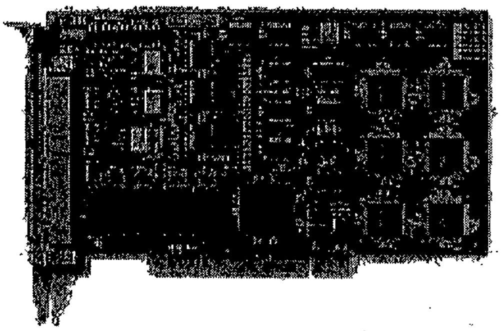
Melec



Stepping & Servo Motor Controller

C-871

Instructions Manual

(For designers' use)

USER'S MANUAL

Please ensure to read and understand this
Instructions Manual before using the product.
Please keep this Instructions Manual at hand
so that it is always available for reference.

PR0700-2

Introduction

This User's Manual describes the method of dealing with "STEPPING AND SERVO MOTOR CONTROLLER C-871", which set weight in specification in order to have a product used safely correctly for the designer of the control system using the stepping motor or servo motor.

Before using this product, carefully read this User's Manual to have a sufficient understanding of the functions.
Keep this User's Manual on hand so that you can refer to it whenever you want.

Description of Safety

Correct operation procedures are essential.
If you use in a wrong way, an unexpected accident may occur to cause personal injuries or damage of your properties.

Many of the possible accidents can be avoided if you have a preliminary knowledge about dangerous situations. For this purpose, this User's Manual describes the precautions if any dangerous situation can be anticipated.

Such descriptions are given in terms of the following symbols and signal words.

▲ WARNING	Death or serious injury may be caused by incorrect handling.
⚠ CAUTION	Slight injury or damage of your properties may be caused by incorrect handling.

Before use

- This product is not designed for use in the equipment related to nuclear power, aerospace equipment, vehicles, marine vessels, medical equipment directly in touch with human body, equipment anticipated to give a serious impact to properties, and other equipment required to provide high reliability.

- This product is provided with a LIMIT (overtravel) signal to prevent mechanical damage.
  This signal is an ACTIVE OFF input. Accordingly, even if the system does not use the LIMIT signal, connect an external power supply for coupler so that the LIMIT signal may be put into the NORMAL ON state to output pulses.
  For details, refer to 14-2.

- This product is surely used for within this description of the specification method of this manual, and the limits of specification.

- Before connecting this product to the expansion slot, certain setting operations must be done for the circuit board. Refer to the following sections for the setting:

- For details of applied functions referred to in this manual, see the separately issued the User's Manual [Applied Functions Part].

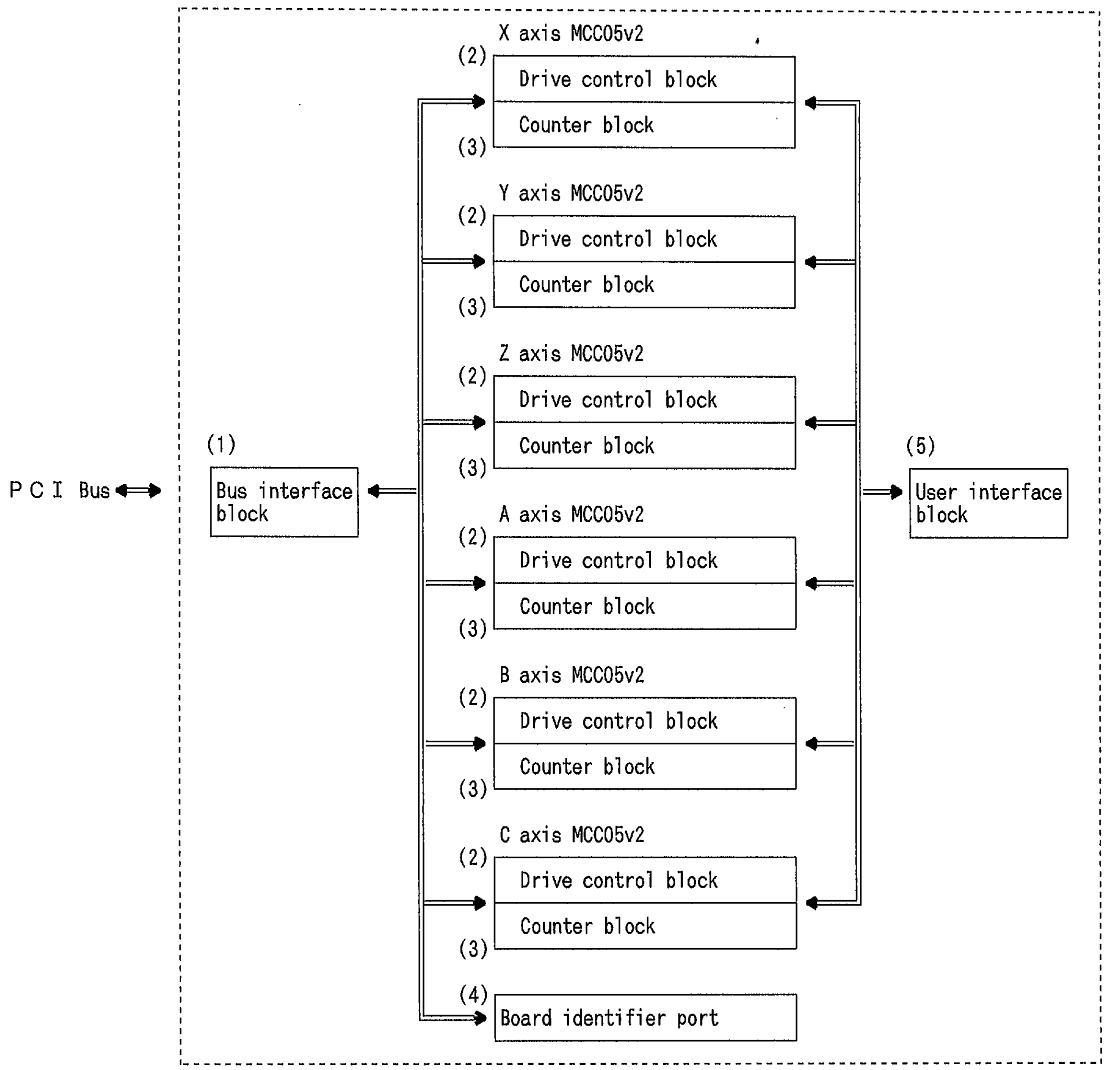
Contents

1. OVERVIEW

The C-871, equipped with 6 independently functioning axes, is a high performance controller that can be directly connected to the PCI bus expansion slot based on PCI Local Bus Specification Rev. 2.1.
It is designed to offer control for both servo and stepping motors. Its board is a compact half-size (107 x 175).
Equipped with our high grade chip controller MCC05v2, the pulse generator permits motor control using easier-to-operate command type.
The multi-function PULSE COUNTER and DIFFERENTIAL COUNTER on the MCC05v2 enable chip controller to count feedback pulses from a servo driver as well as detecting step-out of a stepping motor with the encoder.
Since the C-871 are provided with the 6 independent axes, the first axis, the second axis, the third axis, the fourth axis, the fifth axis and the sixth axis are called X axis, Y axis, Z axis A axis, B axis and C axis respectively.
As a rule, the following description will be given about only the X axis.

2. BASIC CONFIGURATION

2-1. Function Block Diagram



2-2. Description of Blocks

(1) Bus interface block

Interface block with the PCI bus.

This block consists of the interface IC dedicated to PCI bus and a serial EEPROM, etc.

(2) Drive control blocks

These blocks output serial pulses for motor control. The blocks mount pulse generators MCC05v2 for individual 6 axes, so the 6 axes can be independently driven.

For distinction among the 6 axes, these drive control blocks are called X axis MCC05v2, Y axis MCC05v2, Z axis MCC05v2, A axis MCC05v2, B axis MCC05v2 and C axis MCC05v2 respectively.

(3) Counter blocks

This counter block consists of the three types 24 bit counter, that is to say, the ADDRESS COUNTER, general purpose PULSE COUNTER and DIFFERENTIAL COUNTER.

This block allows to read a counting any time as needed, preset a count or interrupt at any counting (or deviation level).

Note: The PULSE COUNTER is used for counting external 90° phase difference clocks and the DIFFERENTIAL COUNTER counts deviation between the output pulse and the 90° phase differentiated external clocks. However, the C-871 is not provided with an external clock input interface, therefore, it can only count pulses from the MCC05v2. Names of each counter is the same as the MCC05v2. If any application using the external clock is needed, use the C-870v1 configured with 4-axes.

(4) Board identifier port

When more than two C-871s are used, the boards must be numbered. This is used only for the port.

(5) User interface block

Interface block for signals of the servo/stepping motor drivers and sensors, etc.

3. SPECIFICATIONS

3-1. PCI Local Bus Specification

(1) PCI Local Bus Specification Rev. 2.1

(2) Bus interface
32 BIT BUS(inside 8 BIT), 5V signaling environment, 33MHz clock

(3) PCI configuration register

31	16	15	0	Offset
Device ID(**1030h**)		Vendor ID(**152Eh**)		00h
Status		Command		04h
Base Class(0Eh)	Sub Class(80h)	Prog.I/F(00h)	Revision ID(00h)	08h
BIST	Header Type(00h)	Latency Timer	Cache Line Size	0Ch
Base Address Register 0 : Reserved(Cannot use)				10h
Base Address Register 1 : **Base Address of C-871**				14h
Base Address Register 2 : Reserved				18h
				1Ch
Reserved				20h
				24h
Cardbus CIS Pointer				28h
Subsystem ID(00h)		Subsystem Vendor ID(00h)		2Ch
Expansion ROM Base Address : Reserved				30h
Reserved				34h
Reserved				38h
Max_Lat	Min_Gnt	Interrupt pin(01h)	Interrupt Line	3Ch

(4) Interrupt
- INTA#
- Resets interrupt by STATUS PORT READ of the interrupt request axis.

(5) Dimensions
5V SHORT CARD (107m x 175mm x 17mm)

3-2. Basic Functions

(1) Drive function
JOG ······················ 1-pulse drive
SCAN ··················· Continuous drive until a stop command is input.
INDEX ·················· Positioning drive up to the specified number of pulses or the specified address.
ORIGIN ················ A series of drive until machine origin detection (9-type detection type).
S-RATE SCAN ········· The same as SCAN. Drive with S-shaped acceleration and deceleration.
S-RATE INDEX ······ The same as INDEX. Drive with S-shaped acceleration and deceleration.

*** "SCAN DRIVE" and "INDEX DRIVE", as used in this manual, do not include "S-RATE SCAN DRIVE" and "S-RATE INDEX DRIVE", respectively.**

(2) Number of drive pulses
JOG ································· 1 pulse/drive
SCAN,S-RATE SCAN ············ Up to infinite pulses/drive
INDEX,S-RATE INDEX ········· 0 to 8,388,607 pulses/drive (when relative is specified)
0 to 16,777,214 pulses/drive (when absolute is specified)

(3) Speed/rate range
Speed range ········· 1Hz to 3.3MHz
Rate range ········· 1030ms/1000Hz to 0.004ms/1000Hz

(4) Speed data Hz unit setting function
The output pulse speed can be set in 1 to 3,333,333Hz.

(5) Acceleration/deceleration time constant(RATE) individual setting function
An acceleration time constant and a deceleration time constant can be separately set.
(Enabled in applied function for S-RATE SCAN and S-RATE INDEX DRIVE.)

(6) DRIVE SPEED change function

You can change a SCAN DRIVE or INDEX DRIVE speed to any desired speed while the drive is taking place. (This feature, however, is not available when different time constants are specified for the acceleration and deceleration at INDEX DRIVE.)

(7) ADDRESS COUNTER function

The ADDRESS COUNTER counts the absolute address of the pulse output from the MCC05v2 and allows to read the count data any time as needed.

(8) PULSE COUNTER function

The PULSE COUNTER can always read out output pulses. Five COMPARE REGISTERs are connected to the PULSE COUNTER, so this permits detecting an optional count value.

(9) Function for fast/slow stop by LIMIT STOP

Two stop types using the LIMIT signal, namely, fast and slow, are available and can be specified by user program.

(10)Function for the servo driver

A function for the END signal and deviation COUNTER RESET signal of the servo driver is provided.

(11)Interrupt generating function

An interrupt can be generated for the PCI bus master.

RDYINT interrupt ········· Interrupt request signal (RDYINT) that is generated upon termination of a command.

CNTINT interrupt ········· Interrupt request signal (CNTINT) that is generated at an optional count value of the PULSE COUNTER.

DFLINT interrupt ········· Interrupt request signal (DFLINT) that is generated at an optional count value of the DIFFERENTIAL COUNTER.

(12)ORIGIN DRIVE function using the limit sensor

The ORIGIN DRIVE using the limit sensor is available, too.

(13)Current speed read function

You can read current speed during the drive.

(14)Setup data read function

It allows you to read the user program specified settings for HSPD, LSPD, RATE, SPEC INITIALIZE and such.

3-3. Ratings

(1) Power supply voltage : ＋5V±5%　　1.1A　max
　　　　　　　　　　　　＋24V±2V　　270mA　max(at EXTV is +24V) (for photo coupler interface)

(2) Ambient temperature : 0℃ to 45℃

(3) Ambient humidity : 80%RH or less (without dew condensation)

(4) Mass : Approx 0.15 kg

3-4. Accessories

Seven short circuit sockets for jumper post (of which four sockets are built in the board).

3-5. Options

Optional functions are prepared for the C-871. For details, Please contact us.

3-6. Applied Functions

For the C-871, Applied functions are available in addition to the basic functions shown in 3-1. to order to meet the requirements of various users' specifications.
For the details on these applied functions, refer to the User's Manual [Applied Functions Part].

(1) Applied Drive Functions

SPECIAL SCAN ……The drive is similar to SCAN DRIVE but the speed can be adjusted during the drive.(Note)
SPECIAL INDEX……The drive is similar to INDEX DRIVE but the speed can be adjusted during the drive.(Note)
SERIAL INDEX ……The drive executes previously set drive patterns continuously without stop.
SPECIAL SERIAL INDEX…SERIAL INDEX DRIVE where a rate can be set for each section.
SENSOR INDEX …Combining the INDEX DRIVE and SENSOR input detection, this drive implements the positioning.
SENSOR SCAN ……DRIVE where SCAN DRIVE and SENSOR input detections are combined to provide positioning.

*** The description of "SCAN DRIVE" and "INDEX DRIVE" in this Manual does not include the application DRIVE.**

(2) INDEX Change Function During Drive

It enables to change the specified pulse number or the address during the INDEX DRIVE.

(3) RATE Change Function During Drive

It enables to change an acceleration/deceleration time constant during the SCAN DRIVE.

(4) Acceleration/Deceleration Time Constant Parameter Setting Function

Acceleration/deceleration time constant can be freely set by parameter.

(5) Speed Data Setting Method Changing Function

Output pulses are generally set in Hz in the Hz setting mode, but it is possible to change this mode to the reference clock magnification setting mode, in which output pulses are set to any integer times of the reference clock.

(6) First Output Pulse Width Selecting Function

The width of the first active pulse after drive start can be selected from any of half period, 100μs fixed period and 20μs fixed period.

(7) Pulse Output Pattern Changing Function

Pulse output pattern is generally separate between CW and CCW, but this can be changed to the direction designated output pattern.

(8) Triangular Drive Prevention Function

In order to avoid the triangular drive which starts decelerating without reaching the high speed in the S-RATE INDEX DRIVE due to shortage of pulse number, this function enables to designate pulse number for the top constant speed in advance and to secure constant speed operating ranges.

(9) END PULSE Drive Function

In order to reduce damping at the end of the INDEX DRIVE and the S-RATE INDEX DRIVE, this function enables to make a continuous drive of designated frequency and of designated pulse number after the end of deceleration up to the low speed.

(10)Origin Drive Direction Changing Function

The precondition for origin drive is that the $\overline{\text{ORG}}$ (or $\overline{\text{NORG}}$) sensor has been installed at the -(CCW) limit side along works, but the origin drive direction changing function enables to install the $\overline{\text{ORG}}$ (or $\overline{\text{NORG}}$) sensor on the +(CW) limit side.

(11)Margin Time Function

In order to prevent the origin drive from malfunctioning due to hunting, this function enables to insert a margin time between the sensor signal detection and the pulse stop.

(12)SOFT LIMIT Function

This function allows you to set up CW or CCW SOFT LIMIT.

(13)DEND ERROR Detection Function

If active level of $\overline{\text{DEND}}$ signal is not returned during the predetermined time span, this function ends the drive forcibly by setting 1 to the error bit of STATUS1 PORT.

(14)Origin Sensor Type Select Function

This function allows you to switch the $\overline{\text{ORG}}$ sensor detection approach from the edge sensing to the level sensing.

(15)ORIGIN ERROR Detection Function

Specifying the maximum number of pulses to be output during the CONSTANT SCAN DRIVE process and JOG DRIVE process, this function can end the drive forcibly if the sensor fails to make detection during that range of pulse number.

(16)PO Input Function

This function offers origin detection utilizing PO (excitation) output signal from stepping motor drivers. When PO input is enabled, ANDing of $\overline{\text{PO}}$ signal and $\overline{\text{ORG}}$ signal is output as $\overline{\text{ORG}}$ signal.

(17)AUTO DRST Output Function

This function automatically outputs $\overline{\text{DRST}}$ signal as the machine origin detection completed.

(18)Special DRST Output Function

This function allows you to constantly generate $\overline{\text{DRST}}$ output.

(19) Asymmetric S-RATE DRIVE function

Acceleration/deceleration constant can be set separately in the S-shaped DRIVE.

(20) S-RATE DRIVE triangular drive workaround function

The DRIVE profile is rounded automatically when there are few output pulses in the S-shaped DRIVE, thereby working around the triangular drive. It should be noted, however, that this is disabled in the asymmetric S-RATE DRIVE.

(21) SPEED/RATE CHANGE speed increase function

The operation from the writing of CHANGE command is performed on the real time basis in the SCAN DRIVE.

(22) AUTO CHANGE function

The SPEED and RATE are changed automatically according to the preset number of output pulses, speed or time.

(23) DRIVE calculation function

The number of acceleration pulses, acceleration time and INDEX DRIVE drive time can be obtained by simulated calculation.

4. I／O PORTS

4-1. I/O Port Table

Axis	Low-order Address	PORT name	R/W
X axis MCC05v2	00_H	DRIVE COMMAND	
	01_H	DRIVE DATA1	w
	02_H	DRIVE DATA2	r
	03_H	DRIVE DATA3	i
	04_H	COUNTER COMMAND	t
	05_H	COUNTER DATA1	e
	06_H	COUNTER DATA2	
	07_H	COUNTER DATA3	
	00_H	STATUS1	
	01_H	DRIVE DATA1	
	02_H	DRIVE DATA2	r
	03_H	DRIVE DATA3	e
	04_H	STATUS2	a
	05_H	STATUS3	d
	06_H	STATUS4	
	07_H	STATUS5	
Y axis MCC05v2	10_H	DRIVE COMMAND	
	11_H	DRIVE DATA1	w
	12_H	DRIVE DATA2	r
	13_H	DRIVE DATA3	i
	14_H	COUNTER COMMAND	t
	15_H	COUNTER DATA1	e
	16_H	COUNTER DATA2	
	17_H	COUNTER DATA3	
	10_H	STATUS1	
	11_H	DRIVE DATA1	
	12_H	DRIVE DATA2	r
	13_H	DRIVE DATA3	e
	14_H	STATUS2	a
	15_H	STATUS3	d
	16_H	STATUS4	
	17_H	STATUS5	
Z axis MCC05v2	20_H	DRIVE COMMAND	
	21_H	DRIVE DATA1	w
	22_H	DRIVE DATA2	r
	23_H	DRIVE DATA3	i
	24_H	COUNTER COMMAND	t
	25_H	COUNTER DATA1	e
	26_H	COUNTER DATA2	
	27_H	COUNTER DATA3	
	20_H	STATUS1	
	21_H	DRIVE DATA1	
	22_H	DRIVE DATA2	r
	23_H	DRIVE DATA3	e
	24_H	STATUS2	a
	25_H	STATUS3	d
	26_H	STATUS4	
	27_H	STATUS5	
A axis MCC05v2	30_H	DRIVE COMMAND	
	31_H	DRIVE DATA1	w
	32_H	DRIVE DATA2	r
	33_H	DRIVE DATA3	i
	34_H	COUNTER COMMAND	t
	35_H	COUNTER DATA1	e
	36_H	COUNTER DATA2	
	37_H	COUNTER DATA3	
	30_H	STATUS1	
	31_H	DRIVE DATA1	
	32_H	DRIVE DATA2	r
	33_H	DRIVE DATA3	e
	34_H	STATUS2	a
	35_H	STATUS3	d
	36_H	STATUS4	
	37_H	STATUS5	

Axis	Low-order Address	PORT name	R/W
B axis MCC05v2	40_H	DRIVE COMMAND	
	41_H	DRIVE DATA1	w
	42_H	DRIVE DATA2	r
	43_H	DRIVE DATA3	i
	44_H	COUNTER COMMAND	t
	45_H	COUNTER DATA1	e
	46_H	COUNTER DATA2	
	47_H	COUNTER DATA3	
	40_H	STATUS1	
	41_H	DRIVE DATA1	
	42_H	DRIVE DATA2	r
	43_H	DRIVE DATA3	e
	44_H	STATUS2	a
	45_H	STATUS3	d
	46_H	STATUS4	
	47_H	STATUS5	
C axis MCC05v2	50_H	DRIVE COMMAND	
	51_H	DRIVE DATA1	w
	52_H	DRIVE DATA2	r
	53_H	DRIVE DATA3	i
	54_H	COUNTER COMMAND	t
	55_H	COUNTER DATA1	e
	56_H	COUNTER DATA2	
	57_H	COUNTER DATA3	
	50_H	STATUS1	
	51_H	DRIVE DATA1	
	52_H	DRIVE DATA2	r
	53_H	DRIVE DATA3	e
	54_H	STATUS2	a
	55_H	STATUS3	d
	56_H	STATUS4	
	57_H	STATUS5	
	70_H	Board identifier	R

4-2. Setting the board identifier port and rotary switch

(1) Board identifier port

This is a port to read out the board number assigned by the rotary switch.

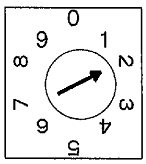
D^7	D^6	D^5	D^4	D^3	D^2	D^1	D^0
Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Rotary switch 2^3 BIT	Rotary switch 2^2 BIT	Rotary switch 2^1 BIT	Rotary switch 2^0 BIT

* "1" when active

- D^0 — Rotary switch 2^0 BIT
- D^1 — Rotary switch 2^1 BIT
- D^2 — Rotary switch 2^2 BIT
- D^3 — Rotary switch 2^3 BIT
- D^7–D^4 — Reserved (0)

(2) Setting the rotary switch

C-871 board numbers are assigned by the rotary switch on the board.
If more than two C-871s are used, numbers must be assigned artificially.
Set them by avoiding possible overlaps. When only one C-871 is used, set the rotary switch to "0". (This setting is made at the time of shipment).
The following sketch shows the example where the board is set to 2.



4-3. DRIVE COMMAND PORT

Used to write a drive command. For the details of commands, refer to Chapter 6.

4-4. DRIVE DATA1, 2, 3 PORT (WRITE)

Various data are written by each drive command.

4-5. DRIVE DATA1, 2, 3 PORT (READ)

Used to read various data.
When reading data by the ADDRESS READ command, SET DATA READ command and ERROR STATUS READ command, confirm BUSY BIT=0 in STATUS1 after writing a command.
Reading the count data of the PULSE COUNTER, DIFFERENTIAL COUNTER and ADDRESS COUNTER is always enabled.

4-6. COUNTER COMMAND PORT

Used to write a command to control the PULSE COUNTER and DIFFERENTIAL COUNTER.
For the details of commands, refer to Chapter 9.

4-7. COUNTER DATA1, 2, 3 PORT (WRITE)

Used to write various data by counter command.

4-8. STATUS1 PORT

Used to read the current status of each axis. This read operation is always enabled.

D^7	D^6	D^5	D^4	D^3	D^2	D^1	D^0
FSEND BIT	SSEND BIT	LSEND BIT	MAN BIT	ERROR BIT	DREND BIT	DRIVE BIT	BUSY BIT

* Each of the bits becomes 1 in the active state.

BUSY BIT : When this bit is 0, it indicates that writing a command to the corresponding axis is enabled. When this bit is 1, it indicates that the corresponding axis is in the driving status or in the data processing status. The command is ignored. A command must be written after confirming BUSY BIT=0.
However, special commands(refer to the paragraph 6-2) can be written even when BUSY BIT=1.

DRIVE BIT : When this bit is 1, it indicates that the corresponding axis is in the driving status.

DREND BIT : When this bit is 1, it indicates that the drive of the corresponding axis has been terminated (Note 1,3). Judge the termination of driving when multiple control.
This bit is reset by writing the next command.

ERROR BIT : This bit indicates that the written command is undefined or has a data error(Note 1,3).
You can check description of the error using the ERROR STATUS READ Command.
This bit is reset by writing the next command.

MAN BIT : This bit is not used on this product. 0 is output (Note3).

LSEND BIT : When DRIVE BIT=1, it indicates a valid CWLM or CCWLM signal has been entered.
When DRIVE BIT=0, it indicates pulse output has been stopped by CWLM or CCWLM signal (includes the output stopped by an applied function SOFT LIMIT).
This bit is reset at start of the next drive (Note2).

SSEND BIT : When DRIVE BIT=1, it indicates the SLOW STOP Command has been entered.
When DRIVE BIT=0, it indicates pulse output has been stopped by the SLOW STOP Command.
This bit is reset at start of the next drive (Note2).

FSEND BIT : When DRIVE BIT=1, it indicates the FSSTOP signal or FSSTOP Command has been entered.
When DRIVE BIT=0, it indicates pulse output has been stopped by the FSSTOP signal or FAST STOP Command. This bit is reset at start of the next drive (Note2).

Note1: It is valid only when BUSY=0.
Note2: Reset by the up edge of the DRIVE signal. Not reset by a command not accompanied by drive.
Note3: **When the power is turned on or the RESET signal is input, each of DREND, ERROR and MAN bits has an undefined value. So, check only whether BUSY bit is 0, run an NOP command and then initialize the DREND, ERROR and MAN bits.**

4-9. STATUS2 PORT

Used to read the input signal state of each axis. Reading is always enabled.
The contents shown below are of the X axis but also applicable to another axis.

D7	D6	D5	D4	D3	D2	D1	D0

* Each of the bits becomes 1 in the active state.

Bit	Signal	Level
D0	$\overline{XDEND}/\overline{PO}$ signal	0=HIGH 1=LOW
D1	$\overline{XORG}$ signal	0=HIGH 1=LOW
D2	$\overline{XNORG}$ signal	0=HIGH 1=LOW
D3	XCWLM signal	0=LOW 1=HIGH
D4	XCCWLM signal	0=LOW 1=HIGH
D5	Unused(0)	
D6	Unused(0)	
D7	$\overline{XDEND}/\overline{PO}$ signal (same to D^0)	

Note: This STATUS PORT is real-time data.

4-10. STATUS3 PORT

Used to read the status information from the PULSE COUNTER and DIFFERENTIAL COUNTER.
Reading is always enabled.

D7	D6	D5	D4	D3	D2	D1	D0

* Each of the bits becomes 1 in the active state.

