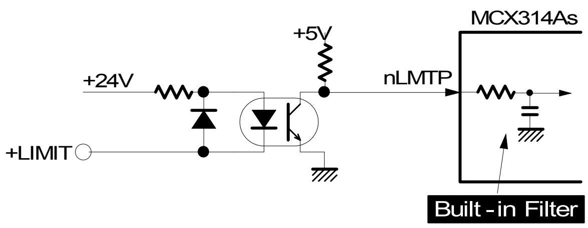
■ Synchronous action

The synchronous action is a function that performs a specified action such as starting or stopping of driving when an activation factor (provocative) occurs within each axis, between two axes, or with a device outside of IC by linking with a provocative.
Ten types of provocatives are available including the passing of the specified position, the starting/stopping of driving, and rising/falling of an input signal. Four types of actions are available, including starting/stopping of driving, saving a position counter value, writing of a drive speed, and so on.



■ Input signal filter

The IC is equipped with an integral type filter in the input step of each input signal. It is possible to set for each input signal whether the filter function is enabled or the signal is passed through. A filter time constant can be selected from eight types.



■ Driving by External Signal

It is possible to control each axis by external signals. The +/− direction fixed pulse driving and continuous pulse driving can be also performed through the external signals. This function is used for JOG or teaching modes, and will share the CPU load.

■ Servo Motor Feedback Signals

Each axis includes input pins for servo feedback signals such as in-positioning. An output signal for clearing a deviation counter is also available.

■ Interrupt Signals

Interrupt signals can be generated when: (1). the start / finish of a constant speed drive during the acceleration/deceleration driving, (2). the end of driving, and (3). the compare result once higher / lower the border-lines of the position counter range. An interrupt signal can be also generated during the interpolation driving.

■ Real Time Monitoring

During the driving, the present status such as logical position, real position, drive speed, acceleration / deceleration, status of accelerating / decelerating and constant driving can be read.

■ 8 or 16 Bits Data Bus Selectable

MCX314As can be connected to either 8-bit or 16-bit CPU Fig. 1.1 is the IC functional block diagram.

It consists of same functioned X, Y, Z and U axes control sections and interpolation counting sections. Fig. 1.2 is the functional block diagram of each axis control section.

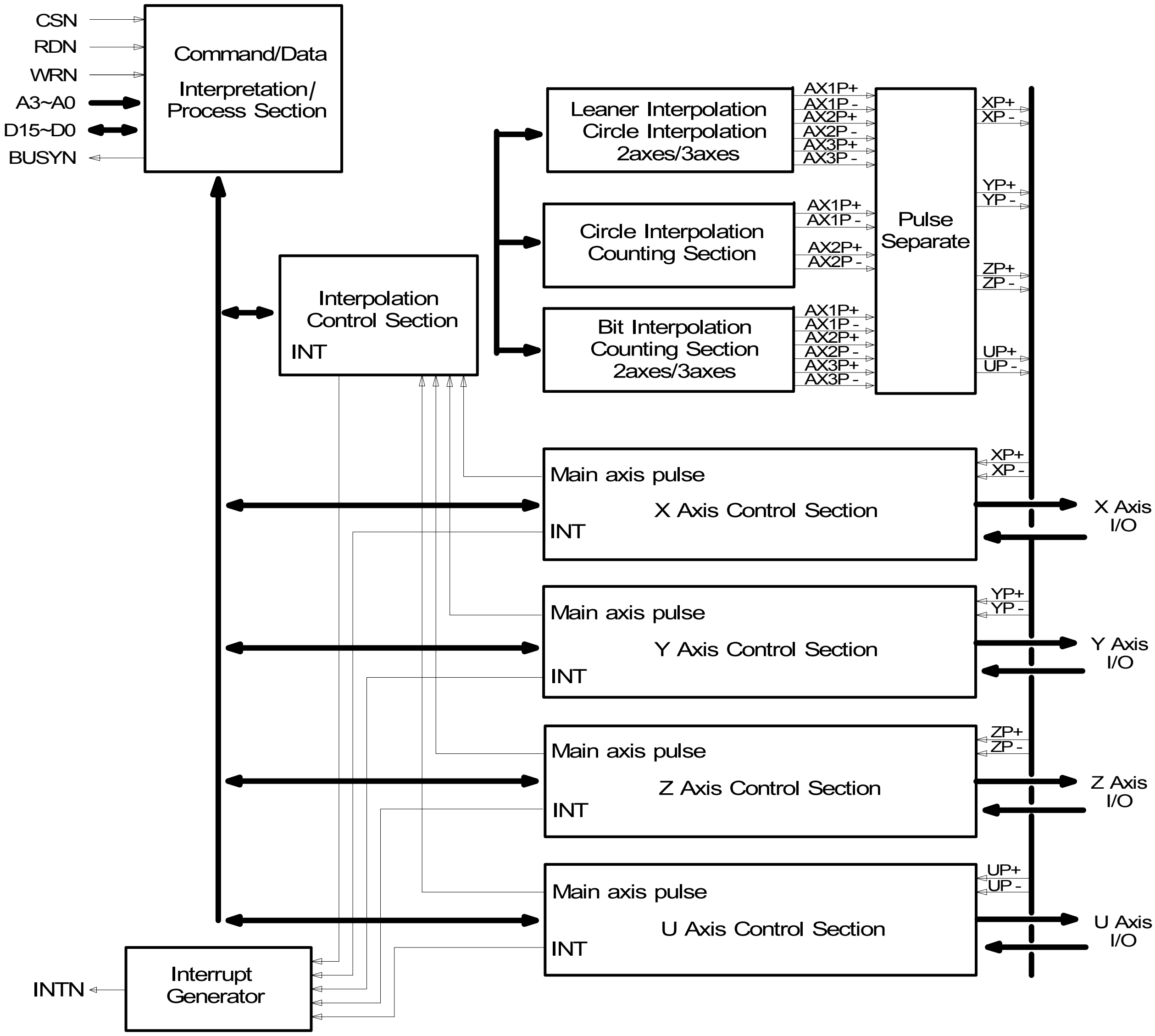


Fig. 1.1 MCX314As Functional Block Diagram

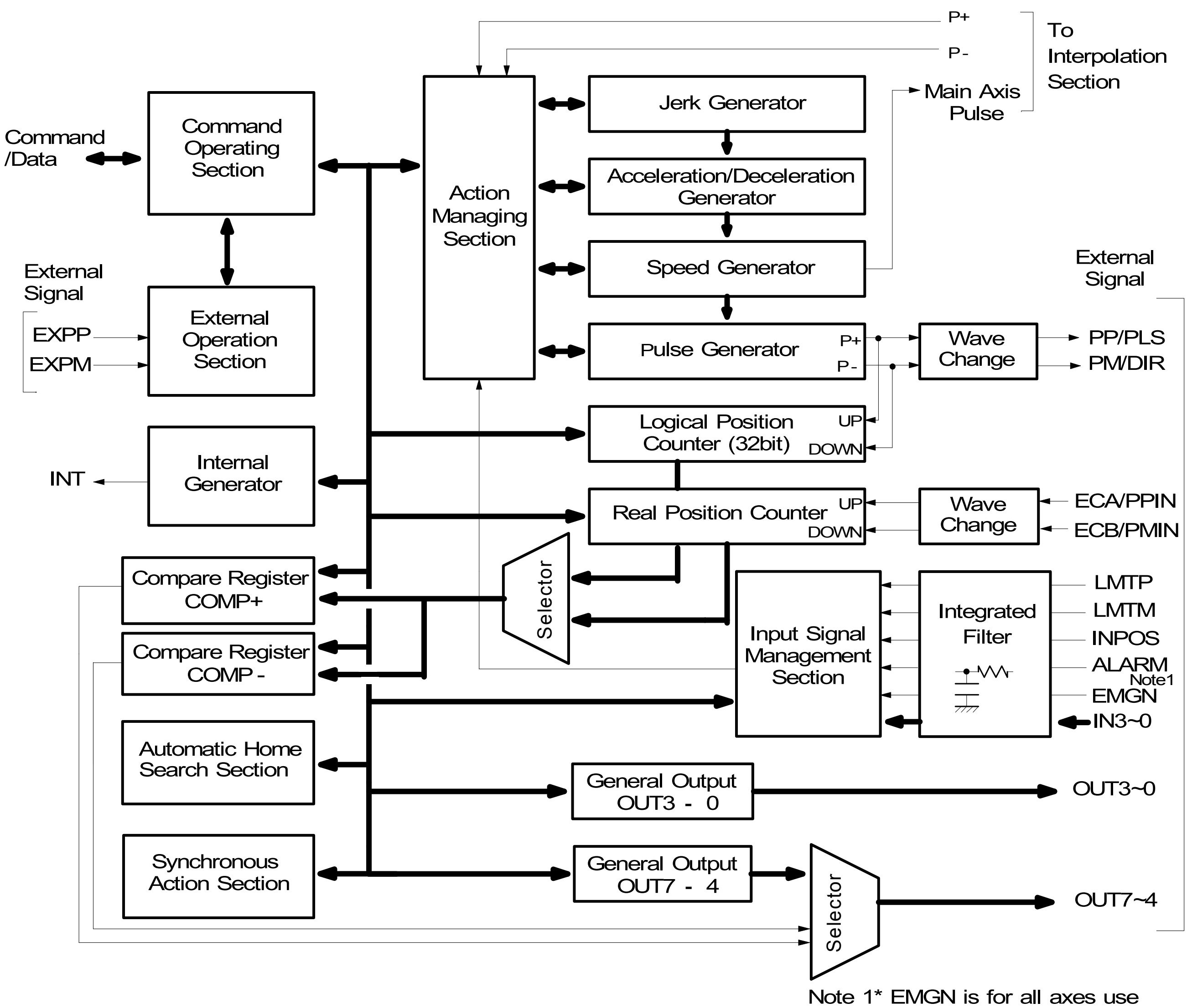


Fig.1.2 Block Diagram of the X,Y,Z and U-axis Control Section (for One Axis Only)

2. The Descriptions of Functions

2.1 Pulse Output Command

There are two kinds of pulse output commands: fixed pulse driving output and continuous pulse driving output.

2.1.1 Fixed Pulse Driving Output

When host CPU writes a pulse numbers into MCX314As for fixed pulse driving and configures the performance such as acceleration / deceleration and speed, MCX314As will generate the pulses and output them automatically. Fixed pulse driving operation is performed at acceleration/deceleration where the acceleration and deceleration are equal. As shown in Fig. 2.1, automatic deceleration starts when the number of pulses becomes less than the number of pulses that were utilized at acceleration, and driving terminates at completion of the output of the specified output pulses. For fixed pulse driving in linear acceleration, the following parameters must be set.

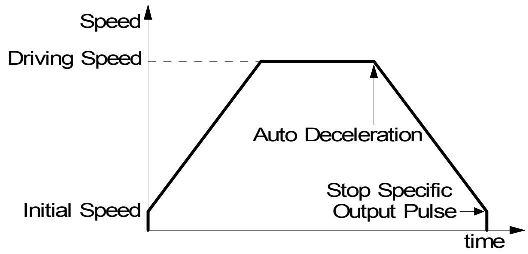


Fig.2.1 Fixed Pulse Driving

Parameter name	Symbol	Comment
Range	R	
Acceleration/Deceleration	A/D	When acceleration and deceleration are equal, the setting of deceleration is not required.
Initial Speed	SV	
Drive Speed	V	
Number of Output Pulse	P	

■ Changing the Number of Output Pulse in Driving

The number of output pulse can be changed in the fixed pulse driving. If the command is for increasing the output pulse, the pulse output profile is shown as Fig. 2.2 or 2.3. If the command is for decreasing the output pulses, the output pulse will be stopped immediately as shown in Fig. 2.4. Furthermore, when in the S-curve acceleration/deceleration driving mode, the output pulse number change will occur to an incomplete deceleration S-curve.

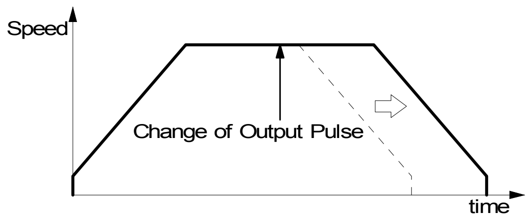


Fig.2.2 Change of Output Pulse Number in Driving

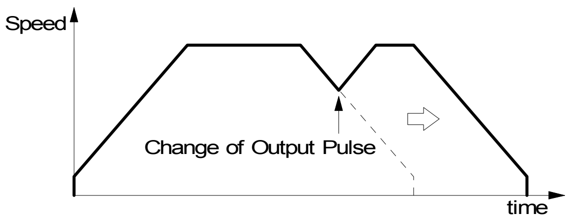


Fig. 2.3 Changing The Number of Output Pulse During Deceleration

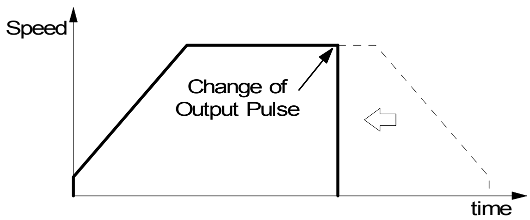


Fig. 2.4 Changing The Pulse Number Less Than Output Pulse Number

■ Manual Setting Deceleration for fixed pulse Acceleration/Deceleration Driving

As shown in Fig. 2.1, generally the deceleration of fixed pulse acceleration /deceleration driving is controlled automatically by MCX314As. However, in the following situations, it should be preset the deceleration point by the users.

- The change of speed is too often in the trapezoidal fixed pulse acceleration/deceleration driving.
- Set an acceleration, a deceleration, an jerk (acceleration increasing rate), and deceleration increasing rate individually for S-curve deceleration fixed pulse driving.
- When use circular interpolation, bit pattern interpolation and continuous interpolation for acceleration and deceleration.

In case of manual deceleration, please set D0 bit of register WR3 to 1, and use command (07h) for presetting deceleration point. As to the other operation, the setting is as same as that of fixed pulse driving.

■ Changing a Drive speed During Driving

In fixed pulse driving under linear acceleration at a constant speed, a drive speed (V) can be changed during driving. However, if a speed of fixed pulse driving is changed at linear acceleration, some premature termination may occur. Therefore, caution is necessary when using the IC by setting a low initial speed.

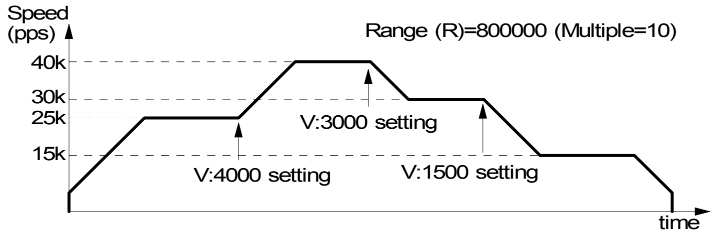


Fig. 2.5 Example of Drive Speed Change During Driving

A drive speed (V) cannot be changed during fixed pulse driving in S-curve deceleration.

■ Offset Setting for Acceleration/Deceleration Driving

The offset function can be used for compensating the pulses when the decelerating speed does not reach the setting initial speed during the S-curve fixed pulse driving. MCX314As will calculate the acceleration / deceleration point automatically, and will arrange the pulse numbers in acceleration equal to that in deceleration. The method is calculating the output acceleration pulses and comparing them with the remaining pulses. When the remaining pulses are equal to or less the pulses in acceleration, it starts the deceleration.

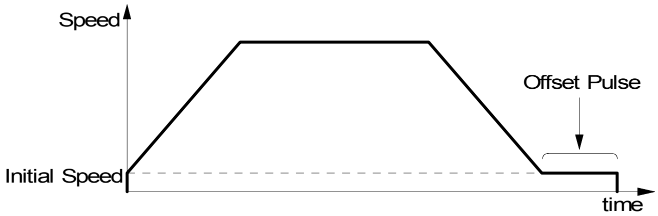


Fig. 2.6 Offset for Deceleration

When setting the offset for deceleration, MCX314As will start deceleration early for the offset. The greater is the positive value set for the offset, the closer the automatic declaration point becomes, increasing the creep pulses at the initial speed at deceleration termination. If a negative value is set for the offset value, output may stop prematurely before the speed reaches the initial speed (see Fig. 2.6).

The default value for offset is 8 when MCX314As power-on reset. It is not necessary to change the shift pulse value in the case of acceleration/deceleration fixed pulse driving. As for fixed driving in non-symmetrical trapezoidal acceleration/deceleration or S-curve acceleration/deceleration, if creep pulses or premature termination occurs at termination of driving due to the low initial speed setting, correct the speed by setting the acceleration counter offset to an appropriate value.

2.1.2 Continuous Pulse Driving Output

When the Continuous Pulse Driving is performed, MCX314As will drive pulse output in a specific speed until stop command or external stop signal is happened. The main application of continuous pulse driving is : home searching, teaching or speed control. The drive speed can be changed freely during continuous pulse driving.

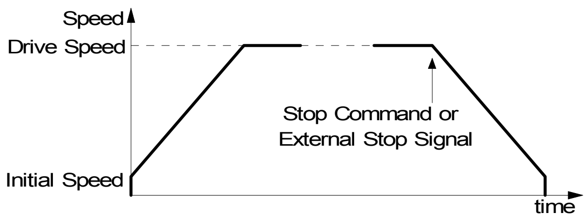


Fig. 2.7 Continuous Pulse Driving

Two stop commands are for stopping the continuous driving. One is “decelerating stop”, and the other is “sudden stop”. Four input pins, IN3~IN0, of each axis can be connected for external decelerating and sudden stop signals. Enable / disable, active levels and mode setting are possible.

■Stop Condition for External Input IN2 to IN0 in Continuous Pulse Driving

Assign an encoder Z-phase signal, a home signal, and a near home signal in nIN2 to nIN0. (Assign an encoder Z phase signal in nIN2.) Enable / disable and logical levels can be set at WR1 of each axis. For the application of high-speed searching, the user can set MCX314As in the acceleration/deceleration continuous pulse driving mode and enable IN2,1,0 in WR1. And then, MCX314As will perform the decelerating stop when the external signal IN2,1,0 is active.

For the application of low-speed searching, the user can set MCX314As in the constant-speed continuous driving and enable IN2,1,0. Then, MCX314As will perform the sudden stop when IN1 is active.
When the automatic home search function of this IC is used, the Z-phase signal, home signal, and near home signal are assigned to nIN2, nIN1, and nIN0 respectively.

Except the parameter of the number of output pulse, the other four parameters for the fixed pulse drive must be set to execute the acceleration/deceleration continuous pulse driving.

2.2 Acceleration and Deceleration

Basically, driving pulses of each axis are output by a fixed pulse driving command or a continuous pulse driving command of the + direction or – direction. These types of driving can be performed with a speed curve of constant speed, linear acceleration, non-symmetrical linear acceleration, S-curve acceleration/deceleration, or non-symmetrical S-curve acceleration/deceleration according to the mode that is set or the operation parameter value.

2.2.1 Constant Speed Driving

When the drive speed set in MCX314As is lower than the initial speed (or a speed higher than the drive speed is set as the initial speed), the acceleration / deceleration will not be performed, instead, a constant speed driving starts.
If the user wants to perform the sudden stop when the home sensor or encoder Z-phase signal is active, it is better not to perform the acceleration / deceleration driving, but the low-speed constant driving from the beginning.
For processing constant speed driving, the following parameters will be preset accordingly.

Parameter name	Symbol	Comment
Range	R	
Initial Speed	SV	Set a value higher than the drive speed (V).
Drive Speed	V	
Number of Output Pulse	P	Not required for continuous pulse driving.

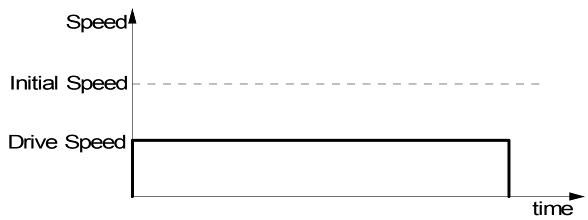


Fig. 2.8 Constant Speed Driving

■ Example for Parameter Setting of Constant Speed

The constant speed is set 980 pps as shown in the right Figure.

Range R = 8,000,000 ; Multiple(M) = 1
Initial Speed SV=980 ; Initial Speed ≥ Drive Speed
; Should be less than initial speed
Drive Speed V=980
Number of output pulses P=2,450

Please refer each parameter in Chapter 6.

2.2.2 Trapezoidal Driving [Symmetrical]

In linear acceleration driving, the drive speed accelerates in a primary linear form with the specified acceleration slope from the initial speed at the start of driving. When the acceleration and the deceleration are the same (symmetrical trapezoid) in fixed pulse driving, the pulses utilized at acceleration are counted. When the remaining number of output pulses becomes less than the number of acceleration pulses, deceleration starts. Deceleration continues in the primary line with the same slope as that of acceleration until the speed reaches the initial speed and driving stops, at completion of the output of all the pulses (automatic deceleration).

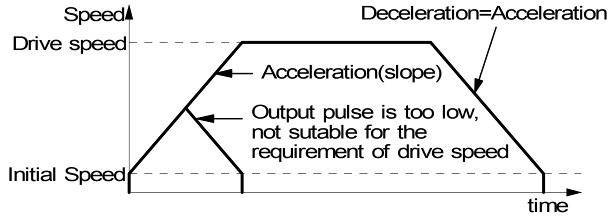


Fig. 2.9 Trapezoidal Driving (Symmetrical)

When the decelerating stop command is performed during the acceleration, or when the pulse numbers of the fixed pulse drive do not reach the designated drive speed, the driving will be decelerating during acceleration, as show in Fig. 2.9. By setting a triangle prevention mode, such triangle form can be transformed to a trapezoid form even if the number of output pulses low. See the section of triangle prevention of fixed pulse driving.

To perform symmetrical linear acceleration driving, bits D2 to 0 of the WR3 register must be set as follows.

Mode setting bit	Symbol	Setting value
WR3/D0	MANLD	0
WR3/D1	DSNDE	0
WR3/D2	SACC	0

See 4.6 for details of the WR3 register.

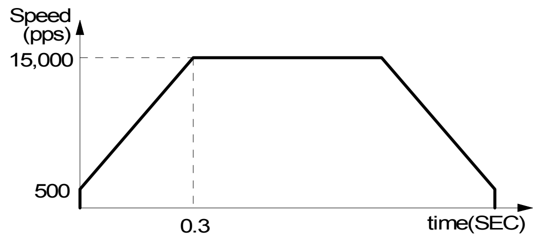
The following parameters must be set.

Parameter name	Symbol	Comment
Range	R	
Acceleration	A	This value is applied to deceleration also.
Initial Speed	SV	
Drive Speed	V	
Number of Output Pulse	P	Not required for continuous pulse driving.

■ The example of setting Trapezoidal Driving

Shown in the figure right hand side, acceleration is form the initial speed 500 PPS to 15,000 PPS in 0.3 sec.

Range R = 4,000,000	; Multiple= 2
Acceleration A=193	; (15,000-500)/0.3 =48,333
	; 48,333/125/M = 193
Initial Speed SV = 250	; 500/M = 250
Drive Speed V = 7,500	; 15,000/M = 7,500



Please refer Chapter 6.

■ Triangle Prevention of Fixed Pulse Driving

The triangle prevention function prevents a triangle form in linear acceleration fixed pulse driving even if the number of output pulses is low. When the number of pulses that were utilized at acceleration and deceleration exceeds 1/2 of the total number of output pulses during acceleration, this IC stops acceleration and enters a constant speed mode.

The triangle prevention function is disabled at resetting. The function can be enabled by setting the WR6/D3 (AVTRI) bit of the extension mode setting command (60h) to 1. See Section 6.16 for details of the extension mode setting command.

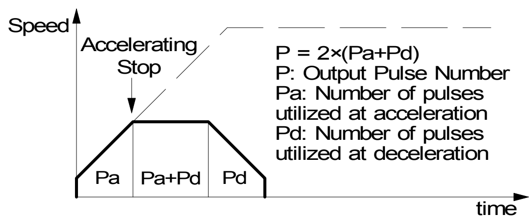


Fig. 2.10 Triangle Prevention of Linear Acceleration Driving

2.2.3 Non-Symmetrical Trapezoidal Acceleration

When an object is to be moved using stacking equipment, the acceleration and the deceleration of vertical transfer need to be changed since a gravity acceleration is applied to the object.

This IC can perform automatic deceleration in fixed pulse driving in non-symmetrical linear acceleration where the acceleration and the deceleration are different. It is not necessary to set a manual deceleration point by calculation in advance. Fig. 2.11 shows the case where the deceleration is greater than the acceleration and Fig. 2.12 shows the case where the acceleration is greater than the deceleration. In such non-symmetrical linear acceleration also, the deceleration start point is calculated within the IC based on the number of output pulses P and each rate parameter.

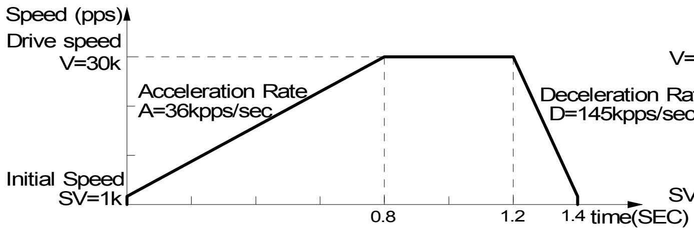


Fig.2.11 Non-Symmetrical Linear Acceleration Driving (acceleration<deceleration)

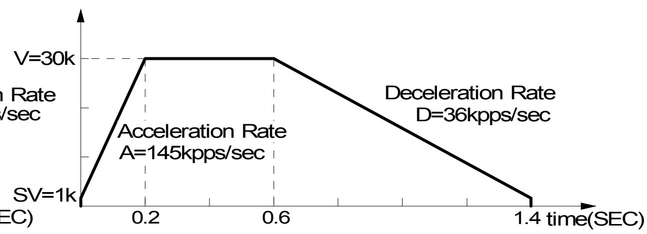


Fig.2.12 Non-Symmetrical Linear Acceleration Driving (acceleration>deceleration)

To perform automatic deceleration for fixed pulse driving of non-symmetrical linear acceleration, bits D1 to 0 of the WR3 register must be set as follows.

Mode setting bit	Symbol	Setting value	Comment
WR3/D0	MANLD	0	Automatic deceleration
WR3/D1	DSNDE	1	The deceleration setting value is applied at deceleration.
WR3/D2	SACC	0	Linear acceleration

The following parameters must be set.

Parameter name	Symbol	Comment
Range	R	
Acceleration	A	
Deceleration	D	
Initial speed	SV	
Drive speed	V	
Number of output pulses	P	Not required at continuous pulse driving

[Note]

• In the case of acceleration > deceleration (Fig. 2.12), the following condition is applied to the ratio of the acceleration and the deceleration.

$$D > A \times \frac{V}{4\times10^6}$$

D: Deceleration rate (pps/sec)
A: Acceleration rate (pps/sec)
V: Drive Speed (pps)

Where CLK=16MHz

For instance, if the driving speed V = 100kps, deceleration D must be greater than 1/40 of acceleration A. The value must not be less than 1/40 of the acceleration.

• If acceleration > deceleration (Fig. 2.12), the greater the ratio of acceleration A to deceleration D becomes, the greater the number of creep pulses becomes (about maximum of 10 pulse when A/D=10 times). When creep pulses cause a problem, solve the problem by ①increasing the initial speed or ②setting a minus value to the acceleration counter offset.

■ Example of setting parameters

See below for the parameter setting of fixed pulse driving in non-symmetrical linear acceleration (acceleration < deceleration).

```
WR3 ← 0002h                      ; Mode setting of the WR3 register

Range R=800000                   ; Multiple=10
Acceleration A=29                ; (30000-1000)/0.8=36250PPS/SEC
                                 ; (36250/125)/10=29
Deceleration D=116               ; (30000-1000)/0.2=145000PPS/SEC
                                 ; (145000/125)/10=116
Initial speed SV=100             ; 1000/10=100
Drive speed V=3000               ; 30000/10=3000
Number of output pulses P=27500  ;
```

2.2.4 S-curve Acceleration/Deceleration Driving

This IC creates an S curve by increasing/reducing acceleration/decelerations in a primary line at acceleration and deceleration of drive speed.
Figure 2.13 shows the operation of S-curve acceleration/deceleration driving where the acceleration and the deceleration are symmetrical. When driving starts, the acceleration increases on a straight line at the specified jerk (K). In this case, the speed data forms a secondary parabolic curve (section a). If the difference between the specified drive speed (V) and the current speed becomes less than the speed that was utilized at the increase of acceleration, the acceleration starts to decrease towards 0. The decrease ratio is the same as the increase ratio and the acceleration decreases in a linear form of the specified jerk (K). In this case, the rate curve forms a parabola of reverse direction (section b).
The speed reaches the specified drive speed (V) or the acceleration reaches 0, the speed is maintained (section c).

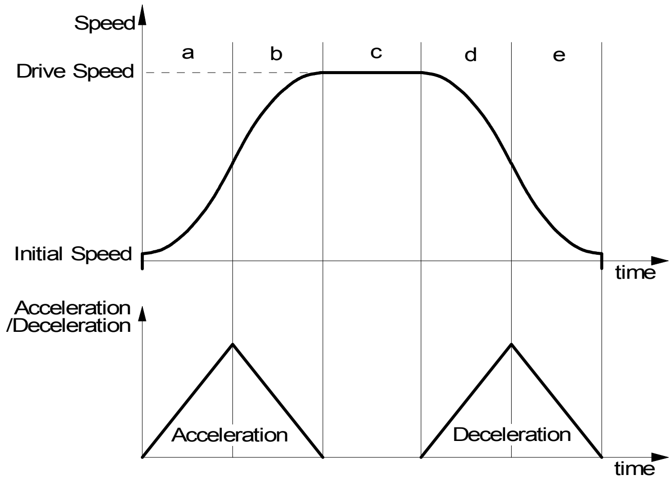


Fig. 2.13 Symmetrical S-Curve Acceleration/Deceleration Driving

In fixed pulse driving of S-curve acceleration/deceleration where acceleration and deceleration are symmetrical, deceleration starts when the number of remaining output pulses becomes less than the number of pulses that were utilized. At the deceleration also, the speed forms an S curve by increasing/decreasing the deceleration in a primary linear form (sections d and e).
The same operation is performed in acceleration/deceleration where the drive speed is changed during continuous pulse driving.

To perform symmetrical S-curve acceleration/deceleration driving, set bits D2, D1, and D0 of the nW3 register as follows.

Mode setting bit	Symbol	Setting value	Comment
WR3/D0	MANLD	0	Automatic deceleration
WR3/D1	DSNDE	0	The acceleration setting value and jerk setting value are used at deceleration.
WR3/D2	SACC	1	S-curve acceleration/deceleration

The following parameters must be set.

Parameter name	Symbol	Comment
Range	R	
Jerk	K	
Acceleration	A	Always set the maximum value, 8000.*1
Initial Speed	SV	
Drive Speed	V	
Number of Output Pulse	P	Not required for continuous pulse driving

*1 : When a low acceleration is set, the acceleration/deceleration does not increase above the set value (A) (functions as a limiter) in acceleration increase or deceleration increase in S-curve acceleration and a straight line appears on the speed curve.

■The Prevention of Triangle Driving Profile

For fixed pulse driving of S-curve acceleration/deceleration where acceleration and deceleration are symmetrical, the following method is applied to maintain a smooth speed curve when the output pulses do not reach the pulses required for accelerating to the drive speed or deceleration stop is applied during acceleration.

If the initial speed is 0, and if the rate of acceleration is a, then the speed at time t in acceleration region can be described as following.

$$v(t) = at^2$$

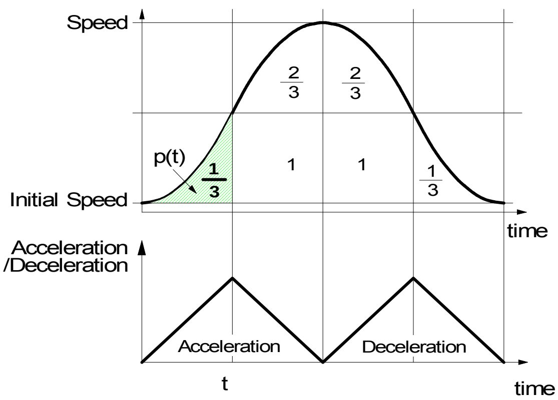


Fig. 2.14 The rule of 1/12 of Parabolic Acceleration/Deceleration

Therefore, the total the number of pulse p(t) from time 0 to t is the integrated of speed.

$$p(t) = 1/3 \times at^3$$

The total output pulse is

$$(1/3+2/3+1+2/3+1+1/3) \times at^3 = 4\ at^3$$

so

$$p(t) = 1/12\ \text{(total pulse output)}$$

Therefore, when the output pulse in acceleration of S-curve is more than 1/12 of total output pulse, MCX314As will stop increasing acceleration and start to decrease the acceleration value.

■ The Decelerating Stop for Preventing the Triangle Driving Profile in S-curve Driving

When the decelerating stop is commanded, or when the external signals IN3~IN0 are active during the S curve acceleration / deceleration driving, the acceleration is decreasing, then the deceleration starts when the acceleration reaches 0.

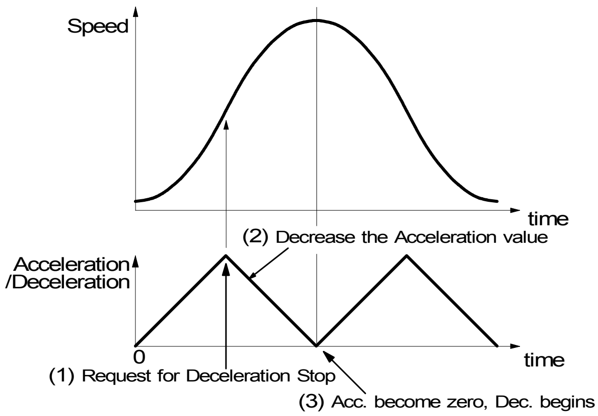


Fig. 2.15 The rule of 1/12 of Parabolic Acceleration/Deceleration

■ Constraints for S-curve Acceleration / Deceleration Driving

a. The drive speed cannot be changed during the fixed pulse S-curve acceleration / deceleration driving.

b. When the fixed-pulse S-curve acceleration / deceleration driving is performed, the change of the numbers of output pulse during the deceleration will not result a normal S-curve driving profile.

c. In case of executing circular interpolation, bit pattern interpolation and continuous interpolation, S-curve acceleration/deceleration cannot be executed normally.

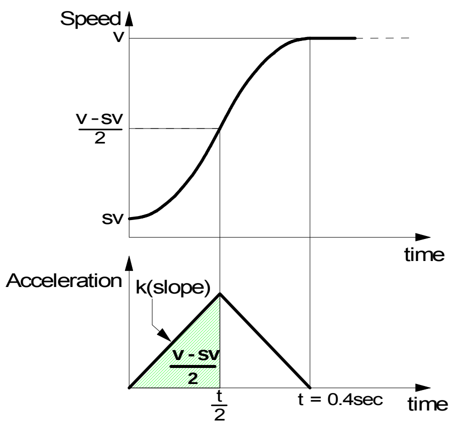
d. If an extremely low value is set as the initial speed for fixed pulse driving of S-curve acceleration/deceleration, premature termination (output of the specified driving pulses is completed and terminated before the speed reaches the initial speed) or creep (output of specified driving pulses is not completed even if the speed reaches the initial speed and the remaining driving pulses are output at the initial speed) may occur.

■ Example of Parameter Setting (Symmetrical S-Curve Acceleration/Deceleration)

As shown in the diagram, in this example, S-curve acceleration is applied to reach the drive speed to 40KPPS from the initial speed of 100PPS in 0.4 seconds.

At acceleration, increase the speed in a straight line according to the specified jerk (k). The integral value (area indicated by diagonal lines) is the speed increase.

Find the jerk (k) to produce the result where the speed reaches a half ((v−sv)/2) of the drive speed (v) from the initial speed (sv) within a half (5/2) of the acceleration time (t=0.4sec). Use the following expression to find a value of K since the area indicated by diagonal lines, which uses k in the left-hand member, is equal to the right-hand member.



$$\frac{k}{2}\left(\frac{t}{2}\right)^2 = \frac{v-sv}{2}$$

$$k = \frac{4(v-sv)}{t^2}$$

$$k = \frac{4(40000-100)}{0.4^2} = 997{,}500\ \text{pps/sec}^2$$

Unit
Jerk k: pps/sec^2
Drive Speed v: pps
Initial Speed sv: pps
Acceleration time t: sec

Therefore, the following parameters must be set in this IC.

WR3 ← 0004h	; Mode setting of the WR3 register
Range R = 800000	; Multiple=10
Jerk K = 627	; 62.5×10^6/k ×Multiple = 62.5×10^6/997500 ×10
Acceleration A = 8000	; Fixed to the maximum value
Initial Speed SV = 10	; 100/10=10
Drive Speed V = 4000	; 40000/10=4000
Number of output pulse P = 25000	; Set when fixed pulse driving is performed
Acceleration counter offset AO = 0	

2.2.5 Non-symmetrical S-Curve Acceleration/Deceleration

As shown in Fig. 2.16, a non-symmetrical S curve can be created by setting an jerk and a deceleration increasing rate individually in S-curve acceleration/deceleration driving. However, for fixed pulse driving, a deceleration point must be specified manually, since automatic deceleration is prohibited. Since a triangle form prevention function (1/12 rule) is not supported either, a drive speed must be set according to the acceleration/
deceleration increasing rate and the number of output pulses.

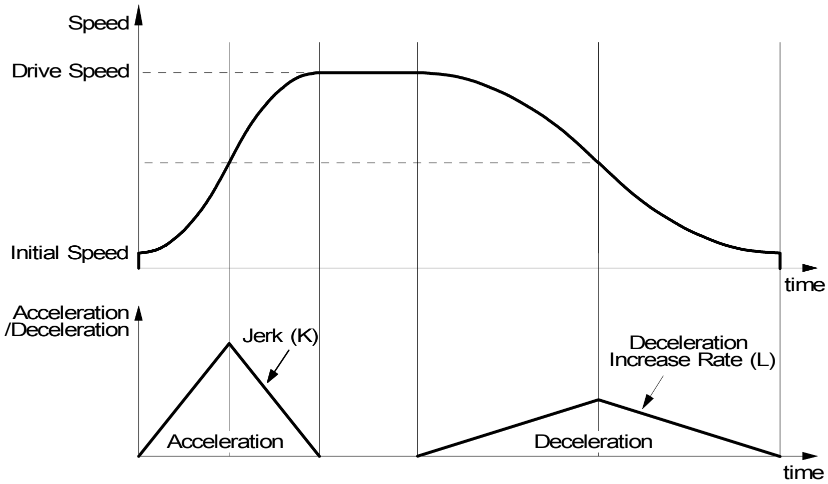


Fig. 2.16 Non-symmetrical S-curve acceleration/deceleration drive

To perform non-symmetrical S-curve acceleration/deceleration driving, set the D2, D1, and D0 bits of the nWR3 register as follows.

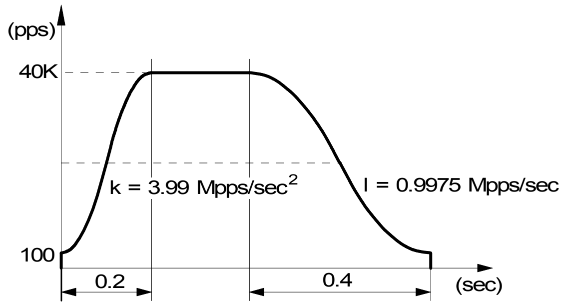
Mode setting bit	Symbol	Setting value	Comment
WR3/D0	MANLD	1	Manual deceleration
WR3/D1	DSNDE	1	The deceleration increasing rate setting value is used at deceleration.
WR3/D2	SACC	1	S-curve acceleration/deceleration

The following parameters must also be set.

Parameter name	Symbol	Comment
Range	R	
Jerk	K	
Deceleration increasing rate	L	
Acceleration	A	The maximum value, 8000, must be set.
Deceleration	D	The maximum value, 8000, must be set.
Initial speed	SV	
Drive speed	V	
Number of Output pulses	P	Not required for continuous pulse driving
Manual deceleration point	DP	Set a value produced by subtracting the number of pulses that were utilized at deceleration from the number of output pulses (P). Not required for continuous pulse driving

■ Example of Parameter Setting (Non-symmetrical S-curve Acceleration/Deceleration)

As shown in the diagram, at acceleration, the drive speed (v) is accelerated up to 40KPPS from the initial speed (sv) of 100PPS in 0.2 seconds. At deceleration, the drive speed (v) is decelerated from 40KPPS to the initial speed (sv) of 100PPS in 0.4 seconds. Using the symmetrical S-curve acceleration/deceleration parameter setting expression that is shown in the previous example, find a jerk and a deceleration increasing rate.



$$\text{Jerk} \quad k = \frac{4(40000 - 100)}{0.2^2} = 3.99 \text{ Mpps/sec}^2$$

$$\text{Deceleration Increase Rate} \quad l = \frac{4(40000 - 100)}{0.4^2} = 0.9975 \text{ Mpps/sec}^2$$

The parameter values that are set in the IC are as follows.

$$\text{Jerk} \quad K = \frac{62.5 \times 10^6}{k} \times \text{Multiple} = \frac{62.5 \times 10^6}{3.99 \times 10^6} \times 10 = 157$$

$$\text{Deceleration Increase Rate} \quad L = \frac{62.5 \times 10^6}{l} \times \text{Multiple} = \frac{62.5 \times 10^6}{0.9975 \times 10^6} \times 10 = 627$$

Since automatic deceleration of non-symmetric S-curve acceleration/deceleration is not supported, set a deceleration point (DP) manually. Since a value produced by subtracting the number of pulses that were utilized (Pd) at deceleration from the number of output pulses (P) is set as the manual deceleration point, initially find the number of pulses that were utilized (Pd) at deceleration.

$$\text{Pulses Utilized at Deceleration} \quad Pd = (v + sv)\sqrt{\frac{v - sv}{l}} = (40000 + 100)\sqrt{\frac{40000 - 100}{0.9975 \times 10^6}} = 8020$$

If the number of output pulses is 20000, the manual deceleration point (DP) will be as follows.

$$\text{Manual deceleration point} \quad DP = P - Pd = 20000 - 8020 = 11980$$

Therefore, the parameter settings for this IC will be as follows.

WR3 ← 0007h	; Mode setting of the WR3 register
Range R=800000	; Multiple=10
Jerk K=157	; $62.5\times10^6/k \times$ Multiple = $62.5\times10^6/3.99\times10^6 \times10$
Deceleration increasing rate L=627	; $62.5\times10^6/L \times$ Multiple = $62.5\times10^6/0.9975\times10^6 \times10$
Acceleration A=8000	; Fixed to the maximum value
Deceleration D=8000	; Fixed to the maximum value
Initial Speed SV=10	; 100/10=10
Drive Speed V=4000	; 40000/10=4000
Number of Output Pulse P=20000	;
Manual deceleration point DP=11980	;
Acceleration counter offset AO=0	

[Note] The above expression used for calculating the number of pulses that were utilized at deceleration is an ideal expression. In the actual IC operation, creep or premature termination occurs depending on the parameter values that are set.

2.2.6 Pulse Width and Speed Accuracy

■ Duty Ratio of Drive Pulse

The period time of + /- direction pulse driving of each axis is decided by system clock SCLK. The tolerance is within ±1SCLK (For CLK=16MHz, the tolerance is ±125nSEC). Basically, the duty ratio of each pulse is 50% as show in Fig. 2.17. When the parameter setting is R=8,000,000 and V=1000 (Multiple=1, V=1000PPS), the driving pulse is 500uSEC on its Hi level and 500uSEC on its Low level and the period is 1mSEC.

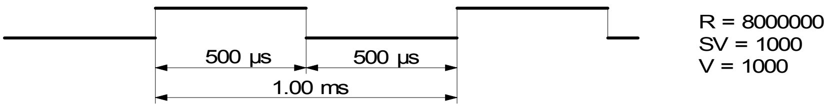


Fig. 2.17 High/Low Level Width of Driving Pulse Output (V=1000PPS)

However, during the acceleration / deceleration driving, the Low level pulse length is shorter than that of Hi level pulse during the acceleration; the Low level pulse is longer than that of Hi level pulse during the deceleration. See Fig. 2.18.

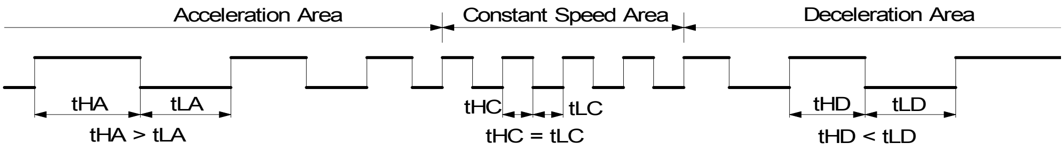


Fig. 2.18 Comparison of Drive Pulse Length in Acceleration/Deceleration

■ The Accuracy of Drive Speed

The clock (SCLK) running in MCX314As is half of external input clock (CLK). If CLK input is standard 16MHz, SCLK will be 8MHz. Therefore, the user had better driving the pulse speed in an exact multiple of SCLK period (125nSEC). Otherwise, the driving pulse will not very stable. The frequency (speed) of driving pulse of MCX314As can be, there are all exact the multiple of 125nSEC. For instance, the only frequencies that can be output are, double：4.000 MHz, triple：2.667 MHz, quadruple：2.000 MHz, five times：1.600 MHz, six times：1.333 MHz, seven times：1.143 MHz, eight times：1.000 MHz, nine times：889 KHz, 10 times：800 KHz, ……. Any fractional frequencies cannot be output. It is not very stable to set any desired drive speed. However, MCX314As can make any drive speed in using the following method.

For instance, in the case of the range setting value:R=80,000 (magnification = 100) and drive speed setting value:V=4900, the speed of driving pulses of 4900×100 = 490 KPPS is output. Since this period is not a multiple integer of the SCLK period, pulses of 490KPPS cannot be output under a uniform frequency. Therefore, as shown in Fig. 2.19, MCX314As combines 16 times and 17 times of SCLK period in a rate of 674:326 to generate an average 490KPPS.

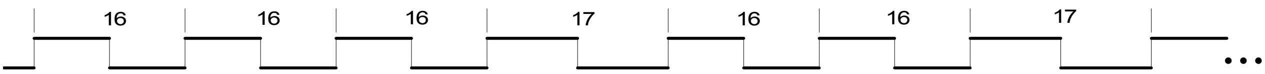


Fig. 2.19 The Driving Pulse of 490KPPS

According to this method, MCX314As can generate a constant speed driving pulse in a very high accuracy. In general, the higher of the drive speed, the lower of the accuracy. But for MCX314As, it still can maintain relative accuracy when the drive speed is high. Actually, the accuracy of driving pulse is still within ±0.1%.

Using oscilloscope for observing the driving pulse, we can find the jitter about 1SCLK (125nSEC). This is no matter when putting the driving to a motor because the jitter will be absorbed by the inertia of motor system.

2.3 Position Control

Fig 2.20 is 1-axis position control block diagram. For each axis, there are two 32 bit up-and-down counters for counting present positions and two comparison registers for comparing the present positions.