Bit	Signal
D0	PLS COMP1 BIT
D1	PLS COMP2 BIT
D2	PLS COMP3 BIT
D3	PLS COMP4 BIT
D4	PLS COMP5 BIT
D5	PLS OVF BIT
D6	DFL COMP1 BIT
D7	DFL COMP2 BIT

PLS*COMP1 BIT : Indicates that the PULSE COUNTER has matched the COMPARE REGISTER1. (Note)
PLS COMP2 BIT : Indicates that the PULSE COUNTER has matched the COMPARE REGISTER2. (Note)
PLS COMP3 BIT : Indicates that the PULSE COUNTER has matched the COMPARE REGISTER3. (Note)
PLS COMP4 BIT : Indicates that the PULSE COUNTER has matched the COMPARE REGISTER4. (Note)
PLS COMP5 BIT : Indicates that the PULSE COUNTER has matched the COMPARE REGISTER5. (Note)
PLS OVF BIT : Indicates that the PULSE COUNTER has overflowed.
DFL*COMP1 BIT : Indicates the DIFFERENTIAL COUNTER≧DFL COMPARE REGISTER1 (excessive deviation). (Note)
DFL COMP2 BIT : Indicates the DIFFERENTIAL COUNTER≦DFL COMPARE REGISTER2 (positioning complete).(Note)

Note: In the initial state, there bits are reset after this status has been read except for when the following state is indicated matching between the PULSE COUNTER and REGISTER, excessive deviation or positioning complete has been indicated. It is enable to reset all the time after status has been read by setting each of the COUNTER INITIALIZE COMMAND.

*** Throughout this manual, the abbreviations "PLS" and "DFL" stands for PULSE and DIFFERENTIAL, respectively.**

4-11. STATUS4 PORT

Used to read the input signal state of each axis. Reading is always enabled.

D7	D6	D5	D4	D3	D2	D1	D0

* Each of the bits becomes 1 in the active state.

Bit	Signal	Level
D0	FSSTOP signal (common to all axes)	0=LOW 1=HIGH
D1	//	
D2	$\overline{ASENSOR}$ input signal (Note1)	0=HIGH 1=LOW
D3	-XEA input signal	0=LOW 1=HIGH
D4	-XEB input signal	0=LOW 1=HIGH
D5	$\overline{XDRST}$ output signal	0=HIGH 1=LOW
D6, D7	Unused (undefined)	

Note1: $\overline{SENSOR}$ input signal is available for the A and Z axes alone. 0 is output for other axes.
For details of $\overline{SENSOR}$ input signal, refer to the User's Manual [Applied Functions Part].
Note2: This status is real time data of input/output pins.

4-12. STATUS5 PORT

This port is used for reading current state of the SOFT LIMIT (An applied function, see Note) and speed change. Data read is always enabled.

D^7	D^6	D^5	D^4	D^3	D^2	D^1	D^0

* Each of the bits becomes 1 in the active state.

- D^0: CCW SOFT LIMIT BIT (Note1)
- D^1: CW SOFT LIMIT BIT (Note1)
- D^2: Unused(0)
- D^3: RDYINT BIT (Note2)
- D^4: Unused(0)
- D^5: SPEED CHANGE BUSY BIT
- D^6: RATE CHANGE BUSY BIT (Note1)
- D^7: INDEX CHANGE BUSY BIT (Note1)

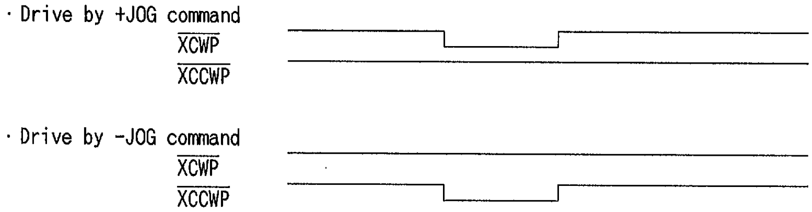
Note1: For details, see User's Manual(Applied Functions Part).
Note2: When interrupt is used, RDYINT request axis is identified by this bit.

5. DETAILS OF DRIVE FUNCTIONS

5-1. JOG DRIVE Function

This function performs 1-pulse drive by the +/- JOG command.

Example of X axis:

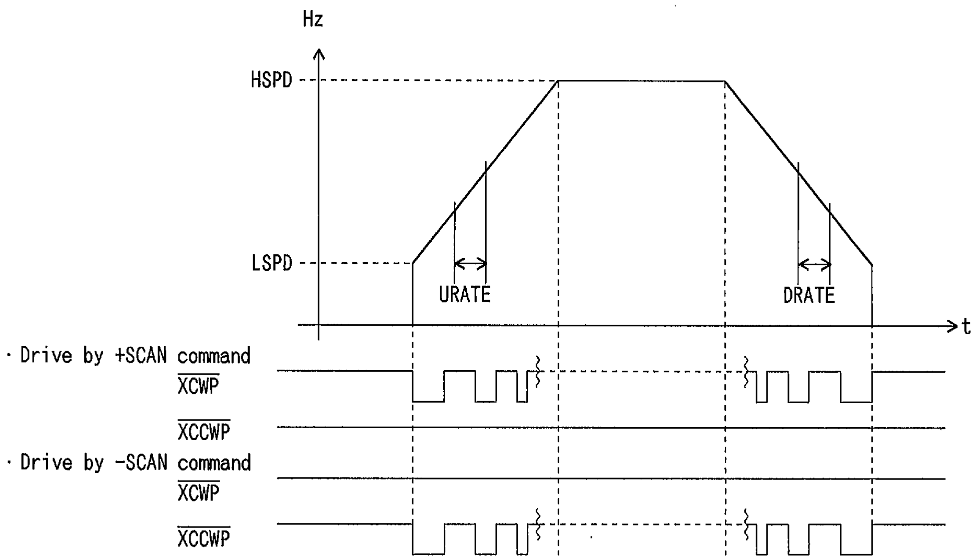


There is not any data required for JOG DRIVE.

5-2. SCAN DRIVE Function

This function accelerating/decelerating drive by the +/- SCAN command.
The drive is stopped by one of the methods described in 5-10., 5-11. and 5-12.

Example of X axis:



The data required for SCAN DRIVE are as follows:

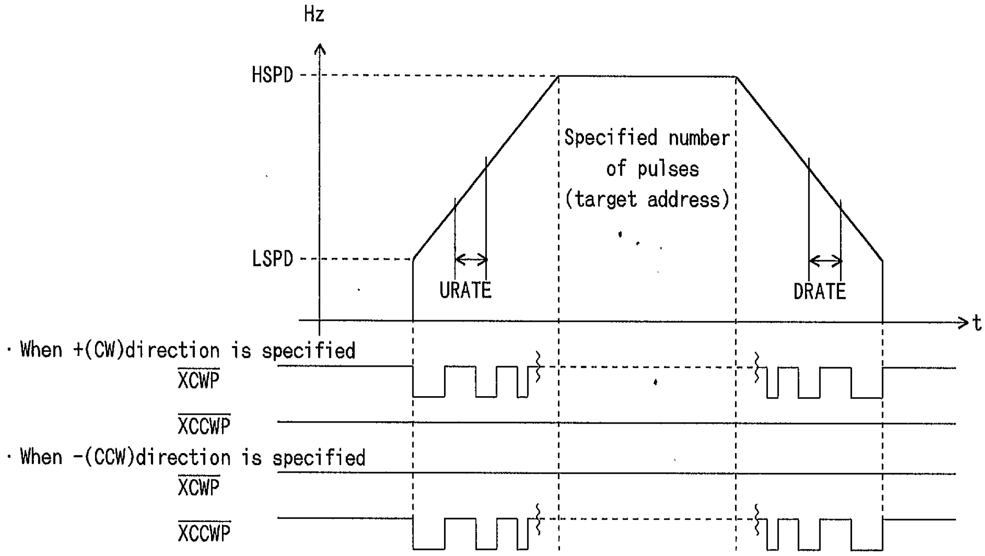
Data name	Setting command
HSPD (HIGH SPEED)	HSPD SET
LSPD (LOW SPEED)	LSPD SET
URATE (acceleration time constant)	RATE SET
DRATE (deceleration time constant)	RATE SET

Note: When LSPD≧HSPD is specified, constant-speed drive is performed with HSPD.

5-3. INDEX DRIVE Function

Accelerating/decelerating drive is performed with the specified number of pulses by the INCREMENTAL INDEX command. (or up to the target address by ABSOLUTE INDEX command).

Example of X axis:



The data required for INDEX DRIVE is as follows.

Data name	Setting command
HSPD (HIGH SPEED)	HSPD SET
LSPD (LOW SPEED)	LSPD SET
URATE (acceleration time constant)	RATE SET
DRATE (deceleration time constant)	RATE SET
Specified number of pulses (target address)	When INDEX DRIVE is started

Note1: When LSPD≧HSPD is specified, constant-speed drive is performed with HSPD.

Note2: When LSPD<HSPD and URATE≠DRATE are specified, the timing up to a pulse output is different from that of URATE=DRATE. For details refer to Chapter 12. Timing.
For this reason, please use the same data unless specially required.

5-4. Drive Speed Change Function

Using the SPEED CHANGE command allows you to change speed of the SCAN or INDEX DRIVE currently taking place (note that this command is valid only for these two drive types).
The drive is accelerated or decelerated to the speed specified by the SPEED CHANGE Command.

Note1: This speed change is not available when the INDEX DRIVE is taking place at URATE≠DRATE.

Note2: Speed change is available within the range of LSPD<Change speed<HSPD.

Note3: When the SPEED CHANGE Command has been executed, you must wait until the command is internally accepted before requesting another speed change using this command. Check the SPEED CHANGE BUSY BIT in the STATUS5 PORT to make sure that the SPEED CHANGE Command is executable.

5-5. Machine Origin Detecting Function (ORIGIN DRIVE)

Drive is performed up to machine origin detection by the ORIGIN command.
The drive up to machine origin detection is performed by a combination of JOG DRIVE, CONSTANT SCAN DRIVE, SCAN DRIVE and ABSOLUTE INDEX DRIVE.
There are 9 types of machine origin detection. For the details of the types and processes, refer to Chapter 7.

The data required for ORIGIN DRIVE is as follows:

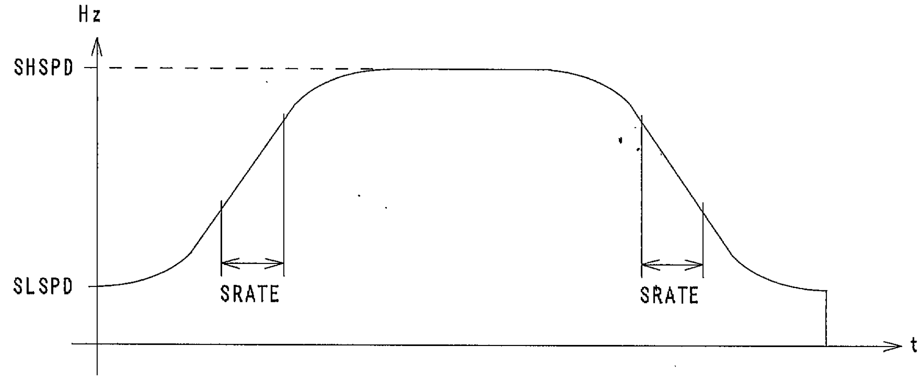
Data name	Setting command
HSPD (HIGH SPEED)	HSPD SET
LSPD (LOW SPEED)	LSPD SET
CSPD (CONSTANT SPEED)	CSPD SET
URATE (acceleration time constant)	RATE SET
DRATE (deceleration time constant)	RATE SET
OFFSET PULSE	OFFSET PULSE SET
LDELAY (LIMIT DELAY TIME)	ORIGIN DELAY SET
SDELAY (SCAN DELAY TIME)	ORIGIN DELAY SET
JDELAY (JOG DELAY TIME)	ORIGIN DELAY SET

5-6. Machine Origin Detecting Function Using Limit Sensor

Two of the machine origin detection sensors can use CCW LIMIT signal as the origin sensor.
This function helps reducing number of sensors.
Refer to Chapter 7 for details of the models and processes.

5-7. S-RATE SCAN DRIVE Function

S-shaped accelerating/decelerating drive is performed by the +/- S-RATE SCAN command.
At the S-shaped accelerating/decelerating drive, the speed difference between SLSPD and SHSPD is divided into 3 equal parts. In the mid-speed area of the 3 equally-divided parts, linear acceleration/deceleration is performed by SRATE.
In the remaining areas, curvilinear smooth acceleration/deceleration is performed.
The drive is stopped by one of the methods described in 5-10., 5-11. and 5-12.



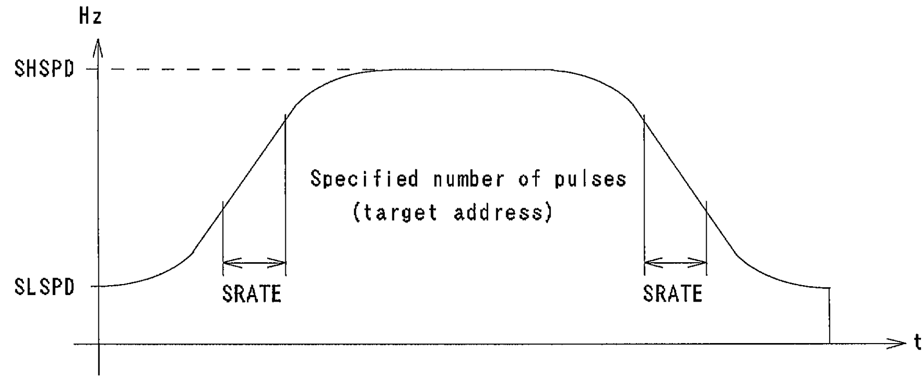
The data required for S-RATE SCAN DRIVE is as follows:

Data name	Setting command
SHSPD (high speed for S-RATE DRIVE)	SHSPD SET
SLSPD (low speed for S-RATE DRIVE)	SLSPD SET
SRATE (acceleration/deceleration time constant for S-RATE DRIVE)	SRATE SET

Note: When SLSPD≧SHSPD is specified, constant-speed drive is performed with SHSPD.

5-8. S-RATE INDEX DRIVE Function

S-shaped accelerating/decelerating drive is performed with the specified number of pulses (or up to the target address) by the S-RATE INCREMENTAL INDEX command (or S-RATE ABSOLUTE INDEX command).
The acceleration/deceleration rate characteristics are the same as those of S-RATE SCAN DRIVE.



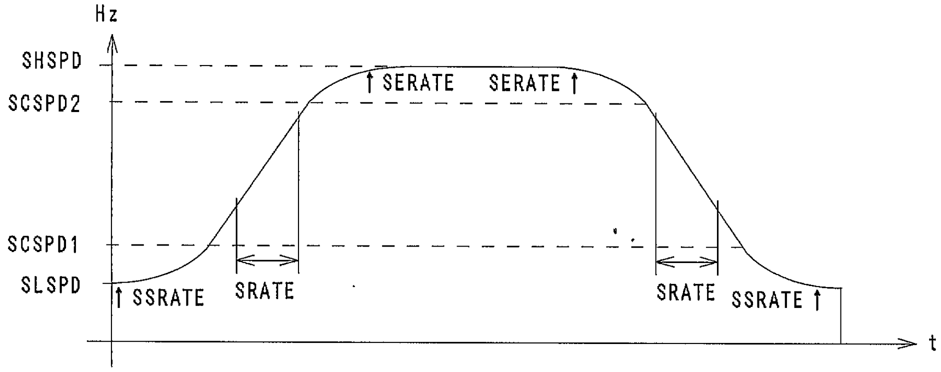
The data required for S-RATE INDEX DRIVE is as follows:

Data name	Setting command
SHSPD (high speed for S-RATE DRIVE)	SHSPD SET
SLSPD (low speed for S-RATE DRIVE)	SLSPD SET
SRATE (acceleration/deceleration time constant for S-RATE DRIVE)	SRATE SET
Specified number of pulses (target address)	When S-RATE INDEX DRIVE is started

Note: When SLSPD≧SHSPD is specified, constant-speed drive is performed with SHSPD.

5-9. S-RATE DRIVE Parameter Adjusting Function

For S-RATE DRIVE, internal parameters can be adjusted. To perform S-RATE DRIVE, the 4 internal parameters of SSRATE, SERATE, SCSPD1 and SCSPD2 are required first.
Usually, these parameters are automatically set to initial values when SRATE, SLSPD and SHSPD are set but can be adjusted to optional values by respective adjust commands.



The adjustable parameters are as follows.

Data name	Adjust command
SSRATE (time constant at a start of acceleration and an end of deceleration)	SSRATE ADJUST
SERATE (time constant at an end of acceleration and at a start of deceleration)	SERATE ADJUST
SCSPD1 (linear rate start speed at acceleration and linear rate end speed at deceleration)	SCSPD1 ADJUST
SCSPD2 (linear rate end speed at acceleration and linear rate start speed at deceleration)	SCSPD2 ADJUST

(1) SSRATE

Explanation of data …… Indicates the momentary time constant at a start of acceleration and at a end of deceleration.
The time constant changes smoothly from SSRATE to SRATE between SLSPD and SCSPD1.

Initial value …………… Automatically set nearly to an eightfold value of SRATE by the SRATE SET command.

Note1: When DRIVE TYPE is the fixed mode, the initial value of SSRATE may be a nonexistent value in the rate data table depending on the SRATE value.
(Even if the eightfold value of SRATE does not exist in the rate data table, the eightfold value of SRATE is adopted as an initial value of SSRATE.)

Note2: When the SRATE value is large, the SSRATE value cannot exceed the maximum settable rate value. The maximum rate value in each drive type is approximately as shown below.

L-TYPE ……………… Approx. 1030ms/1000Hz
M-TYPE ……………… Approx. 51.5ms/1000Hz
H-TYPE ……………… Approx. 5.15ms/1000Hz
Arithmetic mode … When resolution data is supposed to be D
Maximum rate value=1,030 ÷ D(ms/1000Hz)
(For the details of the arithmetic mode,
refer to the User's Manual [Applied Functions Part].)

Adjusting range ………… SSRATE≧SRATE

Note1: When SSRATE<SRATE, is specified, SSRATE=SRATE.

Note2: After the SRATE SET command is executed, the SSRATE value adjusted prior to the execution is invalidated and reset to the initial value.
When DRIVE TYPE has been changed by the SPEC INITIALIZE1 command, the SSRATE value is also reset to the initial value.

(2) SERATE

Explanation of data …… Indicates the time constant at an end of acceleration and at a start of deceleration.
The time constant changes smoothly from SRATE to SERATE between SCSPD2 and SHSPD.

Initial value …………… The same as SSRATE.

Adjusting range ………… The same as SSRATE.

(3) SCSPD1

Explanation of data …… Indicates the linear rate start speed or linear rate end speed by SRATE. Between SCSPD1 and SCSPD2, the time constant indicates a linear rate characteristic because the SRATE value is fixed.

Initial value …………… Set to the following value represented by the following expression by the SLSPD SET or SHSPD SET command.

$$SCSPD1 = SLSPD + (SHSPD - SLSPD) \times \frac{1}{3}$$

Adjusting range ………… $SLSPD \leqq SCSPD1 \leqq SCSPD2$

Note1: When $SCSPD1 < SLSPD$ is specified, $SCSPD1 = SLSPD$.
When $SCSPD1 > SCSPD2$ is specified, $SCSPD1 = SCSPD2$.

Note2: After the SLSPD SET or SHSPD SET command is executed, the SCSPD1 value adjusted prior to the execution is invalidated and reset to the initial value.
When DRIVE TYPE has been changed by the SPEC INITIALIZE1 command, the SCSPD1 value is reset to the initial value.

(4) SCSPD2

Explanation of data …… Indicates the linear rate end speed or linear rate start speed by SRATE. Between SCSPD1 and SCSPD2, the time constant indicates a linear rate characteristic because the SRATE value is fixed.

Initial value …………… Set to the value represented by the following expression by the SLSPD SET or SHSPD SET command.

$$SCSPD2 = SLSPD + (SHSPD - SLSPD) \times \frac{2}{3}$$

Adjusting range ………… $SCSPD1 \leqq SCSPD2 \leqq SHSPD$

Note1: When $SCSPD2 < SCSPD1$ is specified, $SCSPD2 = SCSPD1$.
When $SCSPD2 > SHSPD$ is set, $SCSPD2 = SHSPD$.

Note2: After the SLSPD SET or SHSPD SET command is executed, the SCSPD2 value adjusted prior to the execution is invalidated and reset to the initial value.
When DRIVE TYPE has been changed by the SPEC INITIALIZE1 command, the SCSPD2 value is also reset to the initial value.

5-10. Slow Stop Function

The pulse output can be put into a slow stop by the SLOW STOP command.
After the pulse output is stopped in the above way, SSEND becomes 1.

5-11. Fast Stop Function

WARNING	When you use emergency stop in response to system failure, also turn off power of the drive system. If an error has occurred to the controller or wiring system, the Fast Stop may not work. This may lead to a serious accident. For details, refer to Chapter 19.

The pulse output can be put into an fast stop by the FSSTOP command or FSSTOP signal.
After the pulse output is stopped in the above way, FSEND becomes 1.

FSSTOP signal stops six axes, X, Y, Z, A, B and C immediately.

5-12. LIMIT Stop Function

WARNING	When you use emergency stop in response to system failure, also turn off power of the drive system. If an error has occurred to the controller or wiring system, LIMIT Stop may not work. This may lead to a serious accident. For details, refer to Chapter 19.

CAUTION	If any error has occurred to the system or your setting is incorrect, the machine or workpiece damage and personal injury may occur. To prevent such an accident, be sure to use the LIMIT Stop function for the equipment except on the rotation system.

CAUTION	When you use the LIMIT stop as slow stop, there may be a collision with the limit position of the mechanical device before stop. This may lead to machine or workpiece damage. Note that the stop point will be changed if RATE, HSPD, etc. are changed.

The pulse output can be stopped by the CWLM signal when the pulse output is in the +(CW)direction or by the CCWLM signal when the same output is in the -(CCW)direction. After the pulse output is stopped in the above way, LSEND becomes 1. The LIMIT stop type can be switched between fast stop and slow stop by the SPEC INITIALIZE1 command. "Fast stop" is selected at POWER ON/RESET.

5-13. Function for Servo Driver

The target motor can be switched by the SPEC INITIALIZE1 command. The target motor is a servo motor or a stepping motor. "Stepping motor" is selected as the target motor at POWER ON/RESET.
The signals for servo motor are as follows.

$\overline{\text{DEND}}$ input signal: The deviation COUNTER END signal from the servo driver is input. Even after the completion of pulse output, the driving status is kept until $\overline{\text{DEND}}$=LOW is confirmed, and the command is not terminated with BUSY and DRIVE BIT=1.

$\overline{\text{DRST}}$ output signal: The deviation COUNTER RESET signal is output to the servo driver. If the pulse output is put into an fast stop, the $\overline{\text{DRST}}$ signal=LOW is output for 10ms and the deviation COUNTER of the servo driver is reset. The $\overline{\text{DRST}}$ signal=LOW can also be output optionally for 10ms by the SERVO RESET command.

The above signals are disabled when a stepping motor is selected as the target.
In this case, $\overline{\text{DEND}}$ input and $\overline{\text{DRST}}$ output can be used as a general purpose input and general purpose output, respectively. For their usage, refer to the User's Manual [Applied Functions Part].

5-14. Current Position Reading Function

The current position can read by the ADDRESS READ command.
The guaranteed data range is a pulse area of +8,388,607 to -8,388,607. The current position is reset to 0 at POWER ON/RESET and can also be set to an optional value by the ADDRESS INITIALIZE command.

5-15. Interrupt Request Function

(1) After the termination of a command, an interrupt request (RDYINT signal) can be generated for the initiator. RDYINT signal is generated too at the stop(the termination of a command) of FSSTOP, STOP, and LIMIT, etc.
There are 3 interrupt request generation patterns as shown below.
One of them is selected by the SPEC INITIALIZE1 command. Item 1 is selected at POWER ON/RESET.
  1. Output only when a command accompanied by pulse output is terminated.
  2. Output when every command is terminated(except special commands).
  3. No output in any case.

Note: This function is not effective when the COUNTER command is executed or when special commands are executed. For details of special commands, Refer to the description of 6-2

(2) Interrupt request (CNTINT signal) may occur in response to any given count value of the PULSE COUNTER.
Interrupt request (DFLINT signal) may occur in response to the value above or below any given count value of the DEFERENTIAL COUNTER. For details, refer to the description of PULSE/DEFERENTIAL COUNT COMPARE function in DETAILS OF COUNTER FUNCTION in Chapter 8.

5-16. Speed Data Hz Unit Setting Function

Speed data (HSPD, LSPD, CSPD, SHSPD, SLSPD, and HSPD1 to 10 when an applied function is used) can be set as 3-byte data in Hz.
The data setting range is 1 to 3,333,333 so the speed can be set in the range of 1Hz to 3.3MHz.

*Speed setting example
When 10000 (002710_H) has been set in the HSPD:
HSPD=10000Hz

However, the MCC05v2 output frequency is controlled by counting the reference clocks (40MHz).
For this reason, a frequency that cannot be physically output may appear for the set value of speed data.
Consequently, there will be a difference, specially in a high speed area, between the set value and the real output frequency.
Supposing that the set value of speed data is F', the real output frequency F is represented by the following expression.

$$F=\frac{160,000,000}{\text{Integer part of }\left[\underset{\sim}{160,000,000/F'}\right]}\ (\text{Hz})$$

In the above expression, the part following the decimal point of the wavy-underlined number will be ignored, so that the real output frequency becomes a little higher than the set value.
Take this point into consideration when accuracy is required between the set value and the real output.

5-17. DRIVE TYPE Switching Function

The acceleration/deceleration time constant method when the MCC05v2 is put in accelerating/decelerating drive can be roughly classified into two modes, fixed data mode and arithmetic mode.
For the fixed data mode, the 3 types of L-TYPE, M-TYPE and H-TYPE are available for reasons of output frequency,acceleration/deceleration time constant setting range and speed difference at acceleration/deceleration, etc.
For the arithmetic mode, refer to the User's Manual [Applied Functions Part].
In the fixed data mode, the acceleration/deceleration time constant (URATE, DRATE) is fixed by the data table beforehand, so the user must specify an appropriate time constant by No. of the data table.
For the rate data table, refer to Chapter 18. The speed range, rate range and speed difference at acceleration/deceleration in each type are as follows.

	Fixed data mode			Arithmetic mode
	L-TYPE	M-TYPE	H-TYPE	
Speed range (LSPD,SLSPD)	10Hz～100kHz	10Hz～800kHz	10Hz～3.3MHz	10Hz～3.3MHz
Speed range (except LSPD,SLSPD)	1Hz～100kHz	1Hz～800kHz	1Hz～3.3MHz	1Hz～3.3MHz
Rate range	1000ms/1000Hz ～1.0ms/1000Hz	50ms/1000Hz ～0.05ms/1000Hz	5ms/1000Hz ～0.005ms/1000Hz	1030ms/1000Hz ～0.004ms/1000Hz
Speed difference (Note1)	51Hz/STEP ～62Hz/STEP	1kHz/STEP ～4kHz/STEP	10kHz/STEP ～68kHz/STEP	51Hz/STEP ～68kHz/STEP

Note1: The speed difference indicates a speed difference between before and after speed change at acceleration/deceleration. This speed difference is rather small at a low speed and gradually increases as the speed is accelerated.

5-18. Present Speed Reading Function

Speed data can be read from the DRIVE DATA1, 2 and 3 PORTs during drive, and read data can be converted into present speed by using the following formula:

$$\text{Present Speed} = \frac{160{,}000{,}000}{V} \quad (\text{Hz})$$

where, V : Read data

Note1: The range of speed to be read by using this function is from about 9.5Hz to 3.3MHz because the data length is 3byte. Be careful in reading speeds in the slow area.

Note2: Since the DRIVE DATA1, 2 and 3 PORTs are generally dedicated to read values of the PULSE COUNTER, it is necessary to change the port function to reading speed data in case of reading speed.
To change the function, the SPEED PORT SELECT Command is used.

5-19. Set Data Reading Function

The SET DATA READ Command allows you to read various set data or SPEC INITIALIZE DATA.
This function is helpful for debugging or for applications requiring high reliability since it enables to re-confirm data already set to each axis.

6. DESCRIPTION OF BASIC FUNCTION DRIVE COMMANDS AND OPERATION SEQUENCES

Execute each command to the port (refer to 4-1.) of the axis to which the command is executed.
In the following, a description is give about the X axis MCC05v2. The same is also applicable to the another axis.

6-1. Basic Function DRIVE Command Table

The mark * denotes a command accompanied by pulse output.

	$D^7D^6D^5D^4D^3D^2D^1D^0$	HEX CODE	COMMAND NAME	Execution time
	0 0 0 0 0 0 0 0	0 0	NO OPERATION	MAX 20μs
	0 0 0 0 0 0 0 1	0 1	SPEC INITIALIZE1	MAX 1.2ms(Note1)
	0 0 0 0 0 0 1 0	0 2	PULSE COUNTER INITIALIZE	MAX 25μs
	0 0 0 0 0 0 1 1	0 3	ADDRESS INITIALIZE	MAX 30μs
	0 0 0 0 0 1 0 0	0 4	ADDRESS READ	MAX 25μs
	0 0 0 0 0 1 0 1	0 5	SERVO RESET	MAX 11ms
	0 0 0 0 0 1 1 0	0 6	RATE SET	MAX 60μs(Note1)
	0 0 0 0 0 1 1 1	0 7	LSPD SET	MAX 95μs(Note1)
	0 0 0 0 1 0 0 0	0 8	HSPD SET	MAX 85μs
	0 0 0 0 1 0 0 1	0 9	DFL COUNTER INITIALIZE	MAX 25μs
	0 0 0 0 1 0 1 0	0 A	SET DATA READ	MAX 35μs
		0 B～0 F	Setting is disabled.	(Note3)
*	0 0 0 1 0 0 0 0	1 0	+JOG	(Note2)
*	0 0 0 1 0 0 0 1	1 1	-JOG	(Note2)
*	0 0 0 1 0 0 1 0	1 2	+SCAN	(Note2)
*	0 0 0 1 0 0 1 1	1 3	-SCAN	(Note2)
*	0 0 0 1 0 1 0 0	1 4	INCREMENTAL INDEX	(Note2)
*	0 0 0 1 0 1 0 1	1 5	ABSOLUTE INDEX	(Note2)
		1 6～1 7	Setting is disabled.	――――
		1 8～1 9	Setting is disabled.	(Note3)
	0 0 0 1 1 0 1 0	1 A	CSPD SET	MAX 55μs
	0 0 0 1 1 0 1 1	1 B	OFFSET PULSE SET	MAX 20μs
	0 0 0 1 1 1 0 0	1 C	ORIGIN DELAY SET	MAX 25μs
	0 0 0 1 1 1 0 1	1 D	ORIGIN FLAG RESET	MAX 25μs
*	0 0 0 1 1 1 1 0	1 E	ORIGIN	(Note2)
	0 0 0 1 1 1 1 1	1 F	Setting is disabled.	――――
		2 0～5 F	Setting is disabled.	(Note3)
	0 1 1 0 0 0 0 0	6 0	SRATE SET	MAX 150μs
	0 1 1 0 0 0 0 1	6 1	SLSPD SET	MAX 150μs
	0 1 1 0 0 0 1 0	6 2	SHSPD SET	MAX 150μs
	0 1 1 0 0 0 1 1	6 3	SSRATE ADJUST	MAX 100μs
	0 1 1 0 0 1 0 0	6 4	SERATE ADJUST	MAX 100μs
	0 1 1 0 0 1 0 1	6 5	SCSPD1 ADJUST	MAX 100μs
	0 1 1 0 0 1 1 0	6 6	SCSPD2 ADJUST	MAX 100μs
		6 7～6 F	Setting is disabled.	――――
*	0 1 1 1 0 0 0 0	7 0	+ S-RATE SCAN	(Note2)
*	0 1 1 1 0 0 0 1	7 1	- S-RATE SCAN	(Note2)
*	0 1 1 1 0 0 1 0	7 2	S-RATE INCREMENTAL INDEX	(Note2)
*	0 1 1 1 0 0 1 1	7 3	S-RATE ABSOLUTE INDEX	(Note2)
		7 4～E 1	Setting is disabled.	――――
	1 1 1 0 0 0 1 0	E 2	ERROR STATUS READ	MAX 25μs
		E 3～F 1	Setting is disabled.	――――
		F 2～F 6	Setting is disabled.	(Note3)

Note1: When URATE ≠ DRATE is specified, the execution time as follows.

L-TYPE	M-TYPE	H-TYPE
MAX100ms	MAX 35ms	MAX 15ms

Note2: The execution time cannot be specified. Refer to Chapter 1 2. Timing.
Note3: Applied function DRIVE commands are assigned.
For details, refer to the User's Manual [Applied Functions Part].

6-2. **Special Command Table**

Special commands can always be executed, except just behind not Special command executed in less than 4μs.

$D^7D^6D^5D^4D^3D^2D^1D^0$	HEX CODE	COMMAND NAME	Execution time
1 1 1 1 0 1 1 1	F 7	SPEED CHANGE	(Note)
1 1 1 1 1 0 0 0	F 8	INT MASK	MAX 200ns
1 1 1 1 1 0 0 1	F 9	ADDRESS COUNTER PORT SELECT	MAX 200ns
1 1 1 1 1 0 1 0	F A	DFL COUNTER PORT SELECT	MAX 200ns
1 1 1 1 1 1 0 0	F C	PULSE COUNTER PORT SELECT	MAX 200ns
1 1 1 1 1 1 0 1	F D	SPEED PORT SELECT	MAX 200ns
1 1 1 1 1 1 1 0	F E	SLOW STOP	(Note)
1 1 1 1 1 1 1 1	F F	FAST STOP	(Note)

Note: The execution time cannot be specified. Refer to Chapter 1 2. Timing.

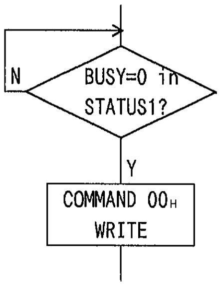
6-3. **NO OPERATION Command**

COMMAND········ 00_H

Function : No function

However, DREND BIT and ERROR BIT in STATUS1 are cleared.

Execution sequence

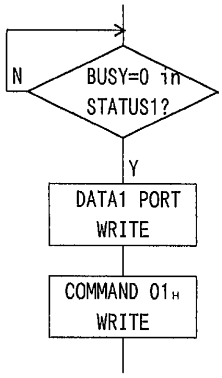


6-4. SPEC INITIALIZE1 Command

COMMAND········· 01H

Function : Defines operation specifications.

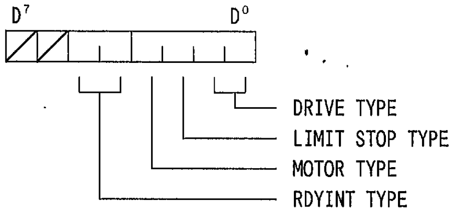
Execution sequence



Drive control specifications are defined in DRIVE DATA1.

The contents of DRIVE DATA1 PORT is as follows.

DATA1 PORT



The bit marked slash may be set to 0 or 1.

The details of each bit are shown below. The bit is set to the underlined side at POWER ON/RESET.

(1) DRIVE TYPE (D^1,D^0)
Specifies DRIVE TYPE.

D^1	D^0	DRIVE TYPE
0	0	L-TYPE
0	1	M-TYPE
1	0	H-TYPE
1	1	Arithmetic mode (Note)

Note: For the arithmetic mode, refer to the User's Manual [Applied Functions Part].

(2) LIMIT STOP TYPE (D^2)
Specifies the type of LIMIT STOP by the CWLM and CCWLM signals.
0: Fast stop　　　1: Slow stop

(3) MOTOR TYPE (D^3)
Specifies the target motor.
0: SERVO　　　1: STEPPING

(4) RDYINT TYPE (D^5,D^4)
Specifies an interrupt request (RDYINT) generation pattern upon termination of a command.

D^5	D^4	Generation pattern
0	0	Generated upon termination of a command accompanied by pulse output.
0	1	Generated upon termination of every command.
1	x	Not output in any case.

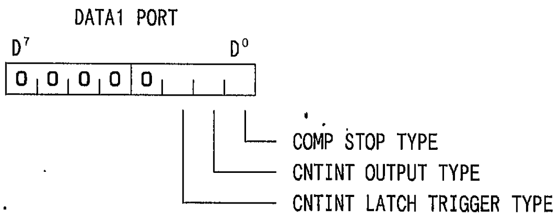
6-5. PULSE COUNTER INITIALIZE Command

COMMAND········ 02H

Function : Defines PULSE COUNTER operation specifications.

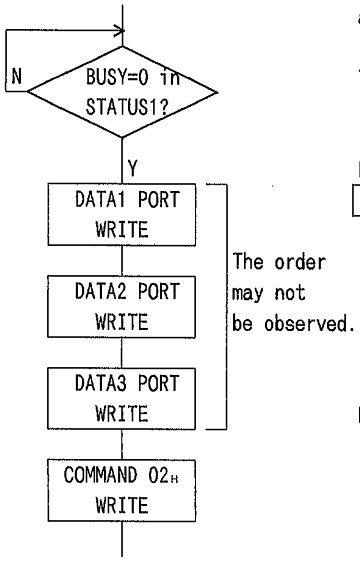
PULSE COUNTER specifications are defined in DRIVE DATA1, DRIVE DATA2 and DRIVE DATA3 PORT.

The contents of DRIVE DATA1 WRITE PORT are as follows.



Note: The bit D^7 to D^3 must to be set 0.

Execution sequence



The details of each bit are shown below. The bit is set to the underlined side at POWER ON/RESET.

(1) COMP STOP TYPE (D^0)
When the "Stop Enable" is selected for the PULSE COUNTER COMP STOP ENABLE, this bit is used for specifying the fast stop or slow stop. (The same specification is selected for the COMPARE REGISTER1 to 5).
0: Fast stop 1: Slow stop

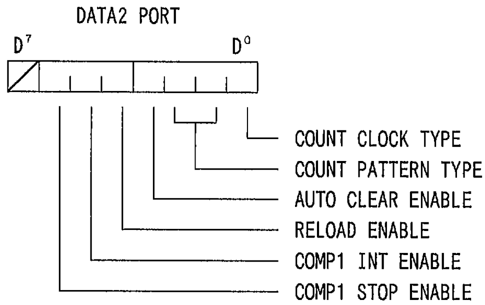
(2) CNTINT OUTPUT TYPE (D^1)
This bit is used for specifying a CNTINT output specification on the PULSE COUNTER.
(The same specification is selected for the COMPARE REGISTER1 to 5).
0: Latches and outputs detection done on each comparator (executing the STATUS3 READ resets the latch).
1: Detection done on each comparator is output without being latched.

Note: Selecting 1 outputs the comparator detection as it is, thus reset with the STATUS3 READ becomes unavailable.

(3) CNTINT LATCH TRIGGER TYPE (D^2)
When "Latch" is selected for the CNTINT output specification of the PULSE COUNTER, this bit is used for selecting a latch type (the same specification is selected for the COMPARE REGISTER1 to 5).
0: Level latch (Even if you may execute the STATUS3 READ while the detection conditions are met, current active state of CNTINT output remains the same).
1: Edge latch (Executing the STATUS3 READ generates CNTINT output reset even if the detection conditions are met).

Note: When the output without latch is selected for CNTINT output, this bit does not have any effect.

The contents of DRIVE DATA2 WRITE PORT are as follows.



The bit marked slash may be set 0 or 1.

The details of each bit are shown below. The bit is set to the underlined side at POWER ON/RESET.

(1) COUNT CLOCK TYPE (D^0)
Selects an operation clock of the PULSE COUNTER.
0: Operated by DRIVE PULSE ($\overline{XCWP}$, $\overline{XCCWP}$) of the X axis MCC05v2.
1: Operated by external clocks from XEA (X axis encoder A phase signal) and XEB (X axis encoder B phase signal).

Note: Don't use external clocks for this pruduct. And if this bit is turned on `1', the PULSE counter is not operated.

(2) COUNT PATTERN TYPE (D^1, D^2)
These bits become effective only when D^0 BIT=1, selecting an external input clock count method.
This bit is invalid at this product.

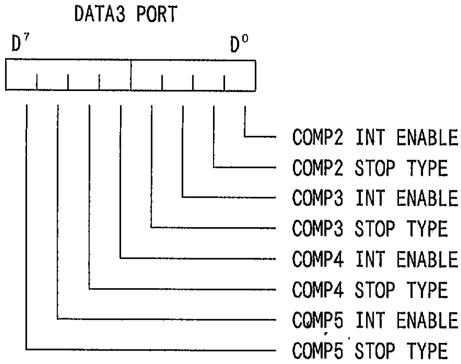
(3) AUTO CLEAR ENABLE (D^3)
Sets the AUTO CLEAR function.
0: No AUTO CLEAR is executed. 1: AUTO CLEAR is executed.

(4) RELOAD ENABLE (D^4)
Sets the RELOAD function.
0: No RELOAD is executed. 1: RELOAD is executed.

(5) COMP1 INT ENABLE (D^5)
Determines whether or not to output the match output XCNTINT of COMPARE REGISTER1.
0: XCNTINT is not output. 1: XCNTINT is output.

(6) COMP1 STOP TYPE (D^6)
Determines whether or not to put the pulse output into stop by the match output of COMPARE REGISTER1.
0: Not put into stop. 1: Put into stop.

The contents of DRIVE DATA3 WRITE PORT are as follows.



The details of each bit are shown below. The bit is set to the underlined side at POWER ON/RESET.

(1) COMP2 INT ENABLE (D^0)
Determines whether or not to output the match output CNTINT of COMPARE REGISTER2.
0: CNTINT is not output. 1: CNTINT is output.

(2) COMP2 STOP TYPE (D^1)
Determines whether or not to put the pulse output into stop by the match output of COMPARE REGISTER2.
0: Not put into stop. 1: Put into stop.

(3) COMP3 INT ENABLE (D^2)
Determines whether or not to output the match output XCNTINT of COMPARE REGISTER3.
0: XCNTINT is not output. 1: XCNTINT is output.

(4) COMP3 STOP TYPE (D^3)
Determines whether or not to put the pulse output into stop by the match output of COMPARE REGISTER3.
0: Not put into stop. 1: Put into stop.

(5) COMP4 INT ENABLE (D^4)
Determines whether or not to output the match output XCNTINT of COMPARE REGISTER4.
0: XCNTINT is not output. 1: XCNTINT is output.

(6) COMP4 STOP TYPE (D^5)
Determines whether or not to put the pulse output into stop by the match output of COMPARE REGISTER4.
0: Not put into stop. 1: Put into stop.

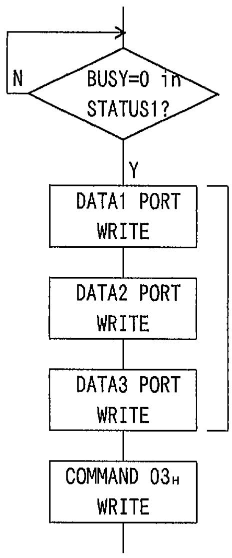
(7) COMP5 INT ENABLE (D^6)
Determines whether or not to output the match output XCNTINT of COMPARE REGISTER5.
0: XCNTINT is not output. 1: XCNTINT is output.

(8) COMP5 STOP TYPE (D^7)
Determines whether or not to put the pulse output into stop by the match output of COMPARE REGISTER5.
0: Not put into stop. 1: Put into stop.

6-6. ADDRESS INITIALIZE Command

COMMAND········ 03H

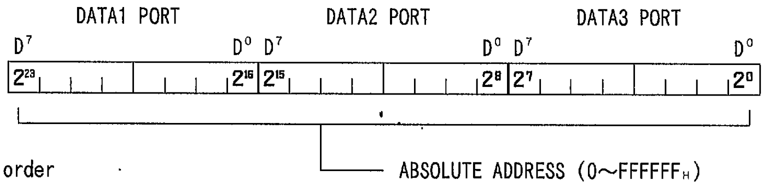
Execution sequence



The order may not be observed.

Function : Defines the current position as the specified absolute address and stores it into memory.
ADDRESS is specified in DRIVE DATA1 PORT, DRIVE DATA2 PORT and DRIVE DATA3 PORT.

The contents of DRIVE DATA1 to DRIVE DATA3 are as follows.



When ADDRESS is negative, it is represented in 2's complement form.

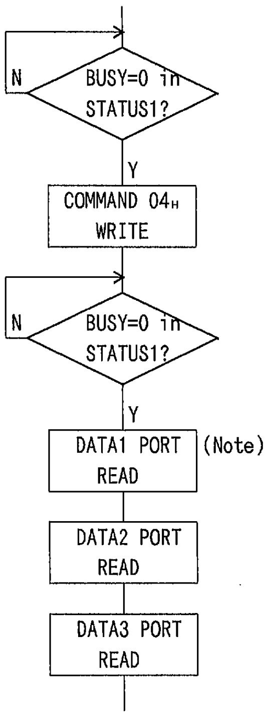
· Setting example of ADDRESS

ADDRESS(decimal)	DATA1 PORT	DATA2 PORT	DATA3 PORT
+8,388,607	7FH	FFH	FFH
+10	00H	00H	0AH
±0	00H	00H	00H
-10	FFH	FFH	F6H
-8,388,607	80H	00H	01H

6-7. ADDRESS READ Command

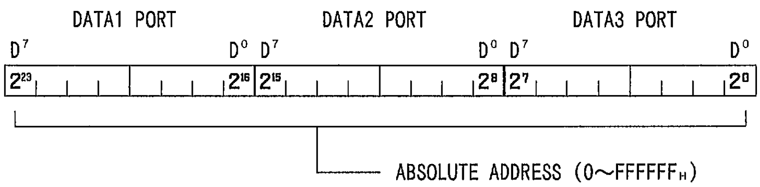
COMMAND········ 04H

Execution sequence



Function : Reads the current position of MOTOR as ABSOLUTE ADDRESS.

ABSOLUTE ADDRESS is read out from DRIVE DATA1 PORT, DRIVE DATA2 PORT and DRIVE DATA3 PORT.
The contents of DRIVE DATA1 to DRIVE DATA3 are as follows.



When ADDRESS is negative, it is represented in 2's complement form.

· Output example of ADDRESS

ADDRESS(decimal)	DATA1 PORT	DATA2 PORT	DATA3 PORT
+8,388,607	7FH	FFH	FFH
+10	00H	00H	0AH
±0	00H	00H	00H
-10	FFH	FFH	F6H
-8,388,607	80H	00H	01H

This command is provided for command compatibility with conventional products. The address data to be read are same as the count data (see 6-39.) of the ADDRESS COUNTER. Apply the COUNT DATA for general use.

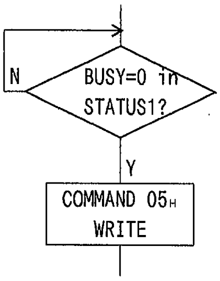
Note: Data must be read in the order of DRIVE DATA1 to DRIVE DATA3 PORT. Usually, DRIVE DATA 1 PORT, DRIVE DATA2 PORT and DRIVE DATA3 PORT are ports exclusively used for read out the counter value of the PULSE COUNTER. The port functions of these ports are switched by writing the ADDRESS READ command and then used for reading address data. The address data reading port functions are recovered to the original port functions by reading DRIVE DATA3 PORT.
Accordingly, when the ADDRESS READ command has been written, be sure to perform a READ operation for DRIVE DATA3 PORT.

6-8. SERVO RESET Command

COMMAND········ 05_H

Function : Outputs the $\overline{DRST}$ signal to the servo driver for 10ms.

Execution sequence



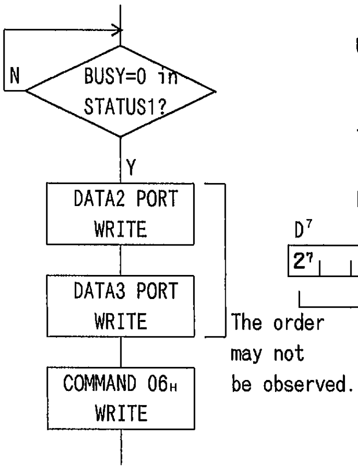
When STEPPING MOTOR is selected, this command becomes equal to the NO OPERATION command.

6-9. RATE SET Command

COMMAND········ 06_H

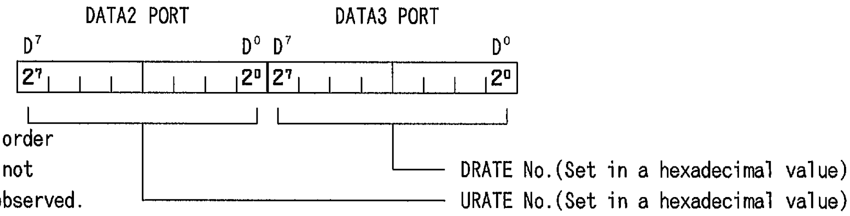
Function : Sets URATE (acceleration time constant) and DRATE (deceleration time constant) required for accelerating/decelerating drive.

Execution sequence



URATE and DRATE are set in DRIVE DATA2 PORT and DRIVE DATA3 PORT respectively in No. of the data table.

The contents of DRIVE DATA2 PORT and DRIVE DATA3 PORT are as follows.



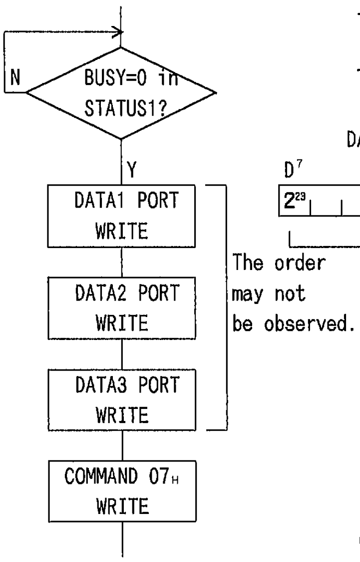
Once the RATE SET command has been executed, it does not need to be set again unless the contents must be changed.
Each No. is 9(100ms/1000Hz) at POWER ON/RESET.

6-10. LSPD SET Command

COMMAND········ 07_H

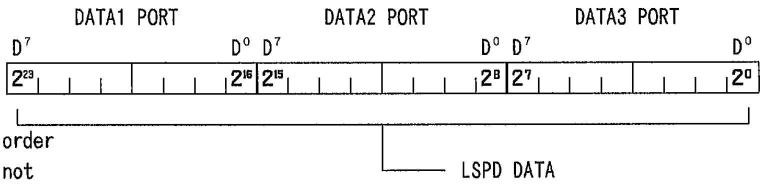
Function : Sets LSPD (LOW SPEED) required for drive.

Execution sequence



LSPD is set in DRIVE DATA1 PORT, DRIVE DATA2 PORT and DRIVE DATA3 PORT in 3-byte data in Hz.

The contents of DRIVE DATA1 PORT to DRIVE DATA3 PORT are as follows.



The LSPD DATA setting range is 10 ($0A_H$) to 3,333,333 ($32DCD5_H$). (Note)

Once the LSPD SET command has been executed, it does not need to be set again unless the contents must be changed.
LSPD is 300Hz at POWER ON/RESET.

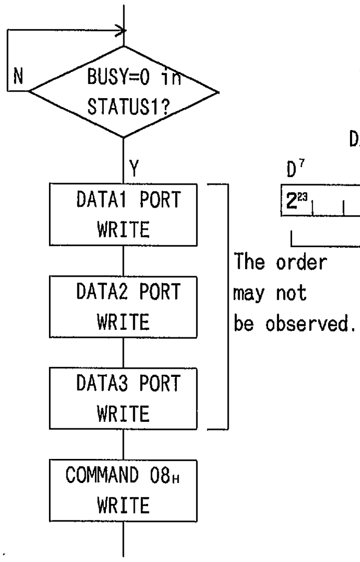
Note: The upper limit of the DATA setting range is the SPEED range shown in 5-17. unlike DRIVE TYPE.

6-11. HSPD SET Command

COMMAND……… 08H

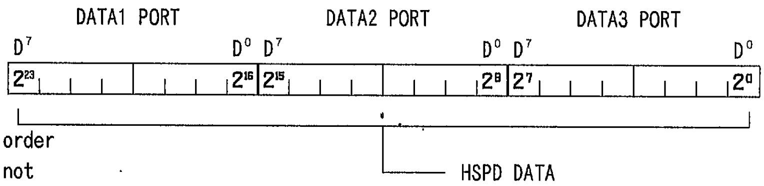
Function : Sets HSPD (HIGH SPEED) required for drive.

Execution sequence



HSPD is set in DRIVE DATA1 PORT, DRIVE DATA2 PORT and DRIVE DATA3 PORT in 3-byte data in Hz.

The contents of DRIVE DATA1 PORT to DRIVE DATA3 PORT are as follows.



The HSPD DATA setting range is 1 (1_H) to 3,333,333 ($32DCD5_H$). (Note)

Once the HSPD SET command has been executed, it does not need to be set again unless the contents must be changed.
HSPD is 3000Hz at POWER ON/RESET.

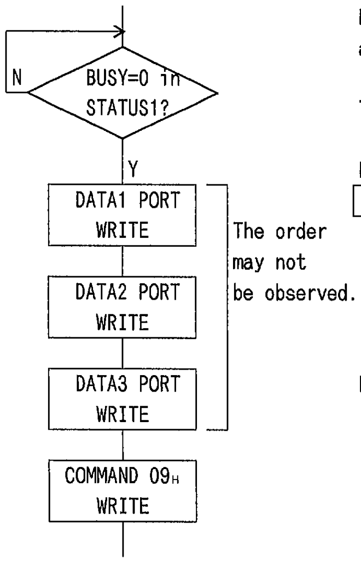
Note: The upper limit of the DATA setting range is the SPEED range shown in 5-17. unlike DRIVE TYPE.

6-12. DFL COUNTER INITIALIZE Command

COMMAND········ 09_H

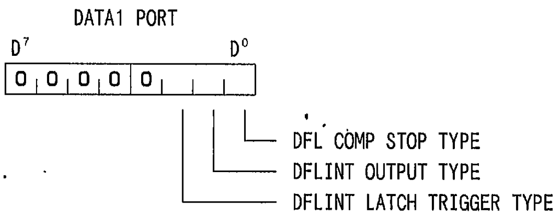
Function : Defines operation specification for the DIFFERENTIAL COUNTER.

Execution sequence



DIFFERENTIAL COUNTER specifications are defined in DRIVE DATA1, DRIVE DATA2 and DRIVE DATA3 PORT.

The contents of DRIVE DATA1 WRITE PORT are as follows.



Note: Applied functions are assigned to D^7 to D^3 bits.
They must be set to 0.

The details of each bit are shown below. The bit is set to the underlined side at POWER ON/RESET.

(1) DFL COMP STOP TYPE (D^0)
When "Stop Enable" is selected for the DIFFERENTIAL COUNTER COMP STOP ENABLE, this bit is used for selecting the fast stop or slow stop through deceleration.
(the same specification applies to the COMPARATORs1 and 2).
<u>0: Fast stop</u>   1: Slow stop

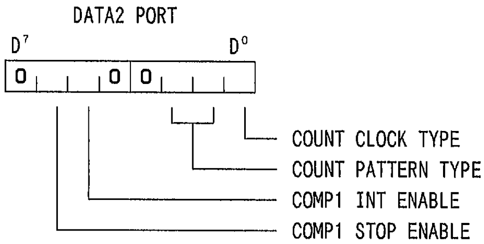
(2) DFLINT OUTPUT TYPE (D^1)
This bit is used for selecting the DFLINT output type on the DIFFERENTIAL COUNTER.
(the same specification applies to the COMPARATORs1 and 2).
<u>0: Detection of each COMPARATOR is latched, then output</u> (executing the STATUS3 READ resets the latch).
1: Detection of each COMPARATOR is output as it is without latch.