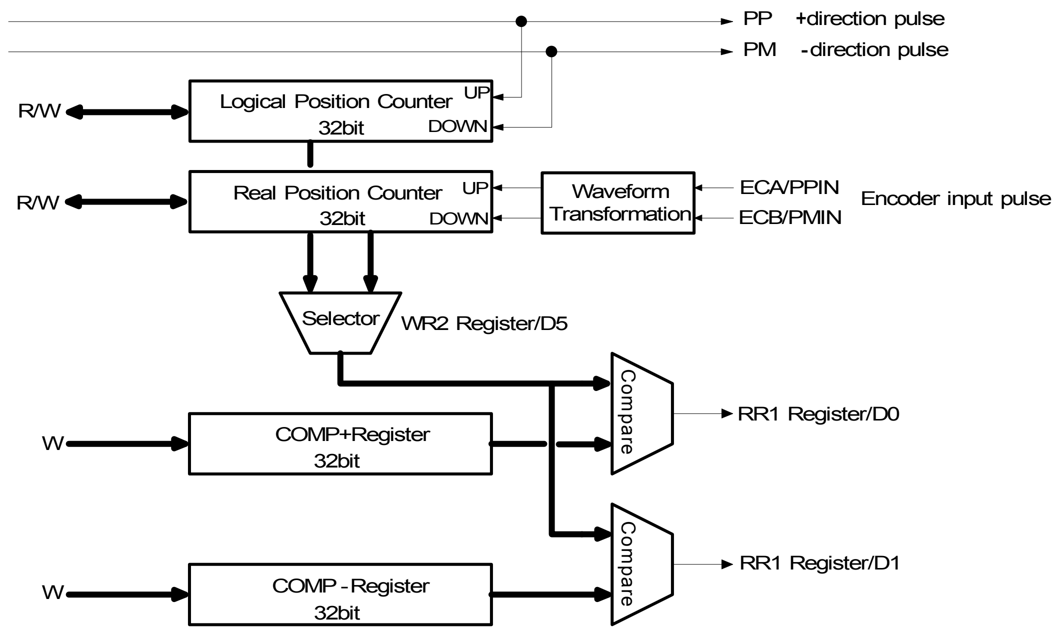


Fig. 2.20 Position Control Block Diagram

2.3.1 Logic Position Counter and Real position Counter

As shown in Fig. 2.20, the logic position counter is counting the driving pulses in MCX314As. When one + direction plus is outputting, the counter will count-up 1; when one - direction pulse is outputting, the counter will count-down 1. The real position counter will count input pulse numbers from external encoder. The type of input pulse can be either A/B quadrature pulse type or Up / Down pulse (CW/CCW) type (See Chapter 2.9.3).

Host CPU can read or write these two counters any time. The counters are signed 32 bits, and the counting range is between -2,147,483,648 ~ + 2,147,483,647. The negative is in 2's complement format. The counter value is random while resetting.

2.3.2 Compare Register and Software Limit

Each axis has, as shown in Fig. 2.20, two 32-bit registers which can compare the logical positions with the real positions. The logical position and real position counters are selected by bit D5 (CMPSL) of register WR2.

The main function of COMP+ Register is to check out the upper limit of logical / real position counter. When the value in the logical / real position counters are larger than that of COMP+ Register, bit D0 (CMP+) of register RR1 will become 1. On the other hand, COMP- Register is used for the lower limit of logical / real position counter. When the value of logical / real position counter become smaller than hat of COMP+ Register, bit D1 (CMP-) of register RR1 will become 1. Fig. 2.21 is an example for COMP+ = 10000, COMP- = -10000, COMP+ and COMP- registers can be used as software +/− limit.

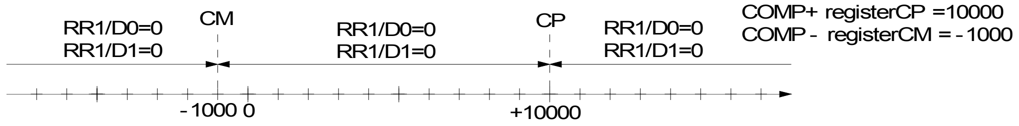


Fig. 2.21 Example of COMP+/ - Register Setting

When D0 and D1bits of register WR2 are set to 1, it enables the software limit. In driving, if the value of logical / real counter is larger than COMP+, the decelerating stop will be performed, and D0 (SLMT+) of register RR2 will change to 1. If the value of logical / actual counter is smaller than that of COMP+, the D0 bit of register RR2 will change to 0 automatically.

Host CPU can write the COMP+ and COMP− registers any time. However, when MCX314As is reset, the register values are random.

2.3.3 Position Counter Variable Ring

A logical position counter and a real position counter are 32-bit up/down ring counters. Therefore, normally, when the counter value is incremented in the + direction from FFFFFFFFh, which is the maximum value of the 32-bit length, the value is reset to the value 0. When the counter value is decremented in the − direction from the value 0, the value is reset to FFFFFFFFh. The variable ring function enables the setting of any value as the maximum value. This function is useful for managing the position of the axis in circular motions that return to the home position after one rotation, rather than linear motions.
To enable the variable ring function, set the D4 (VRING) bit of the WR6 register of the extension mode setting command (60h) to 1 and set the maximum value of the logical position counter in the COMP+ register and the maximum value of the real position counter in the COMP− register.

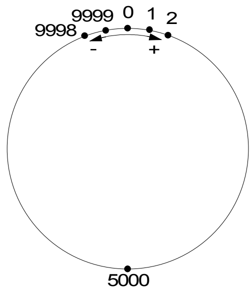


Fig. 2.22 Operation of Position Counter Ring Maximum Value 9999

For instance, set as follows for a rotation axis that rotates one cycle with 10,000 pulses.

① To enable the variable ring function, set 1 in the D4 bit of the WR6 register of the extension mode setting command (60h).
② Set 9,999 (270Fh) in the COMP+ register as the maximum value of the logical position counter.
③ Set 9,999 (270Fh) in the COMP− register when using a real position counter also.

The count operation will be as follows.

Increment in the + direction ···→9998→9999→0→1→···
Decrement in the - direction ···→1→0→9999→9998→···

[Notes]

- The variable ring function enable/disable is set for each axis, however, a logical position counter and a real position counter cannot be enabled/disabled individually.
- If a variable ring function is enabled, a software limit function cannot be used.

2.3.4 Clearing a Real Position Counter Using an External Signal

This function clears a real position counter at rising of the Z-phase active level when Z-phase search is applied in home search.
Normally, home search is performed by assigning a near home signal, a home signal, and an encoder Z-phase signal to nIN0 to nIN2 signals and executing continuous pulse driving. When the specified signal is activated, driving will stop and then the logical position/real position counters are cleared by the CPU. This function is useful for solving the problem of Z-phase detection position slippage that occurs due to a delay of the servo system or the mechanical system even if a low Z-phase search drive speed is set.

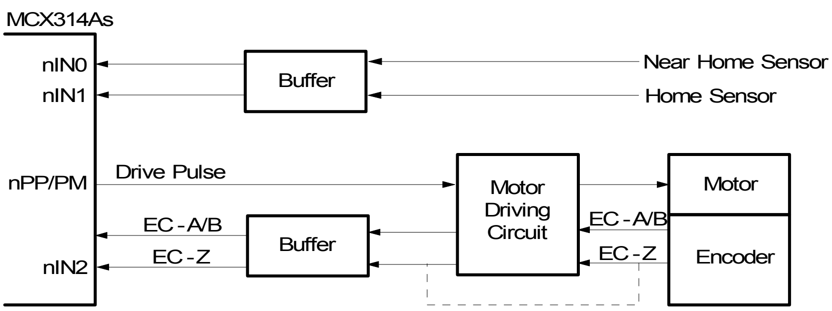


Fig. 2.23 Example of Signal Connection for Clearing The Real Position Counter by The IN2 Signal

To clear a real position counter with a Z-phase signal in encoder Z-phase search, assign the Z-phase signal to nIN2 signal as shown Fig. 2.23. See below for the procedure for setting a mode or a command for Z-phase search accompanied by clearing of the real position counter.

① Set a range and an initial speed.
② Set a Z-phase search drive speed.
If the value set for the drive speed is lower than the initial speed, acceleration/deceleration driving is not performed. If a Z-phase is detected, the driving pulse stops immediately.

③ Validate the IN2 signal and set an active level.
WR1/D5(IN2-E) : 1, D4(IN2-L) : 0(Low active) 1(Hi active)
④ Enable the clearing of the real position counter using the IN2 signal.
Set WR6/D0(EPCLR) to 1 and issue an extension mode setting command (60h). [Note] Other bits of the extension mode command are also set.
⑤ Issue the + direction or - direction continuous pulse driving command.

As a result of the operations described above, driving starts in the specified direction as shown in Fig. 2.24. When the Z-phase signal reaches an active level, the driving pulses stop and the real position counter is cleared at the rising of the Z-phase signal active level.

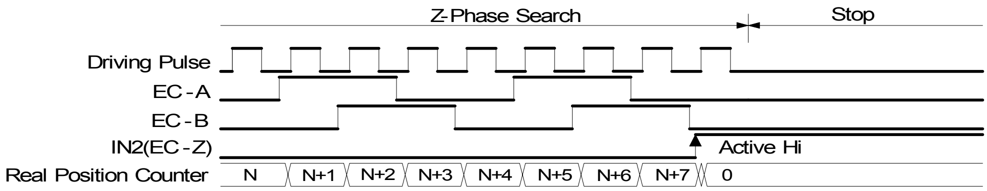


Fig. 2.24 Example of Operation of Clearing The Real Position Counter Using The IN2 Signal

[Notes]
- Only the nIN2 signal can clear the real position counter. The nIN3, nIN1, and nIN0 signals cannot clear the counter.
- When the input signal filter is invalid, an active level width of more than 4CLK cycles is necessary. When the input signal filter is valid, a time more than double the input signal delay time is necessary.
- It is recommended to perform Z-phase search from the one direction to enhance the position detection precision.
- When the nIN2 signal is already set to an active level at the issuing of the extension mode setting command by setting WR6/D0 (EPCLR) to 1, the real position counter is cleared even if the extension mode setting command is issued.

2.4 Interpolation

This 4-axis motion control IC can perform any 2 / 3 axes linear interpolation, any 2 axes circular interpolation and any 2 / 3 axes bit pattern interpolation.
Bits D0, D1 (ax 1), D2, D3 (ax 2) and D4, D5 (ax 3) of register WR5 can be pointed for performing the interpolation. In the process of interpolation driving, all the calculations will follow the main axis (ax1). So, the user has to set the parameters such as initial speed and drive speed of the main axis before performing the interpolation. During the linear interpolation, it is not necessary to set the main axis as "long axis".
After setting all of the parameters for interpolations, and writing the interpolation driving commands to command register WR0, the user can start the interpolation driving. During the interpolation driving, D8 (IDRV) of main status register RR0 will become 1 during the interpolation, and it will become 0 when the interpolation is finished. Also, during the interpolation driving, the bit n-DRV of the interpolating axis will become 1.
The maximum drive speed is 4MPPS for linear, circular or bit pattern interpolation. For continuous interpolation, the maximum drive speed is 2MPPS.

Over Limit Error of Interpolation

When the hardware limit or the software limit of each axis is active during the interpolation driving, the interpolation will stop. It the stop is occurred by errors, RR0 (main status register) will confirm the error bit of the designated interpolating axis. PR0 will become 1, and RR2 (error register) of this axis will be read out.
[Note] In case of circular or bit patter interpolation, the " active" of hardware or software limit, in either + or - direction, will stop the interpolation driving.

In-position Signal for Servo Motor

During the interpolation driving, when the in-position signal (nINP0S) of each driving axis is active, and also when the interpolation is finished, the INP0S signal of the axis is stand-by at its active level, and D8 (I-DRV) of RR0 register returns to 0.

2.4.1 Linear Interpolation

Any 2 or 3 axes of the 4 axes can be set for linear interpolation. To execute the linear interpolation, the user can, according to the present point coordinates, set the finish point coordinates and the interpolation segment(s) for 2 or 3 axes.
Fig. 2.25 shows an example of axis interpolation where linear interpolation is performed from the current coordinates to the finish point coordinates.
For individual axis control, the command pulse number is unsigned, and it is controlled by + direction command or − direction command. For interpolation control, the command pulse number is signed.

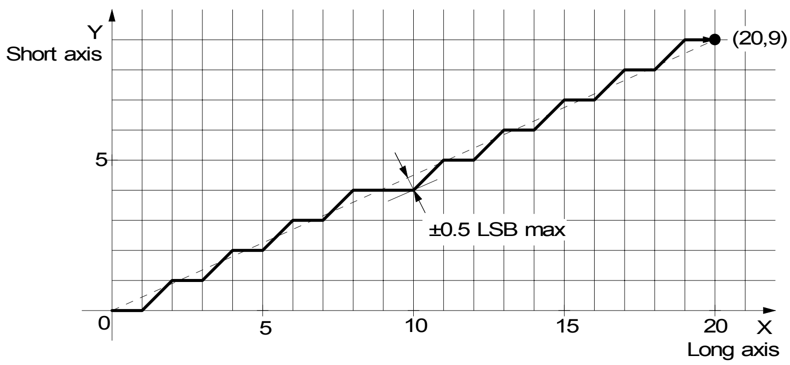


Fig. 2.25 The Position Accuracy for Linear Interpolation

The resolution of linear interpolation is within ±0.5 LSB, as shown in Fig. 2.25

As shown in Fig. 2.26, it is an example for pulse output of the linear interpolation driving. We define the longest distance movement in interpolation is the "long axis". And the other is "short axis". The long axis outputs an average pulse train. The driving pulse of the short axis depends on the long axis and the relationship of the two axes.

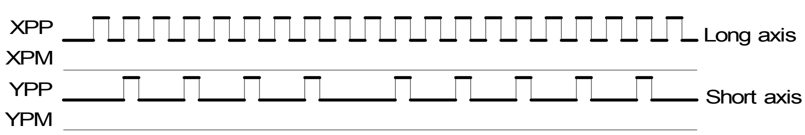


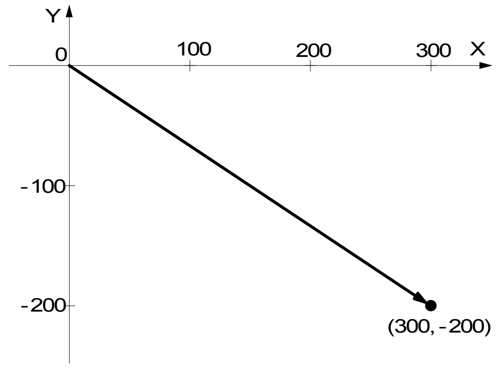
Fig. 2.26 The Example for Pulse Output at Finish Point (X=20, Y=9)

The range for each axis is a 32-bit signed counter, from -2,147,483,648 ~ + 2,147,483,647 (signed 32-bit-2LSB).

■ The example of linear interpolation for 2 axes

Executing linear interpolation drives in X and Y axes from the current position to the finish position (X: +300, Y: −200). The interpolation drive speed is constant: 1000PPS.

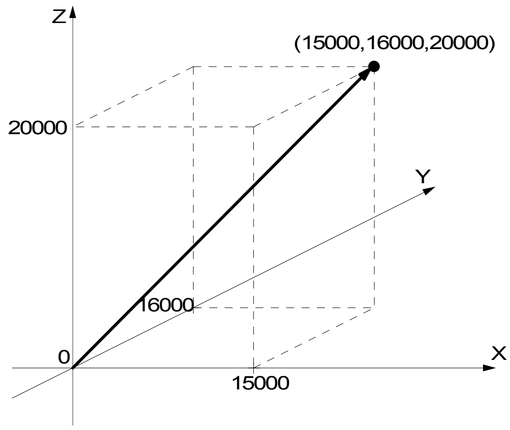
```
WR5 ← 0004h write      ; map ax1 to X axis, ax2 to Y axis

WR6 ← 1200h write      ; range: 8,000,000 (Multiple = 1)
WR7 ← 007Ah write
WR0 ← 0100h write

WR6 ← 03E8h write      ; initial speed :1,000PPS
WR0 ← 0104h write

WR6 ← 03E8h write      ; drive speed: 1,000PPS
WR0 ← 0105h write

WR6 ← 012Ch write      ; finish point of X axis: 300
WR7 ← 0000h write
WR0 ← 0106h write

WR6 ← FF38h write      ; finish point of Y axis: −200
WR7 ← 0000h write
WR0 ← 0206h write

WR0 ← 0030h write      ; linear interpolation driving for 2 axes enabling
```



■ The example of linear interpolation for 3 axes

Executing linear interpolation drive for X, Y and Z axes from the current position to the finish position (X: 15,000, Y: 16,000, Z: 20,000). The initial speed = 500PPS, acceleration / deceleration = 40,000PPS/SEC, drive speed = 5,000PPS.

```
WR5 ← 0024h write      ; define: ax1=X axis, ax2=Y axis, ax3= Z axis

WR6 ← 1200h write      ; range: 8,000,000 (Multiple = 1)
WR7 ← 007Ah write
WR0 ← 0100h write

WR6 ← 0140h write      ; accel./decel. speed: 40,000/SEC
WR0 ← 0102h write      ; 40,000 / 125 / 1 = 320=140h

WR6 ← 01F4h write      ; initial speed : 500PPS
WR0 ← 0104h write

WR6 ← 1388h write      ; drive speed : 5,000PPS
WR0 ← 0105h write

WR6 ← 3A98h write      ; finish point of X axis:15,000
WR7 ← 0000h write
WR0 ← 0106h write

WR6 ← 3E80h write      ; finish point of Y axis:; 16,000
WR7 ← 0000h write
WR0 ← 0206h write

WR6 ← 4E20h write      ; finish point of Z axis; 20,000
WR7 ← 0000h write
WR0 ← 0406h write

WR0 ← 003Bh write      ; deceleration enabling
WR0 ← 0031h write      ; linear interpolation driving for 3 axes enabling
```



2.4.2 Circular Interpolation

Any 2 axes of the 4 axes can be selected for circular interpolation.
The circular interpolation is starting from the current position (start point). After setting the center point of circular, the finish position and the CW or CCW direction, the user can start the circular interpolation.
Note: The coordinates setting value is the relative value of the start point coordinates.

In Fig. 2.27, it explains the definition of CW and CCW circular interpolations. The CW circular interpolation is starting from the start point to the finish position with a clockwise direction; the CCW circular interpolation is with a counter-clockwise direction. When the finish point is set to (0, 0), a circle will come out.

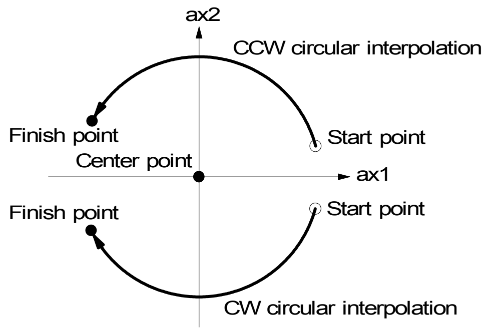


Fig. 2.27 CW / CCW Circular Interpolation

In Fig. 2.28, it explains the long axis and the short axis. First, we define 8 quadrants in the X-Y plane and put the numbers 0~7 to each quadrant. We find the absolute value of ax1 is always larger than that of ax2 in quadrants 0, 3, 4 and 7, so we call ax1 is the long axis (ax2 is the short axis) in these quadrants; in quadrants 1, 2, 5 and 6, ax2 is the long axis (ax1 is the short axis). The short axis will output pulses regularly, and the long axis will output pulses depending on the interpolation calculation.

In Fig. 2.29, it is an example to generate a circle with the center point (-11,0) and the finish point (0,0). Its radium is 11. In Fig. 2.30 shows the pulse output.

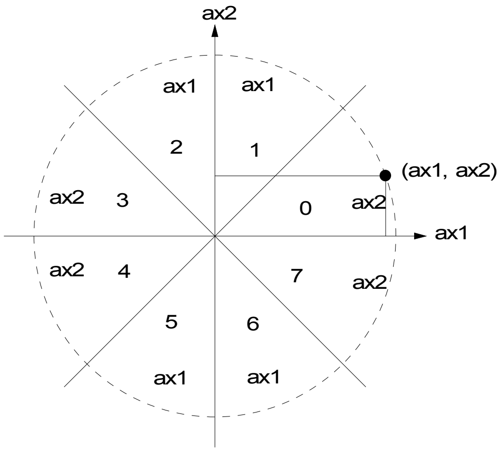


Fig. 2.28 The 0 - 7 Quadrants And Short

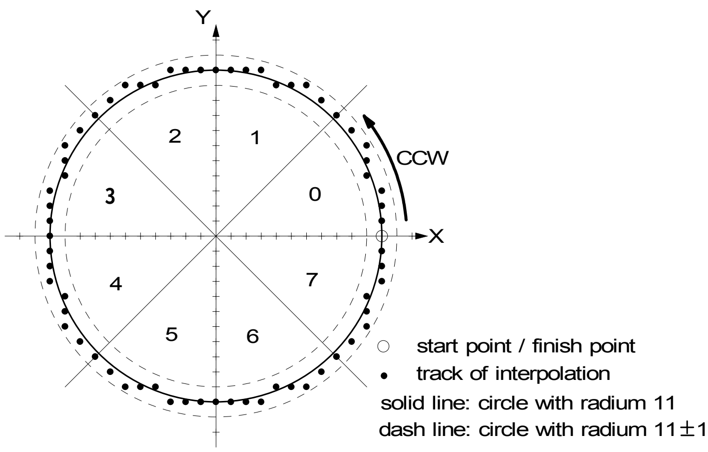


Fig. 2.29 The Example of Circular Interpolation

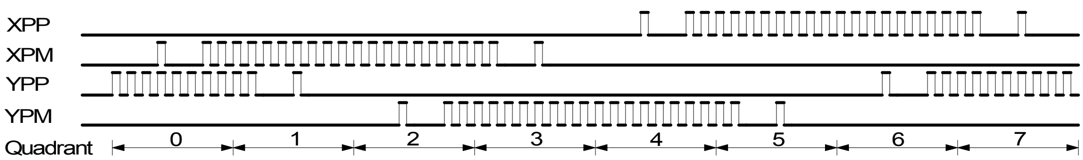


Fig. 2.23 The Example of Pulse Output in Circular Interpolation Driving

The specifiable range of coordinates of the center point and coordinates of the finish point are from −2,147,483,646 to +2,147,483,646 (signed 32-bit - 2LSB). The position tolerance for the specified circular curve is ±1 within the entire interpolation range. The interpolation speed is within the range from 1PPS to 4MPPS.

■ The Finish Point Checking of Circular Interpolation

In the circular interpolation, it assumes that the current position (start point) is (0,0). After the coordinates of the center point is set, the radium will be decided, and the circular tracking will start. The maximum error range of interpolation is with in ±1LSB. Because of the ±1LSB error range, the designated finish point may not on the circular track. When the value of finish point is same as that of short axis, this circular interpolation is finished.

Fig. 2.31 shows an example of CCW interpolation with the start point (0,0), center point (−200,500) and finish point (−702, 299). The finish point is in quadrant 4, and ax2 is the short axis in quadrant 4. So the interpolation is finished when the ax2 is 299.

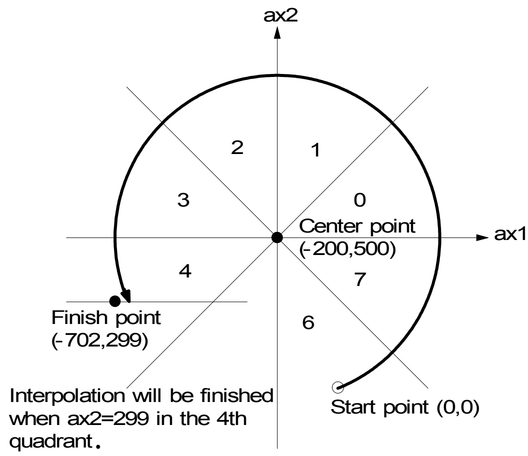
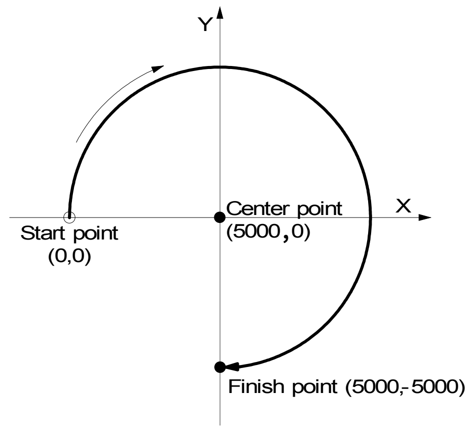


Fig. 2.31 Example of The Finish Point Checking of Circular Interpolation

■ The Example for CW Circular Interpolation

This CW circular interpolation starts from the current point (start point: 0, 0) to the finish point (X: 5000, Y: −5000); the center point is X: 5000, Y: 0. The interpolating speed is constant at 1000PPS in a constant vector speed driving.

```
WR5 ← 0104h write      ; define: ax1:X axis, ax2:Y axis, and with
                       constant linear speed

WR6 ← 0900h write      ; range : 4,000,000 (Multiple: 2)
WR7 ← 003Dh write
WR0 ← 0100h write

WR6 ← 4DC0h write      ; range of constant vector speed for 2 axes
WR7 ← 0056h write      ; 4,000,000 x 1.414 = 5,656,000
WR0 ← 0200h write

WR0 ← 01F4h write      ; initial speed : 500 x 2 = 1000PPS
WR0 ← 0104h write

WR6 ← 01F4h write      ; drive speed : 500 x 2 = 1000PPS
WR0 ← 0105h write

WR6 ← 1388h write      ; center point of X : 5,000
WR7 ← 0000h write
WR0 ← 0108h write

WR6 ← 0000h write      ; center point of Y : 0
WR7 ← 0000h write
WR0 ← 0208h write

WR6 ← 1388h write      ; finish point of X : 5,000
WR7 ← 0000h write
WR0 ← 0106h write

WR6 ← EC78h write      ; finish point of Y : −5,000
WR7 ← FFFFh write
WR0 ← 0206h write

WR0 ← 0032h write      ; CW circular interpolation enabling
```



2.4.3 The Bit Pattern Interpolation

This interpolation driving receives interpolation data that is created by upper-level CPU and transformed to bit patterns in a block of a predetermined size, and outputs interpolation pulses consecutively at the specified drive speed.

Every axis has 2 bit-data buffers for host CPU: one for + direction and the other for - direction. When performing the bit pattern interpolation, the host CPU will write the designated interpolation data, for 2 or 3 axes, into MCX314As.

If a bit in the bit pattern data from CPU is “1” , MCX314As will output a pulse at the time unit; if it is “0”, MCX314As will not output any pulse at the time unit.

For example, if the user want to generate the X-Y profile (see Fig. 2.32), the host CPU must write a set of pattern into those specific registers ---- XPP: the + direction register for X axis, XPM: the − direction register for X axis, YPP and YPM: the + and − directions registers. With in the time unit, MCX314As will check the registers once and decide to output a pulse or not depending on the bit pattern.

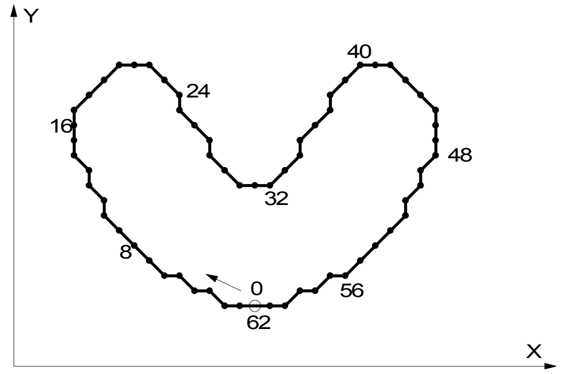


Fig. 2.32 Example for Bit Pattern Interpolation

```
←—— 56 ←——48 ←——40 ←——32 ←—— 24 ←——16 ←——8 ←—— 0
01000000 00000000 00011111 11011011 11110110 11111110 00000000 00000000 :XPP(X+direction)
01111111 11110101 00000000 00000000 00000000 00000000 00101011 11111111 :XPM(X-direction)
00000000 00000000 00000000 11111111 00000000 00001111 11111111 11010100 :YPP(Y+direction)
00001010 11111111 11111100 00000000 00111111 11000000 00000000 00000000 :YPM(Y-direction)
```

Fig. 2.33 shows the register configuration of the 1st axis and movements of bit data of bit pattern interpolation in this IC.

BP1P register and BP1M register are 16 bit-data buffers for bit pattern data form the host CPU. (IF the system uses 8-bit data bus, the host CPU has to write the data by low byte and high byte.) The + direction data should be written into PB1P, and the − direction data into PB1M. Once starting the bit pattern interpolation, the pulse outputting is in the order from D0.

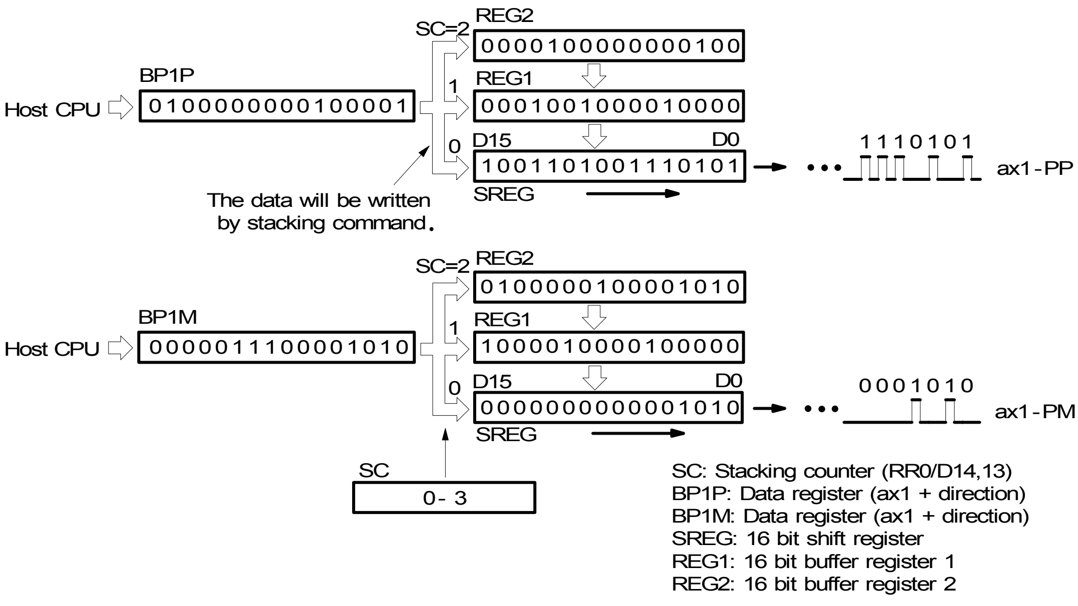


Fig. 2.33 Register Configuration and Movements of Bit Patterns of Bit Pattern Interpolation (for ax1)

Stacking counter (SC) is a 2-bit counter. Its value is between 0 and 3, which can be read from D14,13 of register RR0. SC will decide which register for the data from the host CPU. The initial value of SC is 0. So, when host CPU writes bit pattern data into BP1P or BP1M, the data will be stored in SREG, and then, SC will count up to 1, and the next data from the host CPU will be written into REG1. By this way, the REG2 becomes the register when SC=2. The host CPU is not able to write any bit pattern data into MCX314As when SC=3.

When the bit pattern interpolation pulse is outputting, D0 in SREG will be shifted output first, and then in the order of D1, D…

When all of SREGs have been shifted output, the data in REG1 will be shifted to SREG, the data in REG2 will be shifted to REG1, and the SC will count down to 2. Then, the host CPU is able to write a new data into MCX314As again.

In order to make MCX314As output the bit pattern data continuously, the host CPU should write the data into MCX314As before SC counts down to 0. MCX314As will output a interrupt requirement signal to host CPU when SC counts down from 2 to 1.

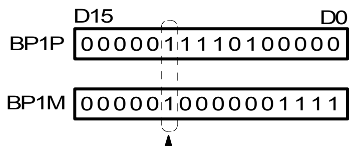
■ The limitation for the speed of bit pattern interpolation driving

The maximum pulse output speed of MCX314As is 4MHz in bit pattern interpolation mode. However, the maximum speed will depend on the data update rate of host CPU if the bit pattern data are more than 48bits.
For example of the X and Y axes bit pattern interpolation, if the host CPU needs 100μsec to update new 16-bit data for X and Y axes. The maximum speed is 16/100μSEC=160KPPS.

■ The ending of bit pattern interpolation

There are 2 ways can terminate the bit pattern interpolation.

① Write a ending code into buffer register of ax1.
The bit pattern interpolation mode will be finished, and stopped if the host CPU write “1” into both + and - directions buffer registers.



The Interpolation Stops Once The + And - Direction Are "1".

When the ending code is executed, the SC will become 0 automatically.

② The host CPU stops writing any command into MCX314As.
When SC=0, and when no other data is updated, MCX314As will stop outputting pulse. Then, the bit pattern interpolation is finished.

■ Utilizing the stop command to pause the interpolation

The interpolation driving will be paused if a sudden stop or decelerating stop command is written into the master axis (ax1) which is executing the bit pattern interpolation. MCX314As will continue the bit pattern interpolation if the host CPU enables the bit pattern interpolation again. If the host CPU wants to finish the interpolation after writing stop command, all of the interpolation bit data in MCX314As must be cleared in using BP register (3Dh).

■ Utilizing hardware limit to interrupt the interpolation

The interpolation driving will be terminated when any hardware limit of any axis is active.
And, if host CPU wants to finish the interpolation, all of the interpolation data in MCX314As must be cleared.

■ Writing the bit pattern data into the register in MCX314As

Either by 16-bit data bus or by 8-bit data bus, the address map of the command buffer for bit pattern interpolation data is show as follows:

The addresses map of register for 16-bit data bus in bit pattern interpolation

Address			Name of register	Content	The register with the same address
A2	A1	A0			
0	0	0			WR0
0	0	1			nWR1
0	1	0	BP1P	ax1 +direction data	nWR2
0	1	1	BP1M	ax1 -direction data	nWR3
1	0	0	BP2P	ax2 +direction data	WR4
1	0	1	BP2M	ax2 -direction data	WR5
1	1	0	BP3P (Note)	ax3 +direction data	WR6
1	1	1	BP3M (Note)	ax3 -direction data	WR7

Note: BP3P and BP3M share the same registers: WR6 and 7.

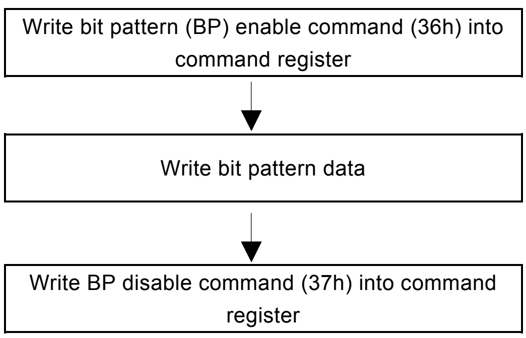
The addresses map of register for 8-bit data bus in bit pattern interpolation

Address				Name of register	Address				Name of register
A3	A2	A1	A0		A3	A2	A1	A0	
0	0	0	0		1	0	0	0	BP2PL
0	0	0	1		1	0	0	1	BP2PH
0	0	1	0		1	0	1	0	BP2ML
0	0	1	1		1	0	1	1	BP2MH
0	1	0	0	BP1PL	1	1	0	0	BP3PL
0	1	0	1	BP1PH	1	1	0	1	BP3PH
0	1	1	0	BP1ML	1	1	1	0	BP3ML
0	1	1	1	BP1MH	1	1	1	1	BP3MH

Note: BPmPL, BPmPH, BPmML, BPmMH represent the following bit groups (m is 1 ~ 3).

BPmPL : the low byte of BPmP (D7 ~ D0)
BPmPH : the high byte of BPmP (D15 ~ D8)
BPmML : the low byte of BPmM (D7 ~ D0)
BPmMH : the high byte of BPmM (D15 ~ D8)

For some addresses of bit pattern data registers are as same as nWR2 ~ nWR7, the host CPU can not write any data into the bit pattern data register since MCX314As has been reset. To write the bit pattern command, the host CPU should be with the following sequence.



[Note] If the host CPU doesn't disable the BP data register, the data in nWR2 ~ nWR5 registers cannot be assessed.

■ The example of bit pattern interpolation

The bit interpolation example is shown in Fig. 2.24. We set X axis as ax1, Y axis as ax2 and a constant speed: 1000PPS in a constant vector speed interpolation driving.

```
      WR5  ←  0104h write     ; Define ax1: X , ax2:Y

      WR6  ←  0900h write     ; setting the master axis speed parameter
      WR7  ←  003Dh write     ; range: 4,000,000 (multiple rate: 2)
      WR0  ←  0100h write

      WR6  ←  4DC0h write     ; range of constant line speed
      WR7  ←  0056h write     ; 4,000,000x1.414=5,656,000
      WR0  ←  0200h write

      WR6  ←  01F4h write     ; Initial speed: 500x2=1000PPS
      WR0  ←  0104h write

      WR6  ←  01F4h write     ; drive speed: 500x2=1000PPS
      WR0  ←  0105h write

      WR0  ←  0039h write     ; clear BP data

      WR0  ←  0036h write     ; enable to write into BP register

      BP1P ←  0000h write     ; command of bit 0~15th
                              ; X axis +direction
      BP1M ←  2BFFh write     ; X axis –direction
      BP2P ←  FFD4h write     ; Y axis +direction
      BP2M ←  0000h write     ; Y axis –direction
      WR0  ←  0038h write     ; BP data stacking, SC=1

      BP1P ←  F6FEhwrite      ; command of bit 16~31th
                              ; X axis +direction
      BP1M ←  0000h write     ; X axis –direction
      BP2P ←  000Fh write     ; Y axis +direction
      BP2M ←  3FC0h write     ; Y axis –direction
      WR0  ←  0038h write     ; BP data stacking, SC=2

      BP1P ←  1FDBh write     ; command of bit 32~47th
                              ; X axis +direction
      BP1M ←  0000h write     ; X axis –direction
      BP2P ←  00FFh write     ; Y axis +direction
      BP2M ←  FC00h write     ; Y axis –direction
      WR0  ←  0038h write     ; BP data stacking, SC=3

      WR0  ←  0034h write     ; enable 2 axis bit pattern
                              ; interpolation, because SC=3

 J1   RR0/D14,13 read         ; until the SC is less than 2
      If D14=D13=1 Jump to J1

      BP1P ←  4000h write     ; command of bit 48~62th
                              ; X axis +direction
      BP1M ←  7FF5h write     ; X axis –direction      (Note 1: If there is more BP data coming then repeat this process)
      BP2P ←  0000h write     ; Y axis +direction
      BP2M ←  0AFFh write     ; Y axis –direction
      WR0  ←  0038h write     ; BP data stacking, SC=3

      WR0  ←  0037h write     ; disable to write into BP register

 J2   RR0/D8 read             ; until ending interpolation drive
      If D8=1 Jump to J2
```

■ Executing bit pattern interpolation by interrupt

During the bit pattern interpolation, MCX314As will generate an interrupt request signal to the host CPU while SC changes the value from 2 to 1. To enable the interrupt, the host CPU must set D15 of register WR5 to 1. Then, INTN of MCX314As will go low once SC changes the value from 2 to 1. The host CPU will check the SC value, and write bit pattern command into the register. The interrupt signal will be released if the host CPU writes the SC stacking command (38h) into MCX314As.

The interrupt signal will be released when the clear command (3dh) is written into the command register. If the interrupt status is keeping on the Low level, it will return to high-Z level after MCX314As finishes the bit pattern interpolation.

2.4.4 Constant Vector Speed

MCX314As is with the constant vector speed control function which can control the resultant speed of two axes to keep the speed in constant.

Fig. 2.34 shows the profile of 2 axes interpolation driving. The vector speed reflects 1.414 times of the individual axis drive speed. So, we have to set the speed of 1.414 times to keep the vector speed for 2-axis driving.

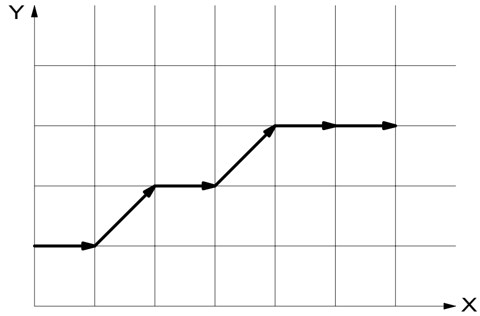


Fig. 2.34 Example of 2-Axis Interpolation

■ Constant Vector Speed Set-up

The user should first set the values of D9 and D8 of register WR5 to 0 and 1. Then, set the range R of salve-axis (ax2) to be 1.414 times of the value of the master-axis (ax1). Therefore, MCX314As will use the range parameter of master-axis if only 1 axis outputs pulses. However, when 2 axes output pulses simultaneously, MCX314As will use the range parameter of slave-axis to implement the pulse period to 1.414 times.

■ Setting Constant Vector Speed for 3 Axes

As same as the setting process of 3 axes, the user should first set the values of D9 and D8 of register WR5 to 0 and 1. Then, set the range R of ax2 to 1.414 times of the value of the master-axis (ax1); then, set the range R of ax3 to 1.732 times of the value of the master axis. After setting the range of constant vector speed for 3 axes, MCX314As will use the range parameter of ax1 if only 1 axis outputs pulses. However, when 2 or 3 axes output pulses simultaneously, MCX314As will use the range parameter of ax2 or ax3 to implement the pulse period. See Fig. 2.36.

User may set the values of D9 and D8 of register WR5 to 0 and 1 for 2-axis constant vector speed driving even in the 3-axis interpolation.

■ The Example of Constant Vector Speed Interpolation for 2 Axes

As shown below, the master-axis (ax1) = X axis, the slave-axis (ax2) = Y axis, and the interpolation is at a constant vector speed: 1000PPS. The result of driving pulse output is shown in Fig. 2.35.

```
WR5 ← 0104h write       ; define ax1: X axis, ax2: Y axis ;
                          constant vector speed

WR6 ← 0900h write       ; setting parameter of the master axis
WR7 ← 003Dh write       ; range: 4,000,000 (multiple=2)
WR0 ← 0100h write

WR6 ← 4DC0h write       ; range of 2-axis constant vector speed
WR7 ← 0056h write       ; 4,000,000x1.414=5,656,000
WR0 ← 0200h write

WR6 ← 01F4h write       ; initial speed: 500x2=1000PPS
WR0 ← 0104h write

WR6 ← 01F4h write       ; drive speed: 500x2=1000PPS
WR0 ← 0105h write
                                        (continue)

WR6 ← 03E8h write       ; finish point of X
WR7 ← 0000h write
WR0 ← 0106h write       ;

WR6 ← 0190h write       ; finish point of Y
WR7 ← 0000h write
WR0 ← 0206h write       ;

WR0 ← 0030h write       ; 2-axis linear
                          interpolation starting
```

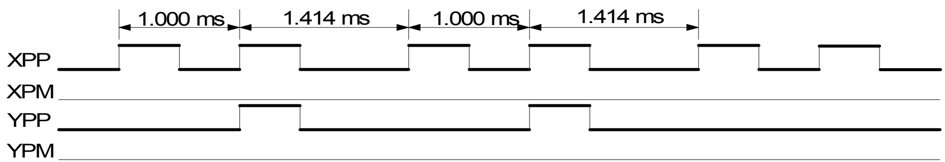


Fig. 2.35 Example of 2-Axis Interpolation at A Constant Vector Speed (Speed =1000PPS)

[Caution] In the process of constant vector speed, the pulse width of high level of output waveform will not be changed, yet kept in the same width. The pulse cycle will be changed to 1.414 or 1.732 times.

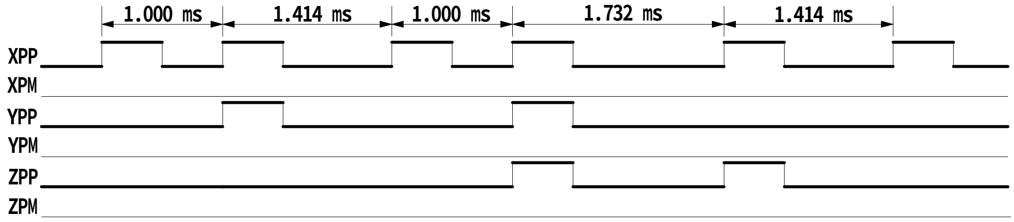


Fig. 2.36 Example of 3-Axis Interpolation at A Constant Vector Speed (speed =1000PPS)

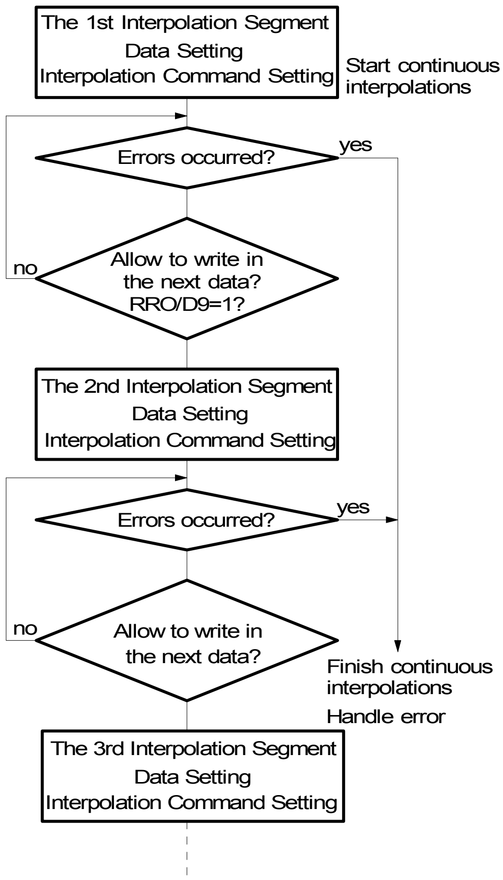
2.4.5 Continuous Interpolation

The continuous interpolation is executing a series of interpolation processes such as linear interpolation → circular interpolation → linear interpolation → ···.

During the continuous interpolation, the driving will not stop; contrarily, the pulses are output continuously. When executing the continuous interpolation, the host CPU has to write the next interpolation segment into MCX314As before the previous interpolation segment is finished.

■ Continuous Interpolation in Using D9 of RR0

If D9 (CNEXT) of register RR0 is 1, MCX314As is ready to accept the next interpolation segment. If D9 is 0, the host CPU is not able to write the next interpolation segment into MCX314As. The D9 will become 1 only when the present command is executed. MCX314As will not accept the next command, and the D9 is 0 if the present command has not been executed.
So, the standard procedure of continuous interpolation is first to write, and enable the interpolation data and command, then check if D9 of RR0 is 1 or 0. And then, repeat writing commands and checking D9. The flow chart is shown at the right side.



■ Interrupt

D14 of register WR5 is used for enable or disable the interrupt during the continuous interpolation. After setting D14 of register WR5 to 1, the interrupt occurs. Pin INTN of MCX314As will be on the Low level to interrupt the host CPU when D9 of register RR0 become 1. The INTN will be on the Hi level if the host CPU writes the next interpolation segment to MCX314As. If the interrupt clear command (3Dh) is written to command register, the INTN signal will return to high-Z level from the Low level.
During the ending of the interpolation, it is forced to be “interrupt disable”, and the INTN signal will return to the high-Z level.

■ Errors Occurring in the Process of Continuous Interpolation

If an error such as over-traveling occurs in the process of continuous interpolation, the drive will stop at the present interpolation segment. The following interpolation segment is still in the command register, but will not be executed. The host CPU has to reload the next command again and enable it.
As shown in the flow chart above, the host CPU has to check the error message before loading the following command. If not, this command will not be executed and will be jumped. So, the user should assure, and check if any error status will occur before the following interpolation segment is loaded.