Note: Selecting 1 outputs detection of each COMPARATOR without latch, so reset of this mode with the STATUS3 READ is not available.

(3) DFLINT LATCH TRIGGER TYPE (D^2)
When "Latch" is selected for the DFLINT output specification, this bit is used for the latch type.
(the same specification applies to the COMPARATORs1 and 2).
<u>0: Level latch</u> (When the detection condition is valid, executing the STATUS3 READ does not change active state of the DFLINT output).
1: Edge latch (When the detection condition is valid, executing the STATUS3 READ resets the DFLINT output).

Note: When through-output is selected for the DFLINT, latch type selection with this bit is disabled.

Contents of the DRIVE DATA2 PORT are as shown to the right.



Note: 2^3 and 2^4 bits must be set to 0.
Since an applied function assigned to 2^7 bit, it must be set to 0, too.

The details of each bit are shown below. The bit is set to the underlined side at POWER ON/RESET.

(1) COUNT CLOCK TYPE (D^0)

This bit is used for selecting the operation clock for the DIFFERENTIAL COUNTER.

0: The DIFFERENTIAL COUNTER is operated with the drive pulse ($\overline{XCWP}$ and $\overline{XCCWP}$) from the X axis MCC05v2 and external clock from the XEA and XEB.

1: The DIFFERENTIAL COUNTER is operated only with external clock from the XEA (X axis encoder A-phase signal) and XEB (X axis B-phase signal).

Note: Don't use external clocks for this pruduct. And if this bit is turned on `1', the DFL counter is not operated. When this bit is turn on `0', the DFL conter is operated only the drive pulse from the MCC05v2. the DFL counter is operated with counting down at the pulse(+(CW)) from the X axis MCC05v2, and counting up at the pulse(+(CCW)) from the X axis MCC05v2. So this operation is the opposite as compared with the ADDRESS and PULSE counter.

(2) COUNT PATTERN TYPE (D^1,D^2)

This bit is used for selecting the external operation clock for the DIFFERENTIAL COUNTER.
This bit is invalid at this product.

(3) COMP1 INT ENABLE (D^5)

Determines whether or not to output the match output DFLINT of COMPARE REGISTER1.

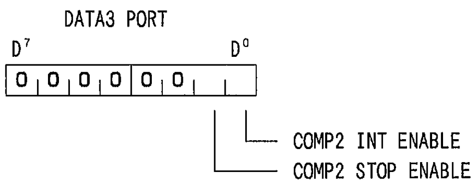
0: DFLINT is not output. 1: DFLINT is output.

(4) COMP1 STOP TYPE (D^6)

Determines whether or not to put the pulse output into stop by the match output of COMPARE REGISTER1.

0: Not put into stop. 1: Put into stop.

Contents of the DRIVE DATA3 PORT are as shown to the right.



Note: 2^2 to 2^7 bits must be set to 0.

(3) COMP2 INT ENABLE (D^0)

Determines whether or not to output the match output DFLINT of COMPARE REGISTER2.

0: DFLINT is not output. 1: DFLINT is output.

(4) COMP3 STOP TYPE (D^1)

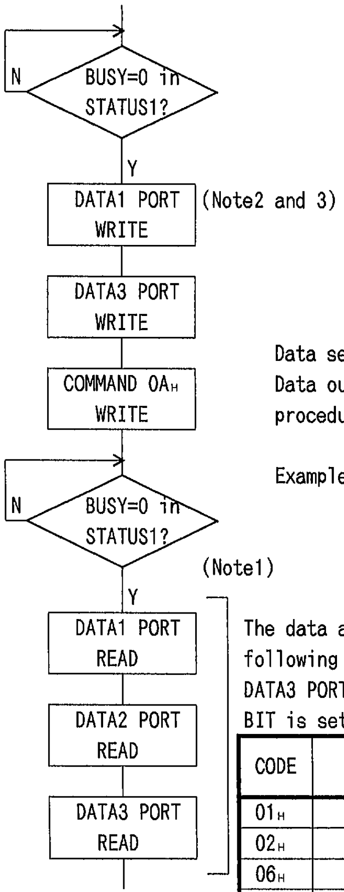
Determines whether or not to put the pulse output into stop by the match output of COMPARE REGISTER2.

0: Not put into stop. 1: Put into stop.

6-13. SET DATA READ Command

COMMAND········ 0A$_H$

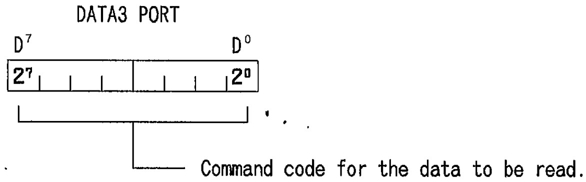
Execution sequence



Function : Reads SPEC DATA, speed data and other data being set on the MCC05v2 of each axis.

On the DRIVE DATA3 PORT (WRITE), specify a command used for reading the target set data. Part of the DRIVE DATA1 PORT(WRITE) is also used. (see Note2 and 3).

The contents of DRIVE DATA3 WRITE PORT are as follows.



Data set on the DRIVE DATA1, 2 and 3 PORTs are read.
Data output procedure using the applicable commands is the same as the data setup procedure which is also implemented using the applicable commands.

Example: When you want to confirm the RATE setup number, write "06$_H$" (RATE SET Command) to the DATA3 PORT and "0A$_H$" (this command) to the COMMAND PORT to read the DATA2 and 3 PORTs. This will output the URATE No. to the DATA2 PORT and the DRATE No. to the DATA3 PORT.

The data available for confirmation are limited to those that have been set with the following commands. Writing a command code other than those listed to the DRIVE DATA3 PORT may not generate the desired data. In this case, the STATUS1 PORT ERROR BIT is set to 1.

CODE	COMMAND NAME	CODE	COMMAND NAME
01$_H$	SPEC INITIALIZE1	29$_H$	PART RATE SET (Note3) *
02$_H$	PULSE COUNTER INITIALIZE	2B$_H$	MARGIN TIME SET *
06$_H$	RATE SET (Note2)	2C$_H$	PEAK PULSE SET *
07$_H$	LSPD SET	2D$_H$	SEND PULSE SET *
08$_H$	HSPD SET	2E$_H$	SESPD SET *
09$_H$	DFL COUNTER INITIALIZE	2F$_H$	SPEC INITIALIZE4 *
0B$_H$	CW SOFT LIMIT SET *	50$_H$	DEND TIME SET *
0C$_H$	CCW SOFT LIMIT SET *	51$_H$	EXTEND ORIGIN SPEC SET *
0E$_H$	DFL DIVISION DATA SET *	52$_H$	CONSTANT SCAN MAX PULSE*
18$_H$	END PULSE SET *	53$_H$	CHANGE POINT DATA SET *
19$_H$	ESPD SET *	54$_H$	CHANGE DATA SET *
1A$_H$	CSPD SET	55$_H$	AUTO CHANGE SET *
1B$_H$	OFFSET PULSE SET	5F$_H$	SPEC INITIALIZE5 *
1C$_H$	ORIGIN DELAY SET	60$_H$	SRATE SET
20$_H$	SPEC INITIALIZE3 *	61$_H$	SLSPD SET
22$_H$	RESOLUTION SET *	62$_H$	SHSPD SET
24$_H$	PART HSPD SET (Note3) *	63$_H$	SSRATE ADJUST
25$_H$	INCREMENTAL DATA SET *	64$_H$	SERATE ADJUST
26$_H$	ABSOLUTE DATA SET *	65$_H$	SCSPD1 ADJUST
27$_H$	PART PULSE SET (Note3) *	66$_H$	SCSPD2 ADJUST

Commands attached with an asterisks (*) are intended for the applied functions.
For details, see the User's Manual [Applied Functions Part].

Note1: Although number of data ports read and the DATA PORT No. vary according to the command used for the reading, read of the DRIVE DATA3 PORT must not be ignored whenever this command has been executed.

Note2: When reading in the ARITHMETIC MODE, an URATE/DRATE must be set to the DRIVE DATA1 PORT(WRITE).

Note3: As for these commands, PART No. must be set to the DRIVE DATA1 PORT(WRITE).

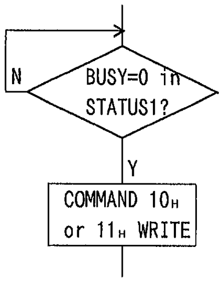
Note4: All data are **output in the same state as they are written** without conducting internal processing such as MIN/MAX process. And, once a piece of data has been written, you cannot change its DRIVE TYPE between the FIXED and ARITHMETIC at its output.

Note5: Initial setting specified at the time of POWER ON/RESET is not read.

6-14. +/-JOG Command

COMMAND········ Drive in +(CW)direction 10_H Drive in -(CCW)direction 11_H

Execution sequence

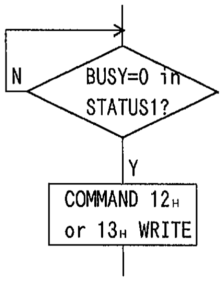


Function : Performs JOG DRIVE.

6-15. +/-SCAN Command

COMMAND········ Drive in +(CW)direction 12_H Drive in -(CCW)direction 13_H

Execution sequence

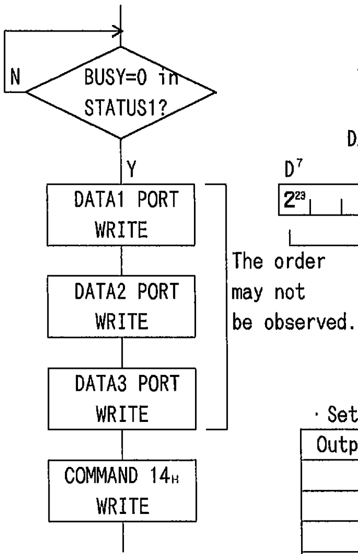


Function : Performs SCAN DRIVE.

6-16. INCREMENTAL INDEX Command

COMMAND········ 14_H

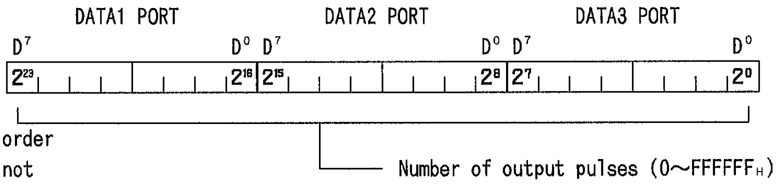
Execution sequence



Function : Performs INDEX DRIVE with relative specified.

The number of output pulses and the direction are specified in DRIVE DATA1 PORT, DRIVE DATA2 PORT and DRIVE DATA3 PORT.

The contents of DRIVE DATA1 PORT to DRIVE DATA3 PORT are as follows.



When set to -(CCW)direction, the number of output pulses is represented in 2's complement form.

· Setting example of the number of output pulses

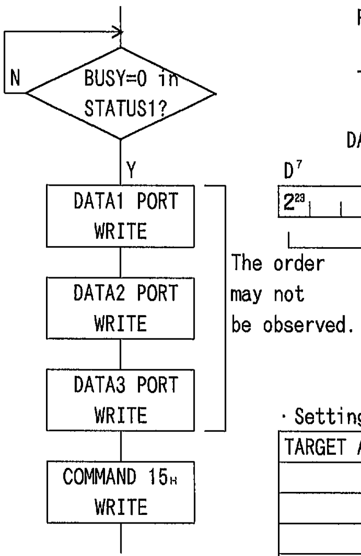
Output pulse (decimal)	DATA1 PORT	DATA2 PORT	DATA3 PORT
+8,388,607	$7F_H$	FF_H	FF_H
+10	00_H	00_H	$0A_H$
±0	00_H	00_H	00_H
-10	FF_H	FF_H	$F6_H$
-8,388,607	80_H	00_H	01_H

6-17. ABSOLUTE INDEX Command

COMMAND········ 15H

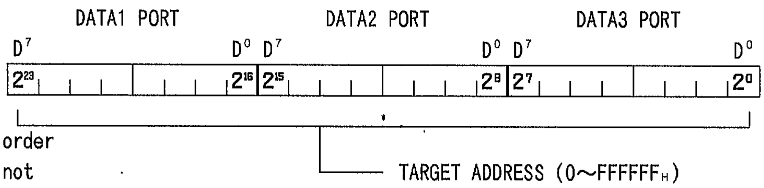
Function : Performs INDEX DRIVE with absolute specified.

Execution sequence



The absolute address of the target location is specified in DRIVE DATA1 PORT, DRIVE DATA2 PORT and DRIVE DATA3 PORT.

The contents of DRIVE DATA1 PORT to DRIVE DATA3 PORT are as follows.



When TARGET ADDRESS is negative, it is represented in 2's complement form.

· Setting example of TARGET ADDRESS

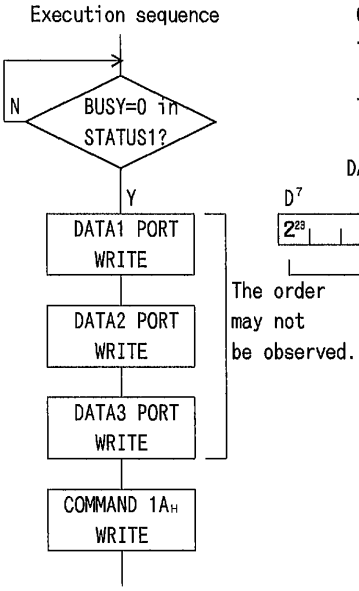
TARGET ADDRESS(decimal)	DATA1 PORT	DATA2 PORT	DATA3 PORT
+8,388,607	7FH	FFH	FFH
+10	00H	00H	0AH
±0	00H	00H	00H
-10	FFH	FFH	F6H
-8,388,607	80H	00H	01H

6-18. CSPD SET Command

COMMAND········ 1AH

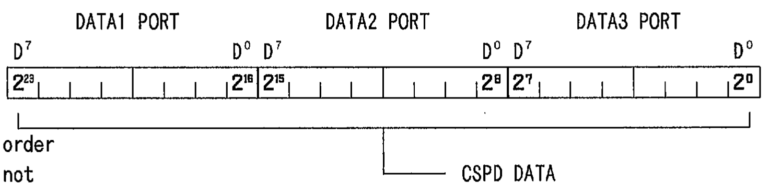
Function : Sets CSPD (constant speed) required for machine origin detection drive.

Execution sequence



CSPD is set in DRIVE DATA1 PORT, DRIVE DATA2 PORT and DRIVE DATA3 PORT in 3-byte data in Hz.

The contents of DRIVE DATA1 PORT to DRIVE DATA3 PORT are as follows.



The CSPD DATA setting range is 1 (1H) to 3,333,333 (32DCD5H). (Note)

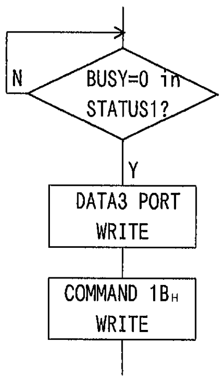
Once the CSPD SET command has been executed, it does not need to be set again unless the contents must be changed.
CSPD is 300Hz at POWER ON/RESET.

Note: The upper limit of the DATA setting range is the SPEED range shown in 5-17. unlike DRIVE TYPE.

6-19. OFFSET PULSE SET Command

COMMAND……… 1BH

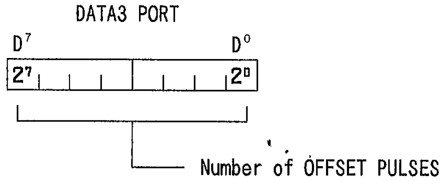
Execution sequence



Function : Sets the number of OFFSET PULSES required for machine origin detection drive.

The number of OFFSET PULSES is set in DRIVE DATA3 PORT.

The contents of DRIVE DATA3 PORT are as follows.

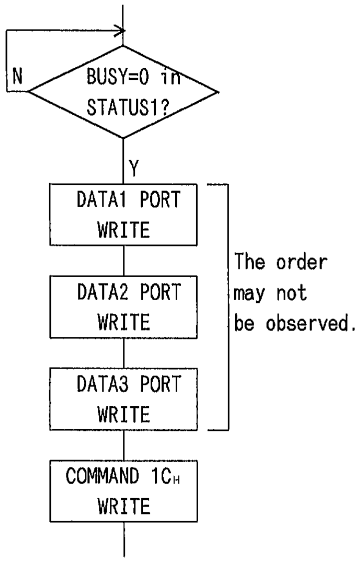


The OFFSET PULSE number setting range is 0(0H) to 255 (FFH).
The number of OFFSET PULSES is set to 0 at POWER ON/RESET.
Once the OFFSET PULSE SET command has been executed, it does not need to be set again unless the contents must be changed.

6-20. ORIGIN DELAY SET Command

COMMAND……… 1CH

Execution sequence



Function : Sets DELAY TIME at machine origin detection drive.

The following data is set in each of DRIVE DATA1 PORT, DRIVE DATA2 PORT and DRIVE DATA3 PORT.

DATA1 PORT ……… LIMIT DELAY TIME (300ms(3CH))
Delay time up to a start of reverse after a stop in the CCW LIMIT.

DATA2 PORT ……… SCAN DELAY TIME (50ms(0AH))
Delay time when the direction is reversed in the CONSTANT SCAN and SCAN DRIVE processes.

DATA3 PORT ……… JOG DELAY TIME (20ms(04H))
Delay time per pulse in the JOG DRIVE process.

In each port, the value in parentheses is set at POWER ON/RESET.

Each data is 00H to FFH and set in units of 5ms.

Example) 00H ……… No delay time
0AH ……… 50ms
FFH ……… 1.275s

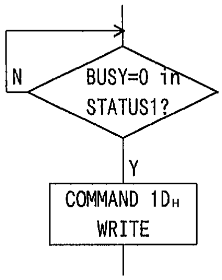
Usually it is not need to change this data. Only something improvement at the machine origin detection or the tact time, etc.

The ORIGIN DELAY SET command does not need to be set again unless the contents must be changed.

6-21. ORIGIN FLAG RESET Command

COMMAND……… 1DH

Execution sequence



Function : Resets the detection flag to be used for machine origin detection drive.

This command is used only when you don't desire to perform ABSOLUTE INDEX DRIVE nearly up to the machine origin when machine origin detection drive is used.

For details, refer to Chapter 7.

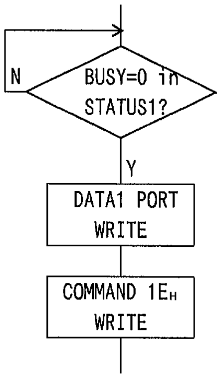
Note: Execute this command before executing the ORIGIN command.

6-22. ORIGIN Command

COMMAND········ $1E_H$

Function : Performs DRIVE up to machine origin detection.

Execution sequence



The ORG type to be executed is specified in DRIVE DATA1 PORT.

ORG-0 ············ 00_H
ORG-1 ············ 01_H
ORG-2 ············ 02_H
ORG-3 ············ 03_H
ORG-4 ············ 04_H
ORG-5 ············ 05_H
ORG-10 ········ $0A_H$
ORG-11 ········ $0B_H$
ORG-12 ········ $0C_H$

If any data other than the above is set, a command error occurs and no operation is performed.

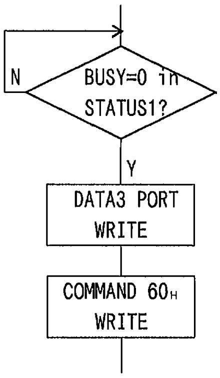
When each bit of LSEND, SSEND and FEND in STATUS1 is 0 and DREND bit is 1 upon termination of DRIVE, the machine origin has been detected normally.(04_H)
When any of the ERROR, LSEND, SSEND and FSEND is set to 1, the machine origin has not been detected. If a detect is interrupted by the $\overline{\text{RESET}}$ entered during a drive, all bits in the STATUS1 are set to 0 (00_H).

6-23. SRATE SET Command

COMMAND········ 60_H

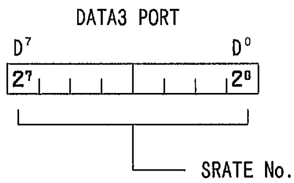
Function : Sets SRATE (acceleration/deceleration time constant) required for S-RATE DRIVE.

Execution sequence



SRATE is set in DRIVE DATA3 PORT in No. of data table.

The contents of DRIVE DATA3 PORT are as follows.



Once the SRATE SET command has been executed, it does not need to be set again unless the contents must be changed.
SRATE No. is set at No.9(100ms/1000Hz) at POWER ON/RESET.

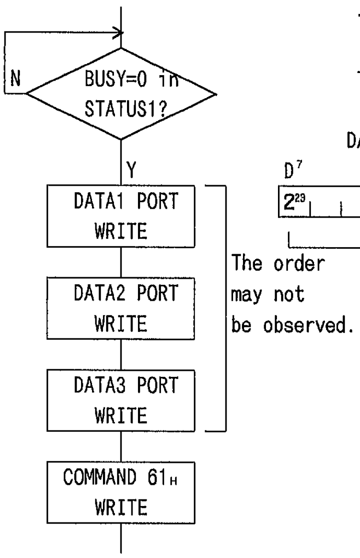
Note: After this command is executed, SSRATE and SERATE are reset to the initial values.
Be careful about this point after SSRATE and SERATE are compensated.

6-24. SLSPD SET Command

COMMAND·········61H

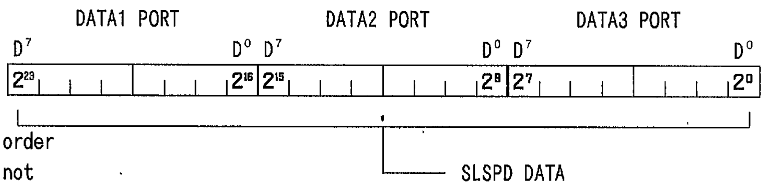
Function : Sets SLSPD (low speed) required for S-RATE DRIVE.

Execution sequence



SLSPD is set in DRIVE DATA1 PORT, DRIVE DATA2 PORT and DRIVE DATA3 PORT in 3-byte data in Hz.

The contents of DRIVE DATA1 PORT to DRIVE DATA3 PORT are as follows.



The SLSPD DATA setting range is 10(0AH) to 3,333,333 (32DCD5H). (Note1)
Once the SLSPD SET command has been executed, it does not need to be set again unless the contents must be changed.
SLSPD is 300Hz at POWER ON/RESET.

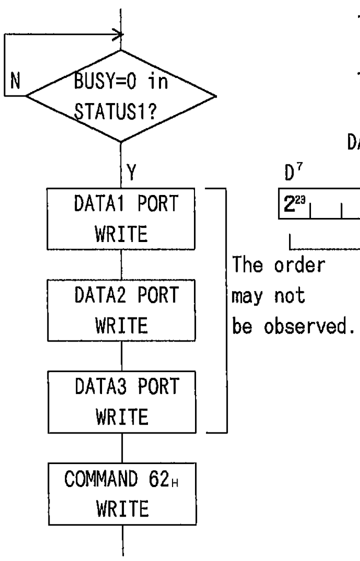
Note1: The upper limit of the data setting range depends on DRIVE TYPE. Refer to 5-17.
Note2: After this command is executed, SCSPD1 and SCSPD2 are reset to the initial values. Be careful about this point after SCSPD1 and SCSPD2 are compensated.

6-25. SHSPD SET Command

COMMAND·········62H

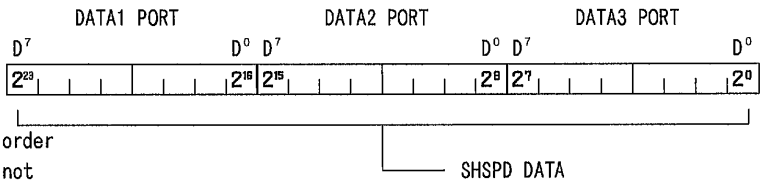
Function : Sets SHSPD (high speed) required for S-RATE DRIVE.

Execution sequence



SHSPD is set in DRIVE DATA1 PORT, DRIVE DATA2 PORT and DRIVE DATA3 PORT in 3-byte data in Hz.

The contents of DRIVE DATA1 PORT to DRIVE DATA3 PORT are as follows.



The SHSPD DATA setting range is 1(1H) to 3,333,333(32DCD5H). (Note1)

Once the SHSPD SET command has been executed, it does not need to be set again unless the contents must be changed.
SHSPD is 3000Hz at POWER ON/RESET.

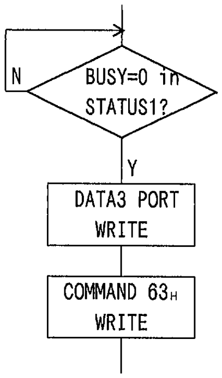
Note1: The upper limit of the data setting range depends on DRIVE TYPE. Refer to 5-17.
Note2: After this command is executed, SCSPD1 and SCSPD2 are reset to the initial values. Be careful about this point after SCSPD1 and SCSPD2 are compensated.

6-26. SSRATE ADJUST Command

COMMAND········ 63H

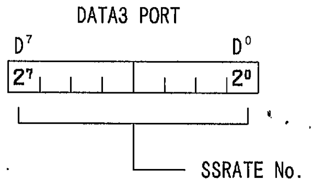
Function : Adjusts SSRATE (time constant at a start of acceleration or at an end of deceleration) required for S-RATE DRIVE.

Execution sequence



SSRATE is set in DRIVE DATA3 PORT in No. of data table.

The contents of DRIVE DATA3 PORT are as follows.



SSRATE is automatically set to an approximately eightfold value of SRATE by the SRATE SET command.
When the specification is satisfied with this data, this command does not need to be executed.
For the details of automatic setting value, refer to 5-9.

Note1: After the SRATE SET command is executed, SSRATE adjusted prior to the execution becomes ineffective and is reset to the initial value. This is also applicable to the case in which DRIVE TYPE has been changed by the SPEC INITIALIZE1 command.

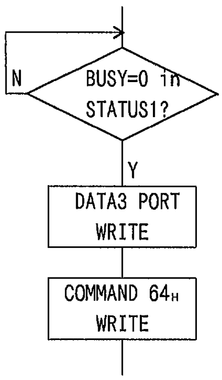
Note2: The SSRATE adjusting range is SSRATE≧SRATE. When SSRATE<SRATE, SSRATE becomes equal to SRATE.

6-27. SERATE ADJUST Command

COMMAND········ 64H

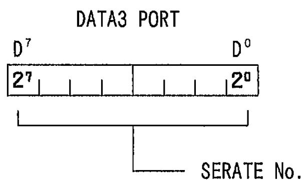
Function : Adjusts SERATE (at an end of acceleration or at a start of deceleration) required for S-RATE DRIVE.

Execution sequence



SERATE is set in DRIVE DATA3 PORT in No. of data table.

The contents of DRIVE DATA3 PORT are as follows.



SERATE is automatically set to an approximately eightfold value of SRATE by the SRATE SET command.
When the specification is satisfied with this data, this command does not need to be executed.
For the details of automatic setting value, refer to 5-9.

Note1: After the SRATE SET command is executed, SERATE adjusted prior to the execution becomes ineffective and is reset to the initial value. This is also applicable to the case in which DRIVE TYPE has been changed by the SPEC INITIALIZE1 command.

Note2: The SERATE adjusting range is SERATE≧SRATE. When SERATE<SRATE, SERATE becomes equal to SRATE.

6-28. SCSPD1 ADJUST Command

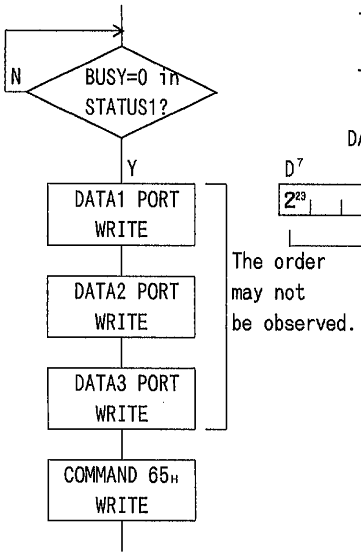
COMMAND········· 65H

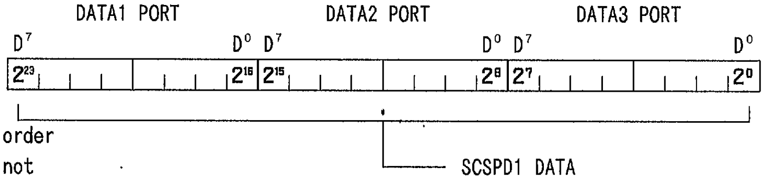
Function : Adjusts SCSPD1 (linear RATE start speed or linear RATE end speed) required for S-RATE DRIVE.

SCSPD1 is set in DRIVE DATA1 PORT, DRIVE DATA2 PORT and DRIVE DATA3 PORT in 3-byte data in Hz.

The contents of DRIVE DATA1 PORT to DRIVE DATA3 PORT are as follows.

Execution sequence





SCSPD1 is reset to the value indicated by the following expression at POWER ON/RESET or by the SLSPD SET command or SHSPD SET command. When the specification is satisfied with the following data, this command does not need to be executed.

$$SCSPD1 = SLSPD + (SHSPD - SLSPD) \times \frac{1}{3}$$

Note1: After the SLSPD SET command or the SHSPD SET command is executed, SCSPD1 set prior to the execution becomes ineffective and is reset to the initial value. This is also applicable to the case in which DRIVE TYPE has been changed by the SPEC INITIALIZE1 command.

Note2: The SCSPD1 adjusting range is SLSPD≦SCSPD1≦SCSPD2.
When SCSPD1<SLSPD is specified, SCSPD1 becomes equal to SLSPD.
When SCSPD1>SCSPD2 is specified, SCSPD1 becomes equal to SCSPD2.

6-29. SCSPD2 ADJUST Command

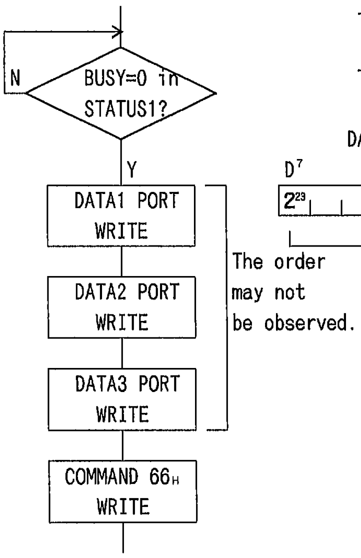
COMMAND········· 66H

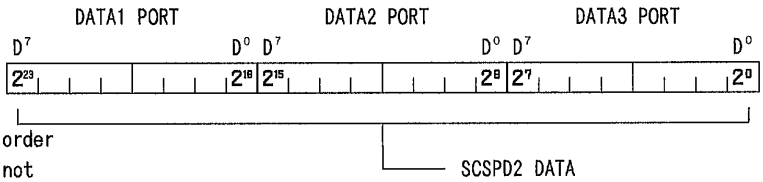
Function : Adjusts SCSPD2 (linear RATE end speed or linear RATE start speed) required for S-RATE DRIVE.

SCSPD2 is set in DRIVE DATA1 PORT, DRIVE DATA2 PORT and DRIVE DATA3 PORT in 3-byte data in Hz.

The contents of DRIVE DATA1 PORT to DRIVE DATA3 PORT are as follows.

Execution sequence





SCSPD2 is reset to the value indicated by the following expression at POWER ON/RESET or by the SLSPD SET command or SHSPD SET command. When the specification is satisfied with the following data, this command does not need to be executed.

$$SCSPD2 = SLSPD + (SHSPD - SLSPD) \times \frac{2}{3}$$

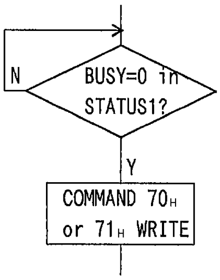
Note1: After the SLSPD SET command or the SHSPD SET command is executed, SCSPD2 set prior to the execution becomes ineffective and is reset to the initial value. This is also applicable to the case in which DRIVE TYPE has been changed by the SPEC INITIALIZE1 command.

Note2: The SCSPD2 adjusting range is SCSPD1≦SCSPD2≦SHSPD.
When SCSPD2<SCSPD1 is specified, SCSPD2 becomes equal to SCSPD1.
When SCSPD2>SHSPD is specified, SCSPD2 becomes equal to SHSPD.

6-30. +/-S-RATE SCAN Command

COMMAND········ Drive in +(CW)direction [70H]   Drive in -(CCW)direction [71H]

Execution sequence



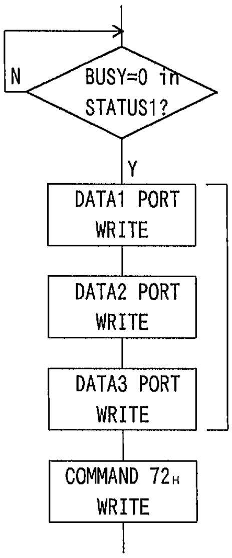
Function : Performs S-RATE SCAN DRIVE.

6-31. S-RATE INCREMENTAL INDEX Command

COMMAND········ [72H]

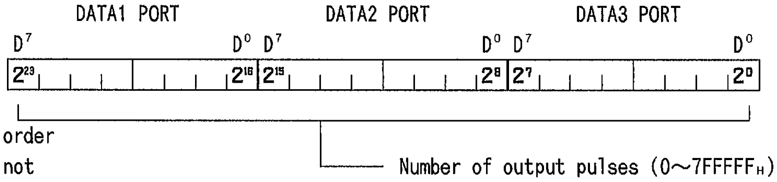
Function : Performs S-RATE INDEX DRIVE with relative specified.

Execution sequence



The order may not be observed.

The number of output pulses and the direction are specified in DRIVE DATA1 PORT, DRIVE DATA2 PORT and DRIVE DATA3 PORT.
The contents of DRIVE DATA1 PORT to DRIVE DATA3 PORT are as follows.



When set to -(CCW)direction, the number of output pulses is represented in 2's complement form.

· Setting example of the number of output pulses

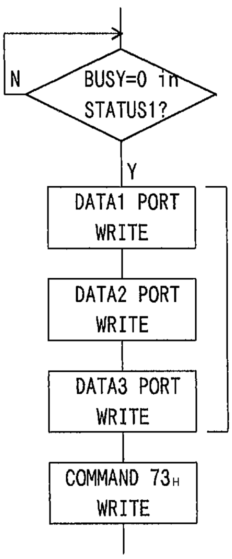
Output pulse (decimal)	DATA1 PORT	DATA2 PORT	DATA3 PORT
+8,388,607	7FH	FFH	FFH
+10	00H	00H	0AH
±0	00H	00H	00H
-10	FFH	FFH	F6H
-8,388,607	80H	00H	01H

6-32. S-RATE ABSOLUTE INDEX Command

COMMAND········ [73H]

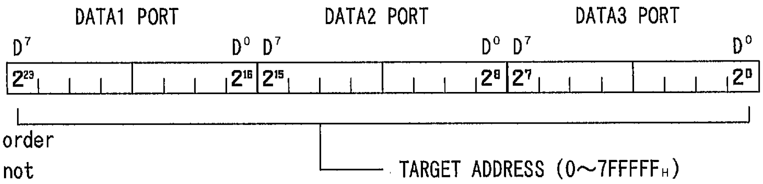
Function : Performs S-RATE INDEX DRIVE with absolute specified.

Execution sequence



The order may not be observed.

The absolute address of the target location is specified in DRIVE DATA1 PORT, DRIVE DATA2 PORT and DRIVE DATA3 PORT.
The contents of DRIVE DATA1 PORT to DRIVE DATA3 PORT are as follows.



When TARGET ADDRESS is negative, it is represented in 2's complement form.

· Setting example of TARGET ADDRESS

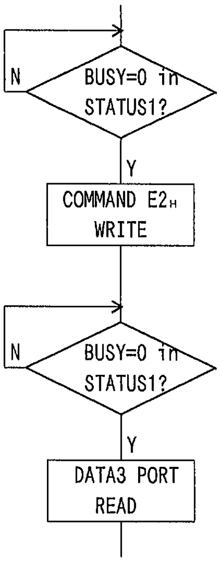
TARGET ADDRESS(decimal)	DATA1 PORT	DATA2 PORT	DATA3 PORT
+8,388,607	7FH	FFH	FFH
+10	00H	00H	0AH
±0	00H	00H	00H
-10	FFH	FFH	F6H
-8,388,607	80H	00H	01H

6-33. ERROR STATUS READ Command

COMMAND········· $E2_H$

Function : When the STATUS1 PORT ERROR BIT is set to 1, this command reads cause of an error.

Execution sequence



Cause of an error is output to the DRIVE DATA3 PORT in hexadecimal value.

00_H ····· Error is not present.
01_H ····· An undefined command has been executed.
02_H ····· The SET DATA READ has been executed by an irrelevant command.
03_H ····· The SPECIAL INDEX has been executed when the URATE≠DRATE specified. *
Or, the SPECIAL SCAN at URATE≠DRATE has been executed when the SOFT LIMIT is enabled.
04_H ····· Section1 condition error in the SERIAL INDEX. *
05_H ····· A SOFT LIMIT error. *
06_H ····· The drive has ended in a DEND ERROR. *
07_H ····· The drive has ended in an ORIGIN ERROR. *
08_H ····· The SENSOR INDEX3 DATA SET has not been executed. *
09_H ····· Data error in writing a command.
·The SET DATA READ has been executed by an undefined command.
·DATA1 PORT data error during the RATE SET in the Arithmetic mode. *
·The RESOLUTION SET has been executed in the fixed data mode. *
·PART No. error in the PART HSPD. *
·PART No. error in the PART PULSE. *
·PART No. error in the PART RATE. *
·PART No. error in the SERIAL INDEX CHECK. *
$0A_H$ ····· The INDEX CHANGE has not been executed because the drive is ended. *
$0D_H$ ····· DEND ERROR and ORIGIN ERROR have occurred. *

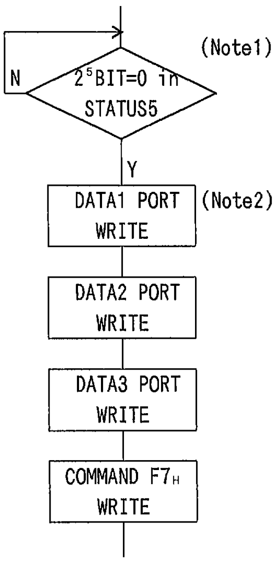
Those errors indicated with asterisks (*) are relevant to applied functions.
For details, see the User's Manual [Applied Functions Part].
As it is so with a STATUS1 PORT ERROR BIT, an error code can be cleared by any command other than this command. When the command is being executed, the error code is cleared after the execution is complete.

6-34. SPEED CHANGE Command

COMMAND········· $F7_H$

Function : Changes speed of the SCAN DRIVE or INDEX DRIVE currently taking place.

Execution sequence



SPEED is set in DRIVE DATA1 PORT, DRIVE DATA2 PORT and DRIVE DATA3 PORT in 3-byte data in Hz.

The contents of DRIVE DATA1 PORT to DRIVE DATA3 PORT are same as HSPD SET command.

Note1: Before writing this command, make sure that the SPEED CHANGE BUSY BIT on the STATUS5 is set to 0.

Note2: Speed data write to the ports must take place in the order of the DATA1, 2 and 3 PORT.
(3-byte data is acquired while write to the DATA3 PORT is taking place.)

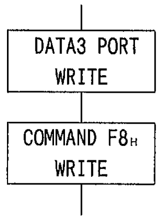
Note3: This command is valid for the SCAN DRIVE and INDEX DRIVE alone.

6-35. INT MASK Command

COMMAND········ F8H

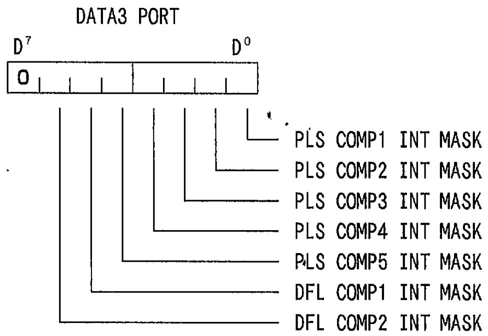
Function : Masks detection of each COMPARATOR at its output.

Execution sequence



INT MASK is set on the DRIVE DATA3 PORT.

Contents of the DRIVE DATA3 PORT is as follows.



Note: D^7 bit must be set to 0.

Detail of the each bit on the DRIVE DATA3 PORT is as follows.
At POWER ON/RESET, the underlined side is selected.

(1) PLS COMP1 INT MASK (D^0)
This bit is used for enabling or disabling masking of output of the detection from the PULSE COUNTER COMPARATOR1.
0: Not masked. 1: Masked.

(2) PLS COMP2 INT MASK (D^1)
This bit is used for enabling or disabling masking of output of the detection from the PULSE COUNTER COMPARATOR2.
0: Not masked. 1: Masked.

(3) PLS COMP3 INT MASK (D^2)
This bit is used for enabling or disabling masking of output of the detection from the PULSE COUNTER COMPARATOR3.
0: Not masked. 1: Masked.

(4) PLS COMP4 INT MASK (D^3)
This bit is used for enabling or disabling masking of output of the detection from the PULSE COUNTER COMPARATOR4.
0: Not masked. 1: Masked.

(5) PLS COMP5 INT MASK (D^4)
This bit is used for enabling or disabling masking of output of the detection from the PULSE COUNTER COMPARATOR5.
0: Not masked. 1: Masked.

(6) DFL COMP1 INT MASK (D^5)
This bit is used for enabling or disabling masking of output of the detection from the DIFFERENTIAL COUNTER COMPARATOR1. (excessive deviation).
0: Not masked. 1: Masked.

(7) DFL COMP2 INT MASK (D^6)
This bit is used for enabling or disabling masking of output of the detection from the DIFFERENTIAL COUNTER COMPARATOR2. (in position)
0: Not masked. 1: Masked.

Note1: Even if "Masking" is specified, the COMPARATOR match stop function is not affected.
For details, refer to 8-5.

Note2: BUSY=0 may not be checked. But, you must not use this command when another command is being written or written after(less than 4 μS) to the DATA3 PORT to rewrite it.

6-36. PORT SELECT Command

(1) ADDRESS COUNTER PORT SELECT Command

COMMAND·········| $F9_H$ | Function : Switches the DRIVE DATA1, 2 and 3 PORTs to the special port for reading count data on the ADDRESS COUNTER.

(2) DFL COUNTER PORT SELECT Command

COMMAND·········| FA_H | Function : Switches the DRIVE DATA1, 2 and 3 PORTs to the special port for reading count data on the DIFFERENTIAL COUNTER.

(3) PULSE COUNTER PORT SELECT Command

COMMAND·········| FC_H | Function : Switches the DRIVE DATA1, 2 and 3 PORTs to the special port for reading count data on the PULSE COUNTER.

(4) SPEED PORT SELECT Command

COMMAND·········| FD_H | Function : Switches the DRIVE DATA1, 2 and 3 PORTs to the special port for reading speed data of output pulse.

These commands are used for switching data to be read from the DRIVE DATA1, 2 and 3 PORTs.
There is no specified command execution sequence. however do not execute this command less than 4 μS after another command is executed.

Desired data will be provided from the DRIVE DATA1, 2 and 3 PORTs within 200ns from execution of a command.

Once executed, each port select command is valid until another port select command is executed.
At POWER ON/RESET, the DRIVE DATA1, 2 and 3 PORTs are set as the special port for reading count data on the PULSE COUNTER.

If one of the following commands is written, the DRIVE DATA1, 2 and 3 PORTs temporarily output data required by the command, then restore once selected the port function. Reading the DRIVE DATA3 PORT restores them the preceding function.
Thus, whenever one of the following commands has been executed, you must read the DRIVE DATA3 PORT.

ADDRESS READ, SET DATA READ, ERROR STATUS READ and SERIAL INDEX CHECK (applied function).

6-37. SLOW STOP Command

COMMAND·········| FE_H | Function : Puts DRIVE into a slow stop.
When DRIVE is of constant speed, this command puts it into a fast stop.

Regarding the execution sequence, there is no special stipulation.
However, since this command stops DRIVE, it is ignored when it is written during BUSY=0.
This command can function only at DRIVE=1 but not at DRIVE=0.

6-38. FAST STOP Command

COMMAND·········| FF_H | Function : Puts DRIVE into a fast stop.

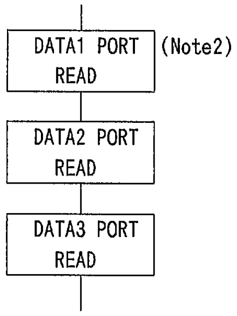
Regarding the execution sequence, there is no special stipulation.
However, since this command stops DRIVE, it is ignored when it is written during BUSY=0.
This command can function only at DRIVE=1 but not at DRIVE=0.

6-39. COUNTER READ

COMMAND········ None

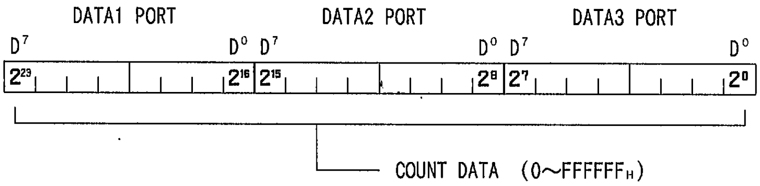
Function : Reads the count data of the PULSE COUNTER or DIFFERENTIAL COUNTER or ADDRESS COUNTER.(Note1)

Execution sequence



This function reads out the count data of the counter data from DRIVE READ PORT after the PORT SELECT command. (Note1)

The contents of DRIVE DATA1 PORT to DRIVE DATA3 PORT are as follows.



When the count data is negative, it is represented in 2's complement form.

· Example of count data

COUNT DATA(decimal)	DATA1 PORT	DATA2 PORT	DATA3 PORT
+8,388,607	$7F_H$	FF_H	FF_H
+10	00_H	00_H	$0A_H$
±0	00_H	00_H	00_H
-10	FF_H	FF_H	$F6_H$
-8,388,607	80_H	00_H	01_H

Note1: The PULSE or DIFFERENTIAL or ADDRESS COUNTER select must be executed before count read by each PORT SELECT Command.(6-36.)

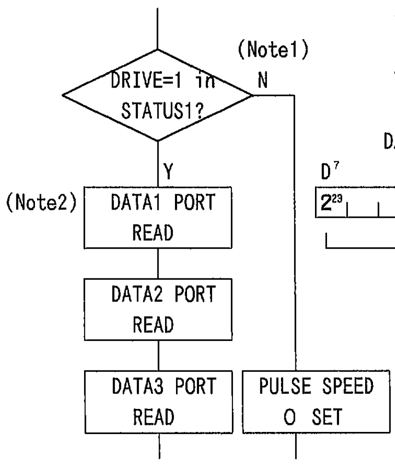
Note2: Be sure to perform a DATA READ operation in the order of DRIVE DATA1 PORT, DRIVE DATA2 PORT and DRIVE DATA3 PORT.
If this sequence is not observed, the data will not be guaranteed.

6-40. SPEED READ

COMMAND········ None

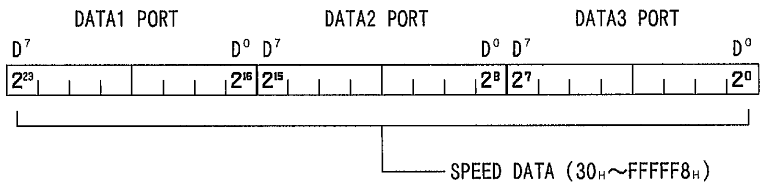
Function : Reads speed data of a drive currently taking place.

Execution sequence



After the SPEED PORT SELECT Command has been executed, this command is used for reading current speed data from the DRIVE DATA1, 2 and 3 PORTs.

The contents of DRIVE DATA1 PORT to DRIVE DATA3 PORT are as follows.



Using this the data obtained through this function, pulse speed can be determined as follows:

$$\text{PULSE SPEED} = \frac{160,000,000}{V} \ (\text{Hz}) \qquad V = \text{READ DATA}$$

(Example) When read data $V = 48(30_H)$

$$\text{PULSE SPEED} = \frac{160,000,000}{48} \fallingdotseq 3.3(\text{MHz})$$

Note1: DRIVE=1 may not be checked. But, note that speed data immediately before the stop will be continuously generated ever after the drive has been stopped.

Note2: Be sure to perform a DATA READ operation in the order of DRIVE DATA1 PORT, DRIVE DATA2 PORT and DRIVE DATA3 PORT.
If this sequence is not observed, the data will not be guaranteed.

7. MACHINE ORIGIN DETECTING FUNCTION

There are 9 MCC05v2 machine origin detection types in all, namely, ORG-0,ORG-1,ORG-2,ORG-3,ORG-4,ORG-5,ORG-10, ORG-11 and ORG-12. The X axis, Y axis , Z axis, A axis, B axis and C axis are provided independently with this function, so mutual interference does not occur.
In each process of ORG-0 to ORG-5 and ORG-11 to ORG-12, the machine origin address once detected is stored in memory, thereby permitting machine origin detection in a short time thereafter. Therefore, a detection flag is prepared inside the MCC05v2.
When this flag is ON, movement is performed nearly up to the machine origin(ORIGIN + OFFSET PULSE) by ABSOLUTE INDEX DRIVE, then DRIVE of the processes shown in and after 7-2. is performed.
When the flag is OFF, DRIVE of each process is directly performed without performing ABSOLUTE INDEX DRIVE.

The following description is given about the X axis but also applicable to the Y axis, Z axis and A axis.

*Detection flag ON condition
① The machine origin was normally detected by ORG DRIVE.
*Detection flag OFF conditions
① POWER ON/RESET
② DRIVE was put into an fast stop by the FSSTOP signal or the FSSTOP command.
(Including an fast stop by the match output of COMPARE REGISTER)
③ ORIGIN DRIVE was stopped by the SLOW STOP command and etc.
(Including DEND ERROR or ORIGIN ERROR was occurred.)
④ When the LIMIT stop type was set to fast stop, DRIVE was stopped by the LIMIT signal.
⑤ ORG DRIVE different from the previous ORG DRIVE was started.
⑥ ADDRESS exceeds the range of +8,388,607 to -8,388,607.
⑦ The ORIGIN FLAG RESET command or SPEC INITIALIZE4 Command was executed.

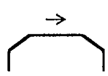
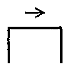
· The machine near-origin address for making a return with the detection flag ON is controlled in the MCC05v2, so the user does not need to consider anything. Even if ADDRESS has been updated by the ADDRESS INITIALIZE command, the machine near-origin address is also update, so the physical location is stored in memory.
· The machine near-origin address depends on the ORG type. In the ORG-0 to ORG-3, ORG-11 and 12 types, the machine near-origin address is the location of "machine origin detection end position + OFFSET PULSE". In the ORG-4,5 types, the machine near-origin address is the location of "$\overline{\text{NORG}}$ signal + OFFSET PULSE". The OFFSET PULSE is specified in the range of 0 to 255 pulses by the OFFSET PULSE SET command. OFFSET PULSE becomes 0 at POWER ON/RESET.

7-1. Machine Origin Detection Types

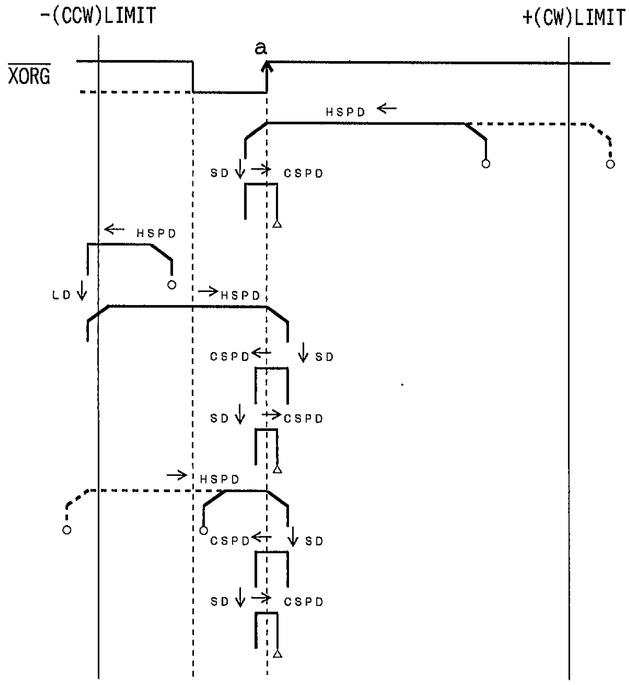
There are 9 machine origin detection types, each of which is characterized as shown in the following table.

Detection Type	Number of sensors	Sensor status at completion	Compensation of backlash	Standard number of processes	Accuracy	Time required
ORG-0,11	1 piece	Sensor OFF	Compensation	2	C	Short
ORG-1	1 piece	Sensor ON	Compensation	2	C	Short
ORG-2,12	1 piece	Sensor OFF	Compensation	4	B	Long
ORG-3	1 piece	Sensor ON	Compensation	4	B	Long
ORG-4	2 piece	Sensor OFF	Compensation	4 or 5	A	Longest
ORG-5	2 piece	Sensor ON	Compensation	4 or 5	A	Longest
ORG-10	2 piece	Sensor ON	No Compensation	2	C	Shortest

· Standard number of processes
Indicates the number of DRIVEs of CONSTANT SCAN, SCAN and JOG started by ORIGIN DRIVE.
Regarding JOG DRIVE, however, the repeated JOG DRIVE process is 1.
· Accuracy
A denotes the top accuracy and the accuracy is lowered in the order of B and C.
In the following process drawings, the meaning of each symbol is as follows.

$\overline{\text{XORG}}$,$\overline{\text{XNORG}}$ ----------------- Indicates a sensor signal. (Low with the sensor ON)
Mark ○ ------------------ Indicates the detection start position.
Mark △ ------------------ Indicates the detection end position.
 ------------------ Indicates SCAN DRIVE and its direction.
 ------------------ Indicates CONSTANT SCAN DRIVE and its direction.
 ------------------ Indicates repeated JOG DRIVE and its direction.
LD ------------------ Indicates a stop during LIMIT DELAY TIME.
SD ------------------ Indicates a stop during SCAN DELAY TIME.
JD ------------------ Indicates a stop during JOG DELAY TIME.

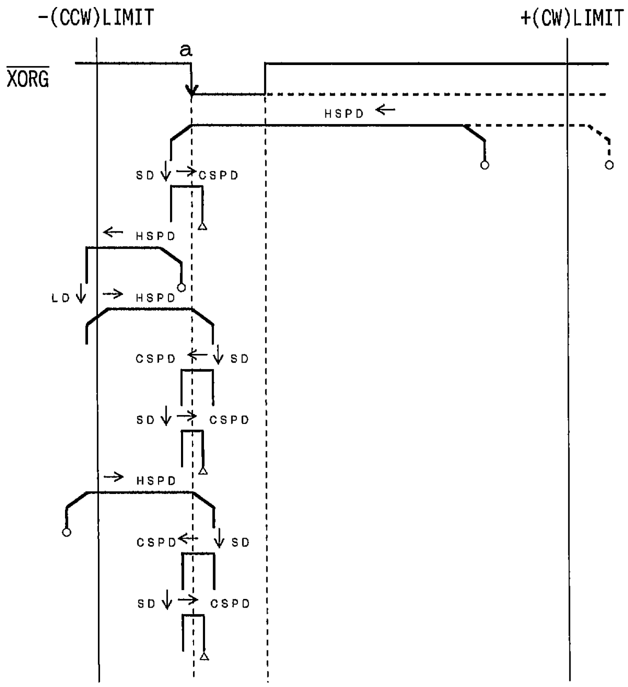
7-2. ORG-0 Type



HSPD : HIGH SPEED
CSPD : CONSTANT SPEED
LD : LIMIT DELAY TIME
SD : SCAN DELAY TIME
○ : Detection start position
△ : Detection end position

This type uses a single sensor. The +(CW)side edge (point a) of the XORG signal is detected.
The ORG sensor is one that holds a single pulse or the -(CCW)side level.