■ Attentions for Continuous Interpolation

a. Before setting the interpolation segment, the user should first set other data such as center point, finish point… for each segment.

b. The maximum speed for the continuous interpolation is 2MHz.

c. The following interpolation segment must be loaded before the previous interpolation segment is finished.

d. The segment driving time should be longer than the time for error checking and the command setting of next segment during the interpolation.

e. It is impossible to operate 2-axis and 3-axis continuous interpolations at the same time.

f. It is not allowed to change the axis assignment during the process of continuous interpolation.

g. In continuous interpolation, if one of 2/3 axes is 0, interpolation is performed correctly, otherwise, 0 cannot be set to the finish point of all axes in 2/3 axes linear interpolation, or to the center point of both axes in circular interpolation, any axis cannot set the data that drive pulse is not output. If suchlike data it set, interpolation cannot be performed correctly.

■ The Example of Continuous Interpolation

Fig. 2.37 shows an example of executing continuous interpolation beginning at point (0,0) from segment 1, 2, 3….to the segment 8. In segment 1, 3, 5, and 7, the linear interpolation will be executed; in segment 2, 4, 6, and 8, the circular interpolation will be executed, and the track is a quadrant circle with radius 1500. The interpolation driving is at a constant vector speed: 1000PPS.

```
WR5 ← 0104h write      ; define ax1: X axis, ax2: Y axis,
                         constant vector speed

WR6 ← 0900h write      ; setting the parameter of master axis
WR7 ← 003Dh write      ; range:4,000,000 (multiple: 2)
WR0 ← 0100h write

WR6 ← 4DC0h write      ; 2-axis constant vector speed
WR7 ← 0056h write      ; 4,000,000x1.414=5,656,000
WR0 ← 0200h write

WR6 ← 01F4h write      ; Initial speed: 500x2=1000PPS
WR0 ← 0104h write

WR6 ← 01F4h write      ; drive speed: 500x2=1000PPS
WR0 ← 0105h write

WR6 ← 1194h write      ; finish point X: 4500
WR7 ← 0000h write
WR0 ← 0106h write

WR6 ← 0000h write      ; finish point Y: 0          Seg1
WR7 ← 0000h write
WR0 ← 0206h write


WR0 ← 0030h write      ; 2-axis linear interpolation

J1  RR0(D4, D5) read     ; if error occurs          Procedure A
    If D4 or D5=1        ; jump to handle error
    Jump to Error

    RR0(D9) read         ; waiting for next segment' s
                           enable signal
```

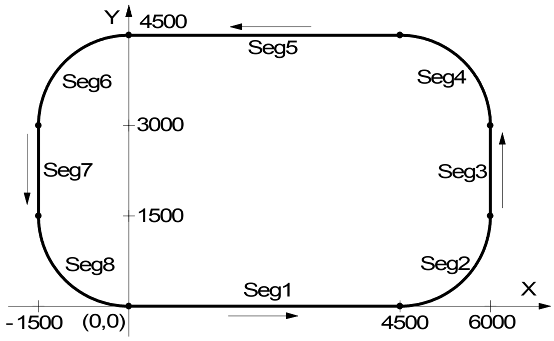


Fig. 2.37 The Example of Continuous Interpolation Axis

```
If D9=0 Jump to J1        ;
 WR6 ← 0000h write        ; center X: 0
 WR7 ← 0000h write
 WR0 ← 0108h write

 WR6 ← 05DCh write        ; center Y: 1500
 WR7 ← 0000h write
 WR0 ← 0208h write

 WR6 ← 05DCh write        ; finish point X:1500             Seg2
 WR7 ← 0000h write
 WR0 ← 0106h write

 WR6 ← 05DCh write        ; finish point Y:1500
 WR7 ← 0000h write
 WR0 ← 0206h write


 WR0 ← 0033h write        ; CCW circular interpolation
```

Procedure A

```
 WR6 ← 0000h write        ; finish point X: 0
 WR7 ← 0000h write
 WR0 ← 0106h write

 WR6 ← 05DCh write        ; finish point Y: 1500            Seg3
 WR7 ← 0000h write
 WR0 ← 0206h write

 WR0 ← 0030h write        ; 2-axis linear interpolation
```

Procedure A

(same procedure for segments 4 ~ 8.)

2.4.6 The Acceleration / Deceleration Control in Interpolation

Different from other IC chips only allowing constant speed for executing the interpolations, MCX314As supports the user to use trapezoidal and S-curve driving (for linear interpolation only).
In the process of interpolation, for executing acceleration / deceleration in continuous interpolation process, the user can enable the deceleration by command (3Bh), or disable deceleration by command (3Ch). The purpose for the deceleration command is to enable the automatic deceleration or manual deceleration function; the purpose of the disable deceleration command is to disable both of them. It will be disable while power-on reset. During the driving, the deceleration enable command cannot be executed.

■ The Acceleration / Deceleration for 2-axis / 3-axis Interpolation

It is possible to perform trapezoidal and S-curve acceleration/deceleration driving during the execution of 2-axis / 3-axis linear interpolation. Either automatic or manual deceleration can be used for decelerating.

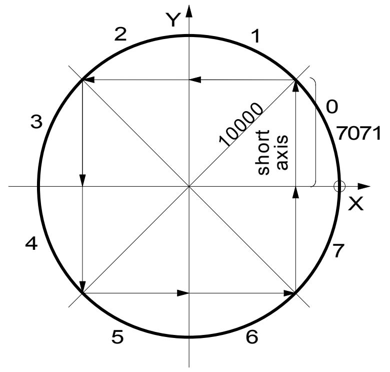
When the manual deceleration is executed, the user can set the maximum absolute value of the axes to be the setting value of master axis decelerating point. For instance, while executing 3-axis linear interpolation of master axis (ax1): X, ax2 : Y and ax3 : Z, the finish point : (X: −20000, Y: 30000, Z: −50000), and the assumed pulse numbers needed for deceleration are 5000. In such situation, the absolute value of Z axis is the largest, so we can set up 50000−5000=45000 to be the manual deceleration point of the master axis: X.

Please refer to the example of 3-axis linear interpolation in 2.4.1.

■ The Acceleration / Deceleration for Circular Interpolation and Bit Pattern Interpolation

In circular interpolation and bit pattern interpolation, only manual deceleration in trapezoidal driving is available; the automatic deceleration in S-curve driving is not available.

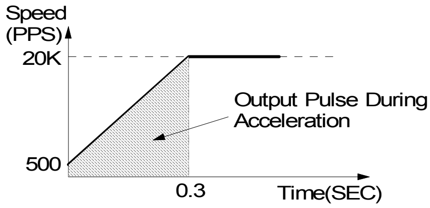
The Figure on the right side shows the circular interpolation of a real circle with radius 1000 in a trapezoidal driving.



The user should calculate the decelerating point before driving because the automatic deceleration will not be active.

In the figure, the circle tracks through all the 8 quadrants: 0~7. In quadrant 0, Y axis is the short axis and it' s displace is about 10000 / $\sqrt{2}$=7071. The total output pulses numbers of the short axis are 7010×8=56568.

Furthermore, if the initial speed is 500PPS, and will be accelerated to 20KPPS after 0.3 SEC, the acceleration will be (20000−500) /0.3 = 65000PPS/SEC. And the output pulses during acceleration will be (500+20000) × 0.3/2=3075. Thus, if we set the deceleration as same as the acceleration, the manual decelerating point will be 56568 - 3075=53493.
[Note] this formula cannot be used in the constant vector speed driving.

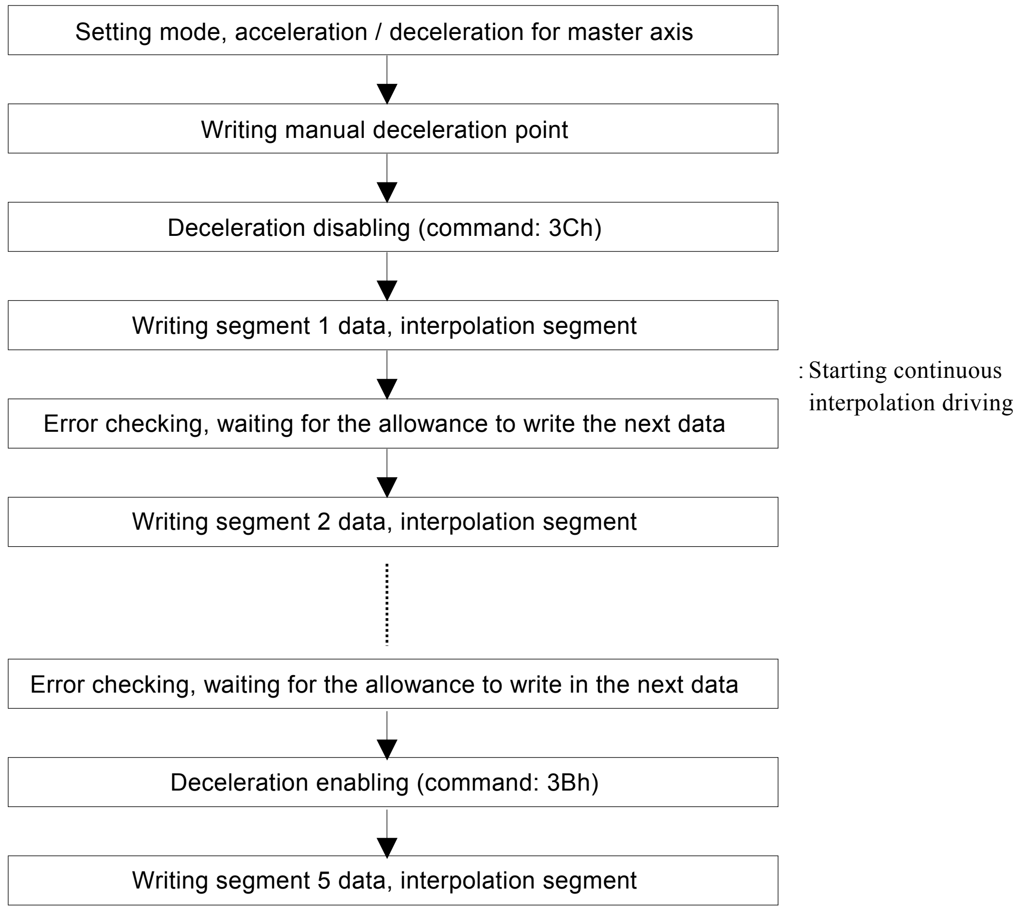



```
WR3 ← 0001h write        ; manual deceleration enabling
WR5 ← 0004h write        ; define ax1: X , ax2: Y

WR6 ← 8480hwrite         ; range: 2,000,000; (multiple: 4)
WR7 ← 001Eh write        ;
WR0 ← 0100h write

WR6 ← 0082h write        ; acceleration :
WR0 ← 0102h write        ; 130x125x4=65000 PPS/SEC

WR6 ← 007Dh write        ; Initial speed:125x4=500PPS
WR0 ← 0104h write

WR6 ← 1388h write        ; drive speed: 5000x4=20000PPS
WR0 ← 0105h write

WR6 ← D8F0h write        ; center point X : −10000
WR7 ← FFFFh write        ;
WR0 ← 0108h write

WR6 ← 0000h write        ; center point Y : 0
WR7 ← 0000h write        ;
WR0 ← 0208h write

WR6 ← 0000h write        ; finish point X : 0
WR7 ← 0000h write        ;
WR0 ← 0106h write

WR6 ← 0000h write        ; finish point Y : 0
WR7 ← 0000h write        ;
WR0 ← 0206h write

WR6 ← D0F5h write        ; manual deceleration point setting : 53493
WR7 ← 0000h write        ;
WR0 ← 0107h write

WR0 ← 003Bh write        ; deceleration enabling
WR0 ← 0033hwrite         ; CCW circular interpolation starting
```

■ The Acceleration / Deceleration for Continuous Interpolation

In continuous interpolation, same as in circular and bit pattern interpolations, only manual deceleration in the trapezoidal driving is available; The automatic deceleration in S-curve driving is not available.
Before performing the continuous interpolation, it is necessary to preset the manual decelerating point; however, this setting point is related to the master axis executing the deceleration in the last segment. The user should disable the deceleration, then start the interpolation driving. Before writing the interpolation command to the final segment which will execute the deceleration, the user should enable the deceleration at first. The deceleration will start if the output pulses are larger than the master axis based pulses in the final segment.

For instance, there are 5 interpolation segments in the process of continuous interpolation. In case, the manual deceleration has to be executed in the last segment, segment 5, the procedure is shown as follows:



The manual deceleration point is related to the master axis' driving pulses which comes from segment 5. For instance, assumed that it needs 2000 pulses for decelerating stop, and the total amount of pulse output form segment 5 is 5000. So, the manual deceleration point will be 5000 − 2000 = 3000.

The cycle of deceleration should be started and finished within the same segment.

2.4.7 Single-step interpolation (from Command or External Signal)

Single-step is defined as: pulse by pulse outputting. Either command or external signal can execute the single-step interpolation. When one pulse is outputting, the master axis interpolation will be set in the constant speed driving.

The Hi level width of each axis' s output pulse is 1/2 of the pulse cycle which is decided by the interpolating master axis' s drive speed. The Low level width is kept until next command or external signal comes. Fig. 2.38 is the example showing the execution of single-step interpolation from an external signal. The master axis' s initial speed is 500PPS, the drive speed is at 500PPS constant speed driving. The Hi level width of output pulse is 1msec.

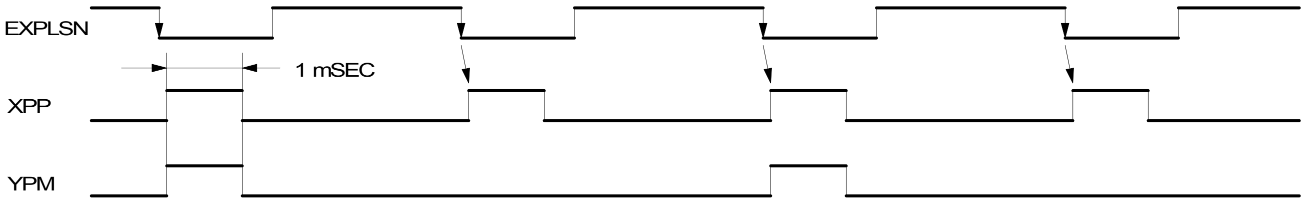


Fig. 2.38 Example of Single Step Interpolation (500PPS) by External Signal (EXPLSN)

■ Command Controlled Single-step Interpolation

The command: 3Ah is for single-step interpolation. The user can set D12 of register WR5 to 1 to enable the command controlled single-step interpolation. The operating procedure is shown as follow.

a. Set D12 of register WR5 to 1.
   It will enable the command controlled single-step interpolation.

b. Set the initial and drive speeds of the master axis in the interpolation process with the same value, and the driving becomes constant speed.
   If the host CPU writes single step command into MCX314As at most 1mSEC, the user should set the drive speed more than 1000PPS.

c. Set interpolation data. (finish point, center point…)

d. Write interpolation command.
   Although the interpolation segment is enabled, there is no pulse output because the single-step is command controlled.

e. Write the single-step interpolation command (3Ah).
   The driving pulses according to the interpolation calculation will be output for each axis. The user may use command 3Ah for single step until the interpolation driving is finished.

If the user wants to stop sending single-steps during the interpolation, he can use the sudden stop command (27h), then wait for more than 1 pulse cycle, and then write the command (3Ah) again to stop the driving.

After this, all the following (3Ah) commands will not be active.

■ External Signal Controlled Single-step Interpolation

The EXPLSN pin (29) is used for the single-step interpolation from the external signal. The user can set D11 of register WR5 to 1 to enable the external signal controlled single-step interpolation. Normally, the EXPLSN input signal is on the Hi level. When it changes to Low, the interpolation step will be output.

The operating procedure is shown as follows.

a. Set D11 of register WR5 to 1.
   It will enable the external signal controlled single-step interpolation.

b. Set the initial and drive speeds of the master axis in the interpolation process to be the same value, and the driving becomes constant speed which should be higher than the Low pulse cycle of EXPLSN.
   This is necessary for this controlled mode. And it will set the MCX314As into a constant speed mode.

c. Set interpolation data. (start point, center point…)

d. Write interpolation command.
   Although the interpolation segment is enabled, there is no pulse output because the single-step is command controlled.

e. EXPLSN input on Low level
   The interpolation pulse will be output from each axis after 2~5 CLK the pulse falling down (when the filer is invalid).

The Low level pulse width of EXPLSN has to be longer than 4CLK (when the filter is invalid. See Section 2.8 for filters). Furthermore, the pulse cycle of EXPLSN has to be longer than the setting speed cycle of master axis.

The user may repeat the Low level of EXPLSN before the interpolation is finished.

If the user wants to stop sending single-steps during the interpolation, he can use the sudden stop command (27h), then wait for more than 1 pulse cycle, and then input pulse on EXPLSN Low level again to stop the driving (the user may try software reset also).

After this, all the following input pulses on EXPLSN Low level will not be active.

[Note] When generating Low pulses of EXPLSN at a mechanical contact point, prevent the occurrence of chattering by enabling the input signal filter (see Section 2.8) of the EXPLSN signal.

2.5 Automatic Home Search

This IC has a function that automatically executes a home search sequence such as high-speed near home search → low-speed home search → encoder Z-phase search → offset driving without CPU intervention. The automatic home search function sequentially executes the steps from step 1 to step 4 that are listed below. Set execution/non-execution and a search direction mode for each step. In steps 1 and 4, search operation is performed at the high-speed that is set in the drive speed. In steps 2 and 3, search operation is performed at the low-speed that is set in the home search speed.

Step number	Operation	Search speed	Detection signal
Step 1	High-speed near home search	Drive speed (V)	nIN0 *1
Step 2	Low-speed home search	Home search speed (HV)	nIN1 *1
Step 3	Low-speed Z-phase search	Home search speed (HV)	nIN2
Step 4	High-speed offset drive	Drive speed (V)	-

*1: By inputting a home signal in both nIN0 and nIN1, high-speed search is enabled by using only one home signal. (See “Example of home search using a home signal only” in Section 2.5.7).

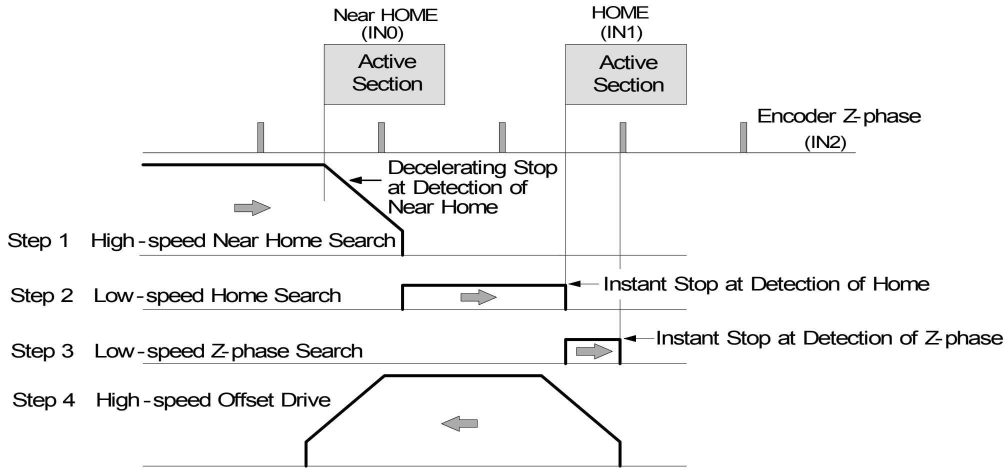


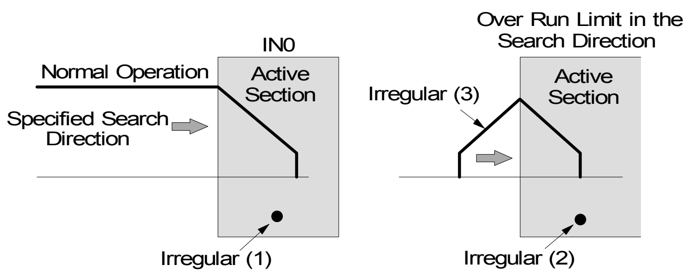
Fig. 2.39 Prototype of Automatic Home Search Using This IC

2.5.1 Operation of Each Step

In each step, it is possible to specify, in mode setting, execution/non-execution and the +/− search direction. If non-execution is specified, the function proceeds with the next step without executing the step.

■Step 1: High-speed near home search

Drive pulses are output in the specified direction at the speed that is set in the drive speed (V) until the near home signal (nIN0) becomes active. To perform high-speed search operation, set a higher value for the drive speed (V) than the initial speed (SV). Acceleration/deceleration driving is performed and when the near home signal (nIN0) becomes active, the operation stops by decelerating.

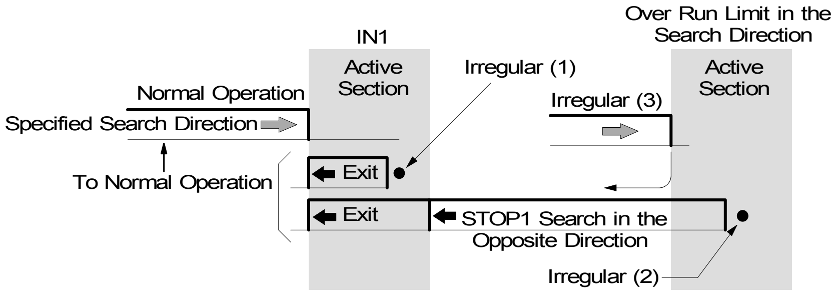


Irregular operation

(1) The near home signal (nIN0) is already active before Step 1 starts. → Proceeds with Step 2.
(2) The limit signal in the detection direction is already active before Step 1 starts. → Proceeds with Step 2.
(3) The limit signal in the detection direction is activated during execution. → Stops driving and proceeds with Step 2.

■Step 2: Low-speed home search

Drive pulses are output in the specified direction at the speed that is set as the home detection speed (HV) until the home signal (nIN1) becomes active. To perform low-search operation, set a lower value for the home search speed (HV) than the initial speed (SV). A constant speed driving mode is applied and the operation stops instantly when the home signal (nIN1) becomes active.



Irregular operation

(1) The home signal (nIN1) is already active before Step 2 starts.

→The motor drives the axis in the direction opposite to the specified search direction at the home search speed (HV) until the home signal (nIN1) becomes inactive. When the home signal (nIN1) becomes inactive, the function executes Step 2 from the beginning.

(2) The limit signal in the search direction is active before Step 2 starts.

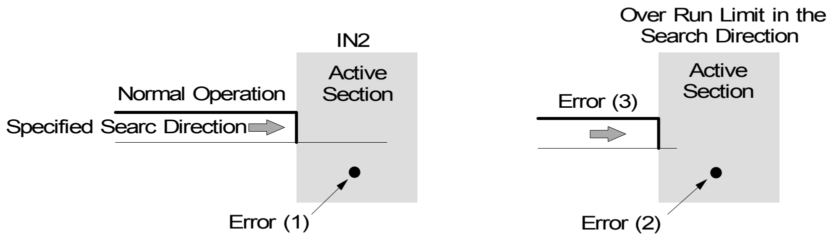
→The motor drives the axis in the direction opposite to the specified search direction at the home search speed (HV) until the home signal (nIN1) becomes active. When the home signal (nIN1) becomes active, the motor drives in the direction opposite to the specified search direction at the home search speed (HV) until the home signal (nIN1) becomes inactive. When the home signal (nIN1) becomes inactive, the function executes Step 2 from the beginning.

(3) The limit signal in the search direction becomes active during execution.

→Driving stops and the same operation as for (2)→ is performed.

■Step 3: Low-speed Z-phase search

Drive pulses are output in the specified direction at the speed that is set as the home search speed (HV) until the encoder Z-phase signal (nIN2) becomes active. To perform low-speed search operation, set a lower value for the home search speed (HV) than the initial speed (SV). A fixed speed driving mode is applied and driving stops instantly when the encoder Z-phase signal (nIN2) becomes active.



As the search condition for stopping driving, the AND condition of the encoder Z-phase signal (nIN2) and the home signal (nIN1) can be applied.

A deviation counter clear signal can be output for a servomotor when the encoder Z-phase signal (nIN2) rises to active. See Section 2.5.2. The real position counter (EP) can be cleared when an encoder Z-phase signal (nIN2) rises to active. See Section 2.3.4.

[Notes]

(1) If the encoder Z-phase signal (nIN2) is already active at the start of Step 3, an error occurs and 1 is set in bit D7 of the nRR2 register. Automatic home search ends. Adjust the mechanical system so that Step 3 always starts from an inactive state with a stable encoder Z-phase signal (nIN2).

(2) If the limit signal in the search direction is already active before the start of Step 3, an error occurs and 1 is set in the search direction limit error bit (D2 or D3) of the nRR2 register. Automatic home search ends.

(3) If the limit signal in the search direction becomes active during execution, search operation is interrupted and 1 is set in the search direction limit error bit (D2 or D3) of the nRR2 register. Automatic home search ends.

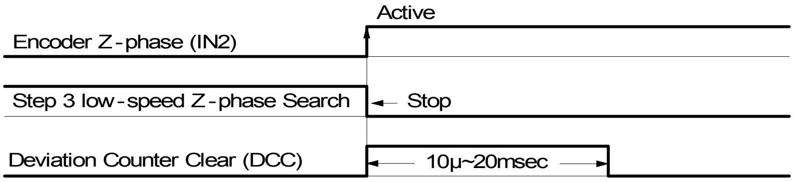
■Step 4: High-speed offset drive

The function outputs as many driving pulses as the output pulse numbers (P) that is set, in the specified direction at the speed that is set in the drive speed (V). Use this step to move the axis from the mechanical home position to the operation home position. Through mode setting, the logical position counter and real position counter can be cleared after moving.

If the drive direction limit signal becomes active before the start or during execution of Step 4, the operation stops due to an error and 1 is set in the search direction limit error bit (D2 or D3) of the nRR2 register.

2.5.2 Deviation Counter Clearing Signal Output

This function outputs a deviation counter clearing (nDCC) signal, by setting a mode, at the activation of the encoder Z-phase signal (nIN2) in Step 3 operation. For deviation counter clearing output, the pin is shared between nDRIVE and DCC output signals. For the clearing pulse, the logical level and the pulse width within the range from 10μsec to 20msec can be specified.



Deviation counter clearing output becomes active at termination of Z-phase search operation in Step 3, and Step 4 starts after completion of clearing pulse output.

Deviation counter clearing pulses can also be output by a single command (deviation counter clear command (63h)) instead of an automatic home search sequence. However, the mode following the deviation counter clearing output must be set in advance using an extension mode setting command (60h).

WR7/D11(DCC-E)	Disable/enable: 1 enable
WR7/D12(DCC-L)	Logical level: 0 or 1
WR7/D15 ~ D13(DCCW2 ~ 0)	Pulse width: 0~7

2.5.3 Setting a Search Speed and a Mode

To perform automatic home search, the following speed parameters and mode must be set.

■ Setting speed parameters

Speed parameter	Command code	Description
Drive speed (V)	05	High-speed search speed that is applied in Steps 1 and 4. The range (R), acceleration (A), and initial speed (SV) must also be set to appropriate values to perform acceleration/deceleration driving. See Section 2.2.2.
Home search speed (HV)	61	Low-speed search speed that is applied in Steps 2 and 3. Set a value lower than the initial speed (SV) to stop operation instantly when the search signal becomes active. See Section 2.2.1.

■ Setting an automatic home search mode

Use an extension mode setting command (60h) to set an automatic search mode. Set each bit of the WR7 register as shown below. To generate an interrupt at termination of automatic home search, set D5 (HMINT) of the WR6 register to 1. Since bit data of the WR6 and WR7 of an extension mode setting command (60h) is written to the internal registers simultaneously, the appropriate values must be set for other bits of the WR6 register.

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
WR7	DCCW2	DCCW1	DCCW0	DCC-L	DCC-E	LIMIT	SAND	PCLR	ST4-D	ST4-E	ST3-D	ST3-E	ST-D	ST2-E	ST1-D	ST1-E
	Deviation Counter Clear Output								Step 4		Step 3		Step 2		Step1	

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
WR6	FL2	FL1	FL0	FE4	FE3	FE2	FE1	FE0	SMODE	0	HMINT	VRING	AVTRI	POINV	EPINV	EPCLR

WR7/D6,4,2,0 STm-E Specify whether operation of each step is executed. 0: Non-execution 1: Execution
Use the WR1 register for logical setting of the input signal that is detected in each step. See Section 4.4

WR7/D7,5,3,1 STm-D Specify search/operation direction of each step. 0: + direction, 1: − direction

WR7/D8 PCLR When this bit is set to 1, the logical position counter and the real position counter are cleared at termination of Step 4.

WR7/D9 SAND When this bit is set to 1, operation of Step 3 stops when the home signal (nIN1) and the encoder Z-phase signal (nIN2) become active.

WR7/D10 LIMIT Set this bit to 1 when setting automatic home search using an overrun limit signal(nLMTP or nLMTM).

WR7/D11 DCC-E This bit enables/disables deviation counter clearing output. 0:Enable, 1:Disable
For deviation counter clearing output, the pin is shared between the nDRIVE and DCC output signals. When this bit is set to 1, the pin is set to deviation counter clearing output.

WR7/D12 DCC-L Specify a deviation counter clearing output logical level. 0:Active High, 1:Active Low

WR7/D15 ~ 13 DCCW2 ~ 0 Specify an active pulse width of deviation counter clearing output.

D15 DCCW2	D14 DCCW1	D13 DCCW0	Clearing pulse width (μSEC)
0	0	0	10
0	0	1	20
0	1	0	100
0	1	1	200
1	0	0	1,000
1	0	1	2,000
1	1	0	10,000
1	1	1	20,000

[Note] CLK=16MHz

WR6/D5 HMINT Use this bit to generate an interrupt signal (INTN) at termination of automatic home search. When this bit is set to 1, the interrupt signal (INTN) becomes Low Active after termination of automatic home search and the RR3/D8 (HMEND) bit of the axis from which the interrupt was generated indicates 1. When the CPU reads the RR3 register of the axis from which interrupt was generated, the bits of the RR3 register are cleared to 0 and the interrupt output signal is reset to Hi-Z.

At resetting, all the mode setting bits of each axis are reset to 0.

2.5.4 Execution of Automatic Home Search and the Status

■ Execution of automatic home search

Automatic home search is executed by an automatic home search execution command (62h). Automatic home search can be executed by writing the command code 62h with the axis assignment to WR0 register after correctly setting an automatic home search mode and speed parameter for each axis. This function can be executed for each axis individually or for all the axes collectively.

■ Suspension of automatic home search

To suspend automatic home search operation, write a drive decelerating stop command (26h) or a drive instant stop command (27h) for the axis. The step currently being executed is suspended and automatic home search terminates.

■ Main status register

Bits D3 to D0 of the main status register RR0 indicate the driving execution of the axis. These bits also indicate execution of automatic home search. When automatic home search of each axis starts, these bits are set to 1 and the state is maintained from the start of Step 1 operation to the end of Step 4 operation. At termination of Step 4, the bits are reset to 0.

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
RR0	-	BPSC1	BPSC0	ZONE2	ZONE1	ZONE0	CNEXT	I-DRV	U-ERR	Z-ERR	Y-ERR	X-ERR	U-DRV	Z-DRV	Y-DRV	X-DRV
									Error of Each Axis				Driving of Each Axis			

The D7 to D4 (n-ERR) bits that indicate an error of each axis sometimes indicate 1 in spite of normal operation when the limit signal in the search direction is set in regular operation of Step 1 or 2. Check these error bits at termination of automatic home search, instead of monitoring during execution of automatic home search.

■ Status register 2

Bits D7 to D0 of status register 2 (RR2) indicate error information and bits D12 to D8 indicate a home search execution state.

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
RR2	-	0	0	HMST4	HMST3	HMST2	HMST1	HMST0	HOME	0	EMG	ALARM	HLMT-	HLMT+	SLMT-	SLMT+

D12–D8: Automatic Home Search Execution State
D7: IN2 Signal Error at Automatic Home Search

The error information bit D7 (HOME) is set to 1 when the encoder Z-phase signal (nIN2) is already active at the start of Step 3 during execution of automatic home search. This bit is cleared when the next drive command or an automatic home search command is written. The bit can also be cleared by a termination status clearing command (25h).

An automatic home search execution state indicates the details of the operation that is currently being executed in automatic home search.

Execution state	Execution step	Operation details
0		Waits for an automatic home search execution command
3	Step 1	Waits for activation of the IN0 signal in the specified search direction
8	Step 2	Waits for activation of the IN1 signal in the direction opposite to the specified search direction (irregular operation)
12	Step 2	Waits for deactivation of the IN1 signal in the direction opposite to the specified search direction (irregular operation)
15	Step 2	Waits for activation of the IN1 signal in the specified search direction
20	Step 3	Waits for activation of the IN2 signal in the specified search direction
25	Step 4	Offset driving in the specified search direction

2.5.5 Errors Occurring at Automatic Home Search

The following table lists the errors that may occur during execution of automatic home search.

Cause of the error	Operation of IC at the occurrence of error	Display at termination
The ALARM signal was activated in any of the Steps 1 to 4	The search driving stops instantly without executing the following steps.	RR0-D7 ~ 4:1, nRR2-D4:1 nRR1-D14:1
The EMGN signal was activated in any of the Steps 1 to 4	The search driving stops instantly without executing the following steps.	RR0-D7 ~ 4:1, nRR2-D5:1 nRR1-D15:1
The limit signal in the positive direction (LMTP/M) is activated in Step 3	The search driving stops instantly/by decelerating without executing the following steps.	RR0-D7 ~ 4:1, nRR2-D3/2:1 nRR1-D13/12:1
The limit signal in the positive direction (LMTP/M) is activated in Step 4	The offset action stops instantly/by decelerating and the operation stops.	RR0-D7 ~ 4:1, nRR2-D3/2:1 nRR1-D13/12:1
The IN2 signal is already active at the start of Step 3	Operation stops without executing the following steps.	RR0-D7 ~ 4:1, nRR2-D7:1

Always check the error bits (RR0-D7 to D4) of each axis after termination of automatic home search. When automatic home search is not performed correctly, the error bit is set to 1. It is not recommended to monitor the error bit of each axis during execution of automatic home search. This is because the error bit indicates 1 in spite of normal operation when the limit signal in the search direction is set in irregular operation of Step 1 or 2.

■ Symptom at sensor failure

This section describes the symptoms when a failure occurs regularly in the sensor circuit such as a home search signal or a limit signal. Analysis of intermittent failures caused by noise around the cable path, loose cable, or unstable operation of the device is difficult and such failures are not applicable to this case. These symptoms may occur due to a logical setting error or signal wiring error at the development of a customer system.

Failure cause		Symptom
Failure in the device of the limit sensor and wiring path	Kept ON	The axis does not advance to the direction and the limit error bit (nRR2-D3.2) is set to 1 at termination.
	Kept OFF	The axis runs into the mechanical terminal point and the home search operation does not terminate.
Failure in the device of the near home (nIN0) sensor and wiring path	Kept ON	Although Step 1 is enabled and automatic home search is started from the signal OFF position, the axis advances to Step 2 without executing Step 1 (high-speed near home search).
	Kept OFF	Operation stops in Step 1 (high-speed near home search) by setting the limit and proceeds with irregular operation of Step 2. The home search result is correct, however, the operation is not normal.
Failure in the device of the home (nIN1) sensor and wiring path	Kept ON	The axis moves in the opposite direction in Step 2 (low-speed home search) and stops by setting the limit. At termination, the error bit (nRR2-D3/2) of the limit of the opposite direction is set to 1.
	Kept OFF	The axis moves in the opposite direction after setting the limit in the specified direction in Step 2 (low-speed home search) and terminates by setting the limit of the opposite direction. At termination, the error limit (nRR2-D3/2) of the limit of the reverse direction is set to 1.
Failure in the device of the Z-phase (nIN2) sensor and wiring path	Kept ON	Operation stops due to an error in Step 3 (low-speed Z-phase search). NRR2-D7 is set to n.
	Kept OFF	Operation stops in Step 3 (low-speed Z-phase search) by setting the limit in the specified direction. The error bit of the limit in the specified direction (nRR2-D3/2) is set to 1 at termination.

2.5.6 Notes on Automatic Home Search

■ Search speed

A home search speed (HV) must be set to a low speed to increase the home search position precision. Set a value lower than the initial speed to stop the operation immediately when the input signal becomes active.
For encoder Z-phase search of Step 3, the relationship between the Z-phase signal delay and the home search speed (HV) becomes important. For instance, if a total of the photo coupler delay time of the Z-phase signal path and delay time of the integral filter incorporated in the IC is the maximum 500μsec, the home search speed must be set so that the encoder Z-phase output is ON for more than 1msec.

■ Step 3 (Z-phase search) starting position

In Z-phase search of Step 3, the function stops search driving when the Z-phase signal (nIN2) changes from active to inactive. Therefore, the Step 3 starting position (that is, Step 2 stop position) must be stable and must be different from this change point. Normally, adjust mechanically so that the Step 3 starting position becomes the 180° opposite side to the encoder Z-phase position.

■ Software limit

Disable the software limit during execution of automatic home search. If software limit is enabled, automatic home search is not performed correctly. Set a software limit after setting a real position counter following normal completion of automatic home search.

■ Logical setting of each input signal

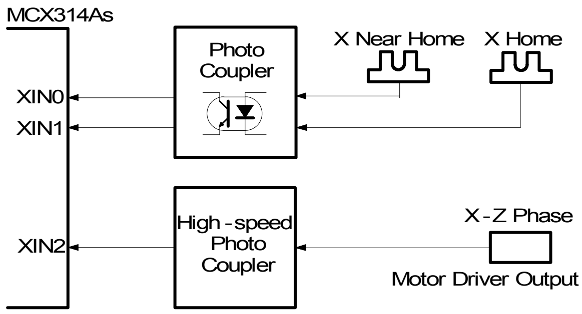
Use the bits (WR1-D2, D4, and D7) of the WR1 register for input signal (nIN0, 1, 2) of active logical setting that is used by automatic home search. At automatic home search, the contents set in the bits (WR1-D1, D3, and D5) that enable/disable each signal are ignored.

2.5.7 Examples of Automatic Home Search

■ Example of home search using a near home, home, or a Z-phase signal

[Operation]

	Input signal and logical level	Search direction	Search speed
Step 1	Near home signal (IN0) signal Low active	−	20,000pps
Step 2	Home (IN1) signal Low active	−	500pps
Step 3	Z-phase (IN2) signal High active	+	500pps
Step 4	(3500 pulse offset driving in the + direction)	+	20,000pps



• For high-speed search in Step 1 and offset driving in Step 4, acceleration/deceleration driving is performed where linear acceleration is applied at the speed within the range from the initial speed 1,000pps to 30,000pps in 0.2 seconds (acceleration speed = 19,000/0.2 = 95,000 pps/sec).

• When Z-phase of Step 3 is High active, deviation counter pulses of 100μsec are output from the XDRIVE/DCC output signal pin. The logical level is High active.

• At completion of Step 4, the logical position counter value and the real position counter value are cleared.

[Parameter and mode setting]

```
WR0 ← 010Fh Write     ; X axis selection
WR1 ← 0010h Write     ; Input signal logical setting: XIN0 and XIN1: Low active, XIN2: High active (See 4.4)

                      ; Sets an extension mode
WR6 ← 5D00h Write     ; Writes an input signal filer mode in WR6 (See 2.8)
                      ; D15 ~ D13  010   filter delay: 512μsec
                      ; D9          0    XIN2 signal: Disables the filter (through)
                      ; D8          1    XIN1,0 signal: Enables the filer

WR7 ← 495Fh Write     ; Writes an automatic home search mode in WR7
                      ; D15 ~ D13  010   Deviation counter clearing pulse width:      100μsec
                      ; D12         0    Deviation counter clearing output logical level: Active High
                      ; D11         1    Deviation counter clearing output:           Enable (output from the XDCC pin)
                      ; D10         0    Uses a limit signal as the home signal:      Disable
                      ; D9          0    Z-phase AND home signal:                     Disable
                      ; D8          1    Logical/real position counter area:          Enable
                      ; D7          0    Step 4 driving direction:                    + direction
                      ; D6          1    Step 4:                                      Enable
                      ; D5          0    Step 3 search direction:                     + direction
                      ; D4          1    Step 3:                                      Enable
                      ; D3          1    Step 2 search direction:                     − direction
                      ; D2          1    Step 2:                                      Enable
                      ; D1          1    Step 1 search direction:                     − direction
                      ; D0          1    Step 1:                                      Enable
WR0 ← 0160h Write     ; Writes an extension mode setting command in the x axis

WR6 ← 3500h Write     ; Range: 8,000,000 (Scaling factor: 10)
WR7 ← 000Ch Write
WR0 ← 0100h Write

WR6 ← 004Ch Write     ; Acceleration speed: 95,000 PPS/SEC
WR0 ← 0102h Write     ; 95000/125/10 = 76

WR6 ← 0064h Write     ; Initial speed: 1000 PPS
WR0 ← 0104h Write

WR6 ← 07D0h Write     ; Speed of Steps 1 and 4: 20000 PPS
WR0 ← 0105h Write

WR6 ← 0032h Write     ; Speed of Steps 2 and 3: 500 PPS
WR0 ← 0161h Write

WR6 ← 0DACh Write     ; Offset driving pulse count : 3500
WR7 ← 0000h Write
WR0 ← 0106h Write

WR0 ← 0162h Write     ; Starts execution of automatic home search
```

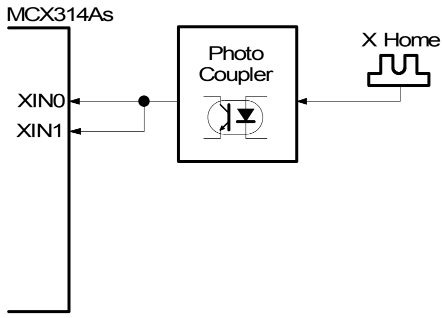
After start of the execution, the function monitors the RR0-D0(X-DRV) bit and terminates automatic home search if the bit is reset to 0 from 1. If automatic home search did not terminate normally due to an error, the RR0-D4(X-ERR) bit is set to 1 after termination. Analyze the error based on the contents of the XRR2-D7, bits D5 to D0, and bits XRR1-D15 to D12.

■ Example of home search using a home search signal only

In this example, high-speed home search is triggered by one home signal that is input to both the IN0 and IN1 pins of this IC.

[Operation]

	Input signal and logical level	Search direction	Search speed
Step 1	Near home (IN0) Signal Low active	−	20,000pps
Step 2	Home (IN1) signal Low active	−	500pps
Step 3	(Not executed)		
Step 4	(3500 pulse offset driving in the + direction)	+	20,000pps



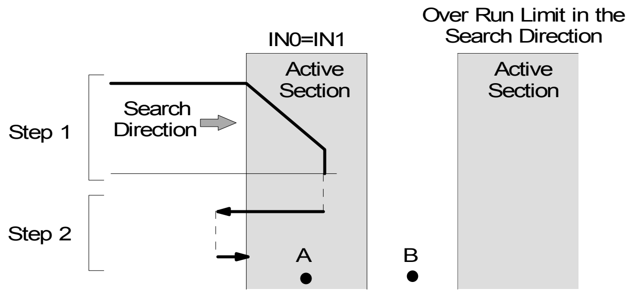
As shown in the table, the same search direction is specified for the signal logical levels of Step 1 and Step 2.(An opposite logical level may also be set.)

High-speed home search is performed in Step 1 and operation stops by decelerating when the home signal becomes active. If the stop position is within the home signal active section, controls escapes in the opposite direction by the irregular operation ① of Step 2 and searches a home in operation of Step 2.

If the Step 1 stop position passed through the home signal active section, the limit in the search direction is set in Step 2. In this case, irregular operation ③ is performed.

When the automatic home search starting position is in point A as shown in the diagram, the function performs irregular operation ① of Step 2 without executing Step 1.

When the starting position is in point B in the diagram, the function performs irregular operation ③ in Step 2 after setting the limit in search direction in Step 1.



[Parameter and mode setting]

```
WR0 ← 010Fh Write     ; Selects X axis
WR1 ← 0000h Write     ; Input signal logical setting: XIN0: Low active, XIN1: Low active (See 4.4)

                      ; Sets an extension mode
WR6 ← 5F00h Write     ; Writes a mode of an input signal filter in WR6 (see 2.8)
                      ; D15 ~ D13  010   Filter delay: 512µsec
                      ; D8           1   XIN1,0 signal: Enables the filter

WR7 ← 014Fh Write     ; Writes an automatic home search mode in WR7
                      ; D15 ~ D13  000
                      ; D12          0
                      ; D11          0   Deviation counter clearing output:      Disable
                      ; D10          0   Uses a limit signal as the home signal: Disable
                      ; D9           0   Z-phase signal AND home signal:         Disable
                      ; D8           1   Clears logical/real position counter:   Enable
                      ; D7           0   Step 4 driving direction:               + direction
                      ; D6           1   Step 4:                                 Enable
                      ; D5           0   Step 3 search direction:
                      ; D4           0   Step 3:                                 Disable
                      ; D3           1   Step 2 search direction:                − direction
                      ; D2           1   Step 2:                                 Enable
                      ; D1           1   Step 1 search direction:                − direction
                      ; D0           1   Step 1:                                 Enable
WR0 ← 0160h Write     ; Writes an extension mode setting command in the X axis

WR6 ← 3500h Write     ; Range: 8,000,000 (multiple: 10)
WR7 ← 000Ch Write
WR0 ← 0100h Write

WR6 ← 004Ch Write     ; Acceleration speed: 95,000 PPS/SEC
WR0 ← 0102h Write     ; 95000/125/10 = 76

WR6 ← 0064h Write     ; Initial speed: 1000 PPS
WR0 ← 0104h Write

WR6 ← 07D0h Write     ; Speed of Steps 1 and 4: 20000 PPS
WR0 ← 0105h Write

WR6 ← 0032h Write     ; Speed of Step 2: 500 PPS
```

```
WR0 ← 0161h Write

WR6 ← 0DACh Write      ; Offset driving pulse count : 3500
WR7 ← 0000h Write
WR0 ← 0106h Write

WR0 ← 0162h Write      ; Starts execution of automatic home search
```

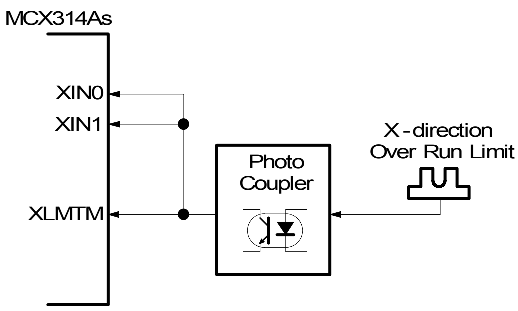
■ Example of home search using a limit signal

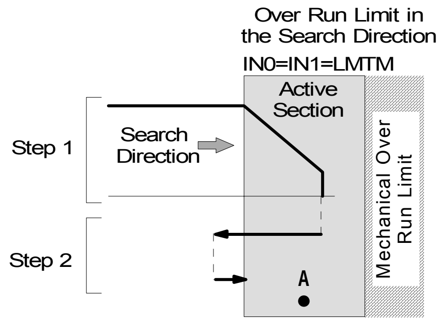
For a simple home search, a limit signal of one side is used as an alternative home signal. However, the following two conditions are applied.

a. When high-speed search operation is performed, decelerating stop must be able to be performed sufficiently within the distance from the limit signal activation position to the mechanical limit position.
b. The automatic home search position is not beyond the limit signal active section in the search direction.

In this example, a limit signal in one direction is used as an alternative home signal.

- Connect XLMTM input to the XIN0 and XIN1 input pins as shown in the diagram on the left-hand side. [Note] This IC pin signal must be connected. If an external signal is connected from the photo coupler of each signal, an operation error may occur due to the photo coupler delay time difference.
- Since high-speed search of Step 1 is performed, set the limit stop mode to a decelerating stop mode. (Section 4.5 WR2/D2 bit)
- Set the same logical level for the XLMTM, XIN0, and XIN1 signals. (Section 4.5 WR2/D4 bit and Section 4.4 WR1/D0 and D2 bits)
- Set WR7/D10 (using limit signals) bit of extension mode setting to 1.





[Operation]

As shown in the diagram on the right-hand side, the function moves the axis to the limit at high speed in the – direction in Step 2. When the – limit signal becomes active, the function stops operation by decelerating and advances to Step 2. The function exits control from the limit in the opposite direction by irregular operation ② of Step 2 and stops operation when Limit Signal Active is detected at low speed in the search direction. When the automatic home search starting position is within the limit (point A in the diagram on the right-hand side), operation starts from Step 2 without execution of Step 1.

[Parameter and mode setting]

```
WR0 ← 010Fh Write      ; Selects X axis
WR1 ← 0000h Write      ; Input signal logical setting: XIN0:Low active, XIN1:Low active (See 4.4)
WR2 ← 0004h Write      ; D4             0 −  Limit signal logic: Low active (see 4.5)
                       ; D2             1    Limit stop mode: Decelerating stop
                       ; Sets an extension mode
WR6 ← 5F00h Write      ; Writes an input signal filter mode in WR6 (See 2.8)
                       ; D15 ~ D13  010    Filter delay:512μsec
                       ; D8             1    XLMTM,XIN1,0 signals: Enables the filter

WR7 ← 054Fh Write      ; Writes an automatic home search mode in WR7
                       ; D15 ~ D13  000
                       ; D12            0
                       ; D11            0    Deviation counter clearing output:          Disable
                       ; D10            1    Using a limit signal as a home signal:      Enable
                       ; D9             0    Z-phase signal AND home signal:             Disable
                       ; D8             1    Clearing the logical/real position counter: Enable
                       ; D7             0    Step 4 driving direction:                   + direction
                       ; D6             1    Step 4:                                     Enable
                       ; D5             0    Step 3 search direction:
                       ; D4             0    Step 3:                                     Disable
                       ; D3             1    Step 2 search direction:                    − direction
                       ; D2             1    Step 2:                                     Enable
                       ; D1             1    Step 1 search direction:                    − direction
                       ; D0             1    Step 1:                                     Enable
WR0 ← 0160h Write      ; Writes an extension mode setting command in the X axis
```

```
WR6 ← 3500h Write      ; Range: 8,000,000 (Multiple: 10)
WR7 ← 000Ch Write
WR0 ← 0100h Write

WR6 ← 004Ch Write      ; Acceleration speed: 95,000 PPS/SEC
WR0 ← 0102h Write      ; 95000/125/10 = 76

WR6 ← 0064h Write      ; Initial speed: 1000 PPS
WR0 ← 0104h Write

WR6 ← 07D0h Write      ; Speed of Steps 1 and 4: 20000 PPS
WR0 ← 0105h Write

WR6 ← 0032h Write      ; Speed of Step 2: 500 PPS
WR0 ← 0161h Write

WR6 ← 0DACh Write      ; Offset driving pulse count: 3500
WR7 ← 0000h Write
WR0 ← 0106h Write

WR0 ← 0162h Write      ; Starts execution of automatic home search
```

[Notes on using limit signals]

- The same search direction must be applied for Steps 1 and 2. For Step 3 (Z-phase search), apply a direction opposite to the direction of Steps 1 and 2. For Step 4 also (offset driving), apply a direction opposite to Steps 1 and 2 and make sure that automatic home search operation stops at the position beyond the limit active section.
- When Step 3 operation is performed, the AND condition between a Z-phase signal and a home signal (IN1) cannot be applied. The extension mode setting bit WR7/D9 (SAND) must be set to 0.

2.6 Synchronous Action

Synchronous action of this IC performs actions such as starting and stopping of driving within each axis and between axes in IC and between IC and an external device. For instance, the following actions can be performed.

Example 1 Starts driving of the Z axis after the Y axis passes the position 15,000.

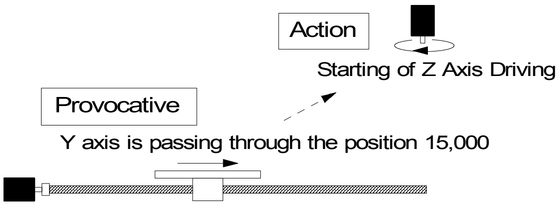


Fig. 2.40 Example of a synchronous action

Example 2 Stops driving of Y and Z axes after the X axes passes the position –320000.

Example 3 Saves position data of the X, Y, and Z axes when an input signal is set.

Normally, such synchronous actions can be performed by coding a program on the CPU side. However, this function is useful when no delay by CPU software execution time is allowed. The synchronous action of this IC is a function that executes a specified action immediately when a specified activation factor occurs. This linked action is performed without CPU intervention, achieving high-precision synchronization.

To perform a synchronous action, set a specified activation factor and a specified action in the synchronous mode registers in the IC.
Specify an activation factor (Provocative) register and other axis activation in the WR6 register, specify an action in the WR7 register, and write a synchronous action mode setting command 64h in the following WR6 register together with axis specification.
Ten activation factors are available as options for the WR6 register and fourteen actions are available as options for the WR7 register.

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
WR6	AXIS3	AXIS2	AXIS1	0	0	0	CMD	LPRD	IN3↓	IN3↑	D-END	D-STA	P≧C-	P<C-	P<C+	P≧C+
	Activation of Other Axes						Provocative									

The active factor and other axis activation bits can be enabled by setting them to 1 and disabled by setting them to 0.

D0	P≥C+	The logical/real position counter value exceeded the value of COMP + register value. (Use the WR2/D5(CMPSL) bit for selection of a logical position real/real position counter.)
D1	P<C+	The logical/real position counter value became less than the COMP + register value.
D2	P<C−	The logical/real position counter value became less than the COMP − register value.
D3	P≥C−	The logical/real position counter exceeded the COMP − register value.
D4	D-STA	Driving started.
D5	D-END	Driving terminated.
D6	IN3↑	The nIN3 signal rose from the Low to the High level.
D7	IN3↓	The nIN3 signal fell from the High to Low level.

D8 LPRD A logical position counter read command (10h) was written.
(Simultaneous read processing is enabled by setting LP save or EP save in the action of the own/another axis.)

D9 CMD A synchronous action activation command (65h) was written.

D15 ~ 13 AXIS3 ~ 1 Specify another axis to be driven by the activation factor of the own axis. 1: Enable

Own axis	D15(AXIS3)	D14(AXIS2)	D13(AXIS1)
X	U axis activation	Z axis activation	Y axis activation
Y	X axis activation	U axis activation	Z axis activation
Z	Y axis activation	X axis activation	U axis activation
U	Z axis activation	Y axis activation	X axis activation

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
WR7	INT	OUT	0	0	VLSET	OPSET	EPSET	LPSET	EPSAV	LPSAV	ISTOP	SSTOP	CDRV -	CDRV+	FDRV -	FDRV+

Action

Each bit for specification of an action is enabled by setting to 1 and disabled by setting to 0.

D0 FDRV+ Activates fixed pulse driving in the + direction.

D1 FDRV- Activates fixed pulse driving in the - direction.

D2 CDRV+ Activates continuous pulse driving in the + direction.

D3 CDRV- Activates continuous pulse driving in the - direction.

D4 SSTOP Stops driving in deceleration.

D5 ISTOP Stops driving immediately.

D6 LPSAV Saves the current logical position counter value (LP) in the synchronous buffer register (BR). LP→ BR

D7 EPSA Saves the current real position counter value (EP) in the synchronous buffer register (BR). EP→ BR

D8 LPSET Sets the values of the WR6 and WR7 registers in the logical position counter (LP). LP← WR6,7
See (3) in Section 2.6.3 Notes on Synchronous Action.

D9 EPSET Sets the values of the WR6 and WR7 registers in the real position counter (EP). EP← WR6,7
See (3) in Section 2.6.3 Notes on Synchronous Action.

D10 OPSET Sets the values of the WR6 and WR7 registers in the pulse count (P). P← WR6,7
See (3) in Section 2.6.3 Notes on Synchronous Action.

D11 VLSET Sets the value of the WR6 register in the drive speed (V). V← WR6
See (3) in Section 2.6.3 Notes on Synchronous Action.

D14 OUT Outputs synchronous pulses as external signals.
An external signal uses a nDCC signal. DCC Enable, logical setting, and a pulse width must be set in advance by using the extension mode setting command (60h). See Sections 2.5.2 and 6.16.

D15 INT Generates an interrupt signal (INTN).
The interrupt signal (INTN) becomes Low Active and the RR3/D9 (SYNC) bit of the axis to which the interrupt occurred indicates 1. When CPU reads the RR3 register of the axis to which the interrupt occurred, the bit of the RR3 register is cleared to 0 and the interrupt output signal is reset to Hi-Z.

At resetting, all the activation factors and actions are set to Disable.

Figure 2.41 shows the flow of synchronous action of the X axis in IC. If the activation factor that was set to Enable becomes active among the ten types of activation factors of the X axis, the action that is set to Enable starts immediately. When other axis activation is enabled, the actions of the other axis that are enabled are also executed simultaneously by the X axis activation factor.

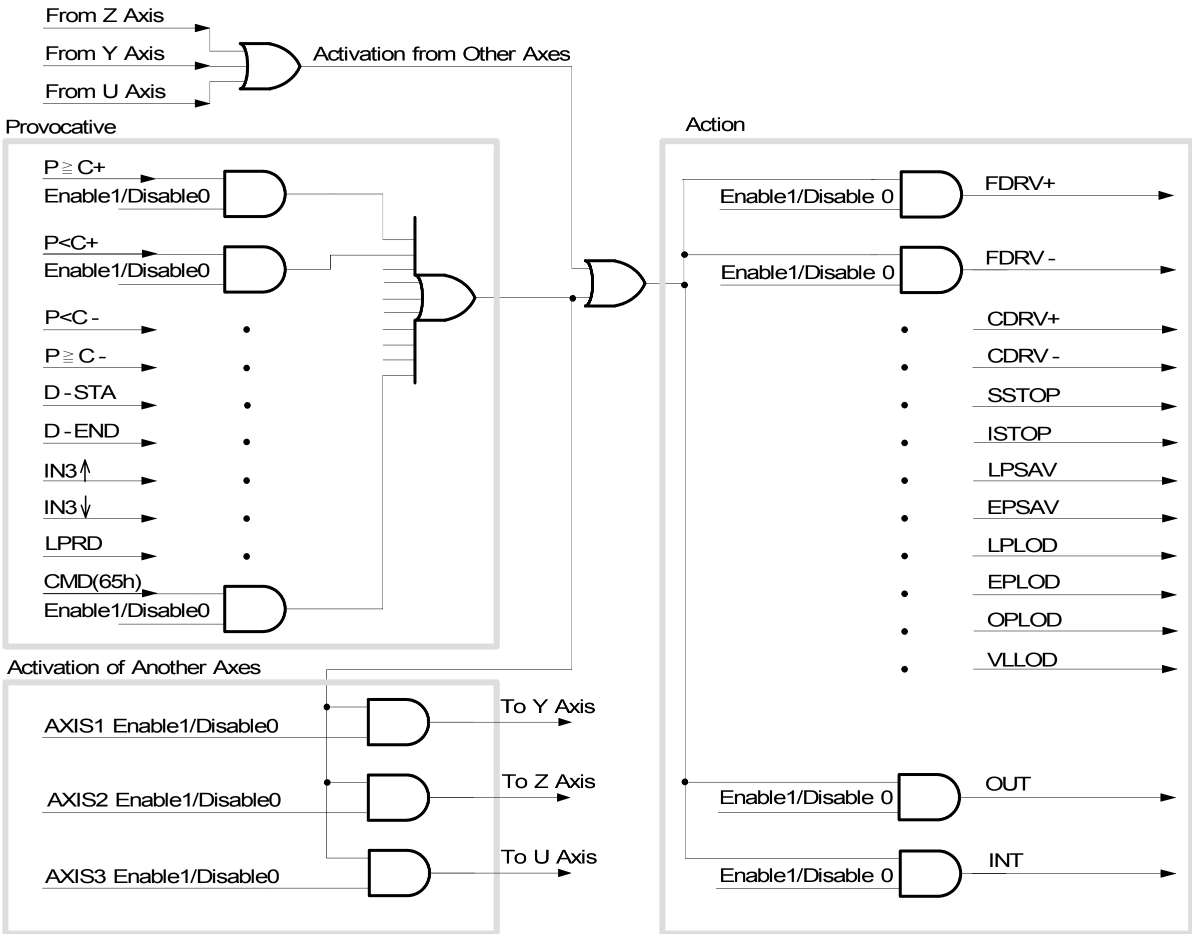


Fig. 2.41 Flow of Synchronous Action (X axis)

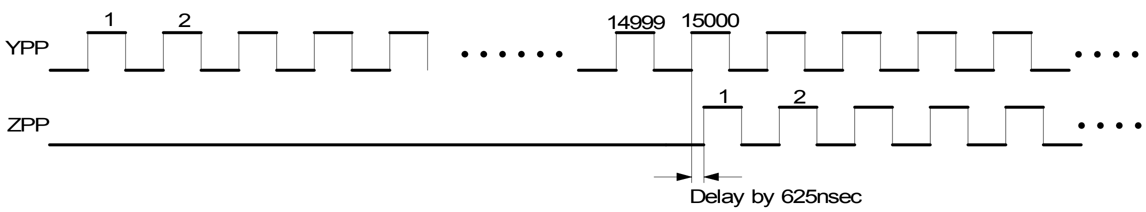
2.6.1 Example of Synchronous Action

■ Example 1　The Y axis is passing through the position 15,000. → Starts Z + direction fixed pulse drive.

Set the parameters and commands for this IC as follows.

WR6 ← 3500h WR7 ← 000Ch WR0 ← 0600h	Y, Z Axes Range: 800,000 (Multiple: 10)	
WR6 ← 0190h WR7 ← 0000h WR0 ← 0602h	Y, Z Axes Acceleration Rate: 400×125×10 = 500KPPS/SEC	
WR6 ← 0032h WR7 ← 0000h WR0 ← 0604h	Y, Z Axes Initial Speed: 50×10 = 500PPS	
WR6 ← 0BB8h WR7 ← 0000h WR0 ← 0605h	Y, Z Axes Drive Speed: 3000×10 = 30KPPS	
WR6 ← C350h WR7 ← 0000h WR0 ← 0206h	Y Axis Output Pulse Number: 50,000	
WR6 ← 2710h WR7 ← 0000h WR0 ← 0406h	Z Axis Output Pulse Number: 10,000	
WR6 ← 3A98h WR7 ← 0 WR0 ← 020Bh	Set 15000 in Y Axis COMP+	
WR6 ← 0 WR7 ← 0 WR0 ← 0609h	Clear Y, Z Axes Logical Counter (LP)	
WR6 ← 2001h WR7 ← 0000h WR0 ← 0264h	Provocative: P≧C+, Activation of Other Axes: Z Action of Own Axis: None	⇦ Set Y Axis Synchronous Action Mode
WR6 ← 0000h WR7 ← 0001h WR0 ← 0464h	Action of Own Axis: + Direction Constant Drive	⇦ Set Z Axis Synchronous Action Mode
WR0 ← 0220h	Start of Y Axis + Direction Constant Drive	

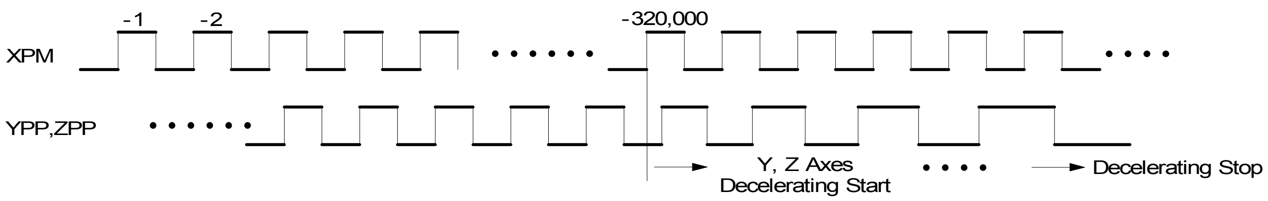
Z axis fixed pulse driving in the + direction starts when the Y axis passes through 15000 pulses after the start of Y axis driving. The delay time from the rise of the 15000th pulse of the Y axis to the rise of the 1st pulse of the Z axis is 5SCLK (625nsec CLK=16MHz).



■ Example 2 The X axis passed through the position –320000. → Stops driving of the X and Y axes.

WR6 ← 3500h WR7 ← 000Ch WR0 ← 0700h	X, Y, Z Axes Range: 800,000 (Multiple: 10)	
WR6 ← 0190h WR7 ← 0000h WR0 ← 0702h	X, Y, Z Axes Acceleration Rate: 400 x 125 x 10 = 500KPPS/SEC	
WR6 ← 0032h WR7 ← 0000h WR0 ← 0704h	X, Y, Z Axes Initial Speed: 50×10 = 500PPS	
WR6 ← 0BB8h WR7 ← 0000h WR0 ← 0705h	X, Y, Z Axes Drive Speed: 3000×10 = 30KPPS	
WR6 ← A120h WR7 ← 0007h WR0 ← 0106h	X Axis Output Pulse Number: 500,000	
WR6 ← 1E00h WR7 ← FFFBh WR0 ← 010Ch	Set -320,000 in X Axis COMP-	
WR6 ← 0 WR7 ← 0 WR0 ← 0109h	Clear X Axis Logical Counter (LP)	
WR6 ← 6004h WR7 ← 0000h WR0 ← 0164h	Provocative: P < C-, Activation of Other Axes: Y, Z Action of Own Axis: None	⇦ Set X Axis Synchronous Action Mode
WR6 ← 0000h WR7 ← 0010h WR0 ← 0664h	Action of Own Axis: Decelerating Stop	⇦ Set Y, Z Axes Synchronous Action Mode
WR0 ← 0622h	Start of Y, Z Axes + Direction Continuous Drive	
WR0 ← 0121h	Start of X Axis - Direction Constant Drive	

In this example, the X axis is started with fixed pulse driving in the – direction after the starting of the Y and Z axes with continuous pulse driving. If the X axis passes through the –320,000th pulse, the X and Y axes stop by decelerating.



When instant stop is specified for the synchronous actions of the Y and Z axes and the X axis passes through the –320,000th pulse, the Y and Z axes stop actions instantly.

■ Example 3 The input signal (XIN3) fell. → Saves position data of the X, Y, and Z axes.

In this example, logical position counter values of the three axes are saved in the buffer registers (BR) of the axes individually at the falling of the XIN3 signal after the start of driving of the X, Y, and Z axes. At the same time, the function sets an interrupt output signal (INTN) to Low Active for the X axis and notifies the effect to the CPU. The CPU reads the buffer contents of each axis after checking that the interrupt is caused by a synchronous action.

WR6 ← 3500h WR7 ← 000Ch WR0 ← 0700h	X, Y, Z Axes Range: 800,000 (Multiple: 10)	
WR6 ← 0190h WR7 ← 0000h WR0 ← 0702h	X, Y, Z Axes Acceleration Rate: 400×125×10 = 500KPPS/SEC	
WR6 ← 0032h WR7 ← 0000h WR0 ← 0704h	X, Y, Z Axes Initial Speed: 50×10 = 500PPS	
WR6 ← 0BB8h WR7 ← 0000h WR0 ← 0705h	X, Y, Z Axes Drive Speed: 3000×10 = 30KPPS	
WR6 ← 0 WR7 ← 0 WR0 ← 0709h	Clear Z, Y, Z Axes Logical Counter (LP)	
WR6 ← 6040h WR7 ← 8040h WR0 ← 0164h	Provocative: XIN3↓ , Activation of Other Axes: Y, Z Action of Own Axis: Saving LP or Occurrence of Interrupt	⇐ Set X Axis Synchronous Action Mode
WR6 ← 0000h WR7 ← 0040h WR0 ← 0664h	Action of Own Axis: Saving LP	⇐ Set Y, Z Axes Synchronous Action Mode
WR0 ← 0722h	Start of Z, Y, Z Axes + Direction Continuous Drive	

↓

The XIN3 input signal fell; interrupt occurred

↓

Check the interrupt by the synchronous action
(Read the XRR3 register and check D9(SYNC)=1)

↓

WR0 ← 0114h RR6 → Read RR7 → Read	X Axis Buffer - Read
WR0 ← 0214h RR6 → Read RR7 → Read	Y Axis Buffer - Read
WR0 ← 0414h RR6 → Read RR7 → Read	Z Axis Buffer - Read

■ Example 4　Continuous actions of fixed pulse driving

By using the synchronous action function, fixed pulse driving can be performed continuously by starting the next driving immediately following termination of driving. In the following example, driving of –5,000 is performed immediately after driving of +15,000 is terminated.

WR6 ← 3500h WR7 ← 000Ch WR0 ← 0100h	X Axis Range: 800,000 (Multiple: 10)	
WR6 ← 0190h WR7 ← 0000h WR0 ← 0102h	X Axis Acceleration Rate: 400×125×10 = 500KPPS/SEC	
WR6 ← 0032h WR7 ← 0000h WR0 ← 0104h	X Axis Initial Speed: 50×10 = 500PPS	
WR6 ← 0BB8h WR7 ← 0000h WR0 ← 0105h	X Axis Drive Speed: 3000×10 = 30KPPS	
WR6 ← 0 WR7 ← 0 WR0 ← 0109h	Clear X Axis Logical Counter (LP)	
WR6 ← 3A98h WR7 ← 0000h WR0 ← 0106h	X Axis Output Pulse Number: 15,000	
WR6 ← 0020h WR7 ← 8402h WR0 ← 0164h	Provocative: Termination of driving Action of Own Axis: P← WR6, WR7 -Direction Fixed Drive Occurrence of interrupt	⇦ Set X Axis Synchronous Action Mode
WR6 ← 1388h WR7 ← 0000h	Output Pulse Number of he Next Drive:5,000	
WR0 ← 0120h	Start of X Axis + Direction	

↓

Terminate of X Axis 15,000 + Direction Fixed Drive

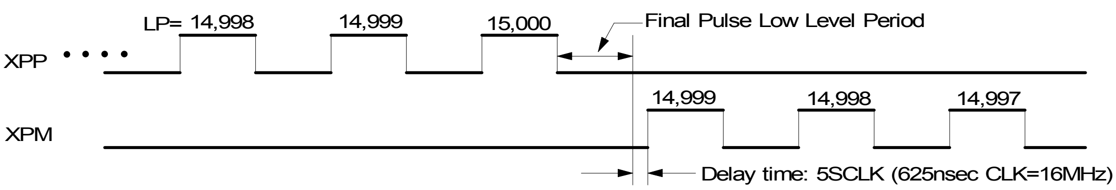
↓

Start of -5,000 - Direction Fixed Drive, Occurrence of interrupt

↓

XRR3← Read	Check the Interrupt by the Synchronous Action Check D9 (SYNC)=1	⇦ Processing in the Interrupt Routine
WR6 ← 0000h WR7 ← 0000h WR0 ← 0164h	Release of Synchronous Action Mode	

The delay time from termination of driving of +15,000 to the start of driving of -5,000 is 5SCLK (625nsec CLK=16MHz).



In the above example, an interrupt is generated as soon as driving of –5,000 starts, canceling the synchronous action mode within the interrupt processing. Without this cancellation, fixed pulse driving in the – direction is performed endlessly.

Even if driving is suspended during driving of the first +15,000 due to the limit in the + direction (LMTP) or emergency stop (EMGN), driving of the next –5,000 is executed. When this becomes a problem in the system operation, the synchronous action function cannot be used.

2.6.2 Synchronous Action Delay Time

A synchronous action delay is a total of a delay from the occurrence of an activation factor and a delay up to the action as shown in the following tables.

■ Delay from the occurrence of an activation factor 1SCLK=125nsec (CLK=16MHz)

Activation factor	Definition of the start of delay		Delay time (SCLK)		
			Min.	Standard	Max.
P≥C+ P<C+ P<C− P≥C−	P=LP	From ↑ of the driving pulse when the LP value satisfies the comparison condition with the CMP+/− register value		1	
	P=EP (A/B phase input)	From↑↓ of the nECA/B input signal when the EP value satisfies the comparison condition with the CMP+/− register value	3		4
D-STA	From ↓ of the WRN signal at writing of a driving command		1		2
D-END	From Low level termination of the last driving pulse			1	
IN3↑	Form ↑ of the nIN3 signal (when the built-in filter is disabled)		0		1
IN3↓	From ↓ of the nIN3 signal (with the built-in filter is disabled)		0		1
LPRD	From ↓ of the WRN signal at writing of the LP read command (10h)		0		1
CMD	From ↓ of the WRN signal at the writing of the synchronous action activation command (65h)		0		1

■ Delay up to an action 1SCLK=125nsec (CLK=16MHz)

Action	Definition of the end of delay	Delay time(SCLK)
FDRV+ FDRV− CDRV+ CDRV−	Up to ↑ of the 1st driving pulse	4
SSTOP	Up to the start of deceleration	*1
ISTOP	Up to the stopping of driving	*1
LPSAV EPSAV	Up to the saving of the LP and EP values in the BR (buffer)	1
LPSET EPSET OPSET VLSET	Up to the saving of the values of WR6 and WR7 in LP, EP, P, and V	1
OUT	Up to ↑ of the nDCC output signal (positive logic)	1
INT	Up to ↓ of the INTN signal	1

*1:Time spent up to termination of one driving pulse that is currently output

For instance, the delay time from ↑ of the IN3 input signal to the saving of the logical position counter value (LP) in the synchronous buffer register (BR) is a total of the IN3 ↑ delay time (0 to 1SCLK) and the LPSAV delay time (1SCLK). The range is from 1SCLK to 2SCLK. When CLK=16MHz, the range is from 125nsec to 250nsec.

2.6.3 Notes on Synchronous Action

(1) After a required synchronous action is activated by specifying an interrupt in the action concurrently, release the synchronous action specification by re-issuing the synchronous action mode setting command 64h. If the action is not released, the action may be performed unexpectedly.

(2) By using the synchronous action function, an endless loop can be formed as shown below.

→ Start Driving of X Axis → Stop → Start Driving of Y Axis → Stop

To stop this endless loop, the activation factor that is enabled and each bit of the operation must be disabled by re-issuing the synchronous action mode setting command 64h. The loop will not be released simply by issuing an instant stop command or the decelerating stop command for the axis that is driving and operation continues.

(3) For action specification D8 (LPSET), D9 (EPSET), D10 (OPSET), and D11 (VLSET), data must be written to WR6 and WR7 before a synchronous action is activated. If continuous synchronous actions coincide with activation of a synchronous action due to the delay in the writing data in WR6 and WR7 from CPU, undefined data may be fetched. Write data to WR6 and WR7 when non-activation of synchronous actions is guaranteed.

(4) When a driving activation action occurs during driving, the action is ignored. When a decelerating stop action or an instant stop action occurs while the axis is inactive, the action is ignored.

2.7 Interrupt

The interrupt is generated from X, Y, Z, or U axis, bit pattern interpolation or continuous interpolation.
There is only one interrupt signal, INTN (33), to the host CPU. So, the signal will be OR calculated, then output, as shown in Fig. 2.42.

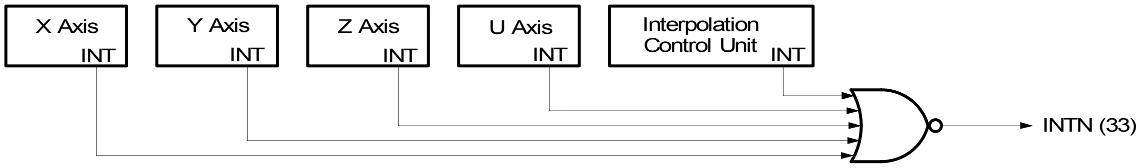


Fig. 2.42 Interrupt Signal Path in IC

Every interrupt can be enabled or disabled individually. When resetting, all interrupt signals are disabled.

■ Interrupt of X, Y, Z, and U Axes

The following table shows the interrupt factors generated by X, Y, Z, and U axes.

Enable / Disable nWR1 Register	Status nRR3 Register	The Factors of Interrupt Happening
D8 (PULSE)	D0 (PULSE)	when one pulse outputs... (The interrupt will be generated at the rising edge of pulse output for + direction driving.)
D9 (P≥C−)	D1 (P≥C−)	once the value of logical / real position counter is larger than or equal to the value of COMP− register (CM)...
D10 (P<C−)	D2 (P<C−)	once the value of logical/real position counter is smaller than the value of COMP− register (CM)...
D11 (P<C+)	D3 (P<C+)	once the value of logical / real position counter is larger than the value of COMP+ register (CM)…
D12 (P≥C+)	D4 (P≥C+)	once the value of logical / real position counter is smaller than or equal to the value of COMP+ register (CM)...
D13 (C-END)	D5 (C-END)	in the acceleration / deceleration driving, when the driving changes from the constant speed region into the decelerating region...
D14 (C-STA)	D6 (C-STA)	in the acceleration / deceleration driving, when the driving changes from the accelerating region into the constant speed region…
D15 (D-END)	D7 (D-END)	when the driving is finished...

Each factor of interrupt can be masked by setting levels in nWR1 register bits: 1- enable and 0 - disable. When interrupt is generated during the driving, and if the interrupt is generated, each bit in nRR3 will be set to 1; INTN will be on the Low level. After the nRR3 status has been read from the host CPU, nRR3 will be cleared from 1 to 0, and INTN will return to the High-Z level.
[Note] Since the entire contents of the 8-bit data bus are cleared by reading the RR3L register, read RR3H before reading RR3L register when using end of automatic home search D8 (HMEND) or synchronous action activation D9 (SYNC).

The following automatic home search end and synchronous action activation interrupts are added. See the related section for the details.

Setting permission/ prohibition	Occurrence confirmation nRR3 register	Interrupt factor
Setting the extension mode command (60h) WR6/D5 (HMINT)	D8 (H-END)	Automatic home search is terminated.

Setting permission/ prohibition	Occurrence confirmation nRR3 register	Interrupt factor
Synchronous action specification command (64h) WR7/D15 (INT)	D9 (SYNC)	A synchronous action is activated due to the specified activation factor.

■ Interrupt from Interpolations

Enable / Disable WR5 Register	Status Check RR0 Register	The Factors of Interrupt Happening *Interrupt Clearing
D14 (CIINT)	D9 (CNEXT)	in continuous interpolation, when MCX314As is available for the interpolation data of next node... *after next interpolation command is written, the interrupt will be cleared.
D15 (BPINT)	D14,13 (BPS1,0)	In bit pattern interpolation, when the value of stack connector (SC) is changed from 2 to 1, and the stack is available for next BP command writing... *after a BP command for the stack is written, the interrupt will be cleared.

When an interrupt is generated during interpolations, this interrupt can be cleared by writing the interrupt clear command (3Dh) INTN will return to the High-Z level automatically once the interpolation is finished.

See Bit pattern /Continuous Interpolation section for executing interpolation by interrupt.

2.8 Input Signal Filter

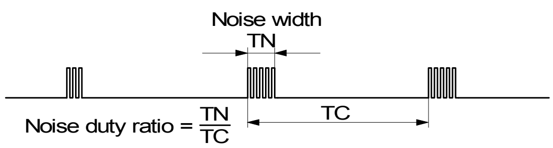
This IC is equipped with an integral type filter in the input stage of each input signal. Figure 2.4 shows the filter configuration of each input signal of the X axis. The same circuit is provided to the X, Z, and U axes also. The time constant of the filter is determined by the T oscillation circuit in the diagram. One time constant can be selected from eight time constants using the bits D15 to D13 (FL2 to FL0) of the WR6 register of the extension mode setting command (06h). Using the bits D12 to D8 (FE4 to 0) of the WE6 register, it is possible to set whether the filter function is enabled or the signal is passed through for a number of input signals. At reset, all the bits in the extension mode are cleared to 0 so that the filter function is disabled for all the input signals and the signals pass.

Select a filter time constant from eight stages as shown in the table below. When a time constant is increased, the removable maximum noise width increases, however, the signal delay time also increases. Therefore, set an appropriate value. Normally, set 2 or 3 for FL2 to FL0.

FL2 ~ 0	Removable maximum. noise width [*1]	Input signal delay time
0	1.75μSEC	2μSEC
1	224μSEC	256μSEC
2	448μSEC	512μSEC
3	896μSEC	1.024mSEC
4	1.792mSEC	2.048mSEC
5	3.584mSEC	4.096mSEC
6	7.168mSEC	8.192mSEC
7	14.336mSEC	16.384mSEC

CLK=16MHz

*1: Noise width



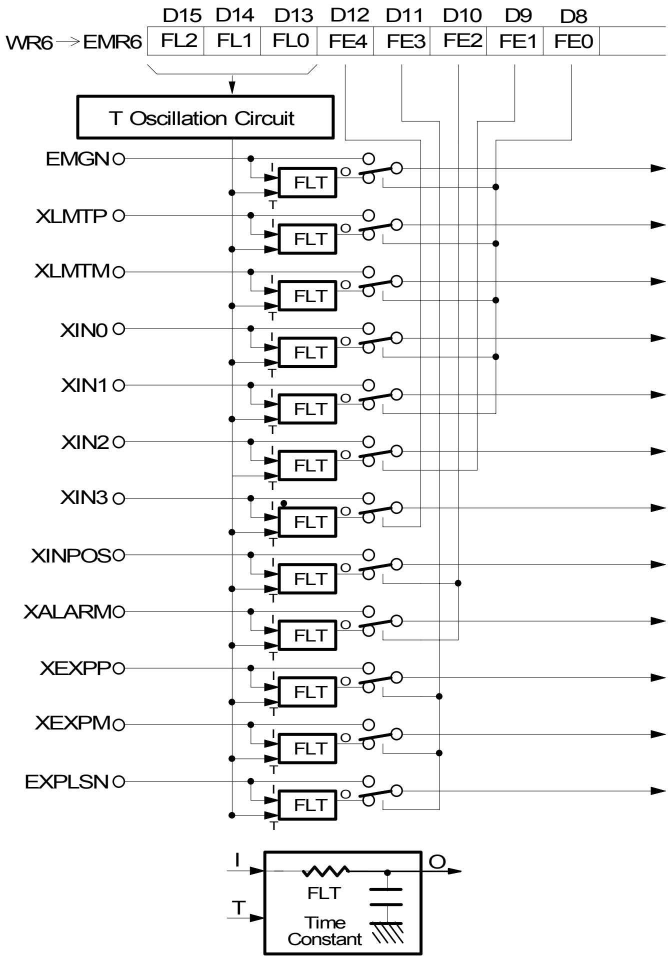


Fig. 2.43 Concept of Input Signal Filter Circuit

As the condition, the noise duty ratio (time ratio under which noise is generated in the signal) must be 1/4 or less.

In bits D12 to D8 (FE4 to FE0) of the WR6 register of an extension mode setting command (60h), set whether the filter function of each input signal is enabled or signals are passed through as shown below. When 1 is set in each bit, the filter function of the signal is enabled.

Specification bit	Filter Enable signal
WR6/D8 (FE0)	EMGN[*2], nLMTP, nLMTM, nIN0, nIN1
WR6/D9 (FE1)	nIN2
WR6/D10 (FE2)	nINPOS, nALARM
WR6/D11 (FE3)	nEXPP, nEXPM, EXPLSN[*3]
WR6/D12 (FE4)	nIN3

*2: The EMGN signal is set using the D8 bit of the WR6 register of the X axis.
*3: The EXPLSN signal is set using the D11 bit of the WR6 register of the X axis.

■Example of setting input signal filters

Set a 512μsec delay filter for EMGN and input signals LMTP, LMTM, IN0, IN1, EXPP, and EXPM of the X and Y axes and specify “through” for other input signals of the X and Y axes.

Set a 2msec delay filter for input signals LMTP, LMTM, IN0, IN1, EXPP, and EXPM of the Z and U axes and specify “through” for other input signals of the Z and U axes.

```
                        ; Set an extension mode for the X and Y axes.
WR6 ← 4900h Write       ; Writes an input signal filter mode in WR6.
                        ; D15 ~ D13  010   Filter delay:512μsec
                        ; D12          0   IN3 signal: Disables the filter (through)
                        ; D11          1   EXPP, EXPM, and EXPLS signals: Enable filter
                        ; D10          0   INPOS and ALARM signals: Disable the filter (through)
                        ; D9           0   IN2 signal: Disables the filter (through)
                        ; D8           1   EMGN, LMTP, LMTM, IN1, and 0 signals: Enables filter
                        ; D7 ~ D0          Mode other than the built-in filter function
                                           (Set an appropriate value. See Section 6.16.)

WR7 ← 0000h Write       ; Set an appropriate value when performing automatic home search. (See Section 2.5)

WR0 ← 0360h Write       ; Writes an extension mode setting command in the X and Y axes.

                        ; Sets an extension mode for the X and Y axes.
WR6 ← 8900h Write       ; Writes an input signal filter mode in WR6.
                        ; D15 ~ D13  100   Filter delay:2msec
                        ; D12          0   IN3 signal: Disables the filter (through)
                        ; D11          1   EXPP and EXPM signals: Enables the filter
                        ; D10          0   INPOS and ALARM signal: Disables the filter (through)
                        ; D9           0   IN2 signal: Disables the filter (through)
                        ; D8           1   LMTP, LMTM, IN1, and 0 signals: Enables the filter
                        ; D7 ~ D0          Mode other than the built-in filter function
                                           (set an appropriate value. See Section 6.16.)

WR7 ← 0000h Write       ; Set an appropriate value when performing automatic home search. (See Section 2.5)

WR0 ← 0C60h Write       ; Writes an extension mode setting command in the Z and U axes.
```

2.9 Other Functions

2.9.1 Driving By External Pulses

Fixed pulse driving and continuous pulse driving can be controlled by either commands or external signals, which can reduce the load of host CPU. By inputting an encoder 2-phase signal of a manual pulsar, jog feed of each axis is enabled.

Each axis has two input signals, nEXPP and nEXPM. In fixed pulse drive mode and a continuous pulse drive mode, the nEXPP signal triggers driving in the + direction and the nEXPM signal in the – direction. nEXPP controls + direction pulse output, and nEXPM controls – direction command. D3 and D4 bits of register WR3 are for the setting in driving. The user should preset the parameters and commands. The default level of nEXPP and nEXPM is normally set on Hi. In manual pulsar mode, the A-phase signal is connected to nEXPP input and the B-phase signal to nEXPM input.

■ Fixed Pulse Driving Mode

Set bits D4 and D3 of register WR3 to 1 and 0 respectively, and set all the parameters of fixed pulse driving. Once nEXPP is falling down to the Low level (↓), the + direction fixed pulse driving will start; once nEXPM is raising to the Hi level (↓), the − direction fixed pulse driving will start. The width of Low level signal must be larger than 4 CLK-cycle. Before this driving is finished, a new Hi-to-Low level falling down of the signal is invalid.

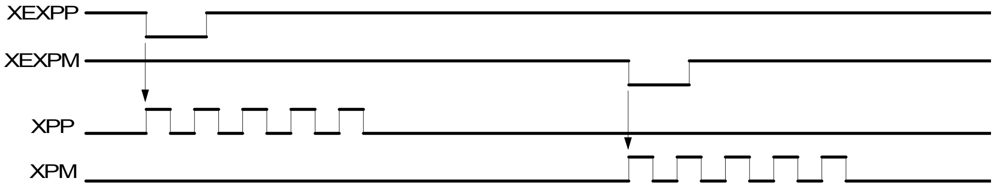


Fig. 2.44 Example of The Constant Pulse Driving by External Signal

■ Continuous Pulse Driving Mode

Set bits D4 and D3 of WR3 register to be 0 and 1 respectively, and set all the parameters of continuous driving. Once nEXPP is falling down to the Low level (↓), the + direction continuous driving will start; once nEXPM is raising to the Low level (↓), the − direction continuous driving will start. When nEXPP and nEXPM returns to the Hi level from the Low level, the decelerating stop will be performed in trapezoidal driving, and the sudden stop in constant speed driving.

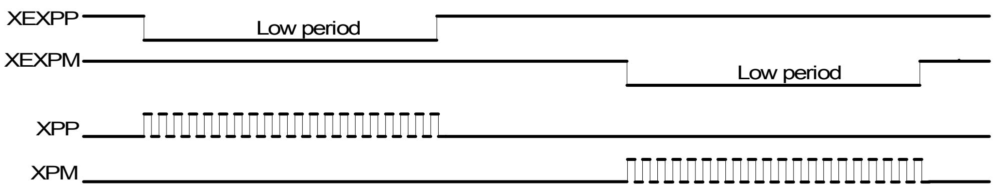


Fig. 2.45 Example of Continuous Driving by External Signal

■ Manual pulsar mode

Set the bits D4 and D3 of the WR3 register to 1 and set the necessary speed parameter for driving and the output pulse number. Connect the A-phase signal of the encoder to nEXPP input and the B-phase signal to nEXPM input. The − fixed pulse driving is activated when the nEXPM signal is at a Low level and the nEXPP signal is at the rising edge. When the output pulse number is set to 1, one drive pulse is output at each of the rising edge and falling edge of the nEXPP signal. If the output pulse number is set to P, the P number of drive pulses is output.

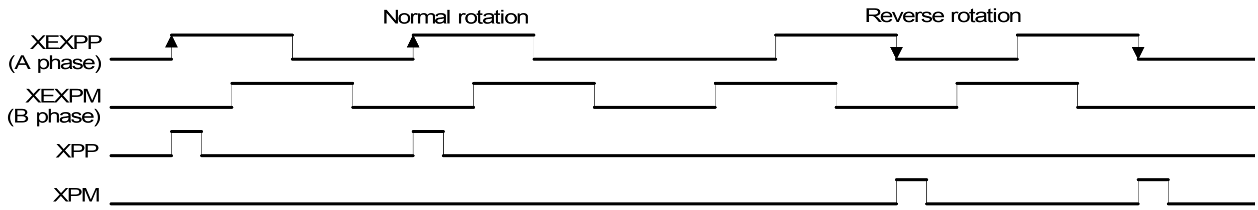


Fig. 2.46 Example Output Pulse 1Driving by A Manual Pulsar

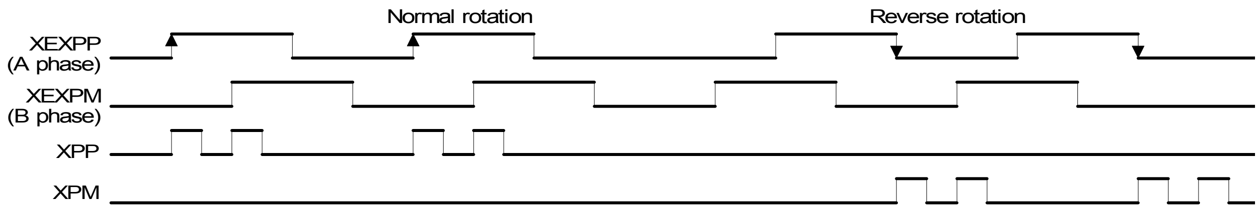


Fig. 2.47 Example of Output Pulse 2 Driving by A Manual Pulsar

Set the speed parameter in the following conditions to complete output of all the P number of drive pulses with a period from the rising edge/falling edge of the nEXPP signal to the next rising edge/falling edge.

$V \geq F \times P \times 2$

V : Drive speed (pps)
P : Output pulses
F : Frequency (Hz) at the maximum speed of the manual pulsar encoder

For instance, under the condition where the maximum frequency of the manual pulsar is F=500Hz and the output pulse is P=1, the drive speed must be V=1000pps or greater. Since acceleration/deceleration driving is not applied, set the initial speed SV to the same value as the drive speed. However, when a stepping motor is used for driving, the drive speed must not exceed the automatic activation frequency of the motor.

2.9.2 Pulse Output Type Selection

There are two types of pulse output--independent 2-pulse type: when the driving is in + direction, the pulse output is from nPP/PLS; when the driving is in − direction, the pulse output is from nPM/DIR; 1-pulse 1- direction type: nPP/PLS is for pulse outputting, and nPM/DIR is for direction signal outputting.

(pulse / direction is set on the positive logical level)

Pulse Output Type	Drive Direction	Pulse Output Waveform	
		nPP/PLS Signal	nPM/DIR Signal
Independent 2-pulse	+Direction	.....	Low level
	−Direction	Low level	.....
1-pulse 1-direction	+Direction	.....	Low level
	−Direction	.....	Hi level

Bit D6 (PLSMD) of register WR2 is used for the selection of pulse output type.

Additionally, bits D7 (PLS-L) and D8 (DIR-L) of register WR2 can be used for pulse outputting, direction and logical level setting.

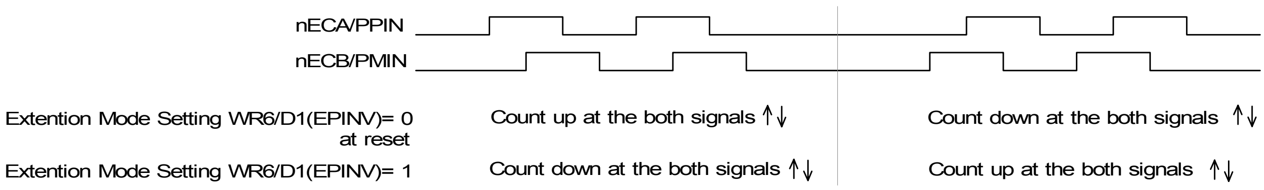
[Note] Please refer to Chapter 14.2 and 14.3 for the pulse signal (nPLS) and direction signal (nDIR) in 1- pulse 1-direction pulse outputting.

2.9.3 Pulse Input Type Selection

For real position counter, A/B quadrature pulse type and Up / Down pulse type can be selected for pulse input.

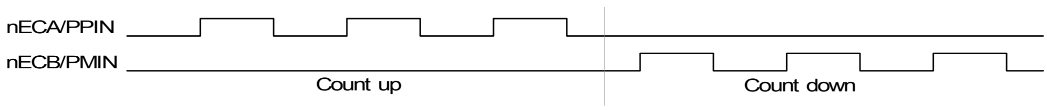
■ A/B quadrature pulse input mode

A A/B quadrature pulse input mode can be set by setting the D9 (PINMD) bit of the WR2 register to 0. In this mode, when A phase is advancing with positive logical pulses, the count is incremented and when the B-phase is advancing, the count is decremented. The count is incremented and decremented at the rising edge and falling edge of both signals. When the real position counter up/down reverse bit (WR6/D1) is set to 1 in extension mode setting, the up/down operation of the real counter is reversed. (See Section 6.16.)
In A/B quadrature pulse input mode, the input pulses can be divided into 1/2 or 1/4.



■ Up/down pulse input mode

By setting the D9 (PINMD) bit of the WR2 register to 1, a counter up/down pulse input mode can be set. nECA/PPIN is count up input and nECB/PMIN is count down input. The counter counts at the rising edge of the positive pulse.



Use the D9 (PINMD) bit of the WR2 register for selecting a pulse input mode and the D11 and D10 (PIND1,0) bits to set the division ratio of encoder 2-phase pulse input.
[Note] Time specification is applied to the pulse width and pulse cycle of input pulses. See Section 13.2.5 Input Pulse of Chapter 13.

2.9.4 Hardware Limit Signals

Hardware limit signals, nLMTP and nLMTM, are used for stopping the pulse output if the limit sensors of + and − directions are triggered.

When the limit signal and also the logical level are active, the command of sudden stop or decelerating stop can be set by bits D3 and D4 (HLMT+, HLMT−), and D2 (LMTMD) of register WR2.

2.9.5 Interface to Servo Motor Drivers

Enable / Disable and logical levels of the input signals for connecting servo motor drivers such as nINPOS (in-position input signal) and nALARM (alarm input signal) can be set by D15~12 bits of register WR2. nINPOS input signal responds to the in-position signal of servo motor driver.