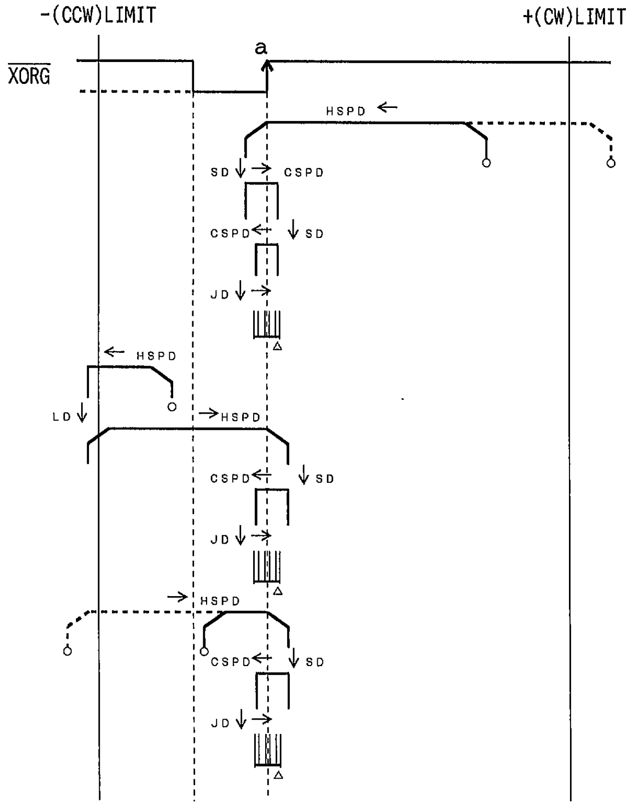
7-3. ORG-1 Type



HSPD : HIGH SPEED
CSPD : CONSTANT SPEED
LD : LIMIT DELAY TIME
SD : SCAN DELAY TIME
○ : Detection start position
△ : Detection end position

This type uses a single sensor. The -(CCW)side edge (point a) of the XORG signal is detected.
The ORG sensor is one that holds a single pulse or the +(CW)side level.

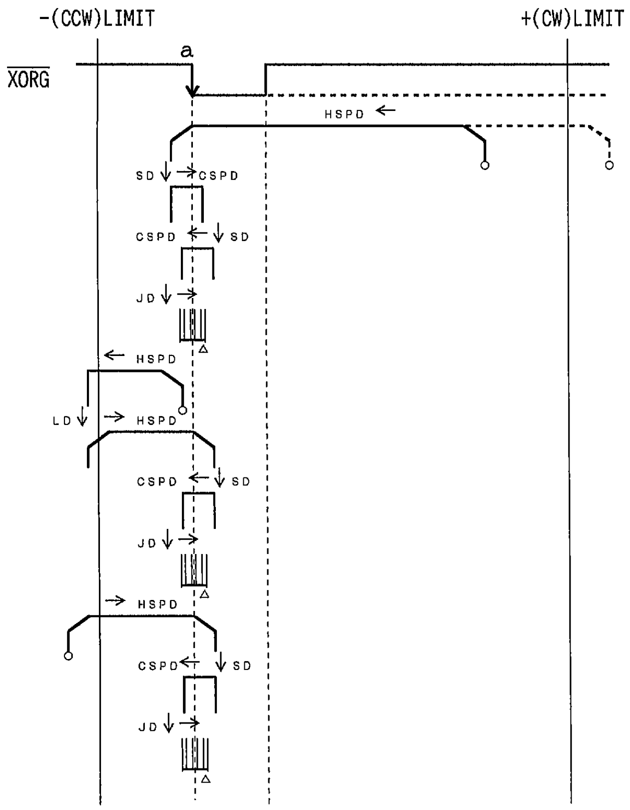
7-4. ORG-2 Type



HSPD : HIGH SPEED
CSPD : CONSTANT SPEED
LD : LIMIT DELAY TIME
SD : SCAN DELAY TIME
JD : JOG DELAY TIME
○ : Detection start position
△ : Detection end position

This type used a single sensor. The +(CW)side edge (point a) of the XORG signal is detected.
The ORG sensor is one that holds a single pulse or the -(CCW)side level.

7-5. ORG-3 Type



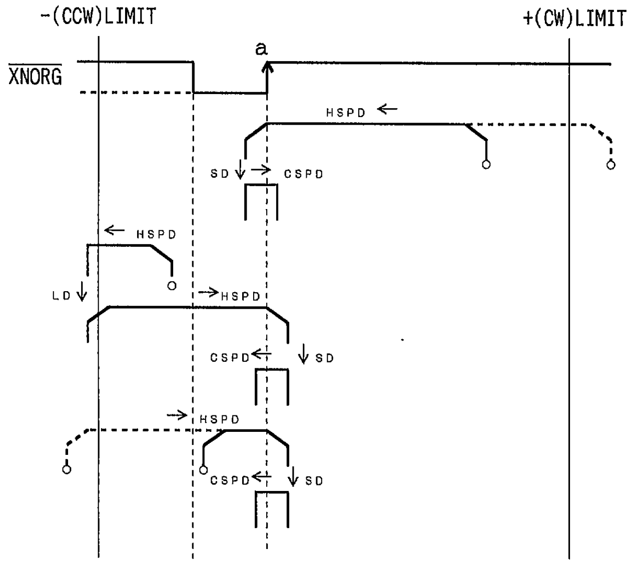
HSPD : HIGH SPEED
CSPD : CONSTANT SPEED
LD : LIMIT DELAY TIME
SD : SCAN DELAY TIME
JD : JOG DELAY TIME
○ : Detection start position
△ : Detection end position

This type uses a single sensor. The -(CCW)side edge (point a) of the XORG signal is detected.
The ORG sensor is one that holds a single pulse or the +(CW)side level.

7-6. ORG-4 Type

The NEAR ORG process is performed first. Next, the ORG process is performed.

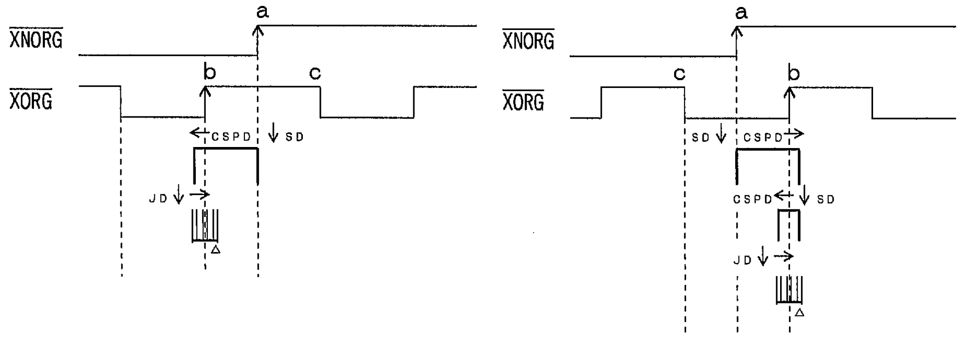
(1) NEAR ORG process



HSPD : HIGH SPEED
CSPD : CONSTANT SPEED
LD : LIMIT DELAY TIME
SD : SCAN DELAY TIME
JD : JOG DELAY TIME
○ : Detection start position
△ : Detection end position

(2) ORG process

- XORG=HIGH upon detection of point a (sensor OFF)
- XORG=LOW upon detection of point a (sensor ON)

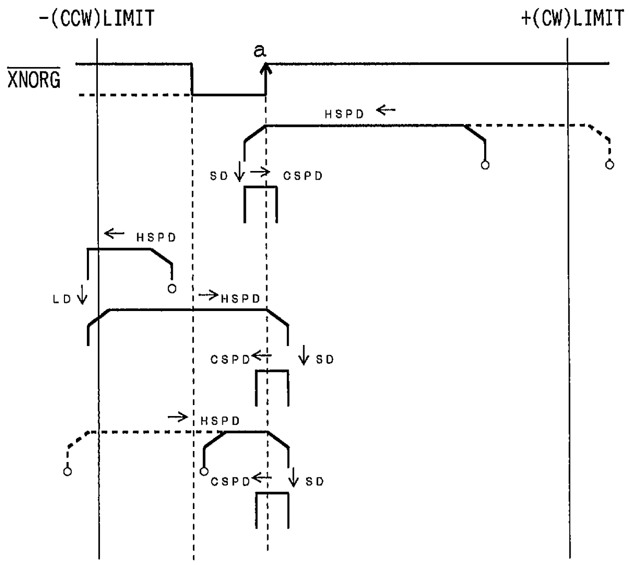


This type uses two sensors. After the +(CW)side edge (point a) of the XNORG signal is detected, the +(CW) side edge (point b) of the XORG signal is detected.
The NORG sensor is one that holds a single pulse or the -(CCW)side level, and the ORG sensor is one that generates a signal such as rotary axis slit cyclically.

7-7. ORG-5 Type

The NEAR ORG process is performed first. Next, the ORG process is performed.

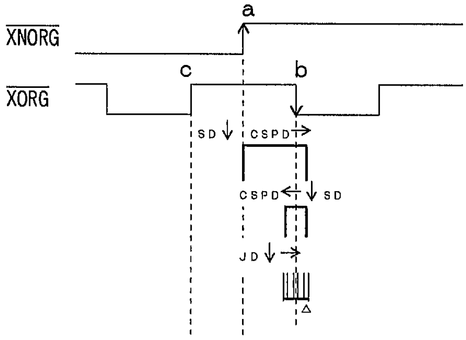
(1) NEAR ORGⅠ process



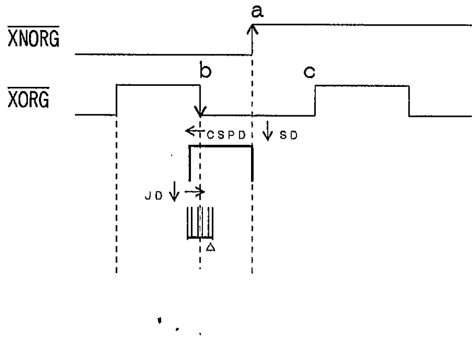
HSPD : HIGH SPEED
CSPD : CONSTANT SPEED
LD : LIMIT DELAY TIME
SD : SCAN DELAY TIME
JD : JOG DELAY TIME
○ : Detection start position
△ : Detection end position

(2) ORG process

· XORG=HIGH upon detection of point a (sensor OFF)

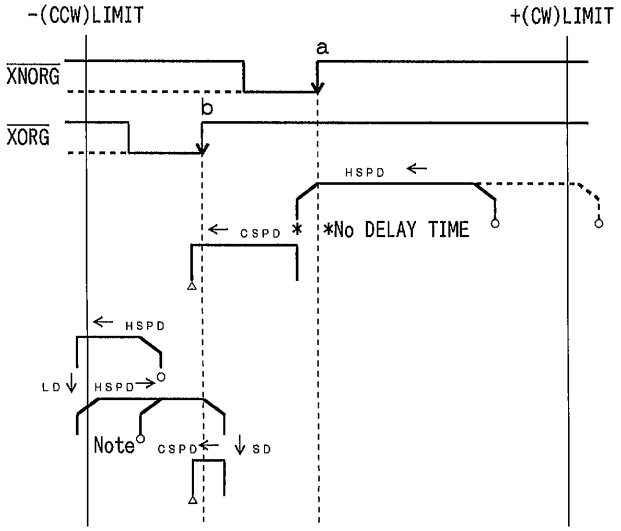


· XORG=LOW upon detection of point a (sensor ON)



This type uses two sensors. After the +(CW)side edge (point a) of the XNORG signal is detected, the -(CCW) side edge (point b) of the XORG signal is detected.
The NORG sensor is one that holds a single pulse or the -(CCW)side level, and the ORG sensor is one that generates a signal such as rotary axis slit cyclically.

7-8. ORG-10 Type



HSPD : HIGH SPEED
CSPD : CONSTANT SPEED
LD : LIMIT DELAY TIME
SD : SCAN DELAY TIME
○ : Detection start position
△ : Detection end position

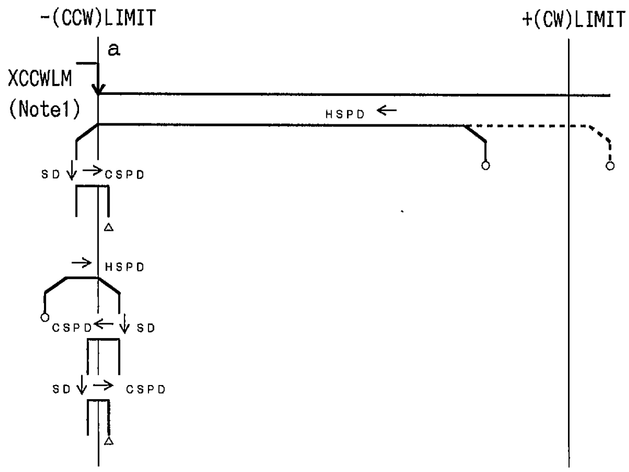
Note: Detection has been started with both XNORG signal and XORG signal ON.

This type uses two sensors. The +(CW)side edge (point a) of the XNORG signal or the +(CW)side edge (point b) of the XORG signal is detected and CONSTANT SCAN DRIVE is performed to point b.
Both NORG sensor and ORG sensor are ones that hold a single pulse or the -(CCW)side level.

7-9. ORG-11 Type

⚠ CAUTION

The drive is stopped at the Slow Stop irrespective of type of the LIMIT stop.
So there may be a collision with the limit position of the mechanical device before stop.
This may lead to machine or workpiece damage.
Note that the stop point will be changed if RATE, HSPD, etc. are changed.



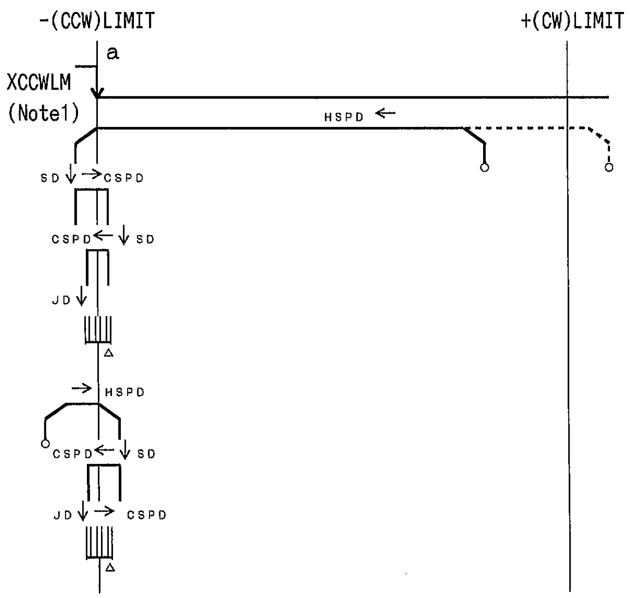
HSPD : HIGH SPEED
CSPD : CONSTANT SPEED
SD : SCAN DELAY TIME
○ : Detection start position
△ : Detection end position

This is a detection done using a single sensor. +(CW) side edge of XCCWLM signal (indicated as "a" in the figure) is detected. -(CCW) LIMIT sensor is used as the ORG sensor.
XCCWLM signal used must be the one that holds a single pulse or level.
Note: XORG signal is also enabled in this type, so make sure that it will not be turned active.

7-10. ORG-12 Type

⚠ CAUTION

The drive is stopped at the Slow Stop irrespective of type of the LIMIT stop.
So there may be a collision with the limit position of the mechanical device before stop.
This may lead to machine or workpiece damage.
Note that the stop point will be changed if RATE, HSPD, etc. are changed.



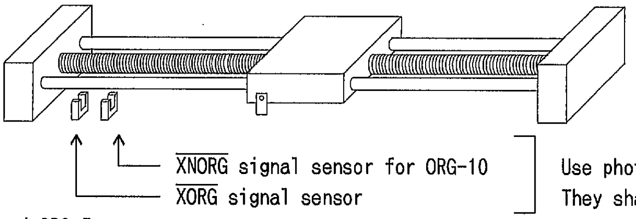
HSPD : HIGH SPEED
CSPD : CONSTANT SPEED
SD : SCAN DELAY TIME
JD : JOG DELAY TIME
○ : Detection start position
△ : Detection end position

This is a detection done using a single sensor. +(CW) side edge of XCCWLM signal (indicated as "a" in the figure) is detected. -(CCW) LIMIT sensor is used as the ORG sensor.
XCCWLM signal used must be the one that holds a single pulse or level.
This type of detection differs from the ORG-11 detection in that it repeats the JOG DIVE in the final process.
Note: XORG signal is also enabled in this type, so make sure that it will not be turned active.

7-11. Sensor Arrangement

(1) Mount the XORG signal sensor for ORG-0, ORG-1, ORG-2 or ORG-3 and the XNORG/XORG signal sensor for ORG-10 on the -(CCW)LIMIT side along the work moving direction.

Example) For a ball screw table

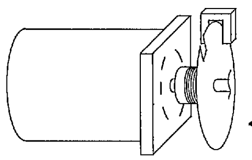


(2) ORG-4 and ORG-5

- Mount the XNORG signal sensor on the -(CCW)LIMIT side along the work moving direction in the same way as (1).
- XORG signal sensor

  When using a stepping motor:

  Mount this sensor on the motor rotary shaft as shown below.



When using a servo motor:

Connect the encoder Z phase (Cϕ) output signals +Z and -Z of the servo driver to the +XZORG and -XZORG of the C-871. For details, refer to Chapter 1 4.

The pulse width of the encoder Z phase (Cϕ) output shall be 10μs or more.

(3) ORG-11,12

These types require the LIMIT sensor alone. XCCWLM signal is used as the origin signal.
Since XORG signal is also enabled, make sure that it will not be turned active.

7-12. Detecting Conditions

(1) For the ORG-0, ORG-1, ORG-2 and ORG-3 types, the XORG signal should be detected for 1ms or more when it passes the ORG sensor at the maximum speed. For the ORG-4, ORG-5 and ORG-10 types, the XNORG signal should be detected for 1ms or more when it passes the NORG sensor at the maximum speed.

(2) For the ORG-4 and ORG-5 types, the distance between point a and point b and the distance between point a and point c should be N pulses or more in terms of the number of pulses.

*N=0.005×CSPD(Hz) (Minimum value of N is 1.)

Example) When CSPD=5KHz, N=0.005×5000=25. Accordingly, the distance should be 25 pulses or more. Practically, give some allowance to it.

(3) Each of the XORG and XNORG signals should have no chattering.(When photo sensors are used, this does not matter.)

(4) The distance between point a and +(CW)LIMIT shown in each process drawing should be enough for a slow stop.

(5) The distance between point a and point b shown in the ORG-10 type should be enough for a slow stop.

(6) With the ORG-11 and 12 detection, sufficient distance must be provided between the point a and the machine limit in CCW direction so that slow stop (through deceleration) is ensured for the drive.

⚠ CAUTION	There may be a collision with the limit position of the mechanical device before stop. his may lead to machine or workpiece damage. Note that the stop point will be changed if RATE, HSPD, etc. are changed.

7-13. Other Functions

The following is additionally prepared as applied functions:

1. The ORIGIN DRIVE direction switch function for the time when you use the sensor in +(CW) side.
2. The MARGIN TIME function for preventing malfunction that can result from hunting.
3. The SENSOR TYPE select function used in the JOG DRIVE.
4. The ERROR DETECT function prepared for the time when the ORIGIN SENSOR detection ended unsuccessfully.
5. The function to produce XDRST signal as the origin detection is completed.
6. The function of ANDing the XORG signal and PO signal from the stepping motor driver.

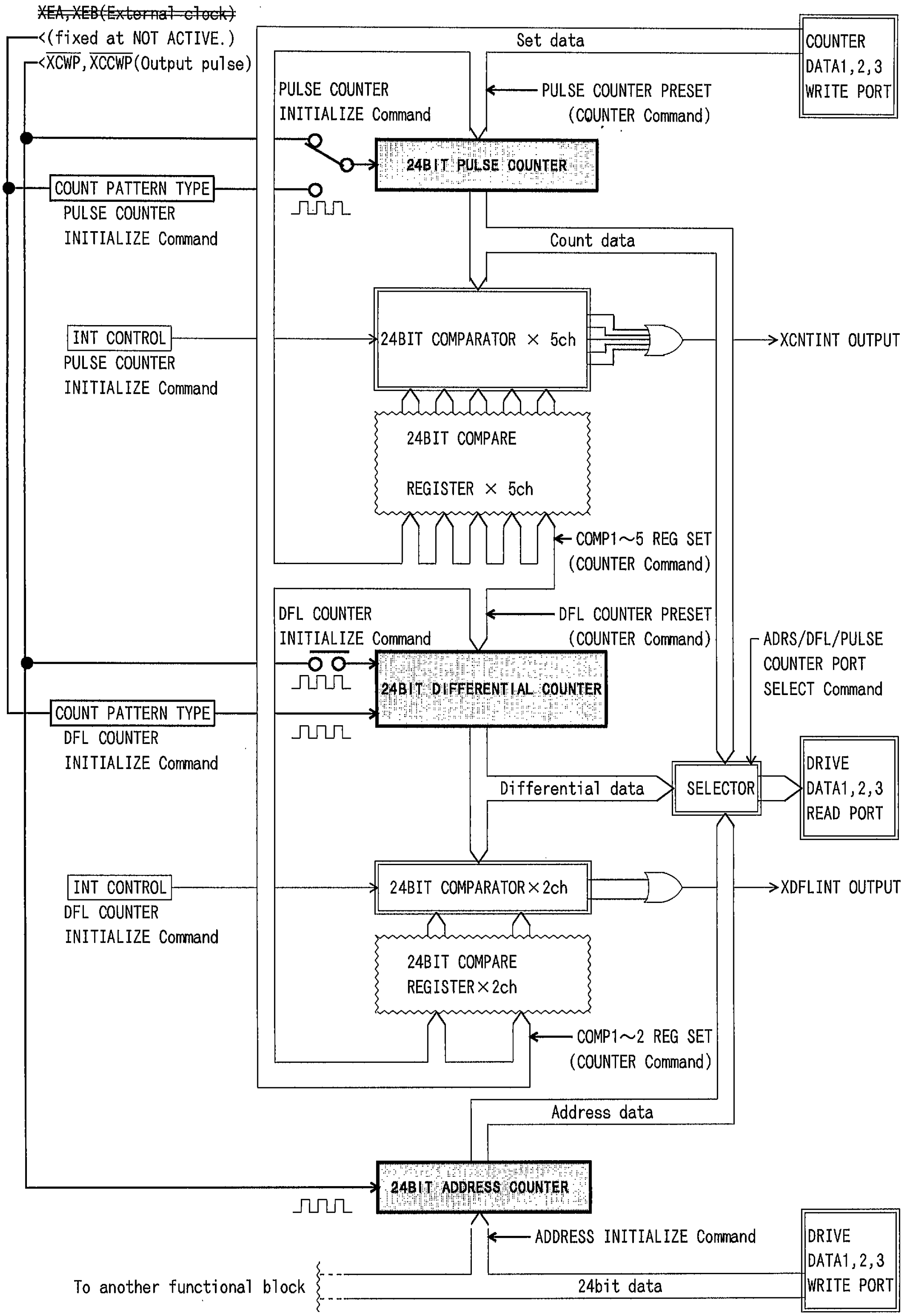
For details of the applied functions, refer to the User's Manual [Applied Functions Part].

8. DETAILS OF COUNTER FUNCTIONS

This chapter describes the counter function of the MCC05v2. However, since this product is not provided with an external clock input interface, some of the functions are invalid.
The invalid functions are covered with double lines (~~functions~~).
The following description is given about the X axis, but also applicable to the another axis.

8-1. Functional Block Diagram

The MCC05v2 has three built-in 24-bit HARD COUNTERs each provided with different functions.



8-2. ADDRESS COUNTER Function

(1) In order to control current address, the ADDRESS COUNTER offers absolute address of the output pulse from the MCC05v2.

(2) The count data can be read from the DRIVE DATA1, 2 and 3 PORTs any time you want (as long as the ADDRESS COUNTER PORT is selected). The ADDRESS READ command can also be used for reading the data.
The data guarantee range is pulse area of +8,388,607 to -8,388,607.

(3) The counter value is reset to 0 at POWER ON/RESET.
Using the ADDRESS INITIALIZE Command allows you to set optional value on the counter.

8-3. PULSE COUNTER Function

(1) Pulse counting function

a. The PULSE COUNTER offers counting of output pulse ~~or external input clock to the MCC05v2~~.

b. Count data can always be read out from DRIVE DATA1, 2 and 3 PORT.
The data guaranty range is a pulse area of +8,388,607 to -8,388607. The data reaches (8,388,608, an overflow occurs. When an overflow occurs, OVF BIT in STATUS3 PORT becomes 1.

c. The counter value is reset to 0 at POWER ON /RESET and can also be set to an optional value by the COUNTER PRESET command.

d. ~~When inputting external signal such as feed back pulses from the servo driver, a 90° phase difference signal or CW/CCW independent clock can be used as an input clock. When the 90° phase difference signal is selected, a count multiplier can also be set.~~
The above input clock selection and count multiplier selection are performed by the PULSE COUNTER INITIALIZE command.
At RESET, output pulses of MCC05v2 are selected as input clocks.

(2) PULSE COUNT COMPARE Function

a. Five COMPARE REGISTERs and COMPARATORs connected to the PULSE COUNTER allows you to detect any count value.

b. Detection of matching between the counter and comparator is done with the STATUS signal or interrupt request signal. You can select the through mode (detection done by the comparator is output as it is) or the latch mode (holds the detection) for the STATUS and interrupt request signals.
In the latch mode, you can reset the STATUS signal or interrupt request signal by reading the STATUS3 PORT. And, you can select a mode that enables the reset or another mode that disables it even when the condition (matching established between the counter and comparator).
For details, refer to 8-5.
The interrupt request signal (XCNTINT) is output from five comparators. You can enable or disable the output on individual comparator basis.

c. The pulse output can also be stopped by match among comparators. You can specify the immediate output stop or gradual stop through deceleration. The immediate stop sets the FSEND BIT to 1 and the SSEND BIT is set to 1 when the gradual stop is specified.

d. The PULSE COUNTER INITIALIZE command is used for controlling the every PULSE COUNT COMPARE function.
Data for the COMPARE REGISTER is set on the COUNTER PORT. The COUNTER PORT is independent from the DRIVE PORT, thus rewriting of the compare data is available any time.

e. Specific functions of COMPARE REGISTER1
Specific functions that are not provided in the other COMPARE REGISTERs are assigned to COMPARE REGISTER1.
The following functions can be automatically performed by a match of COMPARE REGISTER1.
These functions are also controlled by the PULSE COUNTER INITIALIZE command.

*AUTO CLEAR function
The PULSE COUNTER value is cleared to 0 concurrently with a match of COMPARE REGISTER1.

*RELOAD function
The data written in COUNTER DATA1 PORT, COUNTER DATA2 PORT and COUNTER DATA3 PORT is reset to COMPARE REGISTER1 concurrently with a match of COMPARE REGISTER1.

8-4. DIFFERENTIAL COUNTER function

(1) Differential Count Function

a. The DIFFERENTIAL COUNTER offers counting of ~~deviation between~~ output pulse from the MCC05v2 ~~and external input clock, or counting of external clock alone~~.

Note: The DIFFERENTIAL COUNTER count down from +(CW) direction pulse and count up from -(CCW) direction pulse. So this is reverse of ADDRESS and PULSE COUNTER.