When "enable" is set, and when the driving is finished, nINPOS will wait for the "active". Then, the n-DRV bit of main status register PRO will return to 0.

nALARM input signal receives the alarm signal from servo motor drivers. When "enable" is set, nALARM signal will be monitored, and the D4 (alarm) bit of RR2 register is 1 when nALARM is active. The sudden stop will occur in the driving when this signal is active.

These input signals from servo motor drivers can be read by RR4 and RR5 registers.

A deviation counter clear signal (nDCC) is available as a servo motor driver output signal. See Sections 2.4.2 and 2.4.3.

2.9.6 Emergency Stop

Signal EMGN is able to perform the emergency stop function for all of the 4 axes during the driving. Normally, this signal is kept on the Hi level. When it is falling to the Low level, all axes will stop immediately, and the D5 (EMG) bit of register RR2 (each axis) becomes 1. Please be noted that there is no way to select the logical level of EMGN signal.

Please check the following methods to perform the emergency stop function from the host CPU.

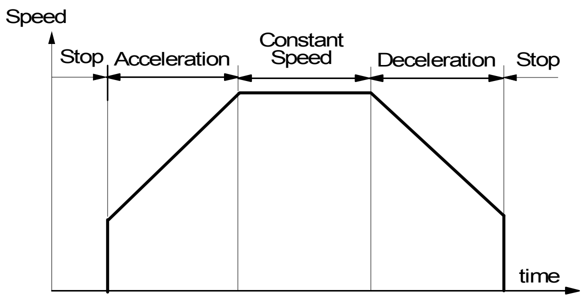
a. Execute the sudden stop commend for all of the 4 axes at the same time…
   Appoint all of the 4 axes, then write the sudden stop command (27h) to register WR0.

b. Reset software limit…
   Write 800h to register WR0 to reset software limit.

2.9.7 Status Output

nDRIVE output signals and bits D3~0 (n-DRV) of register RR0 can be used for drive / stop status output of each axis.

The driving status of acceleration / constant speed / deceleration will be output to bits D2 (ASND), D3 (CNST) and D4 (DSDD), and also the signals nOUT6 / ASND and nOUT7 / DSND will show the levels. However, these output signals and general purpose output signals share the same terminal, D7 (OUTSL) bit of register WR3 should be set 1 for drive status output.



Drive Status	Status Register				Output Signal		
	RR0/n-DRV	nRR1/ASND	nRR1/CNST	nRR1/DSND	nDRIVE/DCC	nOUT6/ASND	nOUT7/DSND
Stop	0	0	0	0	Low	Low	Low
Acceleration	1	1	0	0	Hi	Hi	Low
Constant Speed	1	0	1	0	Hi	Low	Low
Deceleration	1	0	0	1	Hi	Low	Hi

Moreover, in S-curve accelerating/decelerating driving, the state of acceleration / constant speed / deceleration will be also shown to bits D5 (AASND), D6 (ACNST), and D7 (ADSND) of register RR1.

2.9.8 General Purpose Output Signal

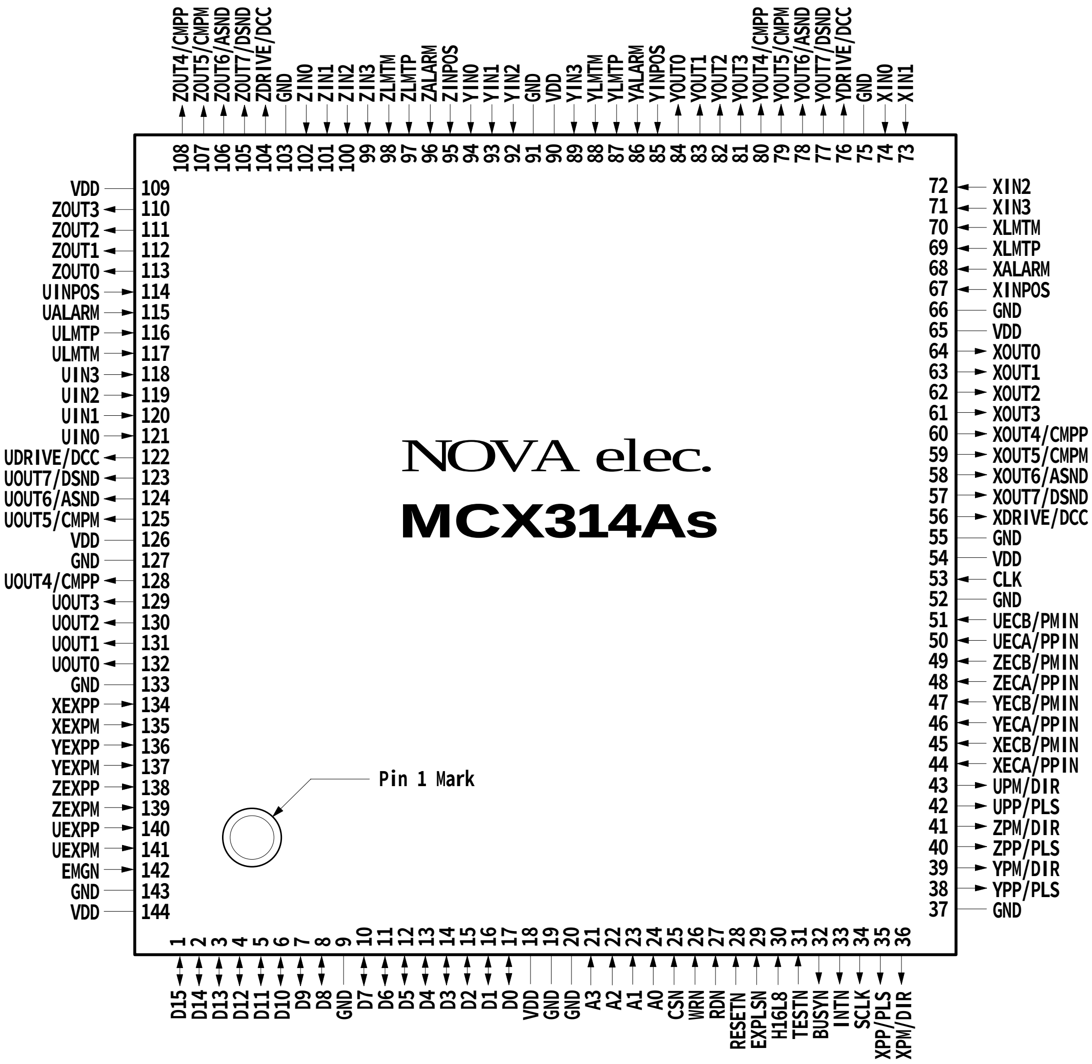
In MCX314As, there are 8 general purpose output pins, nOUT3~0 & nOUT7~4, for each axis. However, during the outputting, nOUT7~4 cannot be used cause they share the same terminals with the position comparison output and drive status output.

NOUT3~0 can be output when the output levels of register WR4 have been set. If the user wants to use nOUT7~4 signals, D7 (OUTSL) of register WR3 should be set in the “general purpose output mode” , then the output levels of D11~8 (OUT7~4) of register WR3 can be set for outputting.

It is possible to use the general purpose output signal for motor driver current-OFF, deviation counting clear and alarm reset…

When resetting, each bit of WR4 and nWR3 registers will be cleared, then, their output levels will be kept.

3. Pin Assignments and Signal Description



See Chapter 15 for the 144-pin LQFP package: 20×20mm, external package: 22×22mm, pin pitch: 0.5mm, pin coating: Sn-Bi (Tin bismuth)

■ Signal Description

Signals X○○○, Y○○○, Z○○○, and U○○○ are input / output signals for X, Y, Z, and U axes, where n stands for X, Y, Z, and U. If the signals are named ○○○N, they are negative-active or low-active.
See the end of this chapter for description of input/output circuits. An integral filter circuit is available in the internal input column of this IC for the input signals with – F – symbol. See Section 2.8 for the filter function.

Signal Name	Pin No.	Input/Output	Signal Description
CLK	53	Input A	Clock: clock signal for internal synchronous loop of MCX314As The standard frequency is 16 MHz. This signal is for drive speed, acceleration / deceleration and jerk. If the frequency setting is not 16 MHz, the setting values of speed and acceleration / deceleration are different.
D15~D0	1~8,10~17	Bi-directional A	DATA BUS: 3-state bi-direction 16-bit data bus When CSN=Low and RDN=Low, these signals are for outputting. Otherwise, they are high impedance inputs. If 8-bit data bus is used, D15~D8 can not be used, and D15~D8 should be pull up to + 5V through high impedance (about 100 kΩ).
A3~A0	21,22,23,24	Input A	Address: address signal for host CPU to access the write / read registers A3 is used only when the 8-bit data bus is used.
CSN	25	Input A	Chip Select: input signal for selecting I/O device for MCX314As Set CSN to the Low level for data reading and writing.
WRN	26	Input A	Write Strobe: its level is Low while data is being written to MCX314As. When WRN is Low, CSN and A3-A0 must be assured. When WRN is up (↑), the data will be latched in the write register, and while WRN is up (↑), the levels of D15~D0 should be assured.
RDN	27	Input A	Read Strobe: its level is Low while data is being read from MCX314As. Only when CSN is on the low level, the selected read register data from A3~A0 address signals can be output from the data bus.
RESETN	28	Input A	Reset: reset (return to the initial setting) signal for MCX314As. Setting RESETN to Low for more than 4 CLK cycles will reset MCX314As. The RESETN setting is necessary when the power is on. [Note] If there is no clock input to MCX314As, setting the RESETN to Low still cannot reset this IC.
EXPLSN	29	Input A – F –	External Pulse: pulse input signal for external pulse interpolation The normal setting is Hi level. When the external pulse interpolation occurs, EXPLSN is down (↓), the interpolation calculation starts, and one pulse for each axis interpolation is output. The width of EXPLSN on the Low level must be more than 4 CLK.
H16L8	30	Input A	Hi=16-bit, Low=8-bit: data bus width selection for 16-bit / 8-bit When the setting is Hi, 16-bit data bus is selected for processing the 16-bit data reading / writing in IC; when the setting is Low, 8-bit data bus (D7~D0) is active for data reading / writing.
TESTN	31	Input A	Test: terminal for internal-circuit test Please open, or connect it to + 5V.
BUSYN	32	Output B	Busy: reflecting the execution of the input command at this moment Once the command is written to MCX314As, the process will take 2 CLK to 4CLK (250nsec for 16MHz) on the Low level. When BUSYN is on the Low level, the other written commands cannot be executed.
INTN	33	Output B	Interrupt: outputting an interrupt signal to the host CPU. If any interrupt factor occurs the interrupt, the level is Low; when the interrupt is released, it will return to the Hi-Z level.
SCLK	34	Output A	System Clock: SCLK=CLK/2 All the signals in MCX314As are controlled and synchronized by internal SCLK. When the output signal of each axis is latched, it can be used as an external signal source. [Note] There is no SCLK output when RESETN is on the Low level.
XPP/PLS YPP/PLS ZPP/PLS UPP/PLS	35 38 40 42	Output A	Pulse +/Pulse: + direction dive pulse outputting When the reset is on the Low level, and while the driving is starting, DUTY 50% (at constant speed) of the plus drive pulses are outputting. + or − pulse mode is selectable. When the 1-pulse 1-direction mode is selected, this terminal is for drive output.
XPM/DIR YPM/DIR ZPM/DIR UPM/DIR	36 39 41 43	Output A	Pulse −/Pulse: − direction dive pulse outputting When the reset is on the Low level, and while the driving is starting, DUTY 50% (at constant speed) of the plus drive pulses are outputting. + or − pulse mode is selectable. When the 1-pulse 1-direction mode is selected, this terminal is direction signal.
XECA/PPIN YECA/PPIN ZECA/PPIN UECA/PPIN	44 46 48 50	Input A	Encoder−A/Pulse +in: signal for encoder phase-A input This input signal, together with phase-B signal, will make the Up / Down pulse transformation to be the input count of real position counter. When the Up / Down pulse input mode is selected, this terminal is for UP pulses input. Once the input pulse is up (↑), the real position counter is counting up.
XECB/PMIN YECB/PMIN ZECB/PMIN	45 47 49	Input A	Encoder-B/Pulse -in: signal for encoder phase-B input This input signal, together with phase-A signal, will make the Up / Down pulse transformation to be the input count of real position counter.

Signal Name	Pin No.	Input/Output	Signal Description
UECB/PMIN	51		When the Up / Down pulse input mode is selected, this terminal is for DOWN pulses input. Once the input pulse is up (↑), the real position counter is counting down.
XDRIVE/DCC YDRIVE/DCC ZDRIVE/DCC UDRIVE/DCC	56 76 104 122	Output A	Drive/Deviation Counter Clear: Drive status output (nDRIVE) and deviation counter clear output (DCC) share the same pin. Drive status display output (nDRIVE) is set to a High level while drive pulses are output. At execution of automatic home search, this signal is set to a High level. The axis with interpolation drive specified is set to a High level while the interpolation driving is executed. The DRIVE signal is set to a High level until nINPOS becomes active, while the nINPOS signal for the serve motor is enabled by mode selection. A deviation counter clear output (DCC) signal is output for a server motor driver. The signal can be output by setting the mode in automatic home search. See Sections 2.5.2 and 2.5.3. At resetting, the drive status display output is set.
XOUT7/DSND YOUT7/DSND ZOUT7/DSND UOUT7/DSND	57 77 105 123	Output A	General Output 7 / Descend: general purpose output signals After the axis is appointed by WR0 register, nOUT7~4 can output the 1/0 data of D11~8 in WR3 register to Hi / Low. They become Low when the IC is reset. When the drive status output mode is engaged, this signal can be used for reflecting the status of deceleration. While the driving command is executed and during the deceleration, it becomes Hi.
XOUT6/ASND YOUT6/ASND ZOUT6/ASND UOUT6/ASND	58 78 106 124	Output A	General Output 6 / Ascend: general purpose output signals (the operation is as same as nOUT7) When the drive status output mode is engaged, this signal can be used for reflecting the status of acceleration. While the driving command is executed and during the acceleration, it becomes Hi.
XOUT5/CMPM YOUT5/CMPM ZOUT5/CMPM UOUT5/CMPM	59 79 107 125	Output A	General Output 5 / Compare−: general purpose output signals (the operation is as same as nOUT7) When the drive status output mode is engaged, it becomes Hi if the value of logical / real position counter is smaller than that of COMP−; it becomes Low if the value of logical / real position counter is larger than that of COMP−.
XOUT4/CMPP YOUT4/CMPP ZOUT4/CMPP UOUT4/CMPP	60 80 108 128	Output A	General Output 4 / Compare+: general purpose output signals (the operation is as same as nOUT7) When the drive status output mode is engaged, it becomes Hi if the value of logical / real position counter is larger than that of COMP+; it becomes Low if the value of logical / real position counter is smaller than that of COMP+.
XOUT3~O YOUT3~0 ZOUT3~0 UOUT3~0	61~64 81~84 110~113 129~132	Output A	General Output 3~0: 4 general output signals for each axis OUT3~0 can output the 1/0 data of D15~0 in WR4 register to Hi / Low. They become Low when the IC is reset. Compared with the setting of nOUT7~4, it is easier cause there is no need to have the appointed axis.
XINPOS YINPOS ZINPOS UINPOS	67 85 95 114	Input A – F –	In-position: input signal for servo driver in-position Enable / disable and logical levels can be set as commands. When “enable” is set, and after the driving is finished, this signal is active and standby. n-DVR bit of main status register returns to 0.
XALARM YALARM ZALARM UALARM	68 86 96 115	Input A – F –	Servo Alarm: input signal for servo driver alarm Enable / disable and logical levels can be set as commands. When it is enable and when this signal is in its active level, the ALARM bit of RR2 register becomes 1.
XLMTP YLMTP ZLMTP ULMTP	69 87 97 116	Input A – F –	OVER Limit +: signal of + direction over limit During the + direction drive pulse outputting, decelerating stop or sudden stop will be performed once this signal is active. When the filter function is disabled, the active pulse width must be 2CLK or more. When it is enable, and when this signal is in its active level, the HLMT+ of RR2 register becomes 1.
XLMTM	70	Input A	OVER Limit −: signal of − direction over limit

Signal Name	Pin No.	Input/Output	Signal Description
YLMTM ZLMTM ULMTM	88 98 117	− F −	During the − direction drive pulse outputting, decelerating stop or sudden stop will be performed once this signal is active. The active pulse width should be more than 2CLK. Decelerating stop / sudden stop and logical levels can be set during the mode selection. When it is enable, and when this signal is in its active level, the HLMT− of RR2 register becomes 1.
XIN3~0 YIN3~0 ZIN3~0 UIN3~0	71~74 89,92~94 99~102 118~121	Input A − F −	Input 3~0: input signal to perform decelerating / sudden stop for each axis These signals can be used for HOME searching. The active pulse width should be more than 2CLK. Enable / disable and logical levels can be set for IN3~IN0. In automatic home search, IN0, IN1, and IN2 are assigned to a near home search signal, a home signal, and an encoder Z-phase signal respectively. The signal status can be read from register RR4 / RR5.
XEXPP YEXPP ZEXPP UEXPP	134 136 138 140	Input A − F −	External Operation +: + direction drive starting signal from external source When the fixed pulse driving is commanded from an external source, +direction driving will start if this signal is down (↓). Otherwise, when the continuous pulse driving is commanded from an external source, + driving will start if this signal is on the Low level. In manual pulsar mode, the encoder A-phase signal is input to this pin.
XEXPM YEXPM ZEXPM UEXPM	135 137 139 141	Input A − F −	External Operation −: − direction drive starting signal from external source When the fixed pulse driving is commanded from an external source, −direction driving will start if this signal is down (↓). Otherwise, when the continuous pulse driving is commanded from an external source, − direction driving will start if this signal is on the Low level. In manual pulsar mode, the encoder B-phase signal is input to this pin.
EMGN	142	Input A − F −	Emergency Stop: input signal to perform the emergency stop for all axes When this signal is on the Low level, including the interpolation driving, every axis will stop the operation immediately. EMG bit of register RR2, of each axis, will become 1. The low level pulse width should be more than 2CLK. [Note] For this signal, its logical levels cannot be selected.
GND	9,19,20,37, 52,55,66, 75,91,103, 127,133,143		Ground (0V) Terminal All of the 13 pins must be connected to 0V.
VDD	18,54,65, 90,109,126, 144		+ 5V Power Terminal. All of the 7 pins must be connected to +5V.

■ Input/ Output Circuit

Input A	More than 10 kΩ~ hundreds of kilo impedance is for internal impedance, which can pull up the VDD to the TTL level input of Smith trigger. CMOS and TTL can be connected. The user should open, or pull up with + 5V if the input is not used. The signal with – F – symbol has an integral filter circuit in the internal input column of this IC. See Section 2.8 for the filter function.
Output A	It is CMOS level output, 4mA driving buffer (Hi level output current IOH=−4mA, VOH=2.4Vmin, Low level output current IOL=4mA, VOL=0.4Vmax). Up to 10 LSTTL can be driven.
Output B	It is open collector type output, 4mA driving buffer, (Low level output current IOL=4mA, VOL=0.4Vmax). Pull up to +5V with high impedance if this output is used.
Bi-directional	Input side is TTL Smith trigger. Because there is no pull high resister for those signals in this IC, the user should pull up the data bus with high impedance. The user should pull up to +5V with high impedance (about 100 kΩ) when bits D15~D8 are not used. Output side is CMOS level output, 8mA driving buffer (Hi level output current IOH=−8mA, VOH=2.4Vmin, Low level output current IOL=8mA, VOL=0.4Vmax).

■ Notes for the Design of Circuitry

a. De-coupling Capacitor

Please connect VDD and GND with one or two De-coupling capacitors (about 0.1μF).

b. Noise Generated by Terminal Induction

The noise will exist because the inductance is in these pins. The user can add a capacitor (10-100pF) to pins to reduce the noise.

c. Reflection on Transfer Path

The load capacity for outputting types A, B, and bi-direction type A is 20-50pf. So, the reflection will happen if the PCB wiring is more than 60cm.

4. Register

This chapter indicates the user how to access all the registers in MCX314As, and what are the mapping addresses of these registers. Please refer to Chapter 2.4.3 for the registers (BP1P/M, BP2P/M, BP3P/M) of bit pattern interpolation.

4.1 Register Address by 16-bit Data Bus

As shown is the table below, when 16-bit data bus is used, the access address of read / write register is 8-bit.

■ Write Register in 16-bit Data Bus

All registers are 16-bit length.

Address A2 A1 A0	Symbol	Register Name	Contents
0 0 0	WR0	Command Register	for setting axis assignment and command
0 0 1	XWR1 YWR1 ZWR1 UWR1	X axis mode register 1 Y axis mode register 1 Z axis mode register 1 U axis mode register 1	for setting the logical levels of external decelerating stop, enable / disable, and the valid / invalid of interrupt for each axis for each axis
0 1 0	XWR2 YWR2 ZWR2 UWR2	X axis mode register 2 Y axis mode register 2 Z axis mode register 2 U axis mode register 2	for setting the limit signal mode, driving pulse mode, encoder input signal mode, and the logical levels and enable / disable of servo motor signal for each axis
	BP1P	BP1P register	for setting the + direction bit data of the first axis in bit pattern interpolation
0 1 1	XWR3 YWR3 ZWR3 UWR3	X axis mode register 3 Y axis mode register 3 Z axis mode register 3 U axis mode register 3	for setting the manual deceleration, individually decelerating, and S-curve acceleration/ deceleration mode for each axis, external operation mode, and general purpose output OUT7~4
	BP1M	BP1M register	for setting the − direction bit data of the first axis in bit pattern interpolation
1 0 0	WR4	Output register	for setting the general output OUT3 ~ 0
	BP2P	BP2P register	for setting the + direction bit data of the second axis in bit pattern interpolation
1 0 1	WR5	Interpolation mode register	for setting axis assignment and the constant vector speed mode, step output mode and interrupt
	BP2M	BP2M register	for setting the − direction bit data of the second axis in bit pattern interpolation
1 1 0	WR6 BP3P	Data writing register 1 BP3P register	for setting the low word 16-bit (D15-D0) for data writing. for setting the + direction bit data of the third axis in bit pattern interpolation
1 1 1	WR7 BP3M	Data writing register 2 BP3M register	for setting the high word 16-bit (D31-D16) for data writing. for setting the − direction bit data of the third axis in bit pattern interpolation

a. Each axis is with WR1, WR2 and WR3 mode registers. Each register is for 4-axis data writing (at the same address). Before those registers have been accessed, the host CPU should specify which axis is going to be accessed by writing a NOP command into WR0.

b. The register for bit pattern interpolation are BP1P~3P and BP1M~3M. After the resetting, the data writing cannot be performed, until the enable command (36h) is engaged by BP register. After the command 36h is enabled, the data writing cannot be performed in nWR1~3. So, the disable command (37h) should be engaged after the bit pattern interpolation data is written.

c. Please be noted that registers WR6 and BP3P / WR7 and BP3M share the same register hardware.

d. The bits of nWR1, nWR2, nWR3, nWR4 and nWR5 will be cleared to 0 after the resetting. It will be unknown for other registers.

■ Read Register in 16-bit Data Bus

All registers are 16-bit length.

Address A2 A1 A0	Symbol	Register Name	Contents
0 0 0	RR0	Main status register	error status, driving status, ready for interpolation, quadrant for circle interpolation and the stack of BP
0 0 1	XRR1 YRR1 ZRR1 URR1	X axis status register 1 Y axis status register 1 Z axis status register 1 U axis status register 1	comparison result, acceleration/deceleration state, and acceleration/deceleration speed increase/decrease state, finishing status
0 1 0	XRR2 YRR2 ZRR2 URR2	X axis status register 2 Y axis status register 2 Z axis status register 2 U axis status register 2	error message automatic home search execution state
0 1 1	XRR3 YRR3 ZRR3 URR3	X axis status register 3 Y axis status register 3 Z axis status register 3 U axis status register 3	interrupt message
1 0 0	RR4	Input register 1	I/O input for X and Y axes
1 0 1	RR5	Input register 2	I/O input for Z and U axes
1 1 0	RR6	Data reading register 1	low word of data register (D15 ~ D0)
1 1 1	RR7	Data reading register 2	high word of data register (D31 ~ D16)

Each axis is with WR1, WR2 and WR3 mode registers. Each register is for 4-axis data writing (at the same address). Before those registers have been accessed, the host CPU should specify which axis is going to be accessed by writing a NOP command into WR0.

4.2 Register Address by 8-bit Data Bus

In case of the 8-bit data bus access, the 16-bit data bus can be divided into high and low word byte. As shown in the table below, xxxxL is the low word byte (D7~D0) of 16-bit register xxxx, xxxxH is the high word byte (D15~8) of 16-bit register xxxx. Only for the command register (WR0L, WR0H), the user must write to the high word byte (WR0L), then to the low word byte (WR0H).

■ Write Register in 8-bit Data Bus

Address A3 A2 A1 A0	Write Register
0 0 0 0	WR0L
0 0 0 1	WR0H
0 0 1 0	XWR1L,YWR1L,ZWR1L,UWR1L
0 0 1 1	XWR1H,YWR1H,ZWR1H,UWR1H
0 1 0 0	XWR2L,YWR2L,ZWR2L,UWR2L,BP1PL
0 1 0 1	XWR2H,YWR2H,ZWR2H,UWR2H,BP1PH
0 1 1 0	XWR3L,YWR3L,ZWR3L,UWR3L,BP1ML
0 1 1 1	XWR3H,YWR3H,ZWR3H,UWR3H,BP1MH
1 0 0 0	WR4L,BP2PL
1 0 0 1	WR4H,BP2PH
1 0 1 0	WR5L,BP2ML
1 0 1 1	WR5H,BP2MH
1 1 0 0	WR6L,BP3PL
1 1 0 1	WR6H,BP3PH
1 1 1 0	WR7L,BP3ML
1 1 1 1	WR7H,BP3MH

■ Read Register in 8-bit Data Bus

Address A3 A2 A1 A0	Read Register
0 0 0 0	RR0L
0 0 0 1	RR0H
0 0 1 0	XRR1L,YRR1L,ZRR1L,URR1L
0 0 1 1	XRR1H,YRR1H,ZRR1H,URR1H
0 1 0 0	XRR2L,YRR2L,ZRR2L,URR2L
0 1 0 1	XRR2H,YRR2H,ZRR2H,URR2H
0 1 1 0	XRR3L,YRR3L,ZRR3L,URR3L
0 1 1 1	XRR3H,YRR3H,ZRR3H,URR3H
1 0 0 0	RR4L
1 0 0 1	RR4H
1 0 1 0	RR5L
1 0 1 1	RR5H
1 1 0 0	RR6L
1 1 0 1	RR6H
1 1 1 0	RR7L
1 1 1 1	RR7H

4.3 Command Register: WR0

Command register is used for the axis assignment and command registration for each axis in MCX314As. The register consists of the bit for axis assignment, bit for setting command code, and bit for command resetting.

After the axis assignment and command code have been written to the register, this command will be executed immediately. The data such as drive speed setting and data writing command must be written to registers WR6 and WR7 first. Otherwise, when the reading command is engaged, the data will be written and set, through IC internal circuit, to registers RR6 and RR7.

When using the 8-bit data bus, the user should write data into the high word byte (H), then low word byte (L).

It requires 250 nSEC (maximum) to access the command code when CLK=16MHz. The input signal BUSYN is on the Low level at this moment. Please don' t write the next command into WR0 before BUSYN return to the Hi level.

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
WR0	RESET	0	0	0	U	Z	Y	X	0							
					Axis Assignment					Command Code						

D6 ~ 0 Command code setting
Please refer to chapter 5 and the chapters following for further description of command codes.

D11 ~ 8 Axis assignment
When the bits of the axis are set to 1, the axis is assigned. The assignment is not limited only for one axis, but for multi-axes simultaneously. It is possible to write the same parameters also. However, the data reading is only for one assigned axis.
Whenever the interpolation is commanded, the bits of the assigned axis (axes) should be set 0.

D15 RESET IC command resetting
When this bit is set to 1, but others are 0, the IC will be reset after command writing. After command writing, the BUSYN signal will be on the Low level within 875 nSEC (When CLK=16 MHz) maximum.

When 8-bit data bus is used, the reset is activated when the command (80h) is written to register WR0H.

RESET bit should be set to 0 when the other commands are written.

4.4 Mode Register1: WR1

Each axis is with mode register WR1. The axis specified by NOP command or the condition before decides which axis' s register will be written.

The register consists of the bit for setting enable / disable and enable logical levels of input signal IN3~IN0 (decelerating stop / sudden stop during the driving) and bit for occurring the interrupt enable / disable.

Once IN3~IN1 are active, when the fixed pulse / continuous pulse driving starts, and also when IN signal becomes the setting logical level, the decelerating stop will be performed during the acceleration / deceleration driving and the sudden stop will be performed during the constant speed driving.

	H															
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
WR1	D-END	C-STA	C-END	P≧C+	P<C+	P<C-	P≧C-	PULSE	IN3-E	IN3-L	IN2-E	IN2-L	IN1-E	IN1-L	IN0-E	IN0-L
	Interrupt Enable/Disable								Driving Stop Input Signal Enable/Disable							

D7,5,3,1 INm-E The bit for setting enable / disable of driving stop input signal INm 0: disable, 1: enable

D6,4,2,0 INm-L The bit for setting enable logical levels for input signal INm 0: stop on the Low level, 1:stop on the Hi level
In automatic home search, the logical level of the INm signal that is used is set in these bits. The Enable/Disable bits (D5, D3, and D1) are set to Disable.

For the following bits, the interrupt is set: 1: enable, 0: disable

D8 PULSE Interrupt occurs when the pulse is up (↑) (drive pulse is set on the positive logical level)

D9 P ≥ C− Interrupt occurs when the value of logical / real position counter is larger than or equal to that of COMP− register

D10 P < C− Interrupt occurs when the value of logical / real position counter is smaller than that of COMP− register

D11 P < C+ Interrupt occurs when the value of logical / real position counter is smaller than that of COMP+ register

D12 P ≥ C+ Interrupt occurs when the value of logical / real position counter is larger than or equal to that of COMP+ register

D13 C-END Interrupt occurs at the end of the constant speed drive during an acceleration / deceleration driving

D14 C-STA Interrupt occurs at the start of the constant speed drive during an acceleration / deceleration driving

D15 D-END Interrupt occurs when the driving is finished

D15~D0 will be set to 0 while resetting.

4.5 Mode Register2: WR2

Each axis is with mode register WR2. The axis specified by NOP command or the condition before decides which axis' s register will be written.

WR2 can be used for setting: (1). external limit inputs, (2). driving pulse types, (3). encoder signal types, and (4). the feedback signals from servo drivers.

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
WR2	INP-E	NP-L	ALM-E	ALM-L	PIND1	PIND0	PINMD	DIR-L	PLS-L	PLSMD	CMPSL	HLMT-	HLMT+	LMTMD	SLMT-	SLMT+

D0 SLMT+ Enable / disable setting for COMP+ register which is used as the + direction software limit 1: enable, 0: disable
Once it is enabled during the + direction driving, if the value of logical / real position counter is larger than that of COMP+, the decelerating stop will be performed. The D0 (SLMT+) bit of register RR2 will become 1. Under this situation, further written + direction driving commends will not be executed.
Note: When an extension mode position counter variable ring is used, a software over run limit cannot be used.

D1 SLMT− Enable / disable setting for COMP− register which is used as the − direction software limit 1: enable, 0: disable
Once it is enabled during the − direction driving, if the value of logical / real position counter is smaller than that of COMP+, the decelerating stop will be performed. The D1 (SLMT−) bit of register RR2 will become 1. Under this situation, further written − direction driving commends will not be executed.

D2 LMTMD The bit for controlling stop type when the hardware limits (nLMTP and nLMTM input signals) are active 0: sudden stop, 1: decelerating stop

D3 HLMT+ Setting the logical level of + direction limit input signal (nLMTM) 0: active on the Low level, 1: active on the Hi level

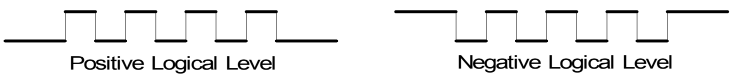
D4 HLMT− Setting the logical level of − direction limit input signal (nLMTM) 0:active on the Low level, 1: active on the Hi level

D5 CMPSL Setting if real position counter or logical position counter is going to be compared with COMP +/− register 0: logical position counter, 1 : real position counter

D6 PLSMD Setting output pulse type 0: independent 2-pulse type, 1: 1-pulse 1-direction type

When independent 2-pulse type is engaged, + direction pulses are output through the output signal nPP/PLS, and − direction pulses through nPM/DIR.
When 1-pulse 1-direction type is engaged, + and − directions pulses are output through the output signal nPP/PLS, and nPM/DIR is for direction signals.
[Note] Please refer to Chapter 13.2 and 13.3 for the output timing of pulse signal (nPLS) and direction signal (nDIR) when 1-pulse 1-direction type is engaged.

D7 PLS-L Setting logical level of driving pulses 0: positive logical level, 1: negative logical level

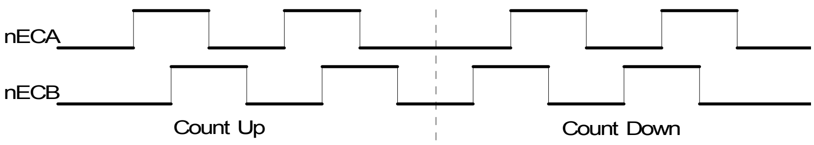


D8 DIR-L Setting logical level of the direction (nPM/DIR) output signal for 1-pulse mode DIR-L

D8 (DIR-L)	+ direction	− direction
0	Low	Hi
1	Hi	Low

D9 PINMD Setting the type of encoder input signals (nECA/PPIN and nECB/PMIN)
0: quadrature pulse input type 1: Up / Down pulse input type

Real position counter will count up or down when encoder input signal is triggered. When quadrature pulse input type is engaged, the “count up” will happen if the positive logical level pulses are input to phase A; the “count down” will happen if the positive logical level pulses are input to phase B. So, it will count up and down when these 2 signals go up (↑) and down (↓).



When Up / Down pulse input type is engaged, nECA/PPIN is for “ count up” input, and nECB/PMIN is for “count down” input. So, it will count up when the positive pulses go up (↑).

D11,10 PIND1,0 The division setting for quadrature encoder input.

D11	D10	Division
0	0	1/1
0	1	1/2
1	0	1/4
1	1	Invalid

Up / down pulse input is not available.

D12 ALM-L Setting active level of input signal nALARM 0: active on the Low level, 1: active on the Hi level

D13 ALM-E Setting enable / disable of servo alarm input signal nALARM 0: disable, 1: enable
When it is enabled, MCX314As will check the input signal. If it is active, D14 (ALARM) bit of RR2 register will become 1. The driving stops.

D14 INP-L Setting logical level of nINPOS input signal 0: active on the Low level, 1: active on the Hi level

D15 INP-E Setting enable/disable of in-position input signal nINPOS from servo driver 0: disable, 1: enable
When it is enabled, bit n-DRV of RR0 (main status) register does not return to 0 until nINPOS signal is active after the driving is finished.

D15~D0 will be set to 0 while resetting.

4.6 Mode Register3: WR3

Each axis is with mode register WR3. The axis specified by NOP command or the condition before decides which axis’ s register will be written.

WR3 can be used for manual deceleration, individual deceleration, S-curve acceleration / deceleration, the setting of external operation mode, and the setting of general purpose output OUT7~4.

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
WR3	0	0	0	0	OUT7	OUT6	OUT5	OUT4	OUTSL	0	0	EXOP1	EXOP0	SACC	DSNDE	MANLD

D0 MANLD Setting manual / automatic deceleration for the fixed pulse acceleration / deceleration riving
0: automatic deceleration, 1: manual deceleration
The decelerating point should be set if the manual deceleration mode is engaged.

D1 DSNDE Setting decelerating rate which is in accordance with the rate of the acceleration or an individual decelerating rate
Set whether jerk or an individual deceleration increasing rate is used as a deceleration increasing rate at S-curve acceleration/deceleration driving.

Value of D1 (DSNDE)	Deceleration speed at linear (trapezoidal) acceleration/deceleration	Deceleration increasing rate at S-curve acceleration/deceleration	Shape of the acceleration/deceleration curve
0	Uses the value of the acceleration speed (A)	Uses the value of the jerk (K)	Symmetry
1	Uses the value of the deceleration speed (D)	Uses the deceleration increase speed rate (L)	Non-symmetry

Set this bit to 0 to perform acceleration/deceleration driving where acceleration and deceleration are symmetrical and set this bit to 1 to perform acceleration/deceleration where acceleration and deceleration are non-symmetrical.
Automatic deceleration cannot be performed for non-symmetrical S-curve acceleration/deceleration fixed pulse driving. In this case, the D0 (MANLD) bit must be set to 1 and a manual deceleration point (DP) must be set.

D2 SACC Setting trapezoidal driving / S-curve driving
0: trapezoidal driving, 1: S-curve driving
Before S-curve driving is engaged, jerk (K) should be set.

D4,3 EXOP1,0 Setting the external input signals (nEXPP, nEXPM) for driving

D4 (EXOP1)	D3 (EXOP0)	
0	0	external signals disabled
0	1	continuous driving mode
1	0	fixed pulse driving mode
1	1	external signals disabled

When the continuous driving mode is engaged, the + direction drive pulses will be output continuously once the nEXPP signal is on the Low level; the − direction pulses will be output continuously once the nEXPM signal is on the Low level. When the fixed pulse driving mode is engaged, the + direction fixed pulse driving starts once the nEXPP signal is falling to the Low level from the Hi level; the − direction pulse driving starts once the nEXPM signal is falling to the Low level from the Hi level.
In manual pulsar mode, fixed pulse driving in the + direction is activated at ↑ f the nEXPP signal when the nEXPM signal is at the Low level. The fixed pulse driving is activated at ↓ of the nEXPP signal when the nEXPM signal is at the Low level.

D7 OUTSL Driving status outputting or used as general purpose output signals (nOUT7~4)

0: nOUT7~4: general purpose output
The levels of D11~8 will be output through nOUT7~4.
1: nOUT4~7: driving status output (see the table below)

Signal Name	Output Description
nOUT4/CMPP	Hi: if logical / real position counter ≥COMP+ register Low : if logical / real position counter <COMP+ register
nOUT5/CMPM	Hi: if logical / real position counter <COMP− register Low: if logical / real position counter ≥COMP− register
nOUT6/ASND	When the driving command is engaged, the level becomes Hi once the driving status is in acceleration.
nOUT7/DSND	When the driving command is engaged, the level becomes Hi once the driving status is in deceleration.

D11~8 OUTm Level setting for output signals OUT7~4 as general purpose output signals
0: Low level output, 1: Hi level output

D15~D0 will be set to 0 while resetting. D15~12, D5 and D6 should be always set 0.

4.7 Output Register: WR4

This register is used for setting the general purpose output signals nOUT3~0. This 16-bit register locates 4 output signals of each axis. It can be also used as a 16-bit general purpose output. It is Low level output when the bit is set 0, and Hi level output when the bit is set 1.

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
WR4	UOUT3	UOUT2	UOUT1	UOUT0	ZOUT3	ZOUT2	ZOUT1	ZOUT0	YOUT3	YOUT2	YOUT1	YOUT0	XOUT3	XOUT2	XOUT1	XOUT0

D15~D0 will be set to 0 while resetting, and nOUT3~0 signals become Low level.

4.8 Interpolation Mode Register: WR5

This register is used for setting axis assignment, constant vector speed mode, 1-step interpolation mode and interrupt during the interpolation.

	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
WR5	BPINT	CIINT	0	CMPLS	EXPLS	0	SPD1	SPD0	0	0	AX31	AX30	AX21	AX20	AX11	AX10
	Interrupt			Step			Constant Vector Speed				3rd Axis		2nd Axis		1st Axis (Master Axis)	

D1,0 AX11,10 ax1 (master axis) assignment for interpolation
Axis codes are shown as follows.

Axis	Code (Binary)
X	0 0
Y	0 1
Z	1 0
U	1 1

1st axis: X, 2nd axis: Y, 3rd axis: Z

D5	D4	D3	D2	D1	D0
1	0	0	1	0	0

第

For ax1 (master axis) will have the basic pulses of starting interpolation calculation, the speed parameter which is for constant / acceleration / deceleration driving should be set before the driving.

D3,2 AX21,20 ax2 assignment according to the codes shown in the table above

D5,4 AX31,30 ax3 assignment for 3-axis interpolation, according to the codes shown in the table above
Setting any value if it is only 2-axis interpolation.

D9,8 LSPD1,0 Constant vector speed mode setting of interpolation driving

D9	D8	Code (Binary)
0	0	constant vector speed invalid
0	1	2-axis constant vector speed
1	0	(setting not available)
1	1	3-axis constant vector speed

When 2-axis constant vector speed mode is engaged, the user should set the range (R) of ax2 to be 1.414 times of the range (R) of master axis (ax1).
When 3-axis constant vector speed mode is engaged, the user should set the range (R) of ax2 to be 1.414 times and the range (R) of ax3 to be 1.732 times of the range (R) of master axis (ax1).

D11 EXPLS When it is 1, the external (EXPLSN) controlled single step interpolation mode is engaged.

D12 CMPLS When it is 1, the command controlled single step interpolation mode is engaged.

D14 CIINT Interrupt enable / disable setting during interpolation 0: disable 1: enable

D15 BPINT Interrupt enable / disable setting during bit-pattern interpolation 0: disable 1: enable

D15~D0 will be set to 0 while resetting.

4.9 Data Register: WR6/WR7

Data registers are used for setting the written command data. The low-word data-writing 16-bit (WD15~WD0) is for register RR6 setting, and the high-word data-writing 16-bit (WD31~WD16) is for register RR7 setting.

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
WR6	WD15	WD14	WD13	WD12	WD11	WD10	WD9	WD8	WD7	WD6	WD5	WD4	WD3	WD2	WD1	WD0

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
WR7	WD31	WD30	WD29	WD28	WD27	WD26	WD25	WD24	WD23	WD22	WD21	WD20	WD19	WD18	WD17	WD16

The user can write command data with a designated data length into the write register. It does not matter to write WR6 or WR7 first (when 8-bit data bus is used, the registers are WR6L, WR6H, WR7L and WR7H).

The written data is binary formatted; 2' complement is for negatives.

For command data, the user should use designated data length.

The contents of WR6 and WR7 are unknown while resetting.

4.10 Main Status Register: RR0

This register is used for displaying the driving and error status of each axis. It also displays interpolation driving, ready signal for continuous interpolation, quadrant of circular interpolation and stack counter of bit pattern interpolation.

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
RR0	-	BPSC1	BPSC0	ZONE2	ZONE1	ZONE0	CNEXT	I-DRV	U-ERR	Z-ERR	Y-ERR	X-ERR	U-DRV	Z-DRV	Y-DRV	X-DRV
									Error Status of Each Axis				Driving Status of Each Axis			

D3 ~ 0 n-DRV Displaying driving status of each axis
When the bit is 1, the axis is an outputting drive pulse. ; when the bit is 0, the driving of the axis is finished. Once the in-position input signal nINPOS for servomotor is active, nINPOS will return to 0 after the drive pulse output is finished. During execution of automatic home search, this bit is set to 1.

D7 ~ 4 n-ERR Displaying error status of each axis
If any of the error bits (D5~D0) of each axis's RR2 register and any of the error-finish bits (D15~D12) of each axis' s RR1 register becomes 1, this bit will become 1.

D8 I-DRV Displaying interpolation driving status
While the interpolation drive pulses are outputting, the bit is 1.

D9 CNEXT Displaying the possibility of continuous interpolation data writing
When the bit is 1, it is ready for inputting parameters for next node and also ready for writing interpolation command data.

D12 ~ 10 ZONEm Displaying the quadrant of the current position in circular interpolation

D12	D11	D10	Quadrant
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

D14,13 BPSC1,0 In bit pattern interpolation driving, it displays the value of the stack counter (SC).

D14	D13	Stack Counter (SC) Value
0	0	0
0	1	1
1	0	2
1	1	3

In bit pattern interpolation driving, when SC = 3, it shows the stack is full. When SC = 2, there is one word (16-bit) space for each axis. When SC = 1, there is a 2-word (16-bit × 2) for each axis. When SC = 0, it shows all the stacks are empty, and the bit-pattern interpolation is finished.

4.11 Status Register 1: RR1

Each axis is with status register 1. The axis specified by NOP command or the condition before decides which axis’ s register will be read.

The register can display the comparison result between logical / real position counter and COMP +/− , the acceleration status of acceleration / deceleration driving, jerk of S-curve acceleration / deceleration and the status of driving finishing.

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
RR1	EMG	ALARM	LMT -	LMT+	IN3	IN2	IN1	IN0	ADSND	ACNST	AASND	DSND	CNST	ASND	CMP -	CMP+
	Status of Driving Finishing															

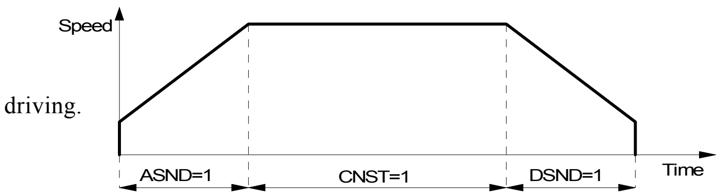
D0 CMP+ Displaying the comparison result between logical / real position counter and COMP+ register
1: logical / real position counter ≥ COMP+ register
0: logical / real position counter < COMP+ register

D1 CMP− Displaying the comparison result between logical / real position counter and COMP− register
1: logical / real position counter ≥ COMP− register
0: logical / real position counter > COMP− register

D2 ASND It becomes 1 when in acceleration.

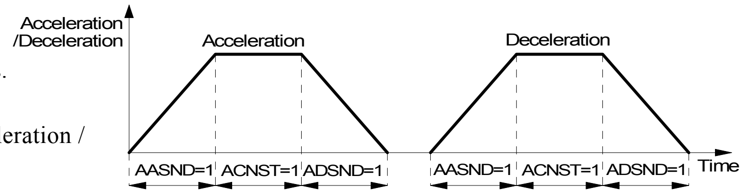
D3 CNST It becomes 1 when in constant speed driving.

D4 DSND It becomes 1 when in deceleration.



D5 AASND In S-curve, it becomes 1 when acceleration / deceleration increases.

D6 ACNST In S-curve, it becomes 1 when acceleration / deceleration keeps constant.



D7 ADSND In S-curve, it becomes 1 when acceleration / deceleration decreases.

D11 ~ 8 IN3 ~ 0 If the driving is stopped by one of external decelerating stop signals (nIN3 ~ 0), it will become 1.

D12 LMT+ If the driving is stopped by +direction limit signal (nLMTP), it will become 1.

D13 LMT− If the driving is stopped by −direction limit signal (nLMTP), it will become 1.

D14 ALARM If the driving is stopped by nALARM from servo drivers, it will become 1.

D15 EMG If the driving is stopped by external emergency signal (EMGN), it will become 1.

■ The Status Bits of Driving Finishing

These bits are keeping the factor information of driving finishing. The factors for driving finishing in fixed pulse driving and continuous driving are shown as follows:

a. when all the drive pulses are output in fixed-pulse driving,
b. when deceleration stop or sudden stop command is written,
c. when software limit is enabled, and is active,

d. when external deceleration signal is enabled, and active,
e. when external limit switch signals (nLMTP, nLMTM) become active,
f. when nALARM signal is enabled, and active, and
g. when EMGN signal is on the Low level.

Above factors "a." and "b." can be controlled by the host CPU, and factor "c." can be confirmed by register RR2 even the driving is finished. As for factors "d." ~ "g.", the error status is latched in RR2 until next driving command or a clear command (25h) is written.

After the driving is finished, if the error factor bits D15~D12 become 1, n-ERR bit of main status register RRO will become 1.

Status bit of driving finishing can be cleared when next driving command is written, or when the finishing status clear command (25h) is used.

4.12 Status Register 2: RR2

Each axis is with status register 2. The axis specified by NOP command or the condition before decides which axis' s register will be read.

This register is for reflecting the error information and automatic home search. When an error occurs, the error information bit (one of D7 to D0) is set to 1. When one or more of D7 to D0 bits of RR2 register are 1, n-ERR bits of main status register RR0 become 1.

	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
	H								L							
RR2	-	0	0	HMST4	HMST3	HMST2	HMST1	HMST0	HOME	0	EMG	ALARM	HLMT-	HLMT+	SLMT-	SLMT+
				Automatic Home Search Execution State					Error Information							

D0 SLMT+ D0 SLMT + During the + direction driving, when logical / real position counter ≥ COMP+ (COMP+ enabled, and used as software limit)

D1 SLMT− During the − direction driving, when logical / real position counter ≤ COMP− (COMPenabled, and used as software limit)

D2 HLMT+ When external +direction limit signal (nLMTP) is on its active level

D3 HLMT− When external −direction limit signal (nLMTM) is on its active level

D4 ALARM When the alarm signal (nALARM) for servo motor is on its active level

D5 EMG When emergency stop signal (EMGN) becomes Low level.

D7 HOME Error occurred at execution of automatic home search. When the encoder Z-phase signal (nIN2) is already active at the start of Step 3, this bit is set to 1.

D12~8 HMST4~0 The home search execution state indicates the contents of the automatic home search execution currently performed. See Section 2.5.4.

In driving, when hardware / software limit is active, the decelerating stop or sudden stop will be executed.

Bit SLMT+ / − will not become 1 during the reverse direction driving.

4.13 Status Register 3: RR3

Each axis is with status register 3. The axis specified by NOP command or the condition before decides which axis' s register will be read.

This register is for reflecting the interrupt factor. When interrupt happens, the bit with the interrupt factor becomes 1. The user should set the interrupt factor through register WR1 to perform the interrupt.

To generate an interrupt from D0 to D7, interrupt Enable must be set for each factor in the WR1 register.

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
RR3							SYNC	HMEND	D-END	C-STA	C-END	P≧C+	P<C+	P<C-	P≧C-	PULSE

Interrupt Factor

D0 PULSE When the drive pulse is up (drive pulse is set on the positive logical level)

D1 P ≥ C− Once the value of logical / real position counter is larger than that of COMP− register

D2 P < C− Once the value of logical / real position counter is smaller than that of COMP− register

D3 P < C+ Once the value of logical / real position counter is smaller than that of COMP+ register

D4 P ≥ C+ Once the value of logical / real position counter is larger than that of COMP+ register

D5 C-END When the pulse output is finished in the constant speed drive during an acceleration / deceleration driving

D6 C-STA When the pulse output is started in the constant speed drive during an acceleration / deceleration driving

D7 D-END When the driving is finished

D8 HMEND Automatic home search terminated. (See Section 2.5)

D9 SYNC Synchronous action was activated. (See Section 2.6)

When one of the interrupt factors occurs an interrupt, the bit of the register becomes 1, and the interrupt output signal (INTN) will become the Low level. The host CPU will read register RR3 of the interrupted axis, the bit of RR3 will be cleared to 0, and the interrupt signal will return to the non-active level.
[Note] For a 8-bit data bus, all the bits are cleared when the RR3L register is read. Therefore, when using the DD8 (HMEND) and D9 (SYNC) bits, read RR3H before reading the RR3L register.

4.14 Input Register: RR4 / RR5

RR4 and RR5 are used for displaying the input signal status. The bit is 0 if the input is on the Low level; the bit is 1 if the input is on the Hi level.

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
RR4	Y-ALM	Y-INP	Y-EX-	Y-EX+	Y-IN3	Y-IN2	Y-IN1	Y-IN0	X-ALM	X-INP	X-EX-	X-EX+	X-IN3	X-IN2	X-IN1	X-IN0

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
RR5	U-ALM	U-INP	U-EX-	U-EX+	U-IN3	U-IN2	U-IN1	U-IN0	Z-ALM	Z-INP	Z-EX-	Z-EX+	Z-IN3	Z-IN2	Z-IN1	Z-IN0

Bit Name	Input Signal	Bit Name	Input Signal
n-IN0	nIN0	n-EX+	nEXPP
n-IN1	nIN1	n-EX-	nEXPM
n-IN2	nIN2	n-INP	nINPOS
n-IN3	nIN3	n-ALM	nALARM

4.15 Data-Read Register: RR6 / RR7

According to the data-read command, the data of internal registers will be set into registers RR6 and RR7. The low word 16 bits (D15 ~ D0) is set in RR6 register, and the high word 16 bits (D31 ~ D16) is set in RR7 register for data reading.

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
RR6	RD15	RD14	RD13	RD12	RD11	RD10	RD9	RD8	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
RR7	RD31	RD30	RD29	RD28	RD27	RD26	RD25	RD24	RD23	RD22	RD21	RD20	RD19	RD18	RD17	RD16

The data is binary formatted; 2' s complement is for negatives.

5. Command Lists

■ Write Commands

Code	Command	Symbol	Data Range	Data Length
00h	Range setting	R	R 8,000,000(multiple=1) ~ 16,000(multiple=500)	4 bytes
01	Jerk setting (Acceleration increasing rate)	K	1 ~ 65,535	2
02	Acceleration setting	A	1 ~ 8,000	2
03	Deceleration setting	D	1 ~ 8,000	2
04	Initial speed setting	SV	1 ~ 8,000	2
05	Drive speed setting	V	1 ~ 8,000	2
06	Output pulse numbers / finish point	P	Output pulse numbers: 0~268,435,455 /finish point: −2,147,483,646 ~ +2,147,483,646	4
07	Manual deceleration point setting	DP	0 ~ 4,294,967,295	4
08	Circular center point setting	C	−2,147,483,646 ~ +2,147,483,646	4
09	Logical position counter setting	LP	−2,147,483,648 ~ +2,147,483,647	4
0A	Real position counter setting	EP	−2,147,483,648 ~ +2,147,483,647	4
0B	COMP + register setting	CP	−2,147,483,648 ~ +2,147,483,647	4
0C	COMP − register setting	CM	−2,147,483,648 ~ +2,147,483,647	4
0D	Acceleration counter offsetting	AO	−32,768 ~ +32,767	2
0E	Deceleration increasing rate setting	L	1 ~ 65,535	2
60	Extension mode setting	EM	(Bit data)	4
61	Home detection speed setting	HV	1 ~ 8,000	2
64	Synchronous action mode setting	SM	(Bit data)	4

[Note] When those parameters are written, the total data length should be completely filled.

[Formula Calculation for Parameters]

$$\text{Multiple} = \frac{8{,}000{,}000}{R}$$

$$\text{Jerk (PPS/SEC}^2) = \frac{62.5 \times 10^6}{K} \times \underbrace{\frac{8{,}000{,}000}{R}}_{\text{Multiple}}$$

$$\text{Acceleration (PPS/SEC)} = A \times 125 \times \underbrace{\frac{8{,}000{,}000}{R}}_{\text{Multiple}}$$

$$\text{Drive Speed (PPS)} = V \times \underbrace{\frac{8{,}000{,}000}{R}}_{\text{Multiple}}$$

$$\text{Deceleration Increasing Rate (PPS/SEC}^2) = \frac{62.5 \times 10^6}{L} \times \underbrace{\frac{8{,}000{,}000}{R}}_{\text{Multiple}}$$

$$\text{Deceleration (PPS/SEC)} = D \times 125 \times \underbrace{\frac{8{,}000{,}000}{R}}_{\text{Multiple}}$$

$$\text{Initial Speed (PPS)} = SV \times \underbrace{\frac{8{,}000{,}000}{R}}_{\text{Multiple}}$$

■ Data Reading Commands

Code	Command	Symbol	Data Range	Data Length
10h	Logical position counter reading	LP	−2,147,483,648 ~ +2,147,483,647	4 bytes
11	Real position counter reading	EP	−2,147,483,648 ~ +2,147,483,647	4
12	Current drive speed reading	CV	1 ~ 8,000	2
13	Acceleration / deceleration reading	CA	1 ~ 8,000	2
14	Synchronous buffer register reading	SB	−2,147,483,648 ~ +2,147,483,647	4

■ Driving Commands

Code	Command
20h	+ direction fixed pulse driving
21	− direction fixed pulse driving
22	+ direction continuous driving
23	− direction continuous driving
24	Drive start holding
25	Drive start holding release / stop status clear
26	Decelerating stop
27	Sudden stop

■ Interpolation Commands

Code	Command
30h	2-axis leaner interpolation
31	3-axis leaner interpolation
32	CW circular interpolation
33	CCW circular interpolation
34	2-axis bit pattern interpolation
35	3-axis bit pattern interpolation
36	BP register writing enabled*
37	BP register writing disabled
38	BP data stack
39	BP data clear
3A	1-step interpolation
3B	Deceleration valid
3C	Deceleration invalid
3D	Interpolation interrupt clear

*BP = bit pattern

■ Other commands

Code	Command
62	Automatic home search execution
63	Deviation counter clear output
65	Synchronous action activation
0F	NOP (for axis switching)

[Note] Please do not write the codes not mentioned above. The unknown situation could happen due to IC internal circuit test.

6. Commands for Data Writing

Data writing is used for setting driving parameters such as acceleration, drive speed, output pulse numbers…

It is possible to write the same data for more than one axis simultaneously if more those axes are assigned.

If the data length is two bytes, WR6 register can be used. If the data is 4 bytes, the high word data can be written into register WR7 and the low word into register WR6. Then, the axis assignment and command code will be written into register WR0 for execution.

Writing data for registers WR6 and WR7 is binary and 2’ s complement for negatives. Each data should be set within the permitted data range. If the setting data out of range, the driving can not be done.

[Note]
a. It requires 250 nSEC (maximum) to access the command code when CLK=16MHz. Please don’ t write the next command or data into WR0 when the present command is written.

b. Except acceleration offset (OA), the other parameters are unknown while resetting. So, please per-set proper values for those driving related parameters before the driving starts.

6.1 Range Setting

Code	Command	Symbol	Data Range	Data Length
00h	Range setting	R	8,000,000 (multiple:1) ~ 16,000 (multiple:500)	4 bytes

“R” is the parameter determining the multiple of drive speed, acceleration / deceleration and jerk. The multiple can be calculated as follows where the range setting value is R.

$$\text{Multiple} = \frac{8{,}000{,}000}{R}$$

For the parameter setting range of drive speed, acceleration / deceleration is 1~8000, if the higher value is needed, the user should have a larger multiple.

In case of increasing the multiple, although the high speed driving is possible, the speed resolution will be decreased. So, the user can set the multiple as small as possible if the setting speed has covered the desired speed. For example, the maximum value of parameter for setting the drive speed (V) is 8000, and the drive speed is set 40KPPS. The user can set V=8000 and R=1,600,000. Because 40K is 5 times of 8000, we set the R=8,000,000/5=1,600,000.

The Range (R) cannot be changed during the driving. The speed will be changed discontinuously.

6.2 Jerk Setting

Code	Command	Symbol	Data Range	Data Length
01h	Jerk setting	K	1 ~ 65,535	2 bytes

A jerk setting value is a parameter that determines the acceleration increase/decrease rate per unit in S-curve acceleration/deceleration. In S-curve acceleration/deceleration driving (WR3/D1=0) where acceleration and deceleration are symmetrical, this jerk is used at deceleration also.

$$\text{Jerk (PPS/SEC}^2) = \frac{62.5 \times 10^6}{K} \times \underbrace{\frac{8{,}000{,}000}{R}}_{\text{Multiple}}$$

“K” is the parameter determining the jerk. The jerk calculation is shown in the following formula:

Because the setting range of jerk is 1 ~ 65,535, the jerk range is shown as follows:

When Multiple = 1, 954 PPS/SEC^2 ~ 62.5 × 10^6 PPS/SEC^2
When Multiple = 500, 477 × 10^3 PPS/SEC^2 ~ 31.25 10^9 PPS/SEC^2

6.3 Acceleration Setting

Code	Command	Symbol	Data Range	Data Length
02h	Acceleration setting	A	1 ~ 8,000	2 bytes

In linear acceleration driving (WR3/D1=0) where acceleration and deceleration are symmetrical, this acceleration speed is used at deceleration.
For S-curve acceleration/deceleration driving, set the maximum value of 8,000 for the parameter.

“A” is the parameter determining the acceleration. The acceleration calculation is shown in the following formula:

$$\text{Acceleration (PPS/SEC)} = A \times 25 \times \underbrace{\frac{8{,}000{,}000}{R}}_{\text{Multiple}}$$

For the range of A is from 1 ~ 8,000, the actual acceleration range is shown as follows:

When Multiple = 1, 125 PPS/SEC ~ 1 × 10^6 PPS/SEC
When Multiple = 500, 62.5 × 10^3 PPS/SEC ~ 500 × 10^6 PPS/SEC

6.4 Deceleration Setting

Code	Command	Symbol	Data Range	Data Length
03h	Deceleration setting	D	1 ~ 8,000	2 bytes

This parameter is used to set a deceleration speed at deceleration in non-symmetrical linear acceleration driving (WR3/D1=1). In non-symmetrical S-curve acceleration driving, set the maximum value of 8000 in this parameter.

"D" is the parameter determining the deceleration. The deceleration calculation is shown in the following formula:

$$\text{Deceleration (PPS/SEC)} = D \times 125 \times \underbrace{\frac{8{,}000{,}000}{R}}_{\text{Multiple}}$$

6.5 Initial Speed Setting

Code	Command	Symbol	Data Range	Data Length
04h	Initial speed setting	SV	1 ~ 8,000	2 bytes

"SV" is the parameter determining the speed of initial speed. The initial speed calculation is shown in the following formula:

$$\text{Initial Speed (PPS)} = SV \times \underbrace{\frac{8{,}000{,}000}{R}}_{\text{Multiple}}$$

For stepper motors, the user should set the initial speed smaller than the self-starting frequency of stepper motors. For a servo motor also, if the value that is set is too low, creep or premature termination may occur at decelerating termination of fixed pulse driving. In this case, take the following measures.

a. Linear acceleration driving with symmetrical acceleration/deceleration

- Set 0 in the acceleration counter offset (A0).
- Enable the triangle form prevention function (extension command 60h WR6/D3(AVTRI) = 1).

b. Linear acceleration driving with symmetrical deceleration/deceleration

- Set 0 in the acceleration offset counter offset (A0).
- Enable the triangle form prevention function (extension command 60h WR6/D3 (AVTRI) = 1).

However, in a case where acceleration > deceleration, the number of creep pulses increases as the ratio of acceleration A and deceleration D increases. In this case, increase the initial speed.

6.6 Drive Speed Setting

Code	Command	Symbol	Data Range	Data Length
05h	Drive speed setting	V	1 ~ 8,000	2 bytes

"V" is the parameter determining the speed of constant speed period in trapezoidal driving. In constant speed driving, the drive speed is the initial speed. The drive speed calculation is shown in the following formula:

$$\text{Drive Speed (PPS)} = V \times \underbrace{\frac{8{,}000{,}000}{R}}_{\text{Multiple}}$$

If the setting drive speed is lower than the initial speed, the acceleration / deceleration will not be performed, and the driving is constant speed. During the encoder Z-phase searching (at a low-peed driving), if the user want to perform the sudden stop once the Z-phase is detected, the drive speed should be set lower than the initial speed.

Drive speed can be altered during the driving. When the drive speed of next constant speed period is set, the acceleration / deceleration will be performed to reach the new setting drive speed, then a constant speed driving starts.

In automatic home search, this drive speed is used for high-speed search speed of Step 1 and high-speed drive speed of Step 4.

[Note]

a. In fixed pulse S-curve acceleration / deceleration driving, there is no way to change the drive speed during the driving. In continuous S-curve acceleration / deceleration driving, the S-curve profile cannot be exactly tracked if the speed alterations during the acceleration / deceleration. it is better to change the drive speed in the constant speed period.

b. In fixed pulse trapezoidal driving, the frequent changes of drive speed may occur residual pulses in the ending of deceleration.

6.7 Output Pulse Number / Interpolation Finish Point Setting

Code	Command	Symbol	Data Range	Data Length
06h	Output pulse number / interpolation finish point setting	P	Output pulse numbers: 0 ~ 4,294,967,295 Finish point: −2,147,483,646 ~ +2,147,483,646	4 bytes

The number of output pulses indicates the total number of pulses that are output in fixed pulse driving. An unsigned 32-bit value is set.

Output pulse number setting:
The parameter "P" is setting total output pulse numbers in fixed pulse driving. The value is absolute, unsigned number. The output pulse numbers can be changed during the driving.

Interpolation finish point setting:
This parameter is also setting the finish point of each axis in linear and circular interpolations. The finish points of these axes should be set by relative numbers in 32-bit data length.

6.8 Manual Decelerating Point Setting

Code	Command	Symbol	Data Range	Data Length
07h	Manual decelerating point setting	DP	0 ~ 4,294,967,295	4 bytes

"DP" is the parameter setting the manual deceleration point in fixed pulse acceleration / deceleration driving when the manual deceleration mode is engaged.

In manual deceleration mode, the user can set the bit D0 of WR3 register to 1. The decelerating point can be set:

Manual Decelerating Point = Output Pulse Number − Pulse Number for Deceleration

6.9 Circular Center Point Setting

Code	Command	Symbol	Data Range	Data Length
08h	Circular center point setting	C	−2,147,483,646 ~ +2,147,483,646	4 bytes

"C" is the parameter setting the center point in circular interpolation. The coordinates of center point should be set the relative number related to the current position.

6.10 Logical Position Counter Setting

Code	Command	Symbol	Data Range	Data Length
09h	Logical position counter setting	LP	−2,147,483,648 ~ +2,147,483,647	4 bytes

"LP" is the parameter setting the value of logic position counter.

Logical position counter counts Up / Down according to the +/− direction pulse output.

The data writing and reading of logical position counter is possible anytime.

6.11 Real position Counter Setting

Code	Command	Symbol	Data Range	Data Length
0Ah	Real position counter setting	EP	−2,147,483,648 ~ +2,147,483,647	4 bytes

"EP" is the parameter setting the value of real position counter.

Real position counter counts Up / Down according to encoder pulse input.

The data writing and reading of real position counter is possible anytime.

6.12 COMP+ Register Setting

Code	Command	Symbol	Data Range	Data Length
0Bh	COMP+ register setting	CP	−2,147,483,648 ~ +2,147,483,647	4 bytes

"CP" is the parameter setting the value of COM+ register.

COMP+ register is used to compare with logical / real position counter, and the comparison result will be output to bit D0 of register RR1 or nOUT4/CMPP signal. Also, it can be used as the + direction software limit.

The value of COMP+ register can be written anytime.

6.13 COMP− Register Setting

Code	Command	Symbol	Data Range	Data Length
0Ch	COMP− register setting	CM	−2,147,483,648 ~ +2,147,483,647	4 bytes

"CM" is the parameter setting the value of COMP − register.

COMP− register is used to compare with logical / real position counter, and the comparison result will be output to bit D0 of RR1 register or nOUT5/CMPM signal. Also, it can be used as the − direction software limit.

The value of COMP− register can be written anytime.

6.14 Acceleration Counter Offsetting

Code	Command	Symbol	Data Range	Data Length
0Dh	Acceleration Counter Offsetting	AO	−32,768 ~ +32,767	4 bytes

"AO" is the parameter executing acceleration counter offset.

The offset value of acceleration counter will be set 8 while resetting.
Set this parameter value to 0 when performing acceleration fixed pulse driving by setting a low initial speed.

6.15 Deceleration Increasing Rate Setting

Code	Command	Symbol	Data range	Data length
0Eh	Deceleration increase set setting	L	1 ~ 65,535	2 bytes

This deceleration increasing rate value is a parameter used to determine a deceleration speed increase/decrease rate per unit time in S-curve acceleration/deceleration driving where acceleration and deceleration are non-symmetrical.

The deceleration increasing rate is calculated as follows where the deceleration increasing rate setting value is L.

$$\underset{(\mathrm{PPS/SEC}^2)}{\text{Deceleration Increasing Rate}} = \frac{62.5\times10^6}{L} \times \underbrace{\frac{8{,}000{,}000}{R}}_{\text{Multiple}}$$

Since the deceleration increasing rate setting value (L) range is from 1 to 65,535, the deceleration increasing rate range will be as follows.

Where multiple = 1,	954 PPS/SEC^2	~	62.5×10^6 PPS/SEC^2
Where multiple = 500,	477×10^3 PPS/SEC^2	~	31.25×10^9 PPS/SEC^2

6.16 Extension Mode Setting

Code	Command	Symbol	Data range	Data length
60h	Extension mode setting	EM		4 bytes

For extension mode setting, set an appropriate value in each bit of the WR6 and WR7 registers that are shown below and write a command code (60h) as well as specification of the axis in the WR0 register. As a result, the contents of the WR6 and WR7 registers are set in the extension mode registers (EM6 and EM7) in the IC. At resetting, all the bits of the extension mode registers (EM6 and EM7) in the IC are cleared to 0.

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
WR6	FL2	FL1	FL0	FE4	FE3	FE2	FE1	FE0	SMODE	0	HMINT	VRING	AVTRI	POINV	EPINV	EPCLR

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
WR7	DCCW2	DCCW1	DCCW0	DCC-L	DCC-E	LIMIT	SAND	PCLR	ST4-D	ST4-E	ST3-D	ST3-E	ST-D	ST2-E	ST1-D	ST1-E
	Deviation Counter Clear Output								Step 4		Step 3		Step 2		Step1	

WR6/D0 EPCLR When driving stops triggered by the nIN2 signal, the real position counter is cleared. When the nIN2 signal is changed to the Active level while this bit is set to 1, the driving stops and the real position counter (EP) is cleared. The WR1/D5(IN2-E) bit must be set to 1 and the Enable level must be set in the WR1/D4(IN2-L) bit. (See Section 4.4).

WR6/D1 EPINV Inverses increase/decrease of the real position counter.

WR6/D1(EPINV)	Input pulse mode	Increase/decrease of the real position counter (EP)
0	A/B phase mode	A Count UP when the A phase is advancing. Count DOWN when the B phase is advancing.
	UP/DOWN pulse mode	Count UP at PPIN pulse input. Count DOWN at PMIN pulse input.
1	A/B phase mode	Count UP when the B phase is advancing. Count DOWN when the A phase is advancing.
	UP/DOWN pulse mode	Count UP at PMIN pulse input. Count DONW at PPIN pulse input.

WR6/D2 POINV Replaces output signals of drive pulse output between nPP (drive pulse in the + direction) and nPM (drive pulse in the – direction). When this bit is set to 1, drive pulses are output to the nPM signal during driving in the + direction and in the – direction, drive pulses are output to the nPP signal.

WR6/D3 AVTRI Prevents triangle forms in linear acceleration (trapezoidal) of fixed pulse driving. 0: Disable, 1: Enable (See Section 2.2.2.)

WR6/D4 VRING Enables the variable ring function of the logical position counter and the real position counter. 0: Disable, 1: Enable (See Section 2.3.3)

WR6/D5 HMINT Generates an interrupt signal (INTN) at termination of automatic home search. When this bit is set to 1, the interrupt signal (INTN) becomes Low Active at termination of automatic home search and the RR3/D8(HMEND) bit of the axes from which the interrupt is generated indicates 1. When the CPU reads the RR3 register of the axis from which the interrupt was generated, the bits of the RR3 registers are cleared to 0 and the interrupt output signal is reset to Hi-Z.

WR6/D7 SMODE Set this bit to 1 when giving priority to the reaching of the specified drive speed in S-curve acceleration/deceleration driving.

WR6/D12~8 FE4~0 Set whether the IC built-in filter function is set to Enable or Disable (through) for each of input signals. 0: Disable (through), 1: Enable

Specification bit	Fitter Enable signal
WR6/D8 (FE0)	EMGN*1, nLMTP, nLMTM, nIN0,nIN1
WR6/D9 (FE1)	nIN2
WR6/D10(FE2)	nINPOS, nALARM
WR6/D11(FE3)	nEXPP, nEXPM, EXPLS*2
WR6/D12(FE4)	nIN3

*1: Set the EMGN signal in the D8 bit of the WR6 register of the X axis.
*2: Set the EXPLS signal in the D11 bit of the WR6 register of the X axis.

WR6/D15~13 FL2~0 Set a time constant of the filter. See Section 2.8 for the details of the input signal filter function.

WR6/D15 ~ 13 (FL2 ~ 0)	Removable maximum noise width	Input signal delay time
0	1.75μSEC	2μSEC
1	224μSEC	256μSEC
2	448μSEC	512μSEC
3	896μSEC	1.024mSEC
4	1.792mSEC	2.048mSEC
5	3.584mSEC	4.096mSEC
6	7.168mSEC	8.192mSEC
7	14.336mSEC	16.384mSEC

Each bit of the WR7 register is used to set an automatic home search mode. See “ Automatic home search mode setting” in Section 2.5.3 for details of each bit.

[Note] For an extension mode setting command, set appropriate values in both the WR6 and WR7 registers since the contents of both WR6 and WR7 registers are set in the extension mode registers (EM6 and EM7) in the IC.

6.17 Home Search Speed Setting

Code	Command	Symbol	Data range	Data length
61h	Home search speed setting	HV	1 ~ 8,000	2 bytes

Set a low-speed home search speed of Steps 2 and 3.
The home search speed can be calculated as follows where the home search speed setting value is HV.

$$\text{Home Detection Speed (PPS)} = \text{HV} \times \underbrace{\frac{8{,}000{,}000}{R}}_{\text{Multiple}}$$

Set a value lower than the initial speed (SV) to stop driving immediately when the search signal becomes active.

See Section 2.5 for details of automatic home search.

6.18 Synchronous Action Mode Setting

Code	Command	Symbol	Data range	Data length
64h	Synchronous action mode setting	SM		4 bytes

For synchronous action mode setting, set an appropriate value in each bit of the WR6 and WR7 registers that are shown below and write a command code (64h) as well as specification of the axis in the WR0 register. As a result, the contents of the WR6 and WR7 registers are set in the synchronous action mode registers (SM6 and SM7) in the IC. At resetting, all the bits of the synchronous action mode registers (SM6 and SM7) in the IC are cleared to 0.

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
WR6	AXIS3	AXIS2	AXIS1	0	0	0	CMD	LPRD	IN3↓	IN3↑	D-END	D-STA	P≧C-	P<C-	P<C+	P≧C+
	Activation of Other Axes						Provocative									

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
WR7	INT	OUT	0	0	VLSET	OPSET	EPSET	LPSET	EPSAV	LPSAV	ISTOP	SSTOP	CDRV-	CDRV+	FDRV-	FDRV+
	Action															

See Section 2.6 for details of each bit and synchronous action.

7. Commands for Reading Data

Data reading commands are used to read the register contents of each axis.

After a data reading command is written into register WR0, this data will be set in registers RR6 and RR7.

The host CPU can reach the data through reading registers RR6 and RR7. Reading data for registers WR6 and WR7 is binary and 2' s complement for negatives.

[Note]
a. It requires 250 nSEC (maximum) to access the command code of data reading where CLK = 16MHz. After the command is written and passed that time, read registers RR6 and 7.

b. The axis assignment is for one axis. If more than one axes are assigned, the data reading priority is X > Y > Z > U.

7.1 Logical Position Counter Reading

Code	Command	Symbol	Symbol	Data length
10h	Logical position counter reading	LP	−2,147,483,648 ~ +2,147,483,647	4 bytes

The current value of logical position counter will be set in read registers RR6 and RR7.

7.2 Real position Counter Reading

Code	Command	Symbol	Data range	Data length
11h	Real position counter reading	EP	−2,147,483,648 ~ +2,147,483,647	4 bytes

The current value of real position counter will be set in read registers RR6 and RR7.

7.3 Current Drive Speed Reading

Code	Command	Symbol	Data range	Data length
12h	Current drive speed reading	CV	1 ~ 8,000	2 bytes

The value of current drive speed will be set in read registers RR6 and RR7.
When the driving stops, the value becomes 0. The data unit is as same as the setting value of drive speed (V).

7.4 Current Acceleration / Deceleration Reading

Code	Command	Symbol	Data range	Data length
13h	Current acceleration / deceleration reading	CA	1 ~ 8,000	2 bytes

The value of current acceleration / deceleration will be set in read registers RR6 and RR7.
When the driving stops, the read data is random number. The data unit is as same as the setting value of acceleration (A).

7.5 Synchronous Action Buffer Register Reading

Code	Command	Symbol	Data range	Data length
14h	Synchronous action buffer register reading	BR	−2,147,483,648 ~ +2,147,483,647	4 bytes

The value of the synchronous action buffer register is set in the RR6 and RR7 read data registers.

8. Driving Commands

Driving commands include the commands for each axis' s drive pulse output and other related commands. After the command code and axis assignment are written in command register WR0, the command will be executed immediately. It is possible to assign more than one axis with same command at the same time.

In driving, bit n-DRV of each axis' s main status register RR0 becomes 1. When the driving is finished, the bit n-DRV will return to 0.

If nINPOS input signal for servo drivers is enabled, bit n-DRV of main status register RR0 will not return to 0 until nINPOS signal is on its active level.

[Note] It requires 250 nSEC (maximum) to access the command code when CLK=16MHz. Please write the next command after this period of time.

8.1 +Direction Fixed Pulse Driving

Code	Command
20h	+Direction Fixed Pulse Driving

The setting pulse numbers will be output through the output signal nPP.

In driving, logical position counter will count-up 1 when one pulse is output.

Before writing the driving command, the user should set the parameters for the outputting speed curve and the correct output pulse numbers (see the table below).

○: Required

Parameter	Speed curve to be output				
	Fixed speed	Symmetrical linear acceleration deceleration/deceleration	Non-symmetrical linear acceleration/deceleration	Symmetrical S-curve acceleration/deceleration	Non-symmetrical S-curve acceleration/deceleration
Range (R)	○	○	○	○	○
Jerk (K)				○	○
Deceleration increasing rate (L)					○
Acceleration (A)		○	○	○ (8000)	○ (8000)
Deceleration (D)			○		○ (8000)
Initial speed (SV)	○	○	○	○	○
Drive speed (V)	○	○	○	○	○
Output pulse number (P)	○	○	○	○	○
Manual deceleration point (DP)					○

8.2 −Direction Fixed Pulse Driving

Code	Command
21h	−Direction Fixed Pulse Driving

The setting pulse numbers will be output through the output signal nPM.

In driving, logical position counter will count-down 1 when one pulse is output.

Before writing the driving command, the user should set the parameters for the outputting speed curve and the correct output pulse numbers.

8.3 +Direction Continuous Pulse Driving

Code	Command
22h	+Direction Continuous Pulse Driving

Before the stop command or external signal is active, the pulse numbers will be continuously output through the output signal nPP.

In driving, logical position counter will count-up 1 when one pulse is output.

Before writing the driving command, the user should set the parameters for the outputting speed curve and the correct output pulse numbers.

8.4 −Direction Continuous Pulse Driving

Code	Command
23h	−Direction Continuous Pulse Driving

Before the stop command or external signal is active, the pulse numbers will be continuously output through the output signal nPM.

In driving, logical position counter will count-down 1 when one pulse is output.

Before writing the driving command, the user should set the parameters for the outputting speed curve and the correct output pulse numbers.

8.5 Drive Status Holding

Code	Command
24h	Holding for driving starting

This command is to hold-on the start of driving.

When this command is used for starting multi-axis driving simultaneously, the user may write other commands after the drive status holding command is registered. The drive start holding release command (25h) can be written to start the driving.

In driving, even this command is written, the driving will not be stopped. The next command will be held.

8.6 Drive Status Holding Release / Finishing Status Clear

Code	Command
25h	Drive status holding release / finishing status clearing

This command is to release the drive status holding (24h), and start the driving.

Also, this command can clear the finishing status bits D15 ~ 8 of register RR1.

This command clears an automatic home search IN2 signal error bit D7 (HOME) of the RR2 register.

8.7 Decelerating Stop

Code	Command
26h	Decelerating stop in driving

This command performs the decelerating stop when the drive pulses are outputting.

If the drive speed is lower than the initial speed, the driving will be suddenly stopped when this command is engaged.
In interpolation driving, for main axis, the decelerating stop and sudden stop commands can be written to stop the driving.
Once the driving stops, this command will not work.

8.8 Sudden Stop

Code	Command
27h	Sudden stop in driving

This command performs the sudden stop when the drive pulses are output. Also, the sudden stop can be performed in acceleration / deceleration driving.

Once the driving stops, this command will not work.

9. Interpolation Commands

Interpolation commands consist of the commands for 2 / 3 axes linear interpolation, CW / CCW circular interpolation, 2 / 3 axes bit pattern interpolation and other related commands. There is no need to make the axis assignment in setting bits D11~8 of command register WR0. Please set 0 in those bits.

Two procedures should be follow before the interpolation command is executed:

a. interpolation accessing axes assignment (set-in bits D5~D0 of register WR5)
b. speed parameter setting for master axis

In interpolation driving, bit D8 (I-DRV) of main status register RR0 becomes 1, and will return to 0 when the driving is finished. In interpolation, the n-DRV bit of interpolating axis becomes 1.

[Note] It requires 250 nSEC (maximum) to access the command code when CLK=16MHz. Please write the next command within this period of time.

9.1 2-Axis Linear Interpolation

Code	Command
30h	2-axis linear interpolation

This command performs 2-axis interpolation from present point to finish point.

Before driving, the finish point of the 2 corresponding axes should be set by incremental value.

9.2 3-Axis Linear Interpolation

Code	Command
31h	3-axis linear interpolation

This command performs 3-axis interpolation from present point to finish point.

Before driving, the finish point of the 3 corresponding axes should be set by incremental value.

9.3 CW Circular Interpolation

Code	Command
32h	CW circular interpolation

This command performs 2-axis clockwise circular interpolation, based on center point, from present point to finish point.

Before driving, the finish point of the 2 corresponding axes should be set by incremental value.

A full circle will come out if the finish position is set (0, 0).

9.4 CCW Circular Interpolation

Code	Command
33h	CCW circulator interpolation

This command performs 2-axis counterclockwise circular interpolation, based on center point, from present point to finish point.

Before driving, the finish point of the 2 corresponding axes should be set by incremental value.

A full circle will come out If the finish position is set (0, 0).

9.5 2-Axis Bit Pattern Interpolation

Code	Command
34h	2-axis bit pattern interpolation

This command performs 2-axis bit pattern interpolation.

Before driving, the +/− direction bit data of the two interpolating axes should be set, and the setting bit data of each axis (each direction) is at most 16 × 3 = 48-bit. Once the data is over than 48-bit, those remaining data can be filled during the driving.

9.6 3-Axis Bit Pattern Interpolation Drive

Code	Command
35h	3-axis bit pattern interpolation

This command performs 3-axis bit pattern interpolation.

Before driving, the +/− direction bit data of the two interpolating axes should be set, and the setting bit data of each axis (each direction) is at most 16 × 3 = 48-bit. Once the data is over than 48-bit, those remaining data can be filled during the driving.

9.7 BP Register Data Writing Enabling

Code	Command
36h	BP register data writing enabling

This command enables the bit pattern data writing registers BP1P/M, BP2P/M and PB3P/M.

After this command is issued, the data writing to register nWR1~nWR5 becomes disabled.

The data written to the bit pattern data writing registers is disabled while resetting.

9.8 BP Register Data Writing Disabling

Code	Command
37h	BP register data writing disabling

This command disables the bit pattern data writing registers BP1P/M, BP2P/M and PB3P/M.

After this command is issued, the data writing to register nWR2~nWR5 becomes enabled.

9.9 BP Data Stack

Code	Command
38h	BP data stacking

This command stacks the data of bit pattern data writing registers BP1P/M, BP2P/M, and BP3P/M.

After this command is issued, stack counter (SC) will plus 1. When stack counter (SC) is 3, this command cannot be issued again.

9.10 BP Data Clear

Code	Command
39h	BP data clearing

This command clears all the bit pattern data, and sets the stack counter (SC) to 0.

9.11 Single Step Interpolation

Code	Command
3Ah	Single step interpolation

This command performs 1-pulse (each step) output in interpolation driving.

When D12 bit of register WR5 is set 1, the single step interpolation can be performed. After this command is issued, single step interpolation starts.

9.12 Deceleration Enabling

Code	Command
3Bh	Deceleration enabling

This command enables the automatic and manual decelerations.

In case of the individual interpolation, the user can issue this command before the driving. However, in continuous interpolation, the user should disable the deceleration than start the driving. This command should be put in the final node, and written before the interpolation command of the final node is written. If each axis has to decelerate individually, execute this command before driving. But for continuous interpolation, disable the deceleration first and enable it until the last node.

The deceleration is disabled while resetting. When the deceleration enabling command is issued, the enabling status is kept until the deceleration disabling command (3C) is written, or the reset happens.

Deceleration enabling / disabling is active in interpolation; automatic and manual decelerations are always active when individual axis is in driving.

9.13 Deceleration Disabling

Code	Command
3Ch	Deceleration disabling

This command disables the automatic or manual deceleration in interpolation.

9.14 Interpolation Interrupt Clear

Code	Command
3Dh	Interpolation interrupt clear

This command clears the interrupt in bit pattern or continuous interpolation.

After the bit D15 of WR5 is set to 1 in bit pattern interpolation, the stack counter (SC) is changed from 2 to 1, and the interrupt will be generated. In continuous interpolation, when the bit D14 of WR5 is set to 1, the interrupt will be generated when it is ready to write the interpolation data for next node.

10. Other Commands

[Notes] The maximum time required for command processing is 250nsec (CLK=16MHz). The following commands must be written before this time has elapsed.

10.1 Automatic Home Search Execution

Code	Command
62h	Automatic home search execution

This command executes automatic home search.
Before execution of the command, the automatic home search mode and correct parameters must be set. See Section 2.5 for details of automatic home search.

10.2 Deviation Counter Clear Output

Code	Command
63h	Deviation counter clear output

This command outputs deviation counter clear pulses from the nDRIVE/DCC output pin.
Before issuing this command, set Output Enable, a pulse logical level, and a pulse width in using the extension mode setting command. See Sections 2.5.2 and 2.5.3 for details.

10.3 Synchronous Action Activation

Code	Command
65h	Synchronous action activation

This command activates synchronous action.
The WR6/D9 (CMD) bit of the activation factor must be set to 1 in advance using the synchronous action mode setting command. See Section 2.6 for details of synchronous action.

10.4 NOP (for Axis Switching)

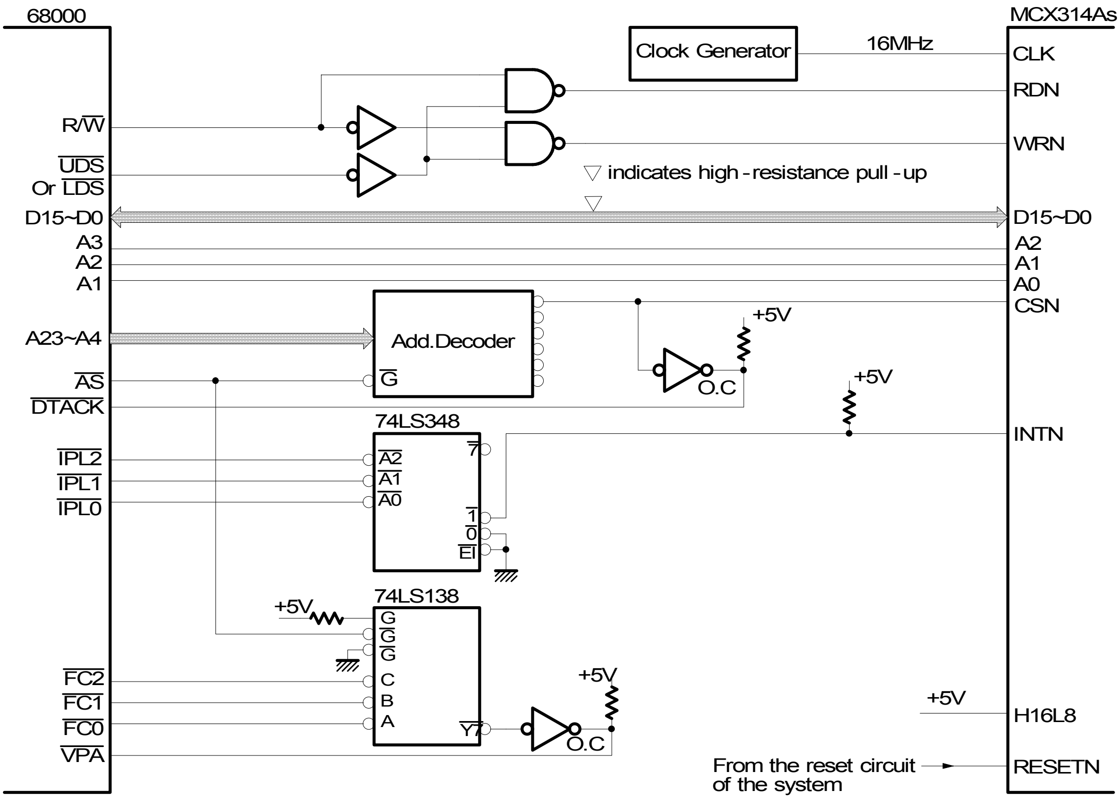
Code	Command
0Fh	NOP (for axis switching)

No execution is performed.

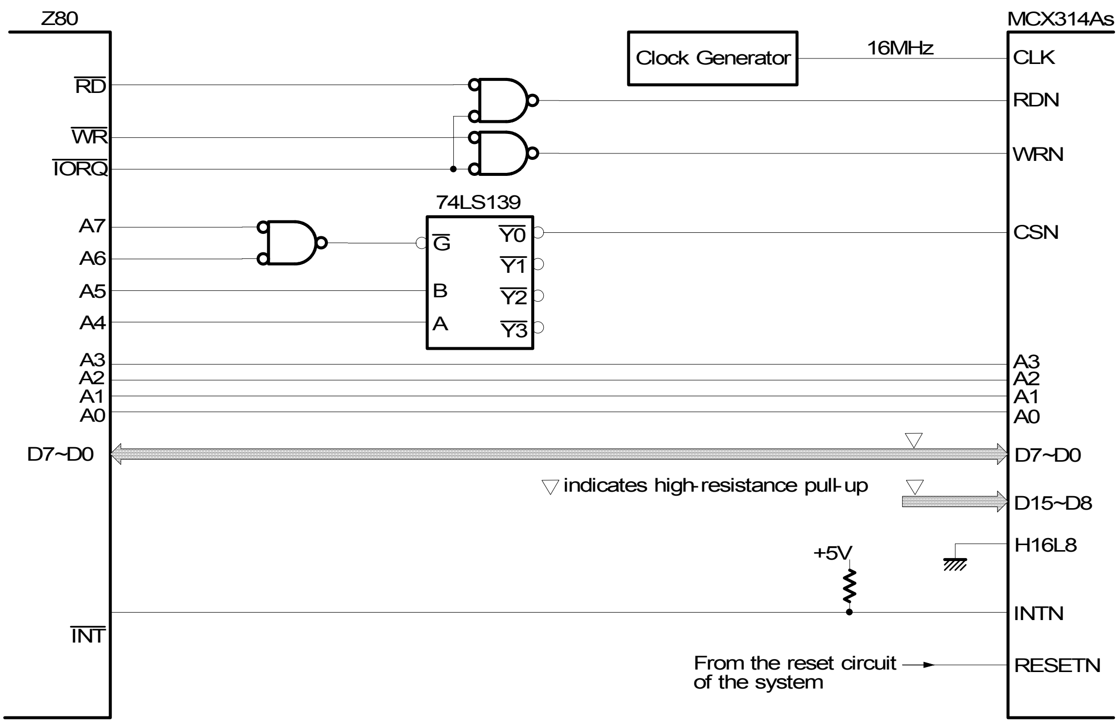
Use this command for switching the axis for selecting the registers from WR1 to WR3 registers RR1to RR3.