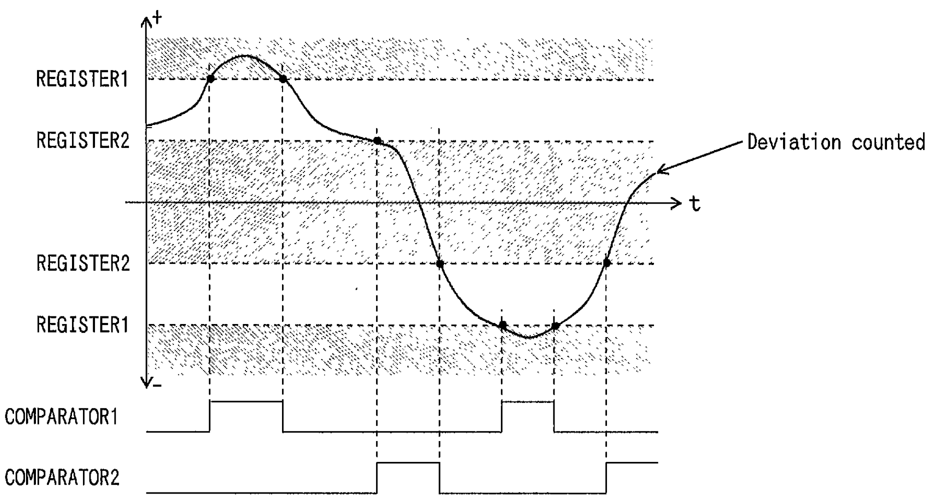
b. Count data can be read from the DRIVE DATA1, 2 and 3 PORTs any time as needed (as long as the DFL COUNTER PORT is selected). Data guarantee range is pulse area of +8,388,607 to -8,388,607.

c. The counter value is reset to 0 at POWER ON/RESET.
Using the DFL COUNTER PRESET Command of the COUNTER Commands allows you to set optional value on the counter.

d. ~~Any one of 90°phase difference signal and respectively independent CW or CCW clock may be used as external clock.~~ For selecting type of external input clock and also for selecting its multiplication factor, the DFL COUNTER INITIALIZE command is used.
At POWER ON/RESET, a single multiplication of 90°phase difference signal is selected and the counter functions as the ~~differential~~ pulse counter.

(2) Differential Count Compare Function

a. The DIFFERENTIAL COUNTER is connected to two COMPARE REGISTERs and COMPARATORs. Its counting (in absolute value) is constantly compared against the value set on these two registers (these registers are different from COMPARE REGISTERs on the PULSE COUNTER).
The COMPARATOR1 is constantly available for detection of the "count in absolute value≧REGISTER1" ~~(excessive deviation)~~ and the COMPARATOR2 is used for detecting the "count in absolute value≦REGISTER2" ~~(positioning complete)~~.



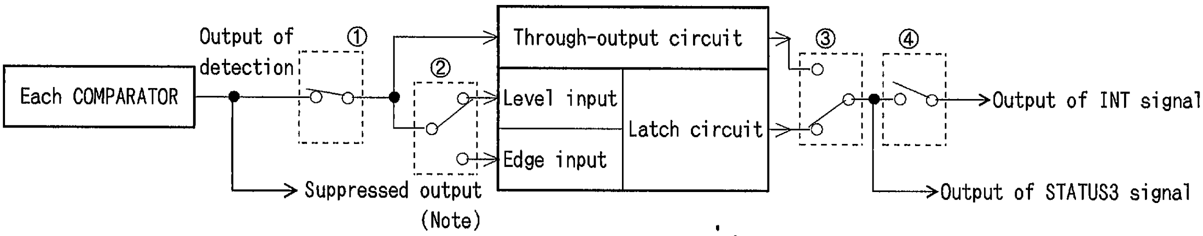
b. Each of above detection is implemented by the STATUS signal or interrupt request signal. You can select the through mode (detection by the comparator is output as it is) or the latch mode (holds the detection) for both the STATUS signal and the interrupt request signal.
In the latch mode, you can reset the STATUS signal or interrupt request signal by reading the STATUS3 PORT. And, you can select a mode that enables the reset or another mode that disables it even when the condition (COUNTER≧REGISTER1 or COUNTER≦REGISTER2).
For details, refer to 8-5.
The interrupt request signal (XDFLINT) is output from two comparators. You can enable or disable the output on individual comparator basis.

c. The pulse output can also be stopped by the COUNTER≧REGISTER1 or COUNTER≦REGISTER2.
You can specify the immediate output stop or gradual stop through deceleration.
The immediate stop sets the FSEND BIT to 1 and the SSEND BIT is set to 1 when the gradual stop is specified.

d. The DFL COUNTER INITIALIZE command is used for controlling the DFL COUNT COMPARE function.
Data for the COMPARE REGISTER is set on the COUNTER PORT. The COUNTER PORT is independent from the DRIVE PORT, thus rewriting of the compare data is available any time.

8-5. Detail of COMPARATOR Functions

Output of the detected condition from five COMPARATORs for the PULSE COUNTER and two COMPARATORs for the DIFFERENTIAL COUNTER are connected to the following functional circuit and, thus, allows control according to the user specification.



①: INT MASK Circuit

This circuit masks output from the COMPARATOR at its exit.
You can specify separate levels of masking for each comparator on the PULSE and DFL counters.
This circuit is turned on or off by the INT MASK command.
As one of the special commands, it provides real-time, fine-tuned control of the masking.

②: LATCH TYPE Switch Circuit

This circuit is used for selecting a latch trigger type when detection by the COMPARATOR is latched before it is output.
The latch type selected commonly applies to five comparators on the PULSE COUNTER.
Likewise, the type selected applies commonly to two comparators on the DIFFERENTIAL COUNTER.
You can differentiate the latch type between the PULSE COUNTER and the DIFFERENTIAL COUNTER.

Latch-output reset condition varies according to the trigger type as described in the following.

- When the level latch is selected:
  When output of the COMPARATOR detection <u>is not taking place</u>, reading the STATUS3 PORT resets the output (returns to the initial state).
- When the edge latch is selected:
  Reading the STATUS3 PORT <u>necessarily resets</u> the output.

This circuit is turned on or off by the PULSE or DFL COUNTER INITIALIZE Command.

③: INT OUTPUT TYPE Switch Circuit

This circuit is used for selecting whether the COMPARATOR detection is to be output as it is (through) or after latch. The output type selected commonly applies to the five comparators on the PULSE COUNTER.
Likewise, the selected type commonly applies to the two comparators on the DIFFERENTIAL COUNTER.
You can differentiate the output type between the PULSE COUNTER and the DIFFERENTIAL COUNTER.
This circuit is turned on or off by the PULSE or DFL COUNTER INITIALIZE command
(initially, the latch is selected).
When the through-output of detection is selected, if you execute the COUNTER Command in the course of the INT output, this output will be turned off for a duration of 50ns.

④: INT Output Enable Circuit

A signal that has passed through ① through ③ circuits above can be unconditionally confirmed from the STATUS3 PORT. This circuit is used for selecting whether this signal is to be externally output as it is (XCNTINT or XDFLINT signal).
Output of the INT signal can be separately enabled or disabled for each of the pulse or DFL comparator.
This circuit is turned on or off by the PULSE or DFL COUNTER INITIALIZE command.

All setups except for the ① INT MASK circuit are done by the COUNTER INITIALIZE command.
Thus, these setups must be complete before pulse output it turned on.

Note: Unlike the INT output or STATUS, comparator's pulse output stop function is directly enabled without going through the above mentioned circuit.

The following shows pages in this manual that bear descriptions relevant to this function:

The PULSE COUNTER INITIALIZE Command -------- page 30
The DFL COUNTER INITIALIZE Command --------- page 36
The INT MASK Command ----------------------- page 48
Timing of each signal ---------------------- page 72, 73

9. DESCRIPTION OF PULSE COUNTER AND DFL COUNTER COMMAND

9-1. Command Table

The HEX code is for the case where all the x bits are set to 0.

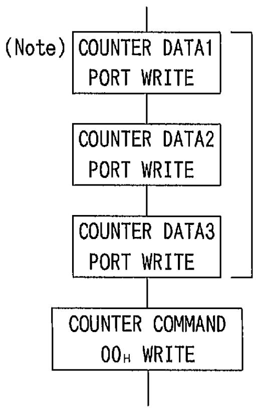
$D^7D^6D^5D^4D^3D^2D^1D^0$	HEX CODE	COMMAND NAME	EXECUTION TIME
x x x x 0 0 0 0	0 0	PULSE COUNTER PRESET	MAX 200ns
x x x x 0 0 0 1	0 1	PULSE COUNTER COMPARE REGISTER1 SET	MAX 200ns
x x x x 0 0 1 0	0 2	PULSE COUNTER COMPARE REGISTER2 SET	MAX 200ns
x x x x 0 0 1 1	0 3	PULSE COUNTER COMPARE REGISTER3 SET	MAX 200ns
x x x x 0 1 0 0	0 4	PULSE COUNTER COMPARE REGISTER4 SET	MAX 200ns
x x x x 0 1 0 1	0 5	PULSE COUNTER COMPARE REGISTER5 SET	MAX 200ns
x x x x 0 1 1 0	0 6	DIFFERENTIAL COUNTER PRESET	MAX 200ns
x x x x 0 1 1 1	0 7	DIFFERENTIAL COUNTER COMPARE REGISTER1 SET	MAX 200ns
x x x x 1 0 0 0	0 8	DIFFERENTIAL COUNTER COMPARE REGISTER2 SET	MAX 200ns

9-2. PULSE COUNTER PRESET Command

COMMAND········ 00_H

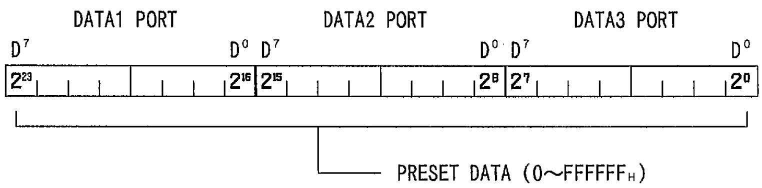
Function : Initializes the count value of the PULSE COUNTER to a specified value.

Execution sequence



Preset data is specified in COUNTER DATA1 PORT, COUNTER DATA2 PORT and COUNTER DATA3 PORT.

The contents of COUNTER DATA1 PORT to COUNTER DATA3 PORT are as follows.



When preset data is negative, it is represented in 2's complement form.
Counter value is 0 at POWER ON/RESET.

· Setting example of preset data

PRESET DATA(decimal)	DATA1 PORT	DATA2 PORT	DATA3 PORT
+8,388,607	$7F_H$	FF_H	FF_H
+10	00_H	00_H	$0A_H$
±0	00_H	00_H	00_H
-10	FF_H	FF_H	$F6_H$
-8,388,607	80_H	00_H	01_H

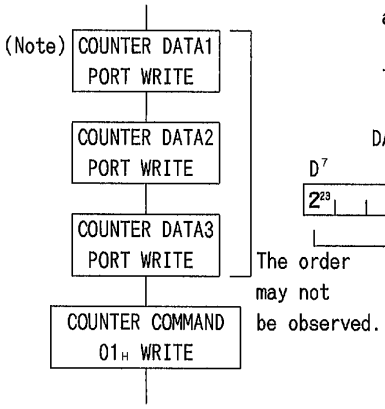
Note: Note that the DATA and COMMAND PORTs described in this chapter are special ports for the counters and their port addresses are different from that of the DRIVE PORTs. For the port address, refer to 4-1.

9-3. PULSE COUNTER COMPARE REGISTER1 SET Command

COMMAND……… 01H

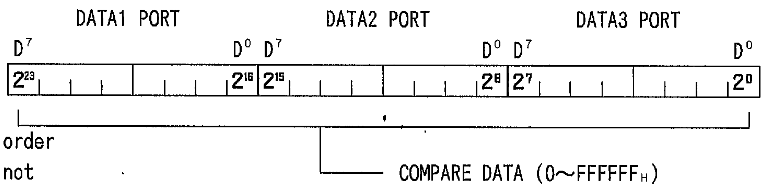
Function : Sets a specified value in COMPARE REGISTER1.

Execution sequence



COMPARE DATA is specified in COUNTER DATA1 PORT, COUNTER DATA2 PORT and COUNTER DATA3 PORT.

The contents of COUNTER DATA1 PORT to COUNTER DATA3 PORT are as follows.



When COMPARE DATA is negative, it is represented in 2's complement form.
COMPARE DATA is 800000H at POWER ON/RESET.

Note: Note that the DATA and COMMAND PORTs described in this chapter are special ports for the counters and their port addresses are different from that of the DRIVE PORTs.
For the port address, refer to 4-1.

9-4. PULSE COUNTER COMPARE REGISTER2 SET Command

COMMAND……… 02H

Function: Sets a specified value in COMPARE REGISTER2.

The execution sequence is the same as that of the COMPARE REGISTER1 SET command.

9-5. PULSE COUNTER COMPARE REGISTER3 SET Command

COMMAND……… 03H

Function: Sets a specified value in COMPARE REGISTER3.

The execution sequence is the same as that of the COMPARE REGISTER1 SET command.

9-6. PULSE COUNTER COMPARE REGISTER4 SET Command

COMMAND……… 04H

Function: Sets a specified value in COMPARE REGISTER4.

The execution sequence is the same as that of the COMPARE REGISTER1 SET command.

9-7. PULSE COUNTER COMPARE REGISTER5 SET Command

COMMAND……… 05H

Function: Sets a specified value in COMPARE REGISTER5.

The execution sequence is the same as that of the COMPARE REGISTER1 SET command.

9-8. DFL COUNTER PRESET Command

COMMAND……… 06H

Function : Modifies the DIFFERENTIAL COUNTER value to the specified one.

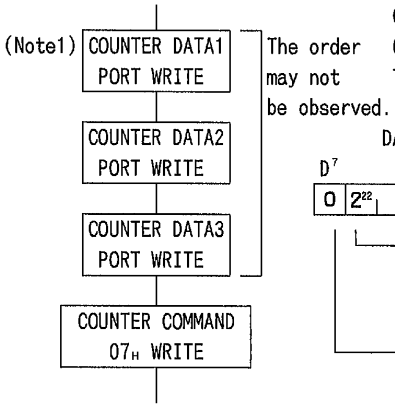
The execution sequence is the same as that of the PULSE COUNTER RESET command.
At POWER ON/RESET, the counter value is set to 0.

9-9. DFL COUNTER COMPARE REGISTER1 SET Command

COMMAND········ 07H

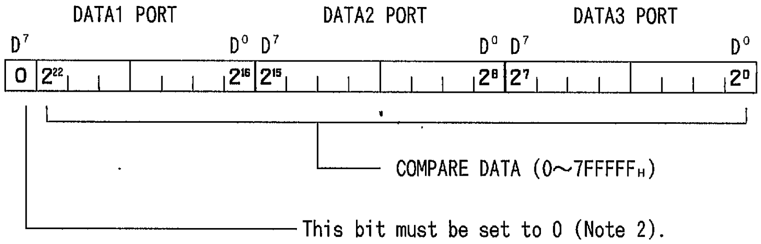
Function : Sets a value specified for the DIFFERENTIAL COUNTER COMPARE1 REGISTER in an absolute value.

Execution sequence



COMPARE DATA is specified in COUNTER DATA1 PORT, COUNTER DATA2 PORT and COUNTER DATA3 PORT.

The contents of COUNTER DATA1 PORT to COUNTER DATA3 PORT are as follows.



At POWER ON/RESET, it is set to 0.

Note1: Note that the DATA and COMMAND PORTs described in this chapter are special ports for the counters and their port addresses are different from that of the DRIVE PORTs. For the port address, refer to 4-1.

Note2: When the signed value detection, an applied function, is selected, this bit is replaced by 2^{23} bit.

9-10. DFL COUNTER COMPARE REGISTER2 SET Command

COMMAND········ 08H

Function : Sets a value specified for the DIFFERENTIAL COUNTER COMPARE2 REGISTER in an absolute value.

The execution sequence is the same as that of the DFL COUNTER REGISTER1 SET Command.

10. INITIAL SPECIFICATION TABLE

The initial specifications at POWER ON/RESET are shown in the following table.

Data name or specification	Initial specification	Corresponding Command
URATE(RATE DATA TABLE No.)	No.9(100ms/1000Hz)	RATE SET
DRATE(RATE DATA TABLE No.)	No.9(100ms/1000Hz)	
LSPD	300Hz	LSPD SET
HSPD	3000Hz	HSPD SET
CSPD	300Hz	CSPD SET
SRATE(RATE DATA TABLE No.)	No.9(100ms/1000Hz)	SRATE SET
SLSPD	300Hz	SLSPD SET
SHSPD	3000Hz	SHSPD SET
DRIVE TYPE	L-TYPE	SPEC INITIALIZE1
LIMIT STOP TYPE	FAST STROP	
MOTOR TYPE	STEPPING	
RDYINT generation pattern	Generated only upon termination of a command accompanied by pulse output.	
PULSE COUNTER operating clock	Output pulse(change invalid)	PULSE COUNTER INITIALIZE
CNTINT generation pattern	Not generated in any case (all of COMP1 to 5)	
COMP1 to 5 STOP TYPE	Not put into stop (all of COMP1 to 5)	
AUTO CLEAR function	Not available	
RELOAD function	Not available	
PLS COMP STOP TYPE	Fast stop	
CNTINT OUTPUT TYPE	Detection of each comparator is latched and output	
CNTINT LATCH TRIGGER TYPE	Level latch	
DIFFERENTIAL COUNTER operating clock	Output pulse Only (change invalid)	DFL COUNTER INITIALIZE
DIFFERENTIAL COUNTER count pattern type	————————	
DFLINT generation pattern	Not generated in any case (both of COMP1 to 2)	
DFL COMP1,2 STOP ENABLE	Not put into stop (both of COMP1 to 5)	
DFL COMP STOP TYPE	Fast stop	
DFLINT OUTPUT TYPE	Detection of each comparator is latched and output	
DFLINT LATCH TRIGGER TYPE	Level latch	
COUNTER SELECT PORT	PULSE COUNTER	Each PORT SELECT
Current address(ADDRESS COUNTER)	0	ADDRESS INITIALIZE
OFFSET PULSE	0	OFFSET PULSE SET
LIMIT DELAY TIME	300ms	ORIGIN DELAY SET
SCAN DELAY TIME	50ms	
JOG DELAY TIME	20ms	
PULSE COUNTER Value	0	PULSE COUNTER PRESET
PULSE COUNTER COMPARE REGISTER1 to 5 value	800000_H	PULSE COUNTER COMPARE REGISTER1～5 SET
DIFFERENTIAL COUNTER value	0	DFL COUNTER PRESET
DIFFERENTIAL COUNTER COMPARE REGISTER1 to 2 value	0	DFL COUNTER COMPARE REGISTER1～2 SET

11. INTERRUPTS

When any one of the following interrupt request signals has occurred, the C-871 gives an interrupt request to the initiator. INTA# interrupt signal pin is used.

11-1. Interrupt Request Signals upon Termination of Command (XRDYINT, YRDYINT, ZRDYINT, ARDYINT, BRDYINT, CRDYINT)

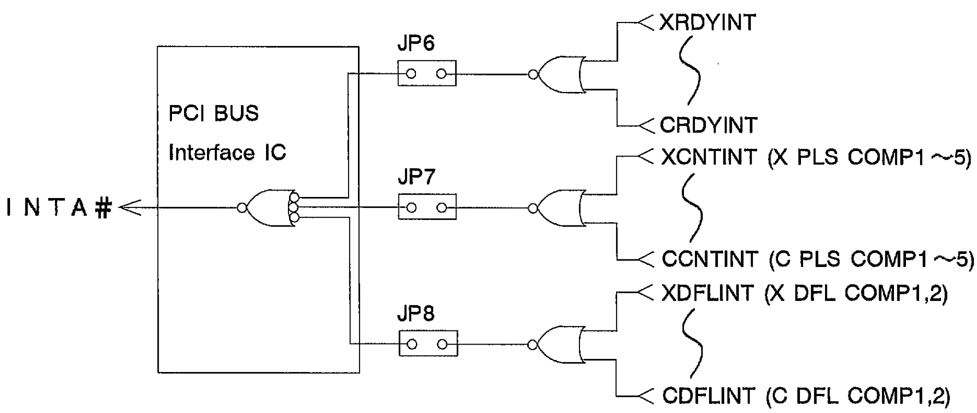
Use the STATUS5 PORT and STATUS1 PORT to identify the interrupt request axis.

11-2. Interrupt Request Signals from PULSE COUNTER and DIFFERENTIAL COUNTER

(XCNTINT,YCNTINT,ZCNTINT,ACNTINT,BCNTINT,CCNTINT,XDFLINT,YDFLINT,ZDFLINT,ADFLINT,BDFLINT,CDFLINT)

CNTINTs and DFLINTs of X, Y, Z, A, B and C axes are OR-ed. The output of each axis is gained by OR-ing of COMP1 to COMP5, DFLCOMP1 to DFLCOMP2. Use the STATUS3 PORT of each axis to evaluate the interrupt request axis and COMPARE REGISTER. When you want to such an interrupt function, it must be set to the edge latch type where each INT signal is reset by the read of the STATUS3 PORT. Refer to the description of 8-5.

11-3. Interrupt Pin Arrangement



* XRDYINT to CRDYINT are output to the STATUS5 PORT of each axis. Refer to the description of 4-12.

11-4. Precautions on Using Interrupts

(1) Before using an interrupt, carefully confirm the User's Manual for your initiator system and the interrupt controller specifications.

(2) When you want to use the interrupt, insert the short circuit socket (an accessory) into the JP6 to JP8 in the above sketch in response to the interrupt request to be used.
So it is not inserted at the time of shipment, Interrupt cannot be used in that state. For the position on the board, refer to the description of 15-2. Board Shape and Dimensions.

(3) Interrupt enable or disable can be specified in RDYINT by the SPEC INITIALIZE1 command, CNTINT by the PULSE COUNTER INITIALIZE command and DFLINT by the DFL COUNTER INITIALIZE command.
For CNTINT and DFLINT, specially, the above specification is possible for each of COMP1 to COMP5 and DFL COMP1 to 2.
Accordingly, be set into the interrupt disable state on no use COMP No.

(4) When you want to use the RDYINT for multiple axis interrupt, perform the following processing in the interrupt processing routine:
"Check the state of the STATUS5 PORT BIT3(RDYINT) of each axis, and read the STATUS1 PORT of only the active axis to reset the RDYINT of that axis."
If the STATUS1 PORT of this axis is read when the STATUS5 PORT BIT3(RDYINT) is not active, the RDYINT of that axis may not occur, depending on the time when it is read.

1 2. T I M I N G

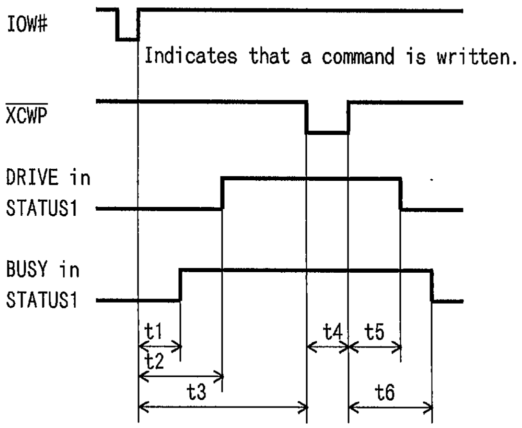
Refer to the description of 12-16 for IOR# and IOW#.

[ ] : Timing of case SOFT LIMIT function in use.(applied function)

Those not in brackets remain unaffected by presence or absence of the SOFT LIMIT function.

12-1. JOG DRIVE Timing

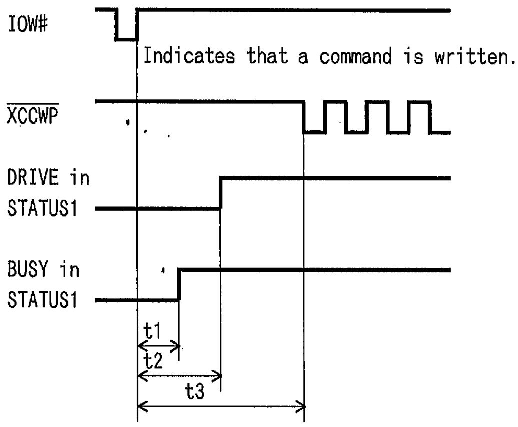
Example) Drive in +(CW)direction for a stepping motor



t1<150ns　　　t4≒100μs
t2<34μs[42μs] t5<23μs[31μs]
t3<36μs[44μs] t6<35μs[43μs]

12-2. SCAN DRIVE/S-RATE SCAN DRIVE Timing

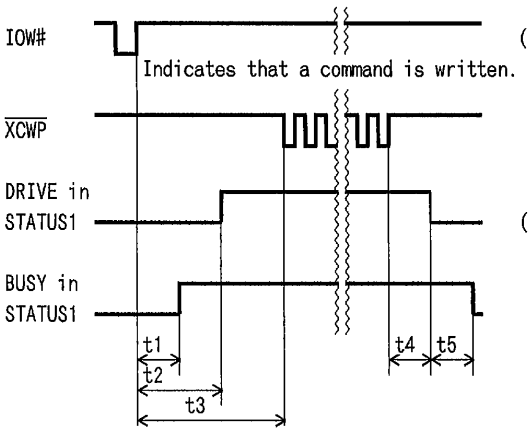
Example) Drive in -(CCW)direction for a stepping motor



t1<150ns
t2<43μs[75μs Note]
t3<63μs[95μs Note]
Note: 1.3ms when URATE≠DRATE is specified.