11. Connection Examples

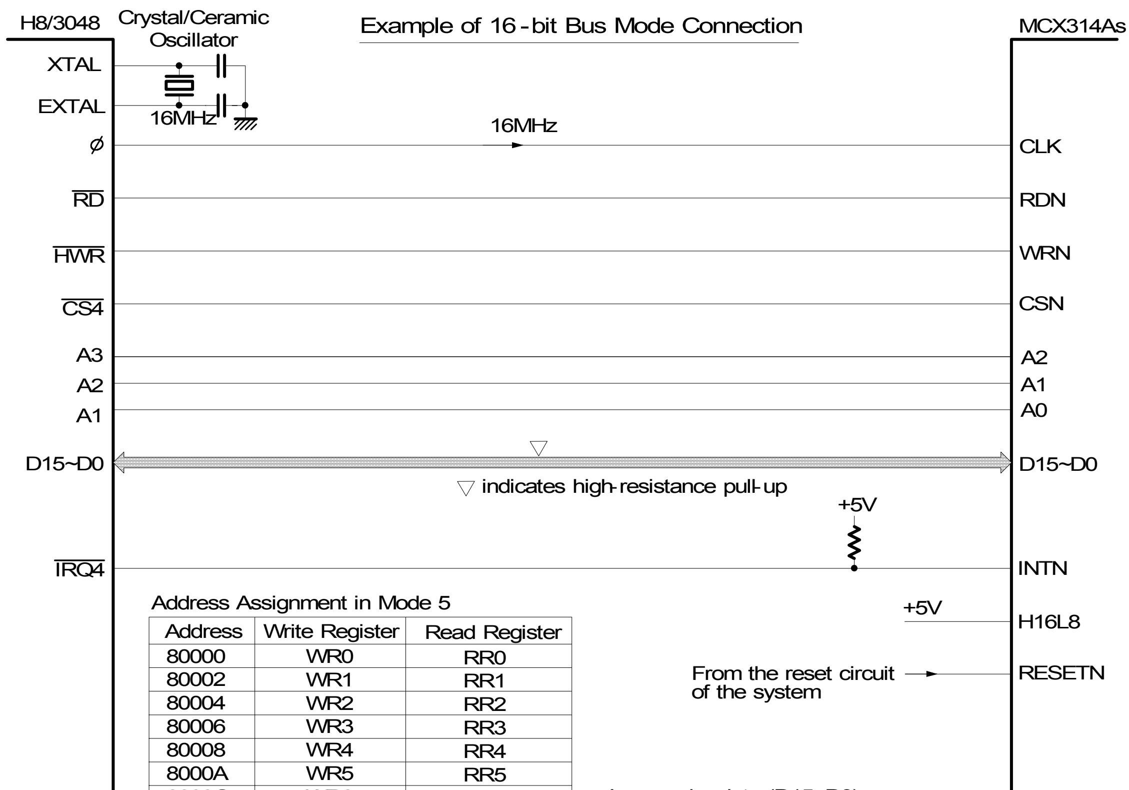
11.1 Connection Example for 68000 CPU



11.2 Connection Example for Z80 CPU

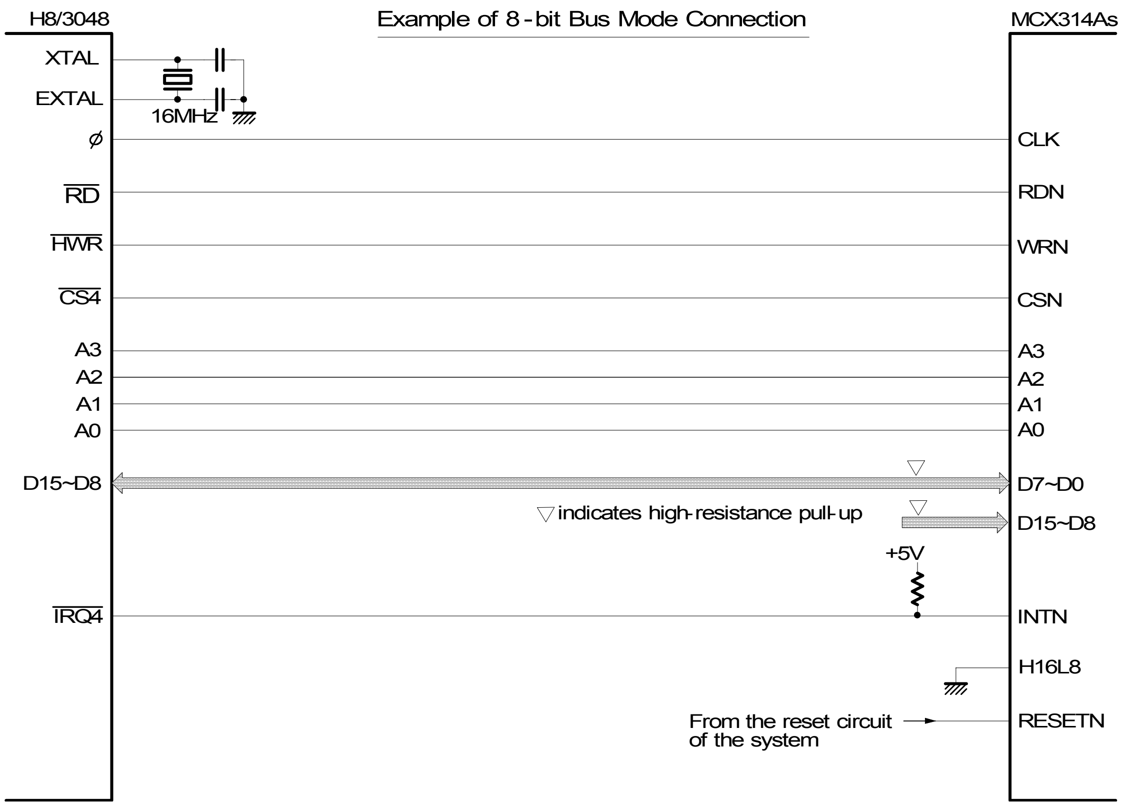


11.3 Example of Connection with H8 CPU



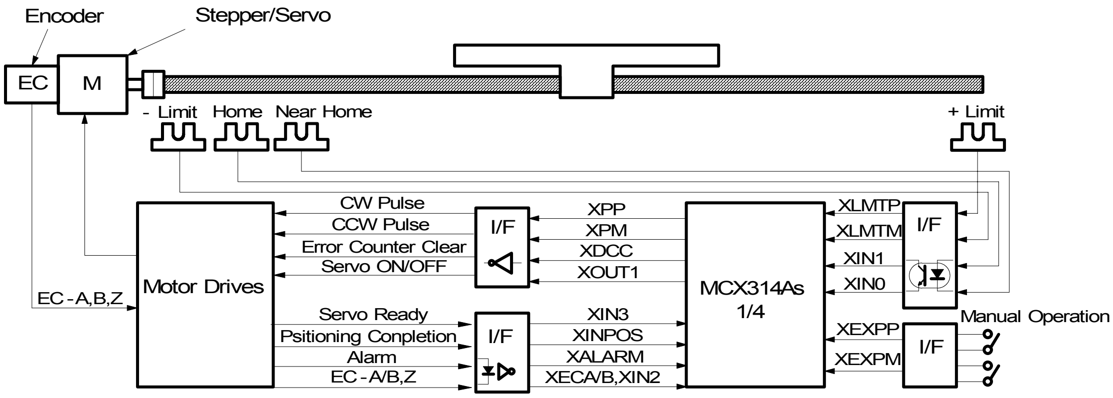
Address Assignment in Mode 5

Address	Write Register	Read Register
80000	WR0	RR0
80002	WR1	RR1
80004	WR2	RR2
80006	WR3	RR3
80008	WR4	RR4
8000A	WR5	RR5
8000C	WR6	RR6
8000E	WR7	RR7



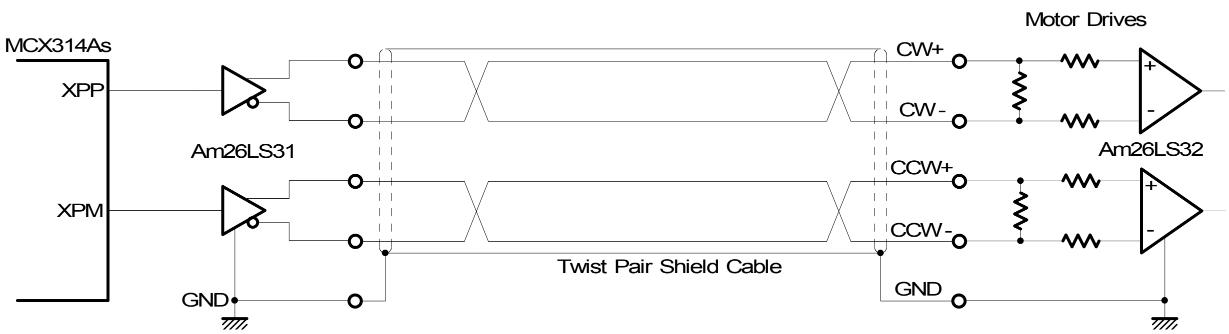
11.4 Connection Example

The figure shown below illustrates the example of 1-axis driving system. 4 axes can be assigned in the same way.

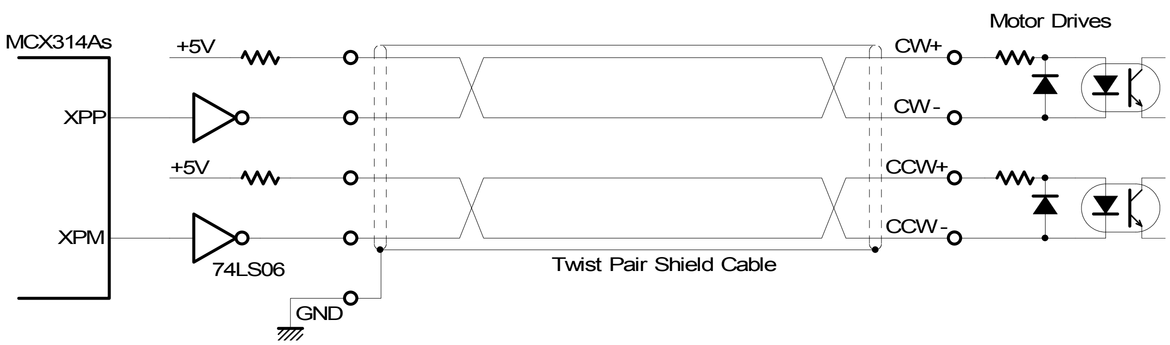


11.5 Pulse Output Interface

■ Output to Motor Drivers in Differential Circuit



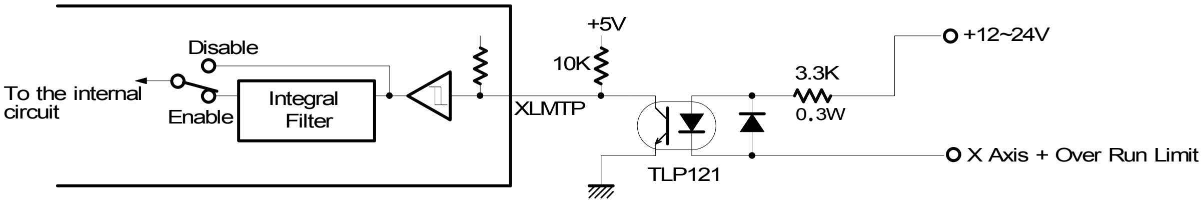
■ Open Collector TTL Output



For drive pulse output signals, we recommend the user to use twist pair shield cable due to the concern of EMC.

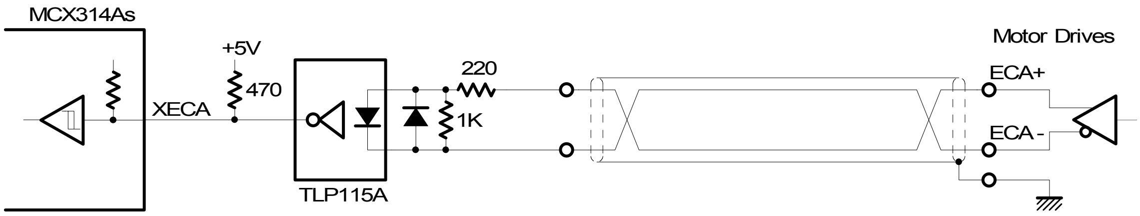
11.6 Connection Example for Input Signals

Limit signals often pick up some noise since complicated cabling is normally involved. A photo coupler alone may not be able to absorb this noise. Enable the filter function in the IC and set an appropriate time constant (FL=2,3).



11.7 Connection Example for Encoder

The following diagram is the example for the encoder signal which is differential line-drive output, Then, this signal can be received through the high speed photo coupler IC which can direct it to MCX314As.



12. Example Program

The example of C program for MCX314As is shown in this section. This is a 16-bit bus configuration program.
This program can be downloaded from our home page (http://www.novaelec.co.jp/). File name：MCX314AML.C

```
#include        <stdio.h>
#include        <conio.h>

// ----- MCX314As register address definition -----

#define     adr     0x2a0       // Basic address

#define     wr0     0x0         //Command register
#define     wr1     0x2         //Mode register 1
#define     wr2     0x4         //Mode register 2
#define     wr3     0x6         //Mode register 3
#define     wr4     0x8         //Output register
#define     wr5     0xa         //Interpolation mode register
#define     wr6     0xc         //Low word bits data writing register
#define     wr7     0xe         //High word bits data writing register

#define     rr0     0x0         //Main status register
#define     rr1     0x2         //Status register 1
#define     rr2     0x4         //Status register 2
#define     rr3     0x6         //Status register 3
#define     rr4     0x8         //Input register 1
#define     rr5     0xa         //Input register 2
#define     rr6     0xc         //Low word bits data reading register
#define     rr7     0xe         //High word bits data reading register

#define     bp1p    0x4         //BP + direction data register for the first axis control
#define     bp1m    0x6         //BP − direction data register for the first axis control
#define     bp2p    0x8         //BP + direction data register for the second axis control
#define     bp2m    0xa         //BP − direction data register for the second axis control
#define     bp3p    0xc         //BP + direction data register for the third axis control
#define     bp3m    0xe         //BP − direction data register for the third axis control

// wreg 1 (axis assignment, data) ----Write register 1 setting

void wreg1(int axis,int wdata)
        {
        outpw(adr+wr0, (axis << 8) + 0xf);//axis assignment
        outpw(adr+wr1, wdata);
        }

// wreg 2 (axis assignment, data) ----Write register 2 setting

void wreg2(int axis,int wdata)
        {
        outpw(adr+wr0, (axis << 8) + 0xf);//axis assignment
        outpw(adr+wr2, wdata);
        }

// wreg 3 (axis assignment, data) -----Write register 3 setting

void wreg3(int axis,int wdata)
        {
        outpw(adr+wr0, (axis << 8) + 0xf);//axis assignment
        outpw(adr+wr3, wdata);
        }

// command (axis assignment, data) -----For writing commands

void command(int axis,int cmd)
        {
        outpw(adr+wr0, (axis << 8) + cmd);
        }

// range(axis assignment, data) -----For range (R) setting

void range(int axis,long wdata)
        {
        outpw(adr+wr7, (wdata >> 16) & 0xffff);
        outpw(adr+wr6, wdata & 0xffff);
        outpw(adr+wr0, (axis << 8) + 0x00);
        }

// acac(axis assignment, data) -----For S-curve Deceleration increasing rate (L) setting

void acac(int axis,int wdata)
        {
        outpw(adr+wr6, wdata);
        outpw(adr+wr0, (axis << 8) + 0x01);
        }
```

```
// dcac(axis assignment, data) ----- For jerk (L) setting

void dcac(int axis,int wdata)
        {
        outpw(adr+wr6, wdata);
        outpw(adr+wr0, (axis << 8) + 0x0e);
        }

// acc(axis assignment, data) -----For acceleration/deceleration (A) setting

void acc(int axis,int wdata)
        {
        outpw(adr+wr6, wdata);
        outpw(adr+wr0, (axis << 8) + 0x02);
        }

// dec( axis assignment, data) -----For deceleration (D) setting

void dec(int axis,int wdata)
        {
        outpw(adr+wr6, wdata);
        outpw(adr+wr0, (axis << 8) + 0x03);
        }

// startv(axis assignment, data) -----For initial speed (SV) setting

void startv(int axis,int wdata)
        {
        outpw(adr+wr6, wdata);
        outpw(adr+wr0, (axis << 8) + 0x04);
        }

// speed(axis assignment, data) -----For drive speed (V) setting

void speed(int axis,int wdata)
        {
        outpw(adr+wr6, wdata);
        outpw(adr+wr0, (axis << 8) + 0x05);
        }

// pulse( axis assignment, data) -----For output pulse output/finish point (P) setting

void pulse(int axis,long wdata)
        {
        outpw(adr+wr7, (wdata >> 16) & 0xffff);
        outpw(adr+wr6, wdata & 0xffff);
        outpw(adr+wr0, (axis << 8) + 0x06);
        }

// decp(axis assignment, data) -----For manual deceleration (DP) setting

void decp(int axis,long wdata)
        {
        outpw(adr+wr7, (wdata >> 16) & 0xffff);
        outpw(adr+wr6, wdata & 0xffff);
        outpw(adr+wr0, (axis << 8) + 0x07);
        }

// center(axis assignment, data) -----For circular center point (C) setting

void center(int axis,long wdata)
        {
        outpw(adr+wr7, (wdata >> 16) & 0xffff);
        outpw(adr+wr6, wdata & 0xffff);
        outpw(adr+wr0, (axis << 8) + 0x08);
        }

// lp(axis assignment, data) -----For logical position counter (LP ) setting

void lp(int axis,long wdata)
        {
        outpw(adr+wr7, (wdata >> 16) & 0xffff);
        outpw(adr+wr6, wdata & 0xffff);
        outpw(adr+wr0, (axis << 8) + 0x09);
        }

// ep(axis assignment, data) -----For real position counter (EP) setting

void ep(int axis,long wdata)
        {
        outpw(adr+wr7, (wdata >> 16) & 0xffff);
        outpw(adr+wr6, wdata & 0xffff);
        outpw(adr+wr0, (axis << 8) + 0x0a);
        }
```

```
// compp(axis assignment, data) -----For COMP+ (CP) setting

void compp(int axis,long wdata)
        {
        outpw(adr+wr7, (wdata >> 16) & 0xffff);
        outpw(adr+wr6, wdata & 0xffff);
        outpw(adr+wr0, (axis << 8) + 0x0b);
        }

// compm(axis assignment, data) -----For COMP − (CM) setting

void compm(int axis,long wdata)
        {
        outpw(adr+wr7, (wdata >> 16) & 0xffff);
        outpw(adr+wr6, wdata & 0xffff);
        outpw(adr+wr0, (axis << 8) + 0x0c);
        }

// accofst(axis assignment, data) ----For acceleration counter shift (AO) setting

void accofst(int axis,long wdata)
        {
        outpw(adr+wr7, (wdata >> 16) & 0xffff);
        outpw(adr+wr6, wdata & 0xffff);
        outpw(adr+wr0, (axis << 8) + 0x0d);
        }

// hsspeed(axis assignment, data) ------------------- Home Search Speed (HV) setting

void hsspeed(int axis,int wdata)
        {
        outpw(adr+wr6, wdata);
        outpw(adr+wr0, (axis << 8) + 0x61);
        }

// expmode(axis assignment, data) ------------------- Expansion Mode (EM) setting

void expmode(int axis,int em6data,int em7data)
        {
        outpw(adr+wr6, em6data);
        outpw(adr+wr7, em7data);
        outpw(adr+wr0, (axis << 8) + 0x60);
        }

// syncmode(axis assignment, data) ------------------ Synchronous Mode (SM) setting

void syncmode(int axis,int sm6data,int sm7data)
        {
        outpw(adr+wr6, sm6data);
        outpw(adr+wr7, sm7data);
        outpw(adr+wr0, (axis << 8) + 0x64);
        }

// readlp(axis assignment) -----For logical position counter (LP) reading

long readlp(int axis)
        {
        long a;long d6;long d7;
        outpw(adr+wr0, (axis << 8) + 0x10);
        d6 = inpw(adr+rr6);d7 = inpw(adr+rr7);
        a = d6 + (d7 << 16);
        return(a);
        }

// readep(axis assignment) -----For real position counter (EP) reading

long readep(int axis)
        {
        long a;long d6;long d7;
        outpw(adr+wr0, (axis << 8) + 0x11);
        d6 = inpw(adr+rr6);d7 = inpw(adr+rr7);
        a = d6 + (d7 << 16);
        return(a);
        }

// wait(axis assignment) -----For waiting for drive stop

void wait(int axis)
        {
        while(inpw(adr+rr0) & axis);
        }

// next_wait() -----Next data setting of waiting for continuous interpolation

void next_wait(void)
        {
        while((inpw(adr+rr0) & 0x0200) == 0x0);
        }
```

```
// bp_wait() ----- Next data setting of waiting for BP interpolation

void bp_wait(void)
        {
        while((inpw(adr+rr0) & 0x6000) == 0x6000);
        }

//   home search() ------------------------------- All axes home search
//
//      ------ X axis home search ---------------------------------
//      Step1   Near home (IN0) signal high-speed search in the – direction at 20,000pps
//      Step2   Home (IN1) signal low-speed search in the – direction at 500pps
//      Step3   Z-phase (IN2) signal low-speed search in the – direction at 500pps
//              Deviation counter clear output at Z-phase search
//      Step4   3500 pulse offset high-speed drive in the + direction at 20,000pps
//
//      ------ Y axis home search ---------------------------------
//      Step1   Near home (IN0) signal high-speed search in the – direction at 20,000pps
//      Step2   Home (IN1) signal low-speed search in the – direction at 500pps
//      Step3   Z-phase (IN2) signal low-speed search in the – direction at 500pps
//              Deviation counter clear output at Z-phase search
//      Step4   700 pulse offset high-speed drive in the + direction at 20,000pps
//
//      ------ Z axis home search ---------------------------------
//      Step1   High-speed search: None
//      Step2   Home (IN1) signal low-speed search in the + direction at 400pps
//      Step3   Z-phase search: None
//      Step4   20 pulse offset drive in the – direction at 400pps
//
//      ------ U axis home search ---------------------------------
//      Step1   High-speed search: None
//      Step2   Home (IN1) signal low-speed search in the – direction at 300pps
//      Step3   Z-phase search: None
//      Step4   Offset drive: None
//

void homesrch(void)
        {
                                        // X and Y axes home search parameter setting
                                        // (See the initial setting of main for mode setting)
        speed(0x3,2000);                // Step1 and 4 High speed: 20000pps
        hsspeed(0x3,50);                // Step2 and 3 Low speed: 500pps
        pulse(0x1,3500);                // X axis offset: 3500 pulse
        pulse(0x2,700);                 // Y axis offset: 700 pulse

                                        // Z axis home search parameter setting
        speed(0x4,40);                  // Step4 drive speed:    400pps
        hsspeed(0x4,40);                // Step2 search speed: 400pps
        pulse(0x4,20);                  // Offset:20 pulses

                                        // U axis home search parameter setting
        hsspeed(0x8,30);                // Step2 search speed: 300pps

        command(0xf,0x62);              // Execution of automatic home search for all the axes
        wait(0xf);                      // Waits for termination of all the axes

        if(inpw(adr+rr0) & 0x0010)      // Error display
                {
                printf("X-axis Home Search Error ¥n");
                }
        if(inpw(adr+rr0) & 0x0020)
                {
                printf("Y-axis Home Search Error ¥n");
                }
        if(inpw(adr+rr0) & 0x0040)
                {
                printf("Z-axis Home Search Error ¥n");
                }
        if(inpw(adr+rr0) & 0x0080)
                {
                printf("U-axis Home Search Error ¥n");
                }
        }
```

```
void main(void)
        {

        int    count;
        outpw(adr+wr0, 0x8000);                 //Software reset
        for(count = 0; count < 2; ++count);

        command(0x3,0xf);                       //------ X and Y axes mode setting ---------

        outpw(adr+wr1, 0x0000);                 //Mode register 1
                                                //D15 ~ 8: 0 All the interrupt disabled
                                                //D7: 0 IN3 signal: Disable
                                                //D6: 0 IN3 signal logic: Low Active
                                                //D5: 0 IN2 signal: Disable
                                                //D4: 0 IN2 signal logic: Low Active
                                                //D3: 0 IN1 signal: Disable
                                                //D2: 0 IN1 signal logic: Low Active
                                                //D1: 0 IN0 signal: Disable
                                                //D0: 0 IN0 signal logic: Low Active

        outpw(adr+wr2, 0xe000);                 //Mode register 2
                                                //D15:1 INPOS input: Enable
                                                //D14:1 INPOS input logic: High active
                                                //D13:1 ALARM input: Enable
                                                //D12:0 ALARM input logic: Low active
                                                //D11:0
                                                //D10:0 Encoder input division: 1/1
                                                //D9: 0 Encoder input mode" 2-phase pulse
                                                //D8: 0 Drive pulse direction logic:
                                                //D7: 0 Drive pulse logic: Positive logic
                                                //D6: 0 Drive pulse mode: 2 pulse
                                                //D5: 0 COMP target: Logical position counter
                                                //D4: 0 – over run limit logic: Low Active
                                                //D3: 0 + over run limit logic: Low Active
                                                //D2: 0 Over run limit stop mode: Decelerating stop
                                                //D1: 0 Software over run limit－:Disable
                                                //D0: 0 Software over run limit＋:Disable

        outpw(adr+wr3, 0x0000);                 //Mode register 3
                                                //D15 ~ 12:0000
                                                //D11:0 General purpose output OUT7:Low
                                                //D10:0 General purpose output OUT6:Low
                                                //D9: 0 General purpose output OUT5:Low
                                                //D8: 0 General purpose output OUT4:Low
                                                //D7: 0 Drove state output:Disable
                                                //D6: 0
                                                //D5: 0
                                                //D4: 0 External operation signal operation:Disable
                                                //D3: 0
                                                //D2: 0 Acceleration/deceleration curve: Linear acceleration (trapezoid)
                                                //D1: 0 Acceleration/deceleration symmetry/non-symmetry: Symmetry
                                                //D0: 0 Fixed pulse drive deceleration: Automatic deceleration

        expmode(0x3,0x5d08,0x497f);             //Extension mode
                                                //[Input signal filter and others]
                                                //W6/D15 ~ 13:010 Input signal filter delay:512μ
                                                //W6/D12:1 IN3 signal filter: Enable
                                                //W6/D11:1 EXPP, EXPM, and EXPLS filter:Enable
                                                //W6/D10:1 INPOS and ALARM signal filter:Enable
                                                //W6/D9: 0 IN2 signal filter:Disable
                                                //W6/D8: 1 EMGN,LMTP/M,IN1, and 0 filter:Enable
                                                //W6/D7: 0
                                                //W6/D6: 0
                                                //W6/D5: 0 Automatic home search termination interrupt: Prohibit
                                                //W6/D4: 0 LP/EP variable ring function:Disable
                                                //W6/D3: 1 Triangle form prevention at linear acceleration:Enable
                                                //W6/D2: 0 Pulse output switching:Disable
                                                //W6/D1: 0 EP increase/decrease inversion:Disable
                                                //W6/D0: 0 EP clear by IN2 signal:Disable

                                                //[Automatic home search mode]
                                                //W7/D15 ~ D13    010 Deviation counter clear pulse width: 100μsec
                                                //W7/D12          0 Deviation counter clear output logical level: Hi
                                                //W7/D11          1 Deviation counter clear output: Enable
                                                //W7/D10          0 Use of limit signal as a home signal: Disable
                                                //W7/D9           0 Z-phase signal AND home signal: Disable
                                                //W7/D8           1 Logical/real position counter clear: Enable
                                                //W7/D7           0 Step 4 driving direction:  ＋direction
                                                //W7/D6           1 Step 4:                    Enable
                                                //W7/D5           1 Step 3 search direction:   －direction
                                                //W7/D4           1 Step 3:                    Enable
                                                //W7/D3           1 Step 2 search direction:   －direction
                                                //W7/D2           1 Step 2:                    Enable
                                                //W7/D1           1 Step 1 search direction:   －direction
                                                //W7/D0           1 Step 1:                    Enable
```

```
                                    //------ X and Y axes operation parameter initial setting --
    accofst(0x3,0);                 // AO = 0
    range(0x3,800000);              // R = 800000 （Multiple = 10）
    acac(0x3,1010);                 // K = 1010     (Jerk = 619KPPS/SEC2)
    dcac(0x3,1010);                 // L = 1010     (Deceleration increasing rate = 619KPPS/SEC2)
    acc(0x3,100);                   // A = 100      (Accleration/deceleration = 125KPPS/SEC)
    dec(0x3,100);                   // D = 100      (Deceleration = 125KPPS/SEC)
    startv(0x3,100);                // SV= 100      (Initial speed = 1000PPS)
    speed(0x3,4000);                // V = 4000     (Drive speed = 40000PPS)
    pulse(0x3,100000);              // P = 100000 (Output pulse number = 100000)
    lp(0x3,0);                      // LP= 0        (Logical position counter = 0)
    ep(0x3,0);                      // EP= 0        (Real/logical position counter = 0)

    command(0xc,0xf);               //------ Z and U axes mode setting ---------

    outpw(adr+wr1, 0x0000);         //Mode register 1
                                    //D15 ~ 8: 0 Prohibit all interrupts
                                    //D7: 0 IN3 signal: Disable
                                    //D6: 0 IN3 signal logic: Low Active
                                    //D5: 0 IN2 signal: Disable
                                    //D4: 0 IN2 signal logic: Low Active
                                    //D3: 0 IN1 signal: Disable
                                    //D2: 0 IN1 signal logic: Low Active
                                    //D1: 0 IN0 signal: Disable
                                    //D0: 0 IN0 signal logic: Low Active

    outpw(adr+wr2, 0x0000);         //Mode register 2
                                    //D15:0 INPOS input:Disable
                                    //D14:0 INPOS input logic:Low Active
                                    //D13:0 ALARM input:Disable
                                    //D12:0 ALARM input logic:Low Active
                                    //D11:0
                                    //D10:0 Encoder input division:1/1
                                    //D9: 0 Encoder input mode: 2-phase pulse
                                    //D8: 0 Driver pulse direction logic:
                                    //D7: 0 Driver pulse logic: Positive logic
                                    //D6: 0 Drive pulse mode: 2-pluse
                                    //D5: 0 COMP target:Logical position counter
                                    //D4: 0 － over run limit logic: Low Active
                                    //D3: 0 ＋ over run limit logic:Low Active
                                    //D2: 0 Over run limit stop mode: Decelerating stop
                                    //D1: 0 Software limit:Disable
                                    //D0: 0 Software limit＋:Disable

  outpw(adr+wr3, 0x0000);           //Mode register 3
                                    //D15 ~ 12:0000
                                    //D11:0 General purpose output OUT7:Low
                                    //D10:0 General purpose output OUT6:Low
                                    //D9: 0 General purpose output OUT5:Low
                                    //D8: 0 General purpose output OUT4:Low
                                    //D7: 0 Drive state output:Disable
                                    //D6: 0
                                    //D5: 0
                                    //D4: 0 External operation signal operation:Disable
                                    //D3: 0
                                    //D2: 0 Accleration/deceleration curve: Linear acceleration (trapezoid)
                                    //D1: 0 Accleration/deceleration symmetry/non-symmetry: Symmetry
                                    //D0: 0 Deceleration of fixed pulse driving: Automatic deceleration

                                    //Since the automatic home search is different between the Z axis and U
                                      axis,
                                    //set the following extension modes individually.

    expmode(0x4,0x5d08,0x01c4);     //Z axis extension mode
                                    //[Input signal filter and others]
                                    //W6/D15 ~ 13:010 Input signal filter delay:512μ
                                    //W6/D12:1 IN3 signal filter:Enable
                                    //W6/D11:1 EXPP,EXPM, and EXPLS filters:Enable
                                    //W6/D10:1 INPOS and ALARM signal filters:Enable
                                    //W6/D9: 0 IN2 signal filter:Disable
                                    //W6/D8: 1 EMGN,LMTP/M,IN1, and 0 filters:Enable
                                    //W6/D7: 0
                                    //W6/D6: 0
                                    //W6/D5: 0 Automatic home search termination interrupt：Prohibit
                                    //W6/D4: 0 LP/EP variable link function:Disable
                                    //W6/D3: 1 Triangle form prevention at linear acceleration:Enable
                                    //W6/D2: 0 Switching pulse output:Disable
                                    //W6/D1: 0 EP increase/decrease inversion:Disable
                                    //W6/D0: 0 EP clear by the IN2 signal:Disable
```

```
                                        //[Automatic home search mode]
                                        //W7/D15 ~ D13   000 Deviation counter clear pulse width：
                                        //W7/D12         0 Deviation counter clear output logical level：
                                        //W7/D11         0 Deviation counter clear output：  Disable
                                        //W7/D10         0 Using an over run limit signal as a home signal: Disable
                                        //W7/D9          0 Z-phase signal AND home signal：      Disable
                                        //W7/D8          1 Logical/Real position counter clear：Enable
                                        //W7/D7          1 Step 4 drive direction:    －direction
                                        //W7/D6          1 Step 4:                    Enable
                                        //W7/D5          0 Step 3 search direction:
                                        //W7/D4          0 Step 3:                    Disable
                                        //W7/D3          0 Step 2 search direction:  ＋direction
                                        //W7/D2          1 Step 2:                    Enable
                                        //W7/D1          0 Step 1 search direction:
                                        //W7/D0          0 Step 1:                    Disable

        expmode(0x8,0x5d08,0x010c);     //U axis extension mode
                                        //[Input signal filter and others]
                                        //W6/D15 ~ 13:010 Input signal filter delay:512μ
                                        //W6/D12:1 IN3 signal filter:Enable
                                        //W6/D11:1 EXPP, EXPM, and EXPLS filters:Enable
                                        //W6/D10:1 INPOS and ALARM signal fiters:Enable
                                        //W6/D9: 0 IN2 signal filter:Disable
                                        //W6/D8: 1 EMGN,LMTP/M, IN1, 0 and filters:Enable
                                        //W6/D7: 0
                                        //W6/D6: 0
                                        //W6/D5: 0 Automatic home search termination interrupt：Prohibit
                                        //W6/D4: 0 LP/EP variable range function:Disable
                                        //W6/D3: 1 Prevention of triangle forms at linear acceleration:Enable
                                        //W6/D2: 0 Replacement of pulse output:Disable
                                        //W6/D1: 0 EP Increase/decrease inversion:Disable
                                        //W6/D0: 0 EP clearing by the IN2 signal:Disable

                                        //[Automatic home search mode]
                                        //W7/D15 ~ D13   000 Deviation counter clear pulse width：
                                        //W7/D12         0 Deviation counter clear output logical level：
                                        //W7/D11         0 Deviation counter clear output：  Disable
                                        //W7/D10         0 Use of an over run limit signal as a home signal:   Disable
                                        //W7/D9          0 Z-phase signal AND home signal：      Disable
                                        //W7/D8          1 Logical/real position counter clear：Enable
                                        //W7/D7          0 Step 4 drive direction:
                                        //W7/D6          0 Step 4:                    Disable
                                        //W7/D5          0 Step 3 search direction:
                                        //W7/D4          0 Step 3:                    Disable
                                        //W7/D3          1 Step 2 search direction:  －direction
                                        //W7/D2          1 Step 2:                    Enable
                                        //W7/D1          0 Step 1 search direction:
                                        //W7/D0          0 Step 1:                    Disable

                                        //------ Z and U axes operation parameter initialization --
        accofst(0xc,0);                 // AO = 0
        range(0xc,800000);              // R = 800000 (Multiple = 10)
        acac(0xc,1010);                 // K = 1010     (Jerk = 619KPPS/SEC2)
        dcac(0xc,1010);                 // L = 1010     (Deceleration increasing rate = 619KPPS/SEC2)
        acc(0xc,100);                   // A = 100      (Accleration/deceleration = 125KPPS/SEC)
        dec(0xc,100);                   // D = 100      (Deceleration = 125KPPS/SEC)
        startv(0xc,50);                 // SV= 50       (Initial speed = 500PPS)
        speed(0xc,40);                  // V = 40       (Drive speed = 400PPS)
        pulse(0xc,10);                  // P = 10       (Output pulse number = 10)
        lp(0xc,0);                      // LP= 0        (Logical position counter = 0)

                                        //------ General purpose output register initialization --
        outpw(adr+wr4, 0x0000);         // 00000000 00000000

                                        //------ Interpolation mode register initialization --
        outpw(adr+wr5, 0x0124);         // 00000001 00100100
                                        // ax1=x, ax2=y, ax3=z, Constant vector speed

                                        //------ Start driving ----------------

//      homesrch();                     //------ Home search for all axes --------------

                                        //------ X and Y axes linear acceleration driving ----
        acc(0x3,200);                   // A = 200      (Accleration/deceleration = 250KPPS/SEC)
        speed(0x3,4000);                // V = 4000     (Drive speed = 40000PPS)
        pulse(0x1,80000);               // xP = 80000
        pulse(0x2,40000);               // yP = 40000
        command(0x3,0x20);              // + fixed pulse drive
        wait(0x3);                      // Waits for termination of driving

                                        //------ X axis non-symmetrical linear acceleration driving ----
        wreg3(0x1, 0x0002);             //Acceleration/deceleration individual (non-symmetrical) mode
        acc(0x1,200);                   // xA = 200     (Accleration/deceleration = 250KPPS/SEC)
        dec(0x1,50);                    // xD = 50      (Deceleration = 62.5KPPS/SEC)
        speed(0x1,4000);                // xV = 4000    (Drive speed = 40000PPS)
        pulse(0x1,80000);               // xP = 80000
        command(0x1,0x20);              // + fixed pulse drive
        wait(0x1);                      // Waits for termination of driving
        wreg3(0x1, 0x0000);             // Release of acceleration/deceleration individual mode

                                        //------ X and Y axes S-curve acceleration/deceleration driving ----
```

```
wreg3(0x3, 0x0004);            //S-curve mode
acac(0x3,1010);                // K = 1010     (Jerk = 619KPPS/SEC2)
acc(0x3,200);                  // A = 200      (Accleration/deceleration = 250KPPS/SEC)
speed(0x3,4000);               // V = 4000     (Drive speed = 40000PPS)
pulse(0x1,50000);              // xP = 50000
pulse(0x2,25000);              // yP = 25000
command(0x3,0x21);             //Fixed pulse drive
wait(0x3);
wreg3(0x3, 0x0000);            //Release of S-curve acceleration/deceleration

                               //------ Z-axis constant speed drive ----
startv(0x4,40);                // SV= 40       (Initial speed = 400PPS)
speed(0x4,40);                 // V = 40       (Drive speed = 400PPS)
pulse(0x4,700);                // P = 700
command(0x4,0x20);             // + fixed pulse drive
wait(0x4);                     // (Moving 700 pulses in the + direction at 400pp)
pulse(0x4,350);                // P = 350
command(0x4,0x21);             // Fixed pulse drive
wait(0x4);                     // (Moves 350 pulses in the – direction at 400pps)

                               //------ X and Y axes linear interpolation driving ----
outpw(adr+wr5, 0x0124);        // ax1=x, ax2=y,ax3=z, Constant vector speed
range(0x1,800000);             // ax1/R = 800000 (Multiple = 10)
range(0x2,1131371);            // ax2/R = 800000×1.414
speed(0x1,100);                // ax1/V = 100 (Drive speed = 1000PPS constant speed)
pulse(0x1,5000);               // xP = +5000   (Finish point X = +5000)
pulse(0x2,-2000);              // yP = -2000   (Finish point Y = −2000)
command(0x0,0x30);             // 2-axes linear interpolation
wait(0x3);

                               //------ X and Y axes circular interpolation driving ----
outpw(adr+wr5, 0x0124);        // ax1=x, ax2=y,ax3=z, Constant vector speed
range(0x1,800000);             // ax1/R = 800000 (Multiple = 10)
range(0x2,1131371);            // ax2/R = 800000×1.414
speed(0x1,100);                // ax1/V = 100 (Drive speed = 1000PPS constant speed)
center(0x1,-5000);             // xC = -5000   (Center point X = −5000)
center(0x2,0);                 // yC = 0       (Center point Y =   0    )
pulse(0x1,0);                  // xP = 0       (Finish point X =   0) full circle
pulse(0x2,0);                  // yP = 0       (Finish point Y =   0)
command(0x0,0x33);             // CCW circular interpolation
wait(0x3);

                               //------ X and Y axes bit pattern interpolation (Figure 2.32 Example) ----
speed(0x1,1);                  // ax1/V = 1 (Drive speed = 10PPS constant speed)
command(0,0x36);               // Enables bit pattern data write

outpw(adr+bp1p, 0x0000);       // 0 ~ 15   Writes bit patter data
outpw(adr+bp1m, 0x2bff);
outpw(adr+bp2p, 0xffd4);
outpw(adr+bp2m, 0x0000);
command(0,0x38);               // Stack

outpw(adr+bp1p, 0xf6fe);       // 16 ~ 31   Writes bit data
outpw(adr+bp1m, 0x0000);
outpw(adr+bp2p, 0x000f);
outpw(adr+bp2m, 0x3fc0);
command(0,0x38);

outpw(adr+bp1p, 0x1fdb);       // 32 ~ 47   Writes bit data
outpw(adr+bp1m, 0x0000);
outpw(adr+bp2p, 0x00ff);
outpw(adr+bp2m, 0xfc00);
command(0,0x38);

command(0,0x34);               // Starts 2-axes BP interpolation driving

bp_wait();                     // Waits for data write

outpw(adr+bp1p, 0x4000);       // 48 ~ 63   Writes bit data
outpw(adr+bp1m, 0x7ff5);
outpw(adr+bp2p, 0x0000);
outpw(adr+bp2m, 0x0aff);
command(0,0x38);

command(0,0x37);               // Disables bit pattern data write

wait(0x3);                     // Waits for termination of driving

                               //------ X and Y axes continuous interpolation (Figure 2.37 Example) ----
speed(0x1,100);                // ax1/V = 100 (Drive speed = 1000PPS constant speed)

pulse(0x1,4500);               // Seg 1
pulse(0x2,0);
command(0,0x30);

next_wait();                   //Waits for next data set
center(0x1,0);                 // Seg 2
center(0x2,1500);
pulse(0x1,1500);
pulse(0x2,1500);
command(0,0x33);
```

```
next_wait();
pulse(0x1,0);                        // Seg 3
pulse(0x2,1500);
command(0,0x30);

next_wait();
center(0x1,-1500);                   // Seg 4
center(0x2,0);
pulse(0x1,-1500);
pulse(0x2,1500);
command(0,0x33);

next_wait();
pulse(0x1,-4500);                    // Seg 5
pulse(0x2,0);
command(0,0x30);

next_wait();
center(0x1,0);                       // Seg 6
center(0x2,-1500);
pulse(0x1,-1500);
pulse(0x2,-1500);
command(0,0x33);

next_wait();
pulse(0x1,0);                        // Seg 7
pulse(0x2,-1500);
command(0,0x30);

next_wait();
center(0x1,1500);                    // Seg 8
center(0x2,0);
pulse(0x1,1500);
pulse(0x2,-1500);
command(0,0x33);

wait(0x3);

                                     //------ Synchronous action (Section 2.61 – Example 1) ----
                                     //Starts fixed pulse drive of the Z axis in the + direction
                                     //if the Y axis passes through position 15000.

range(0x6,800000);                   // R = 800000 (Multiple = 10)
acc(0x6,400);                        // A = 400      (Accleration/deceleration = 500KPPS/SEC)
startv(0x6,50);                      // SV= 50       (Initial speed = 500PPS)
speed(0x6,3000);                     // V = 3000     (Drive speed = 30KPPS)
pulse(0x2,50000);                    // yP = 50000 (Y axis output pulse number)
pulse(0x4,10000);                    // zP = 10000 (Z axis output pulse number)
compp(0x2,15000);                    // yCP+ = 15000 (Y axis CMP+)
lp(0x6,0);                           // LP= 0        (Logical position counter = 0)
syncmode(0x2,0x2001,0x0000);         // Y axis synchronous action mode
                                     //      Activation factor:P≥C+, activation of other axis:Z
                                     //      Automatic activation: None
syncmode(0x4,0x0000,0x0001);         // Z axis synchronous action mode
                                     //      Own axis action: + direction fixed pulse drive

command(0x2,0x20);                   // Starts Y axis + fixed pulse drive
wait(0x6);                           // Waits for termination of Y and Z axes

}
```

13. Electrical Characteristics

13.1 DC Characteristics

■ Absolute Maximum Rated

Item	Symbol	Value	Unit
Power Voltage	V_{DD}	−0.3 ~ +7.0	V
Input voltage	V_{IN}	−0.3 ~ V_{DD}+0.3	V
Input Current	I_{IN}	±10	mA
Reservation Temperature	T_{STG}	−40 ~ +125	°C

■ Recommend Operation Environment

Item	Symbol	Value	Unit
Power Voltage	V_{DD}	4.75 ~ 5.25	V
Ambient Temperature	Ta	0 ~ +85	°C

If the user wishes to operate the IC below 0°C, please make contact with our R&D engineer.

■ DC Characteristics

(Ta = 0 ~ +85°C, V_{DD} = 5V ±5%)

Item	Mark	Condition	Min.	Typ.	Max.	Unit	Remark
High level input voltage	V_{IH}		2.2			V	
Low level input voltage	V_{IL}				0.8	V	
High level input current	I_{IH}	V_{IN} = V_{DD}	−10		10	μA	
Low level input current	I_{IL}	V_{IN} = 0V	−10		10	μA	D15~D0 Input signal
		V_{IN} = 0V	−200		−10	μA	Input signal besides D15~D0
High level output voltage	V_{OH}	I_{OH} = −1μA	V_{DD}−0.05			V	Note 1
		I_{OH} = −4mA	2.4			V	Output signal besides D15~D0
		I_{OH} = −8mA	2.4			V	D15~D0 Output signal
Low level output voltage	V_{OL}	I_{OL} = 1μA			0.05	V	
		I_{OL} = 4mA			0.4	V	Output signal besides D15~D0
		I_{OL} = 8mA			0.4	V	D15~D0 Output signal
Output leakage current	I_{OZ}	V_{OUT}=V_{DD} or 0V	−10		10	μA	D15 ~ D0,BUSYN,INTN
Smith hysteresis voltage	V_H			0.3		V	
Consuming current	I_{DD}	I_{IO}=0mA,CLK=16MHz		70	112	mA	

Note1 : BUSYN and INTN output signals have no items for high level output voltage due to the open drain output.

■ Pin Capacity

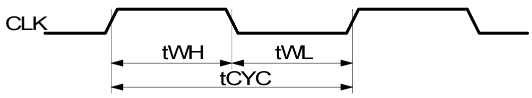
Item	Mark	Condition	Min.	Typ.	Max.	Unit	Remark
Input/ Output capacity	C_{IO}	Ta=25°C, f=1MHz			10	pF	D15 ~ D0
Input capacity	C_I				10	pF	Other input pins

13.2 AC Characteristics

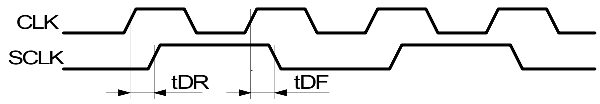
(Ta = 0 ~ +85°C, VDD = 5V ± 5%, Output load condition: 85 pF + 1 TTL)

13.2.1 Clock

CLK Input Signal



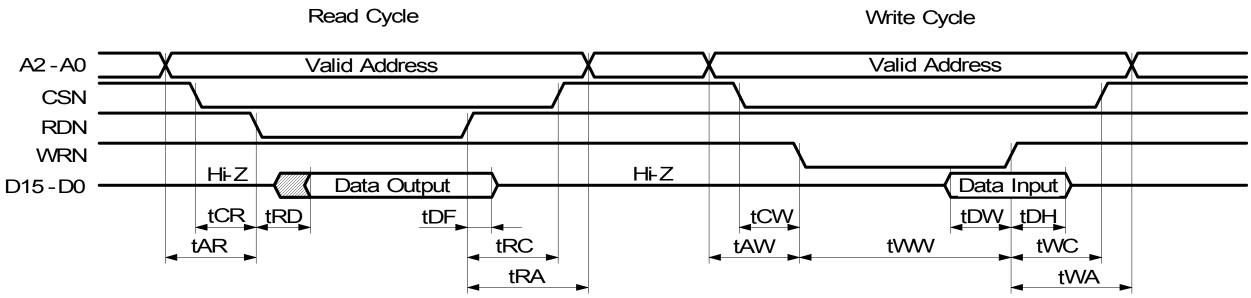
SCLK Output Signal



SCLK will not be output during reset.

Symbol	Item	Min.	Max.	Unit
tCYC	CLK Cycle	62.5		nS
tWH	CLK Hi Level Width	20		nS
tWL	CLK Low Level Width	20		nS
tDR	CLK ↑ → SCLK↑Delay Time 21 nS		19	nS
tDF	CLK ↑ → SCLK↓Delay Time 23 nS		25	nS

13.2.2 Read / Write Cycle

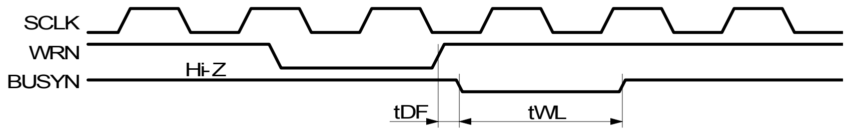


a. The figure shown above is used for 16-bit data bus accessing (H16L8 = Hi). For 8-bit data bus (H16L8 = Low), the address signals shown in the figure become A3~A0, and data signals become D7~D0.

b. At a read cycle, the data signal (D15~D0) becomes an output state as soon as both RDN and CSN become low and stays in the output state during tDF even if RDN is reset to High. Avoid the occurrence of bus conflict (collision).

Symbol	Item	Min.	Max.	Unit
tAR	Address Setup Time (to RDN ↓)	0		nS
tCR	CSN Setup Time (to RDN ↓)	0		nS
tRD	Output Data Delay Time (from RDN ↓)		26	nS
tDF	Output Data Hold Time (from RDN ↑)	0	26	nS
tRC	CSN Hold Time (from RDN ↑)	0		nS
tRA	Address Hold Time (from RDN ↑)	0		nS
tAW	Address Setup Time (to WRN ↓)	0		nS
tCW	CSN Setup Time (to WRN ↓)	0		nS
tWW	WRN Low Level Width	50		nS
tDW	Setup Time of Input Data (to WRN ↑)	21		nS
tDH	Hold Time of Input Data (from WRN ↑)	0		nS
tWC	CSN Hold Time (from WRN ↑)	0		nS
tWA	Address Hold Time (from WRN ↑)	5		nS

13.2.3 BUSYN Signal



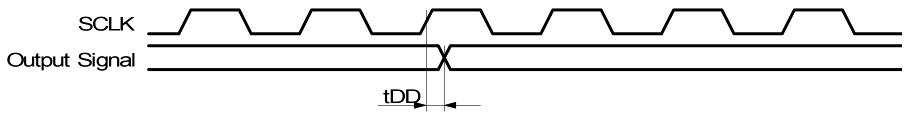
It is low when BUSYN is active. And BUSYN is low after 2 SCLK cycles when WRN ↑ active.

Symbol	Item	Min.	Max.	Unit
tDF	WRN ↑ → BUSYN ↓ Delay Time		32	nS
tWL	BUSYN Low Level Width		tCYC×4 +30	nS

tCYC is a cycle of CLK.

13.2.4 SCLK/Output Signal Timing

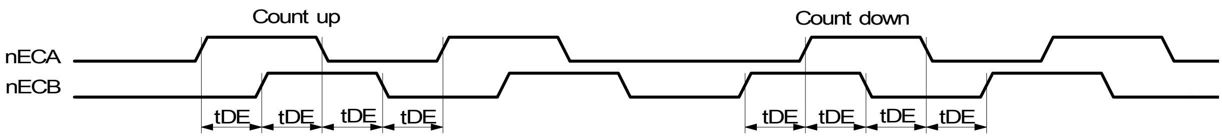
The following output single is synchronized with SCLK output signal. The level at ACLK ↑ will be changed.
Output signals : nPP/PLS, nPM/DIR, nDRIVE, nASND, nDSND, nCMPP, and nCMPM.



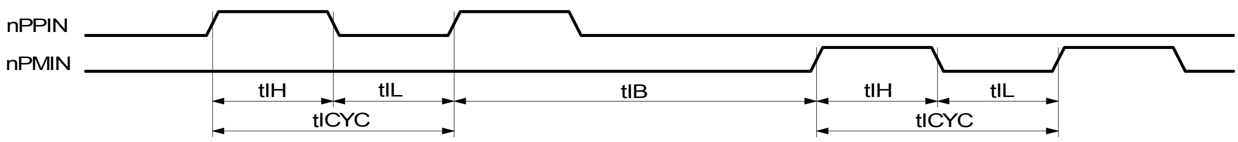
Symbol	Item	Min.	Max.	Unit
tDD	SCLK ↑ → Output Signal ↑ ↓Delay Time	0	20	nS

13.2.5 Input Pulses

■ Quadrature Pulses Input Mode (A/B phases)



■ Up / Down Pulses Input Mode



a. In A/B quadrature pulse input mode, when nECA and nECB input pulses are changed, the value of real position counter will be changed to the value of those input pulses changed after the period of longest SCLK4 is passed.

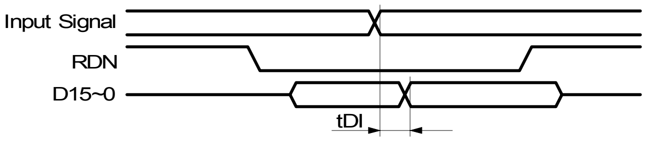
b. In UP/DOWN pulse input mode, the real position counter will become the value of those input pulses changed, after the period between the beginning of nPPIN, nPMIN ↑ and the time of SCLK 4 cycle is passed.

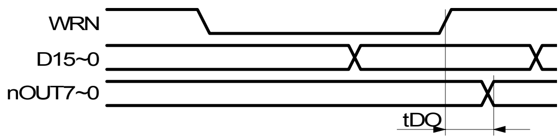
Symbol	Item	Min.	Max.	Unit
tDE	nECA and nECB Phase Difference Time	tCYC×2 +20		nS
tIH	nPPIN and nPMIN Hi Level Width	tCYC×2 +20		nS
tIL	nPPIN and nPMIN Low Level Width	tCYC×2 +20		nS
tICYC	nPPIN and nPMIN Cycle	tCYC×4 +20		nS
tIB	nPPIN ↑ ⟷ nPMIN ↑ between Time	tCYC×4 +20		nS

tCYC is a cycle of CLK.

13.2.6 General Purpose Input / Output Signals

The figure shown at the lower left hand side illustrates the delay time when input signals nIN3 ~ 0, nEXPP, nEXPM, nINPOS, and nALARM are read through RR4 and RR5 registers.
The figure shown at the lower right hand side illustrates the delay time when writing general output signal data into nWR3 and nWR4.

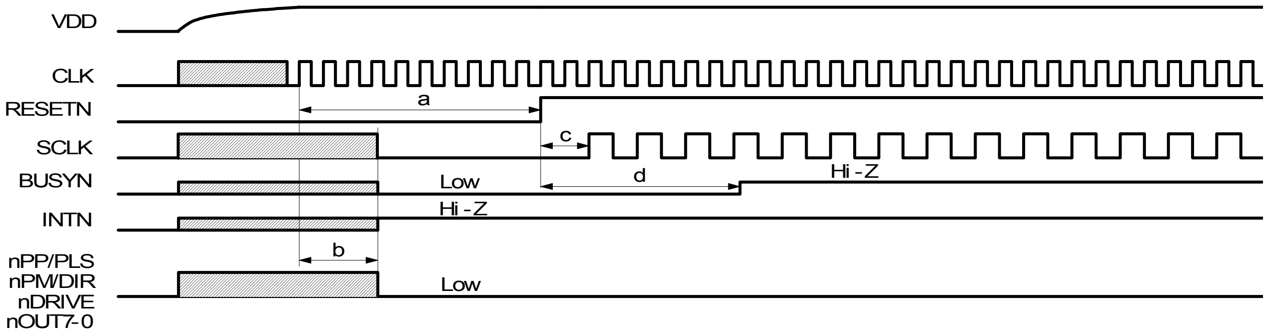




Symbol	Item	Min.	Max.	Unit
tDI	Input Signal → Data Delay Time		32	nS
tDO	WRN ↑ → nOUT7~0 Setup Time		32	nS

14. Timing of Input / Output Signals

14.1 Power-On Reset



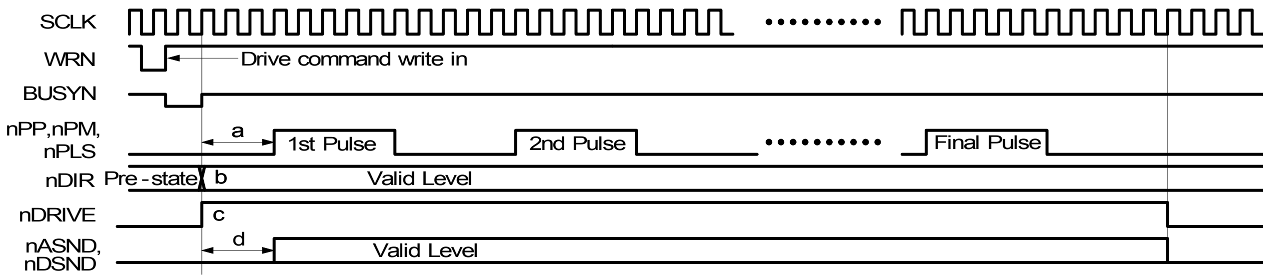
a. The reset signal input to pin RESETN will keep on the Low level for at least 4 CLK cycles.

b. When RESETN is on the Low level for 4 CLK cycles maximum, the output signals of MCX314As are decided.

c. SCLK will be output after 2 CLK cycles when RESTN return to the Hi level.

d. BUSYN keeps on the Low level for 8 CLK cycles when RESTN is on the Hi level.

14.2 Fixed Pulse or Continuous Pulse Driving



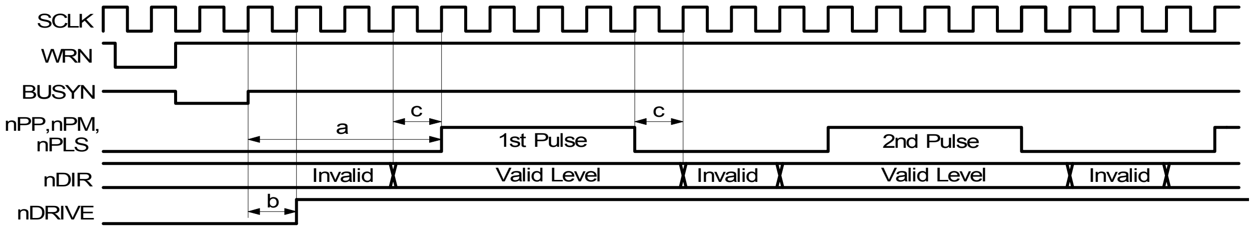
a. This first driving pulses (nPP, nPM, and nPLS) will be output after 3 SCLK cycles when BUSYN is ↑.

b. The nDIR (direction) signal is valid after 1 SCLK cycle when BUSYN is ↑.

c. The dDRIVE becomes Hi level when BUSYN is ↑.

d. The nASND and nDSND are on invalid level after 3 SCLK cycles when BUSYN is ↑.

14.3 Interpolation

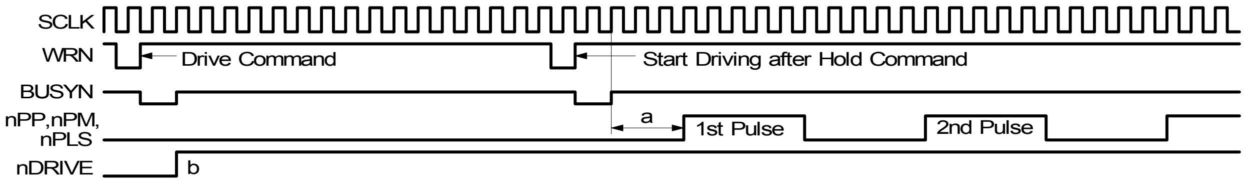


a. The first pulses (nPP, nPM, and nPLS) of interpolation driving will be output after 4 SCLK cycles when BUSYN is ↑.

b. nDRIVE will become Hi level after 1 SCLK cycle when BUSYN is ↑.

c. DIR signal keeps the active level in 1 SCLK cycle before and after the Hi level pulse outputting.

14.4 Start Driving after Hold Command

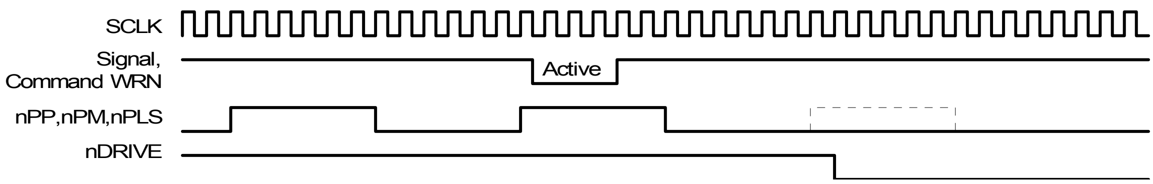


a. The pulses (nPP, nPM, and nPLS) of each axis will start outputting after 3 SCLK cycles when BUSYN is ↑.

b. nDRIVE will become Hi level when BUSYN is ↑ for each axis.

14.5 Sudden Stop

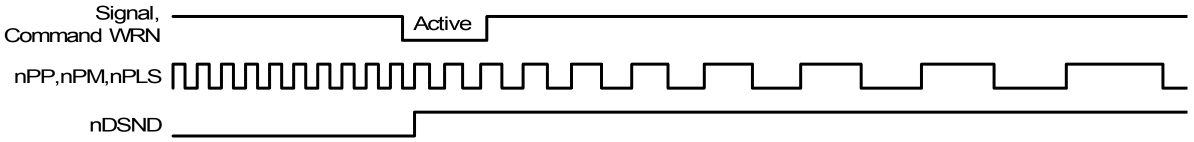
The following figure illustrates the timing of sudden stop. The sudden stop input signals are EMGN, nLMTP/M (When the sudden stop mode is engaged), and nALARM.
When sudden stop input signal becomes active, or the sudden stop command is written, it will stop the output of pulses immediately.
The width of external signals input for sudden stop must be more than 1 SCLK cycle. The stop function will not be active if the width is less 1 SCLK cycle.



An instant stop input signal requires a pulse width of CLK 2 cycles or more even if the input signal filter is disabled.
When the input signal filter is enabled, the input signal will be delayed according to the constant value at filtering.

14.6 Decelerating Stop

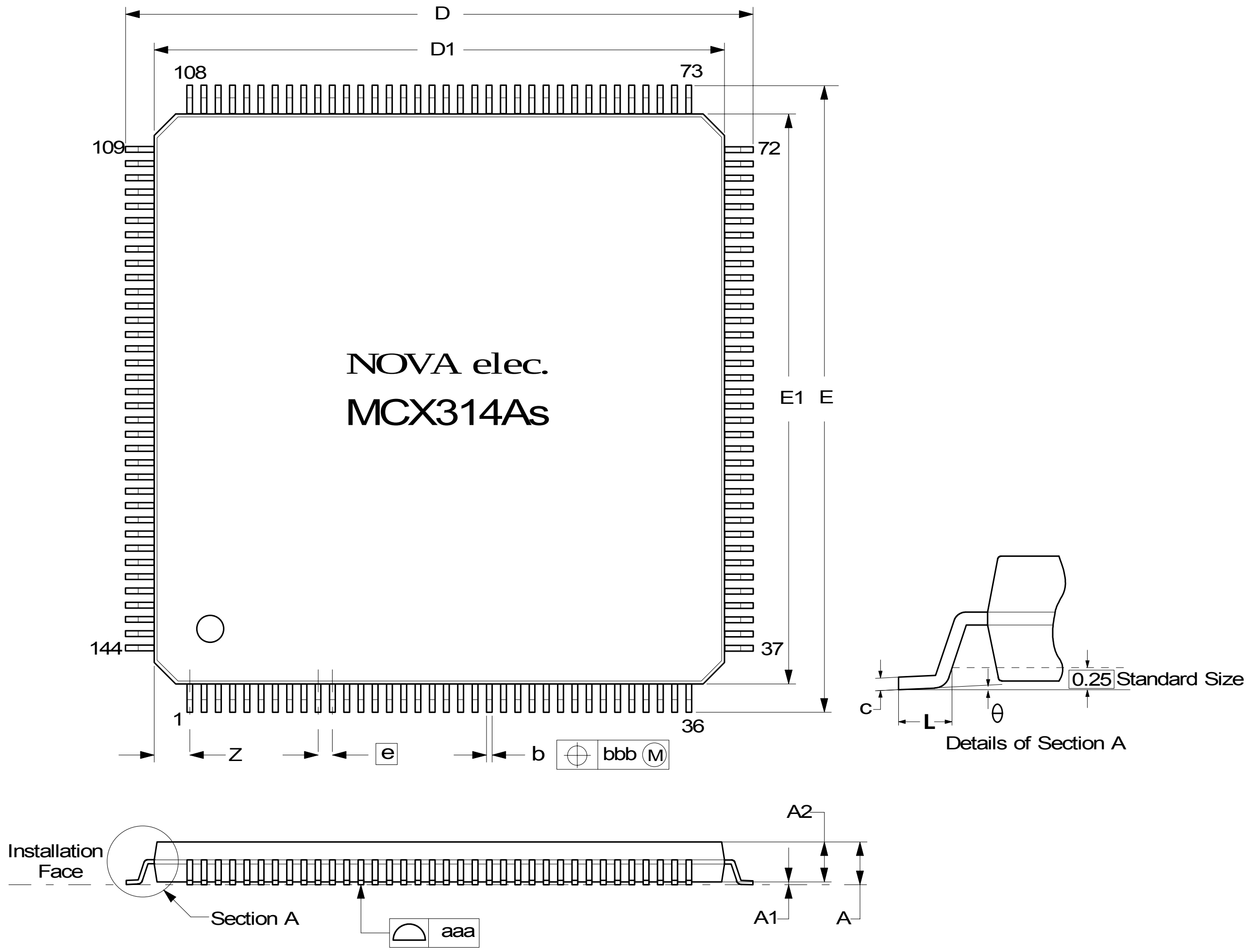
The following figure illustrates the timing of decelerating stop input signal and decelerating commands. The decelerating stop signal are nIN3 ~ 0 and nLMTP/M (When the decelerating mode is engaged).
When speed decelerating signals become active, or the decelerating stop command is written, the decelerating stop function will be performed.



If the input signal filter is disabled, the input signal is delayed according to the value of the time constant of the filter.

15. Package Dimensions

Unit: mm (inch)



Symbol	Size mm (inch)			Description
	Minimum	Standard	Maximum	
A	—	—	1.6 (0.063)	Height from the installation face to the top end of the package main unit
A1	0.05 (0.002)	0.1 (0.004)	0.15 (0.006)	Height from the installation face to the bottom end of the package main unit
A2	1.35 (0.053)	1.4 (0.055)	1.45 (0.057)	Height from the top to the bottom of the package main unit
b	0.17 (0.007)	0.22 (0.009)	0.27 (0.011)	Pin width
c	0.09 (0.004)	0.145 (0.006)	0.2 (0.008)	Pin thickness
D	21.8 (0.858)	22 (0.866)	22.2 (0.874)	Maximum length in the package length direction including pins
D1	19.8 (0.780)	20 (0.787)	20.2 (0.795)	Length of the package main unit excluding pins
E	21.8 (0.858)	22 (0.866)	22.2 (0.874)	Maximum length in the package width direction including pin
E1	19.8 (0.780)	20 (0.787)	20.2 (0.795)	Width of the package main unit excluding pins
e	0.5 (0.020)			Pin pitch standard size
L	0.45 (0.018)	0.6 (0.024)	0.75 (0.030)	Length of the flat section of the pins that contacts the installation face
Z	1.25 TYP (0.049TYP)			Length from the center of the outer-most pin to the outer-most pin section of the package main unit
θ	0°	—	10°	Angle of the pin flat section for the installation face
aaa	0.08 (0.003)			Uniformity of the bottom of the pin (permissible value in the vertical direction)
bbb	0.08 (0.003)			Permissible value of the pin center position error (horizontal direction)

16. Storage and Recommended Installation Conditions

16.1 Storage of this IC

Note the following items in regard to the storage of this IC.
(1) Do not throw or drop the IC. Otherwise, the packing material could be torn, damaging the confidentiality.
(2) Store the IC under the temperature 30°C or lower and humidity 90%RH or lower and use the IC within 12 months.
(3) If the IC usage date has expired, remove any dampness by backing it under the temperature 125°C for 20 hours. If dampproofing is damaged before expiration, apply damp removal processing also.
(4) Apply device corruption prevention using static electricity before applying dampness removal processing.
(5) After opening the dampproof package, store the IC under 30°C/60%RH or lower and install it within seven days. Make sure that backing processing is applied before installation of the IC that is left in the storage for a time that exceeds the expiration period as indicated above.

16.2 Standard Installation Conditions by Soldering Iron

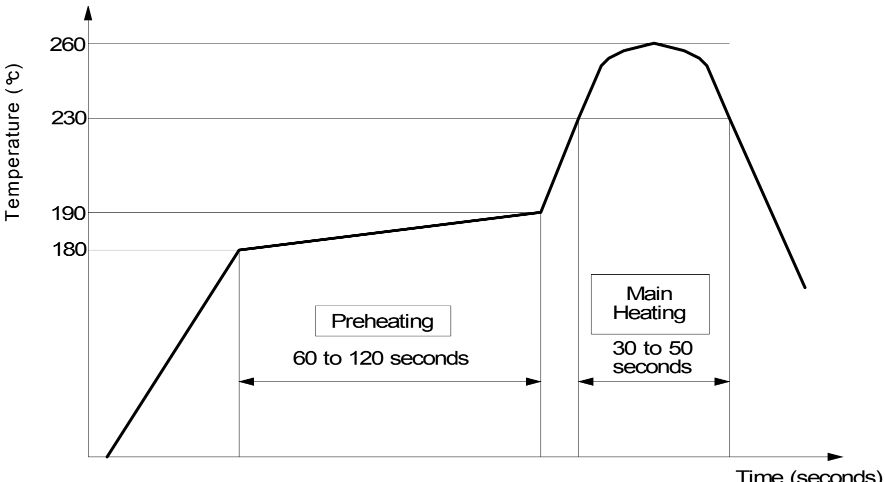
The standard installation conditions for the IC by soldering iron are as follows.
(1) Installation method: Soldering iron (heating the lead section only)
(2) Installation conditions: (a)350°C for 3 seconds or less
: (b)260°C for 10 seconds or less

16.3 Standard Installation Conditions by Solder Reflow

The standard installation conditions for the IC by solder reflow are as follows.

(1) Installation method :	(a) Hot-air solder reflow (including the far/middle infrared solder reflow concurrent use)
:	(b) Far/middle infrared solder reflow
(2) Preheating conditions :	180 ~ 190°C for 60 ~ 120 seconds
(3) Solder reflow conditions :	(a) Maximum 260°C
	(b) 230°C or higher for 30 ~ 50 seconds or less
(4) Solder reflow count :	Up to twice within the permissible storage period

The temperatures in the installation conditions are based on the package surface temperature. The temperature profile indicates the upper limit of the heat-proof temperature. Install the IC within the following profile.



Standard Solder Reflow Heat - Proof Profile

17. Specifications

- ■ Control Axis — 4 axes
- ■ Data Bus — 16/8 bits selectable

Interpolation Functions

■ 2-axes / 3-axes Linear Interpolation

- ● Interpolation Range Each axis — −2,147,483,646 ~ +2,147,483,646
- ● Interpolation Speed — 1 PPS ~ 4 MPPS
- ● Interpolation Accuracy — ± 0.5 LSB (Within the range of whole interpolation)

■ Circular Interpolation

- ● Interpolation Range Each axis — −2,147,483,646 ~ +2,147,483,646
- ● Interpolation Speed — 1 PPS ~ 4 MPPS
- ● Interpolation Accuracy — ± 1 LSB (Within the range of whole interpolation)

■ 2 axes / 3 axes Bit Pattern Interpolation

- ● Interpolation Speed — 1PPS ~ 4 MPPS (Dependent on CPU data writing time)

■ Related Functions of Interpolation ●Can select any axis ●Constant vector speed ●Continuous interpolation ●Single step interpolation (Command/external signals)

Common Specifications of Each Axis

■ Drive Pulses Output (When CLK = 16 MHz)

● Pulse Output Speed Range	1PPS ~ 4MPPS	
● Pulse Output Accuracy within	± 0.1% (according to the setting speed)	
● S-curve Jerk	954 ~ 62.5 x 10^6 PPS/S^2	(Multiple = 1)
	477 x 10^3 ~ 31.25 x 10^9PPS/S^2	(Multiple =500)
● Accelerating / Decelerating Speed	125 ~ 1 x 10^6 PPS/S	(Multiple = 1)
	62.5×10^3 ~ 500 x 10^6 PPS/S	(Multiple = 500)
● Initial Speed	1 ~ 8,000PPS	(Multiple = 1)
	500PPS ~ 4×10^6 PPS	(Multiple = 500)
● Drive Speed	1 ~ 8,000PPS	(Multiple = 1)
	500PPS ~ 4×10^6 PPS	(Multiple = 500)
● Output-pulse Number	0 ~ 4,294,967,295 / unlimited	
● Speed Curve	Constant speed, symmetrical/non-symmetrical linear acceleration, symmetrical/ non-symmetrical parabola S-curve acceleration/deceleration drive	

- ● Index Pulse Drive Deceleration Mode auto (non-symmetrical linear acceleration is also allowed) / manual
- ● Output-pulse numbers and drive speeds changeable during the driving
- ● Triangle form prevention of linear acceleration fixed pulse drive and S-curve acceleration/deceleration fixed pulse drive
- ● Independent 2-pulse system or 1-pulse 1-direction system selectable
- ● Logical levels of drive pulse selectable, output pin switchable

■ Encoder Input

- ● A/B quadrature pulse style or Up/Down pulse style selectable
- ● Pulse of 1, 2 and 4 divisions selectable (A/B quadrature pulse style)

■ Position Counter

- ● Logic Position Counter (for output pulse t) range −2,147,483,648 ~ +2,147,483,647
- ● Real Position Counter (for feedback pulse) range −2,147,483,648 ~ +2,147,483,647

Variable ring counter function, real position counter increase/decrease inversion function, and real position counter clear function by the IN2 signal
Data read and write possible

■ Comparison Register

- ● COMP + Register Position comparison range −2,147,483,648 ~ +2,147,483,647
- ● COMP − Register Position comparison range −2,147,483,648 ~ +2,147,483,647
- ● Status and signal outputs for the comparisons of position counters
- ● Software limit functioned

■ Automatic home search

- ● Automatic execution of Step 1 (high-speed near home search) → Step 2 (low-speed home search) → Step 3 (low-speed encoder Z-phase search) → Step 4 (high-speed offset drive). Enable/Disable of each step and search direction selectable
- ● Deviation counter clear output : Clear pulse width within the range of 10μ ~ 20msec and logical level selectable

■ **Synchronous Action**

● Activation factor: Position counter ≥ COMP+ variation, position counter < COMP+ variation, position counter < COMP − variation, position counter ≥ COMP − variation, start of driving, termination of driving, IN3 signal↑, IN3 signal↓, LP read command, activation command.

● Action: Start of +/− fixed pulse drive, start of +/− continuous pulse drive, drive decelerating stop, drive instant stop, saving position counter values, setting position counter values, setting an output pulse number, setting a drive speed, external signal output (DCC), and interrupt

Any action of other axes can be activated from the factor of the own axis.

■ **Interrupt** (Interpolations Excluded)

● The factors of occurring interrupt:

..the drive-pulse outputting
..the start / finish of a constant-speed drive during the acceleration / deceleration driving
..the end of the driving
..the volume of position counter ≥ the volume of COMP−
..the volume of position counter < the volume of COMP−
..the volume of position counter ≥ the volume of COMP+
..the volume of position counter < the volume of COMP+
..terminating of automatic home search, synchronous action

Enable / disable for these factors selectable

■ **External Signal for Driving**

● EXPP and EXPM signals for fixed pulse / continuous drive
● Driving in manual pulsar mode (encoder input)

■ **External Deceleration / Sudden Stop Signal**

● IN0 ~ 3 4 points for each axis

Enable / disable and logical levels selectable

■ **Servo Motor Input Signal**

● ALARM (Alarm), INPOS (In Position Check), DCC (Pin shared between deviation counter clear output and DRIVE)

Enable / disable and logical levels selectable

■ **General Output Signal**

● OUT0 ~ 7 8 points for each axis (wherein 4 points use with drive status output signal pin)

■ **Driving Status Signal Output**

● ASND (speed accelerating), DSND (speed decelerating), CMPP (position ≥ COMP+), CMPM (position < COMP−)

Drive status and status registers readable

■ **Limit Signals Input**

● 2 points, for each + and − side

Logical levels and decelerating / sudden stop selectable

■ **Emergency Stop Signal Input**

● EMG, 1 point for 4 axes

■ **Contents of integral type filters**

● Equipped with integral filters in the input column of each input signal. One time constant can be selected from eight types.

■ **Electrical Characters**

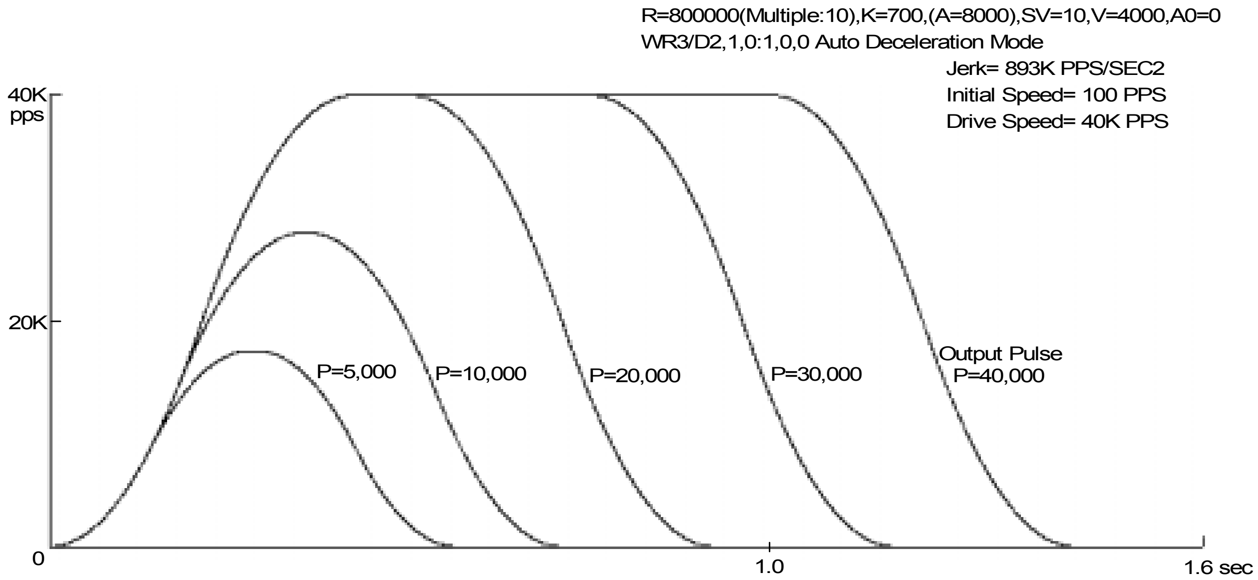
● Temperature Range for Driving: 0 ~ + 85°C (32°F ~185°F)
● Power Voltage for Driving: +5V ± 5 %
● Consumption current: 70mA typ (112mA max)
● Input / Output Signal Level: CMOS, TTL connectable
● Input Clock Pulse: 16,000 MHz (Standard)

■ **Package** 144-pin plastic LQFP, pitch = 0.5mm lead free item
Dimension : 20 × 20 × 1.4 mm (External size including pins : 22 × 22 × 1.6 mm)

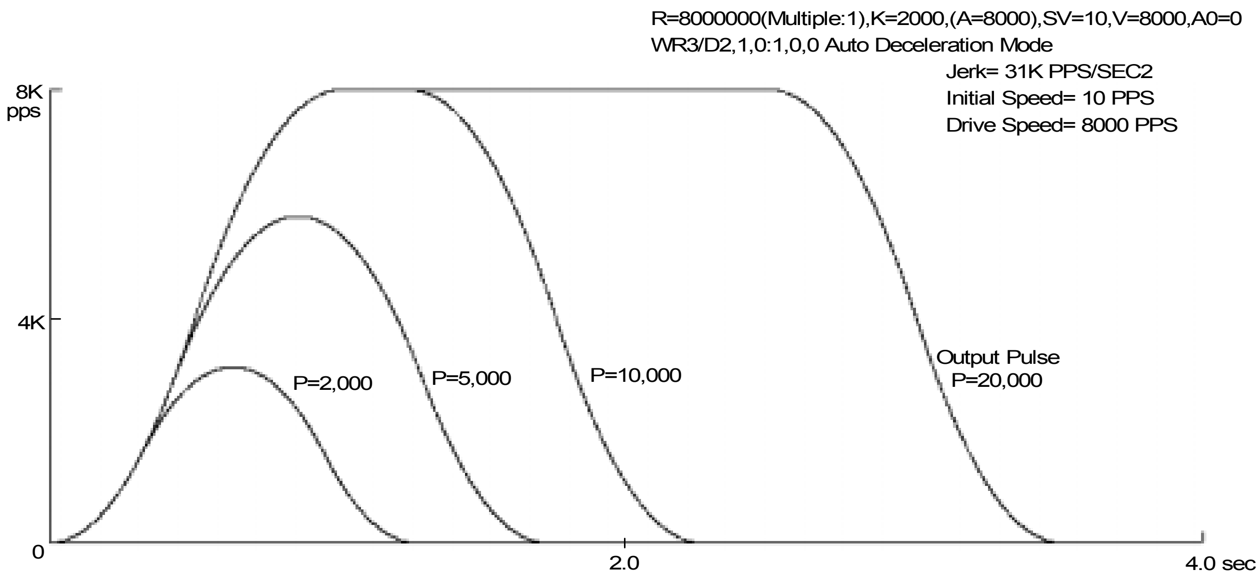
Appendix A Speed Profile of Acceleration/Deceleration Drive

The following curves are based on the test records from MCX314As output drive pulses and speed curve traces.

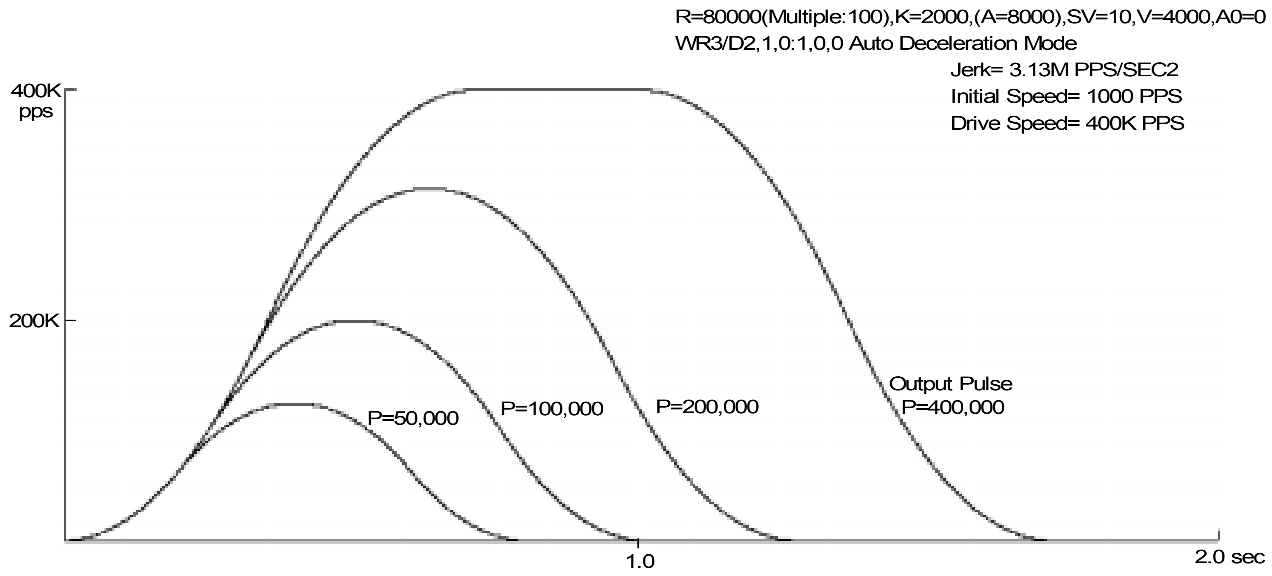
■ 40KPPS Symmetrical S-curve Acceleration/Deceleration



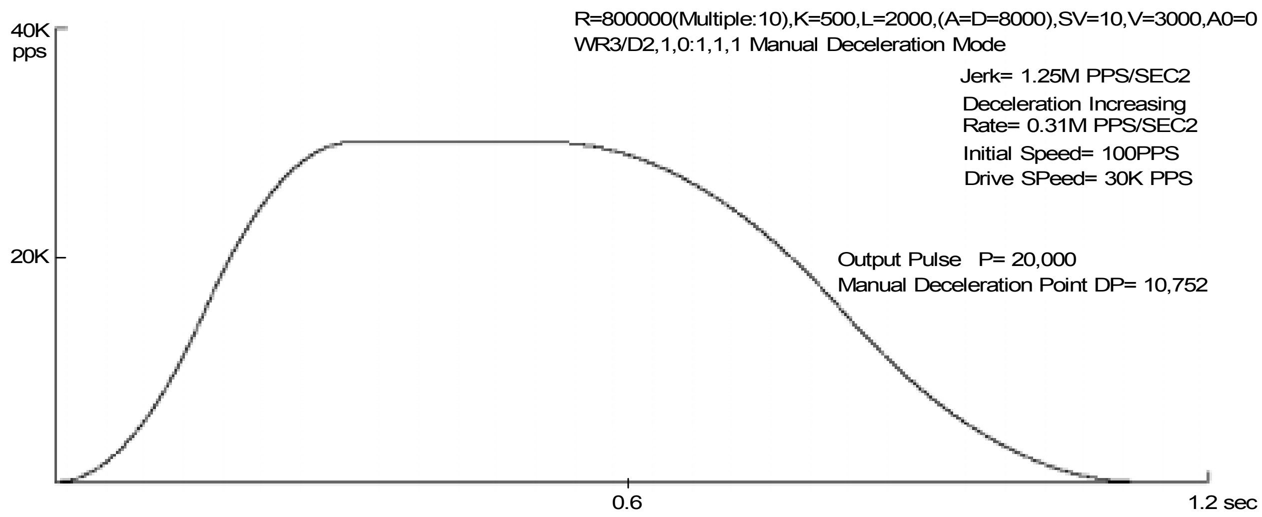
■ 8000PPS Symmetrical S-curve Acceleration/Deceleration



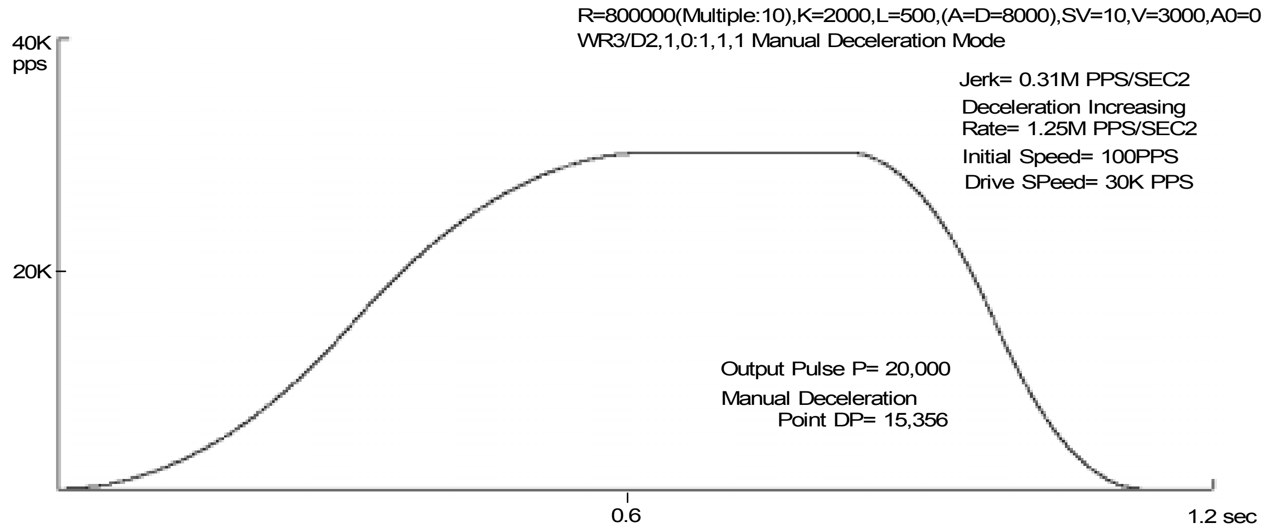
■ 400KPPS Symmetrical S-curve Acceleration/Deceleration



■ 40KPPS Non-Symmetrical S-curve Acceleration/Deceleration (1)



■ 40KPPS Non-Symmetrical S-curve Acceleration/Deceleration(2)

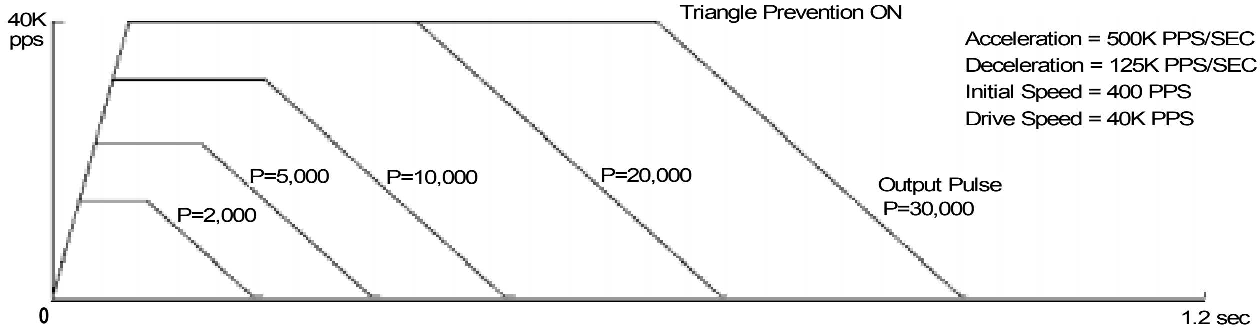


■ 40KPPS Non-Symmetrical Trapezoid Acceleration/Deceleration

a. Acceleration/Deceleration Ratio 4:1

R=800000(Maltiple:10),A=400,D=100,SV=40,V=4000,A0=0
WR3/D2,1,0:0,1,0 Auto Deceleration Mode 60H/WR6/D3:1
Triangle Prevention ON

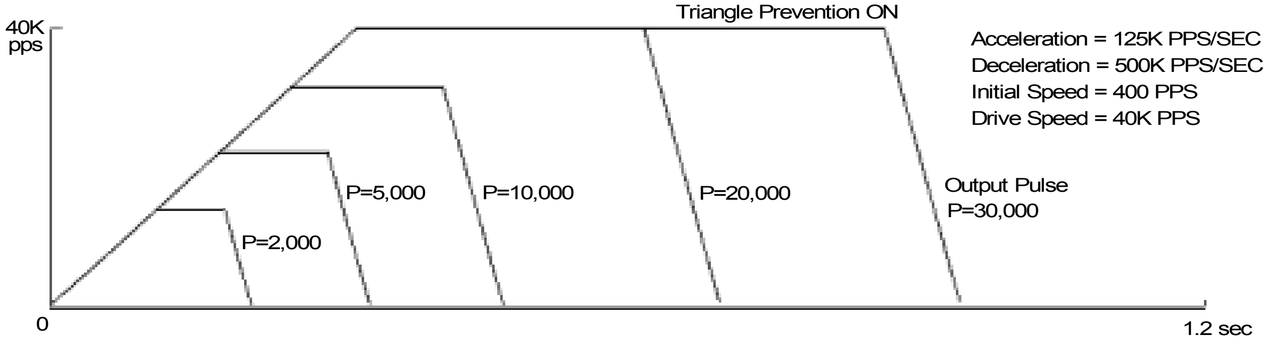
Acceleration = 500K PPS/SEC
Deceleration = 125K PPS/SEC
Initial Speed = 400 PPS
Drive Speed = 40K PPS



b. Acceleration/Deceleration Ratio 1:4

R=800000(Maltiple:10),A=100,D=400,SV=40,V=4000,A0=0
WR3/D2,1,0:0,1,0 Auto Deceleration Mode 60H/WR6/D3:1
Triangle Prevention ON

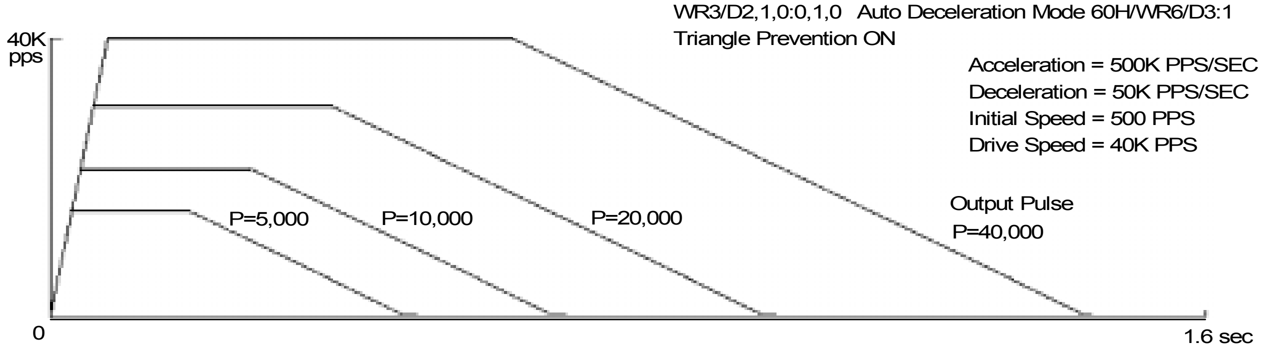
Acceleration = 125K PPS/SEC
Deceleration = 500K PPS/SEC
Initial Speed = 400 PPS
Drive Speed = 40K PPS



c. Acceleration/Deceleration Ratio 10:1

R=800000(Maltiple:10),A=400,D=40,SV=50,V=4000,A0=0
WR3/D2,1,0:0,1,0 Auto Deceleration Mode 60H/WR6/D3:1
Triangle Prevention ON

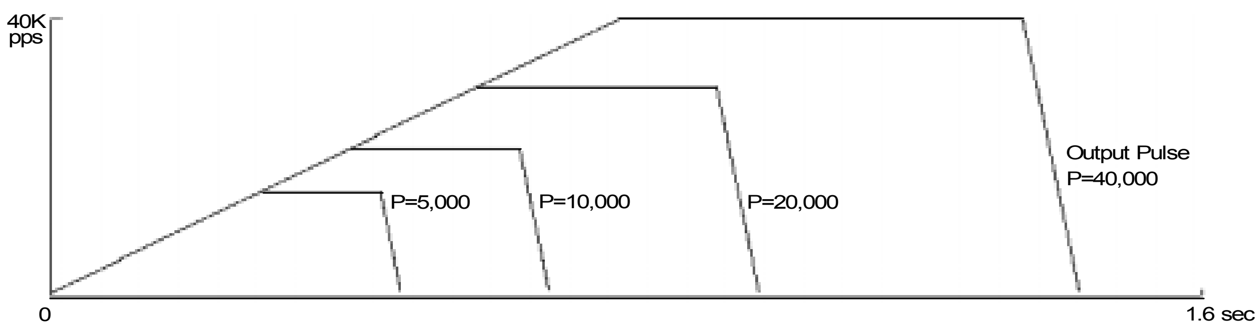
Acceleration = 500K PPS/SEC
Deceleration = 50K PPS/SEC
Initial Speed = 500 PPS
Drive Speed = 40K PPS



d. Acceleration/Deceleration Ratio 1:10

R=800000(Maltiple:10),A=40,D=400,SV=50,V=4000,A0=0
WR3/D2,1,0:0,1,0 Auto Deceleration Mode 60H/WR6/D3:1
Triangle Prevention ON

Acceleration = 50K PPS/SEC
Deceleration = 500K PPS/SEC
Initial Speed = 500 PPS
Drive Speed = 40K PPS



Appendix B Common Items/Differences with MCX314

The following table lists the common items and differences between MCX314 and MCX314As (lead free product)

Item	MCX314As
With/without lead	Without lead (*1)
Package shape	Different from MCX314
Pin assignment	Same as MCX314
Electrical characteristics of each signal	Same as MCX314
All the functions available to MCX314	Same as MCX314
Improvements on deficiencies of MCX314	(1) Improvement on deficiency by writing at continuous interpolation pulses (2) Improvement on deficiency by specification of a circular interpolation finish point (3) Improvement of UP/DOWN pulse input counting error (4) Improvement of extreme creep at S-curve acceleration/deceleration
New function	(1) Automatic home search (2) Automatic acceleration/deceleration of non-symmetrical trapezoidal driving (3) Input signal integral filter (4) Synchronous action (5) 32-bit pulse (6) 32-bit circular/linear interpolation pulse (7) Non-symmetrical complete S-curve acceleration/deceleration (8) Others (clearing position counter by Z-phase input, real position counter increase/decrease inversion, manual pulsar, deviation counter clearing pulse output, logical/real position counter variable ring)
Replacement with MCX314 (hardware)	Not possible (substrate pattern change is necessary)
Replacement with MCX314 (software)	Possible (Notes 1 and 2)

*1: See Chapter 16 for the installation method.
*2: MCX314As package exterior size: 20 × 20mm, pin pitch: 0.5mm, 144 pins, Sn-Bi (Tin bismuth) coated pin

Note 1: Acceleration/deceleration curve

In MCX314As, a part of the circuit for generating acceleration/deceleration is modified to reduce creep in S-curve acceleration/deceleration. Therefore, the acceleration/deceleration curve is not exactly the same as for MCX314. For instance, the fixed drive termination time is slightly different. When applying control with delicate timing using linear/S-curve acceleration/deceleration of MCX314, replace the IC after sufficient evaluation of acceleration/deceleration curve of MCX314As.

Note 2: Writing interpolation data

In MCX314, a 4 byte-length is specified for interpolation finish point/center numeric data write processing. However, the function operates normally even if a 3-byte length is specified. For instance, when –2 is set, the function operates normally even if FFFh and 00FFh are written in WR6 and WR7 respectively. However, in MCX314As, a 4-byte length must be adhered to due to introduction of a 32-bit format. When –2 is set as indicated in the example, FFFEh and FFFFh must be written in WR6 and WR7 respectively.