12-3. INDEX DRIVE, S-RATE INDEX DRIVE Timing

Example) Drive in +(CW)direction for a stepping motor



(1) When URATE=DRATE is specified:

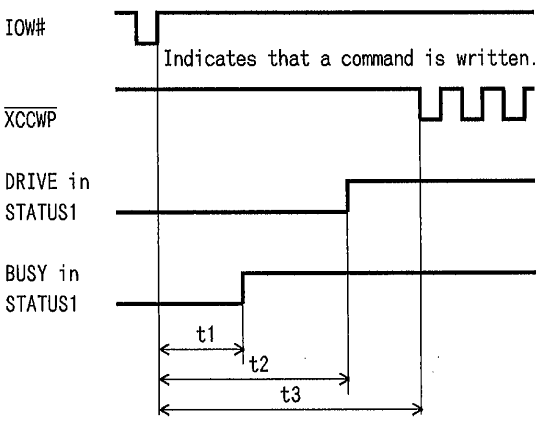
t1<150ns　　　t4<35μs[44μs]
t2<71μs[79μs]　　t5<20μs
t3<93μs[101μs]

(2) When URATE≠DRATE is specified:

t1<150ns　　　t4<35μs[44μs]
t2<1.3ms (L-TYPE)　t5<18μs
630μs(M-TYPE)
400μs(H-TYPE)
t3≒t2+22μs

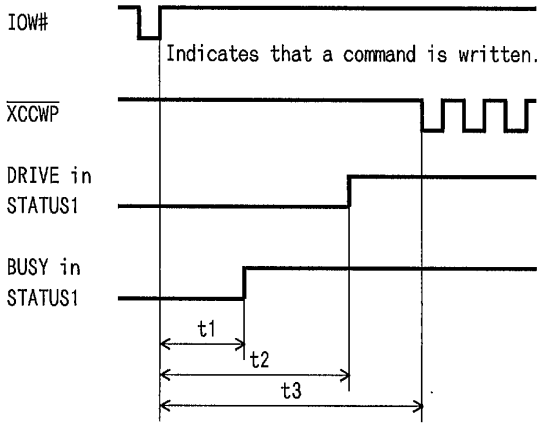
12-4. ORIGIN DRIVE Timing

Example 1) Drive in -(CCW)direction without ABSOLUTE INDEX DRIVE(RETURN DRIVE up to the near-origin address)



t1<150ns
t2<70μs
t3<82μs

Example 2) Drive in -(CCW)direction with ABSOLUTE INDEX DRIVE (RETURN DRIVE up to the near-origin address)



(1) When URATE=DRATE is specified:

$t1 < 150ns$

$t2 < 71\mu s$

$t3 < 93\mu s$

(2) When URATE≠DRATE is specified:

$t1 < 150ns$

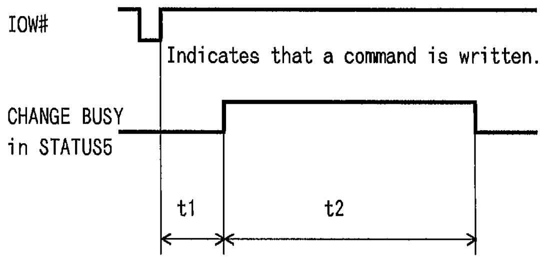
$t2 < 1.3ms$ (L-TYPE)

$630\mu s$(M-TYPE)

$400\mu s$(H-TYPE)

$t3 \fallingdotseq t2 + 22\mu s$

12-5. SPEED CHANGE Timing



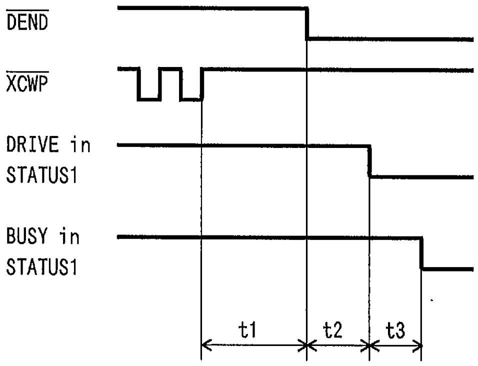
$t1 < 200ns$

$51\mu s < t2 < 53ms$

Note: t2 is determined by the specified RATE.
In the Fixed mode, t2 becomes shorter as a specified RATE No. increases, and in the Arithmetic mode, it becomes shorter as the RATE data becomes smaller (in both cases, speed change rate is increased).
If, however, a pulse cycle greater than t2 at writing of the CHANGE command, t2 becomes equal with or longer than the pulse cycle.

12-6. DEND Signal Confirmation Timing

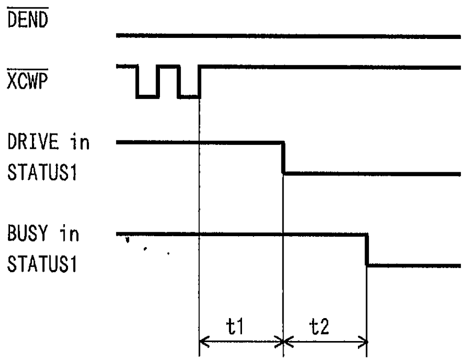
Example 1) End of DRIVE in +(CW)direction for a servo motor



t1:Depends on servo driver characteristics.
t2<41μs
t3<20μs

Note: When the target is a servo motor, the requirement for BUSY in STATUS1=0 is that DEND=0 must be input. This is also applicable to a normal end of DRIVE, a slow stop and an fast stop. Accordingly, note that the time required for BUSY in STATUS1=0 is different from that for the case where the target is a stepping motor.

Example 2) End of DRIVE in +(CW)direction when the target is a servo motor and the DEND signal is active or the target is a stepping motor

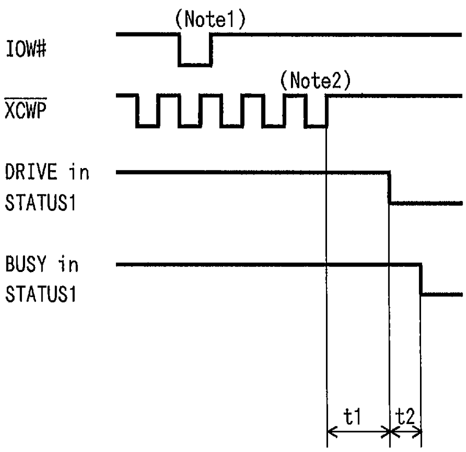


t1<33μs[41μs]
t2<20μs

Note: When the target is a stepping motor, DEND=0/1 does not matter.

12-7. SLOW STOP Timing

Example) Drive in +(CW)direction when the target is a stepping motor

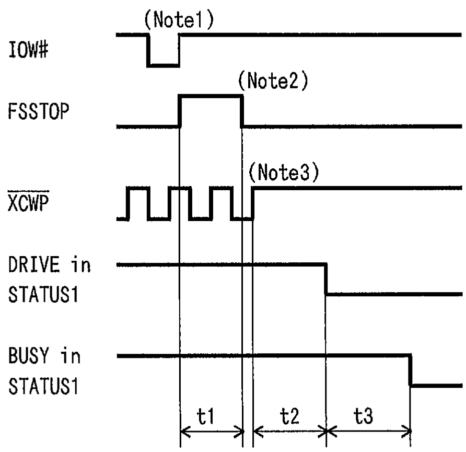


t1<38μs[42μs]
t2<20μs

Note1: Indicates that the SLOW STOP command is written.
Note2: The number of pulses to be output after receipt of the SLOW STOP command is 1 pulse or less at constant-speed DRIVE, or the number of pulses required for a slow stop at acceleration/decelerating DRIVE.

12-8. FAST STOP Timing(1)

Example) DRIVE in +(CW)direction when the target is a stepping motor

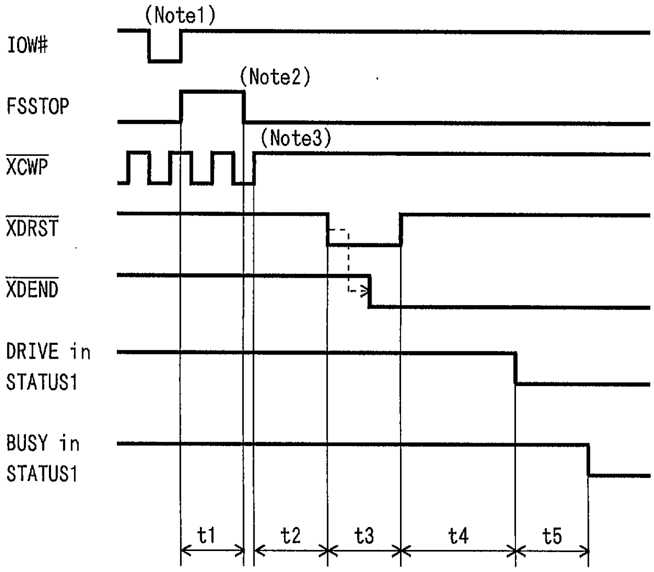


t1≧200μs
t2<38μs[46μs] or T(Note4)
t3<20μs

Note1: Indicates that the FAST STOP command is written.
Note2: Either command or signal may be used.
Note3: The number of pulses to be output after receipt of the FAST STOP command is 1 pulse or less. (The pulse width is secured.)
Note4: When T represents 1/2 of the pulse cycle as the drive is stopped, t2 takes the indicated value or T which ever is longer.

12-9. FAST STOP Timing(2)

Example) DRIVE in +(CW)direction when the target is a servo motor



t1≧200μs　　　t3≒10ms　　　t5<20μs

t2<26μs(Note4)　　　t4<15μs[24μs]($\overline{DEND}$ is assumed to be turned on while $\overline{DRST}$ is being produced.)

Note1: Indicates that the FAST STOP command is written.

Note2: Either command or signal may be used.

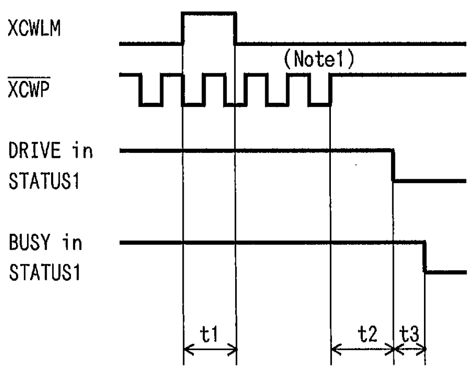
Note3: The number of pulses to be output after receipt of the FAST STOP command is 1 pulse or less. (The pulse width is secured.)

Note4: When T represents 1/2 of the pulse cycle as the drive is stopped, t2 takes the indicated value or T which ever is longer.

12-10. LIMIT STOP Timing

(1) When the LIMIT stop type is slow stop:

Example) DRIVE in +(CW)direction when the target is a stepping motor



t1≧400μs

t2<38μs[42μs]

t3<20μs

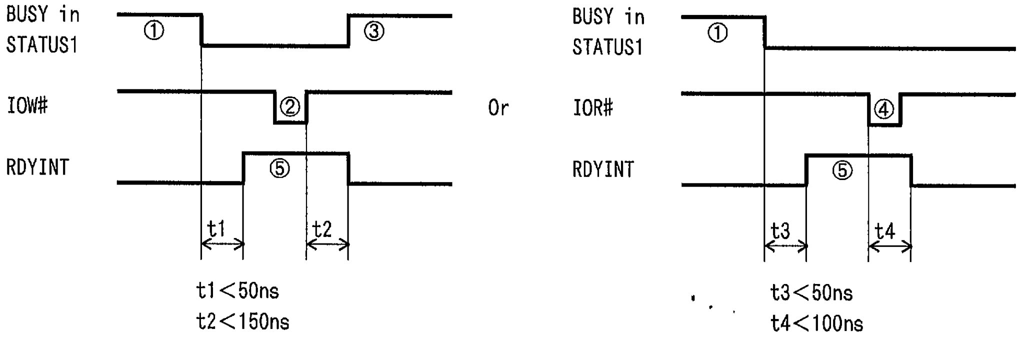
Note1: The number of pulses to be output after receipt of the LIMIT signal is 1 pulse or less at constant-speed DRIVE or the number of pulses required for slow stop at accelerating/decelerating DRIVE.

(2) When the LIMIT stop type is fast stop:

Complies with 12-8. or 12-9. Timing.

At this time, the FSSTOP signal is replaced with CWLM or CCWLM and the input signal width is set to 400μs.

12-11. RDYINT Timing

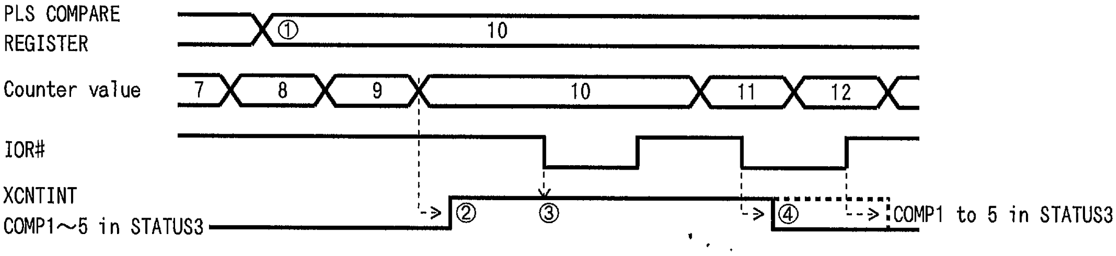


①: Indicates that a command is executed.
②: Indicates a new command is written.
③: Indicates that a new command is executed.
④: Indicates that STATUS1 is read out.
⑤: Indicates that an interrupt signal is generated. Interrupt signal enable or disable is specified by the SPEC INITIALIZE1 command. For details, refer to 6-4.

12-12. CNTINT Timing

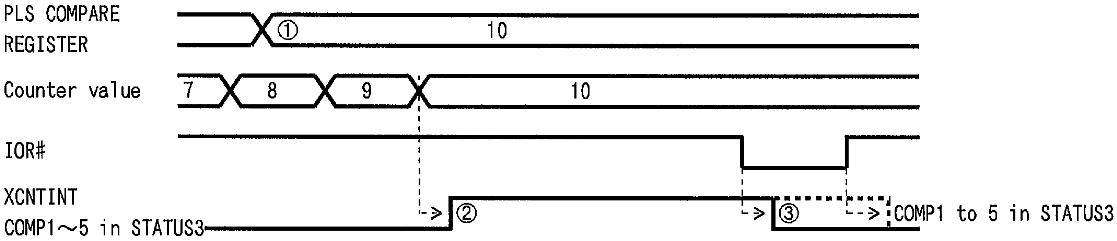
The interrupt request signal (XCNTINT) and STATUS signals (COMP1 to COMP5) are output and reset at the following timing.

(1) When the CNTINT LATCH TRIGGER TYPE=0 is specified by the PULSE COUNTER INITIALIZE Command.



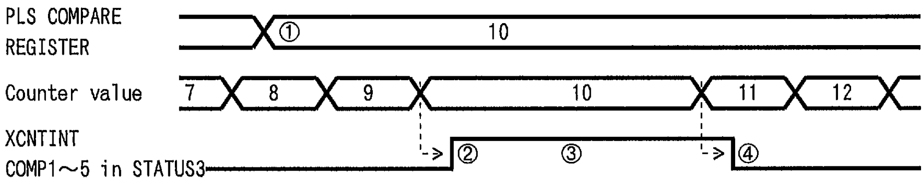
①: Indicates that the interrupt generation counter value is written by the COMPARABLE REGISTER SET command. (In the example, the interrupt generation counter value is set to 10.)
②: When the counter value reaches the value set in ①, a XCNTINT output is generated.
③: When COMPARABLE REGISTER matches the counter value,the XCNTINT output and STATUS signal are not cleared.
④: The XCNTINT output is cleared by STATUS3 PORT is accessed when COMPARABLE REGISTER not matches the counter value.

(2) When CNTINT LATCH TRIGGER TYPE=1 is specified by PULSE COUNTER INITIALIZE command.



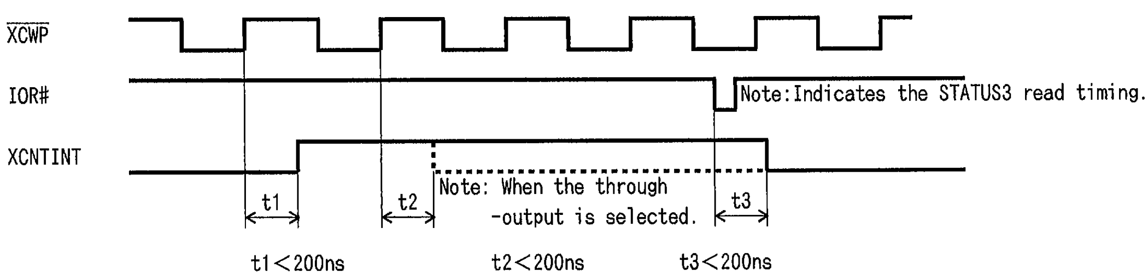
①: COMPARE REGISTER SET Command causes writing of the interrupt generating counter value.
②: As the counter value reaches the level specified in ① above, XCNTINT is output.
③: CNTINT output is maintained until the STATUS3 PORT is accessed (reading the STATUS3 PORT clears XCNTINT even when the counter value is in agreement with the COMPARE REGISTER).

(3) When CNTINT OUTPUT TYPE=1 (through-output) is specified by PULSE COUNTER INITIALIZE Command.



①: COMPARE REGISTER SET Command causes writing of the interrupt generating counter value.
②: As the counter value reaches the level specified in ① above, XCNTINT is output.
③: When the counter value is in agreement with the COMPARE REGISTER, XCNTINT is output.
④: When the counter value is not matching, XCNTINT is cleared without requiring access to the STATUS3 PORT.

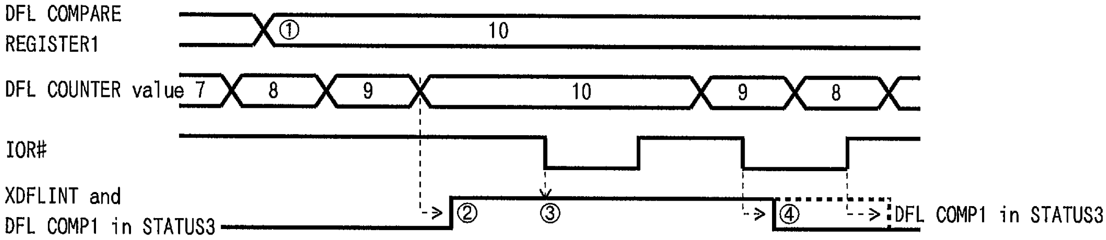
Example: When C-871 X axis drive pulse is used as the operation clock for the drive in +(CW) direction.



12-13. DFLINT Timing (DFL COMP1: An example. Detecting condition: DIFFERENTIAL COUNTER≧COMPARE REGISTER1)

The interrupt request signal(XDFLINT) and STATUS signal are produced and reset at the following timing.

(1) When the DFLINT LATCH TRIGGER TYPE=0 is specified by the DFL COUNTER INITIALIZE Command.



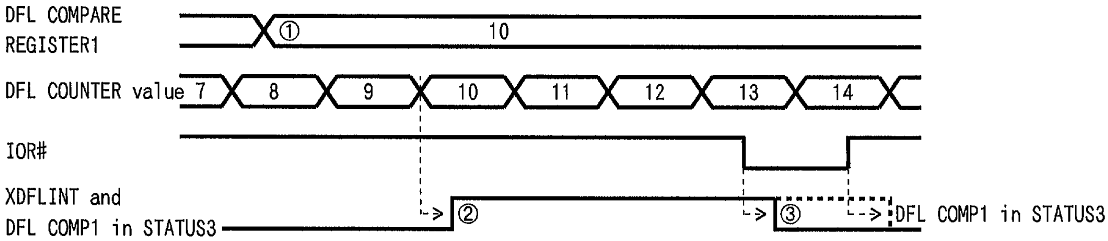
①: COMPARE REGISTER SET Command causes writing of the interrupt generating counter value.
(in this example, the interrupt generating counter value is set to 10).

②: As the counter value reaches the level specified in ① above, XDFLINT is output.

③: When the counter value is in agreement with the detection condition, accessing the STATUS3 PORT does not clear XDFLINT output.

④: When the counter value is not meeting the detect condition, reading the STATUS3 PORT clears XDFLINT.

(2) When the DFLINT LATCH TRIGGER TYPE=1 is specified by DFL COUNTER INITIALIZE Command.

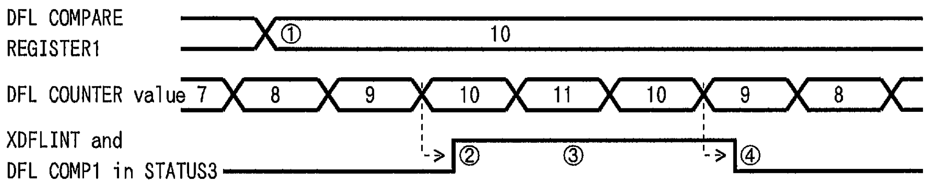


①: COMPARE REGISTER SET Command causes writing of the interrupt generating counter value.

②: As the counter value reaches the level specified in ① above, XDFLINT is output.

③: Output of XDFLINT is maintained until the STATUS3 PORT is accessed
(reading the STATUS3 PORT clears XDFLINT output even if the counter value meets the detect condition).

(3) When the DFLINT OUTPUT TYPE=1 (through-output) is specified by DFL COUNTER INITIALIZE Command.



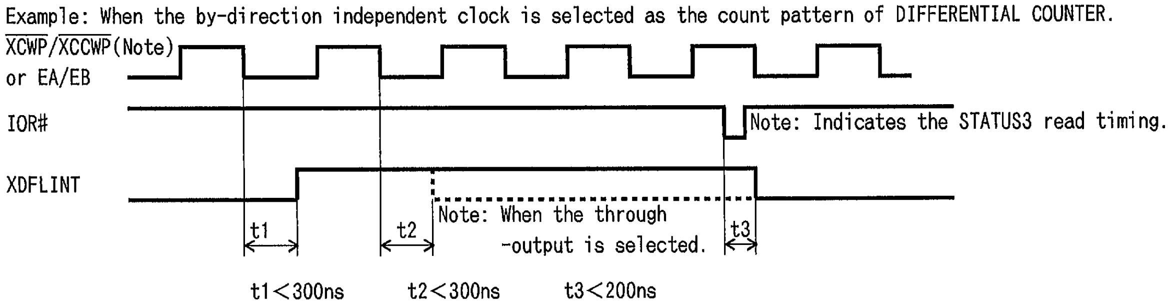
①: COMPARE REGISTER SET Command causes writing of the interrupt generating counter value.

②: As the counter value reaches the level specified in ① above, XDFLINT is output.

③: When the counter value is in agreement with the detect condition, XDFLINT is output.

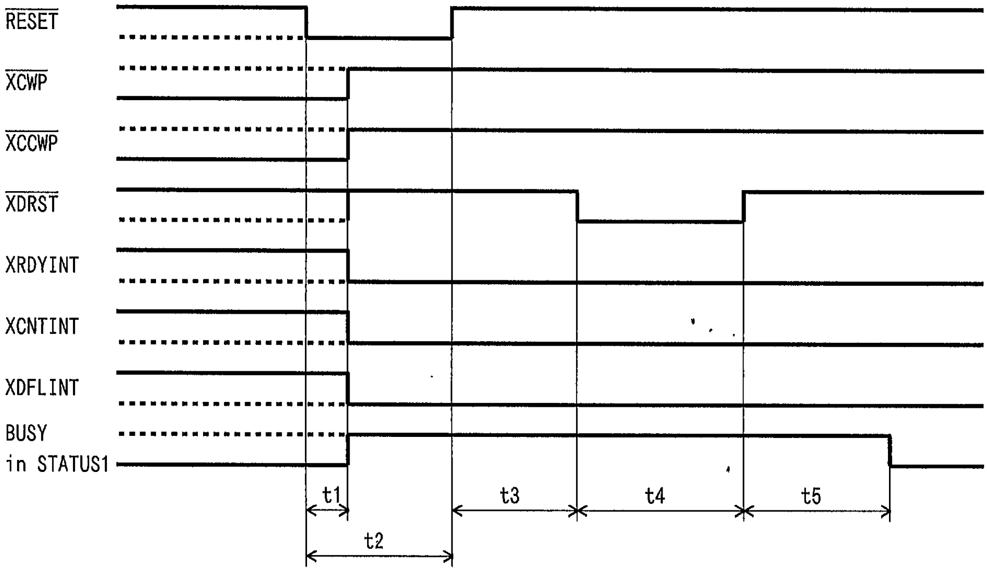
④: When the counter value is not matching, XDFLINT is cleared without requiring access to the STATUS3 PORT.

Example: When the by-direction independent clock is selected as the count pattern of DIFFERENTIAL COUNTER.



Note: Time from startup edge in case of XCWP/XCCWP.

12-14. RESET Timing

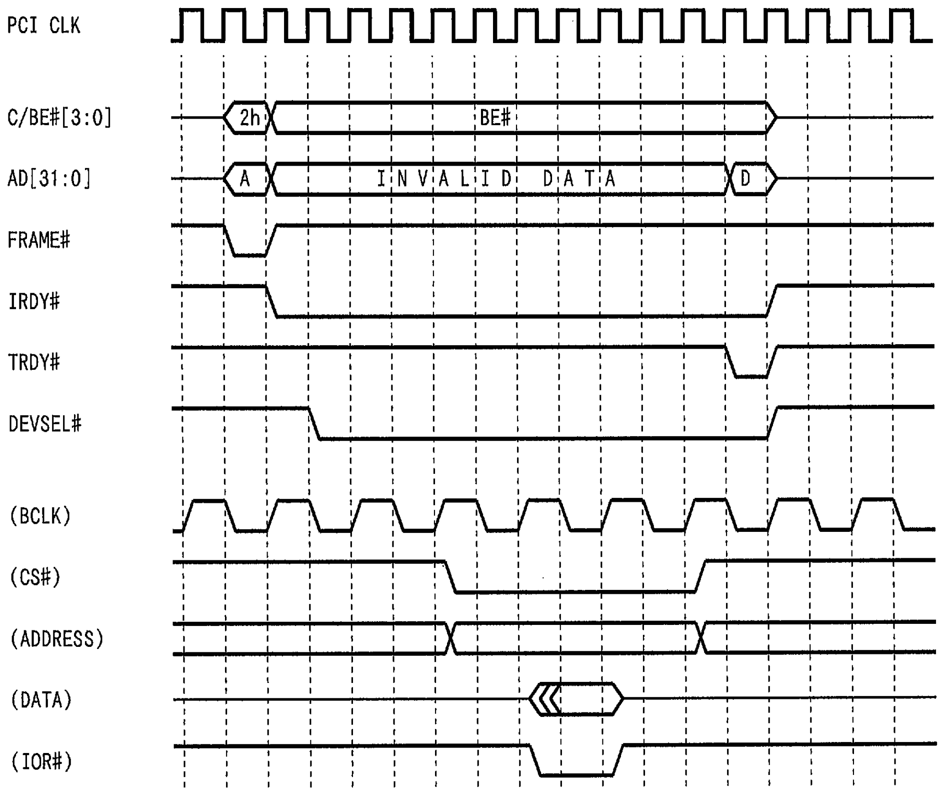


$t1 < 5ms$	(Note 500ns)	$t4 \fallingdotseq 10.6ms$
$t2 \geqq 5ms$	(Note 5.0μs)	$t5 < 5\mu s$
$t3 < 5ms$	(Note 50μs)	

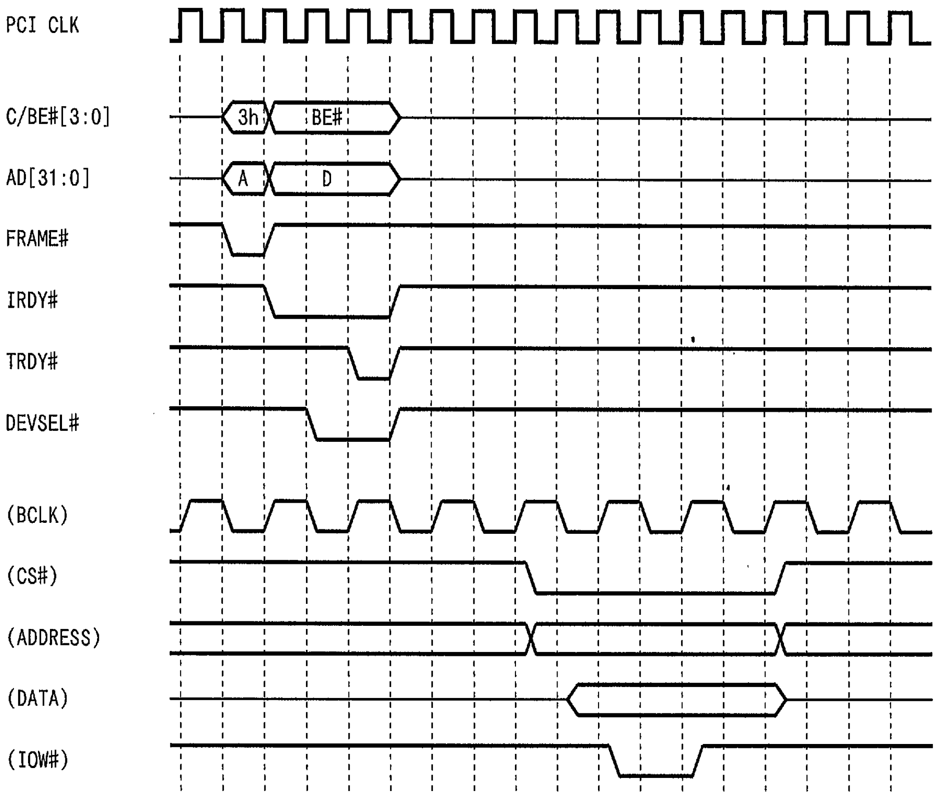
Note: When the system is reset on the PC side.

12-15. BUS Timing

"READ TIMING ()" denotes the internal timing.



"WRITE TIMING ()" denotes the internal timing.

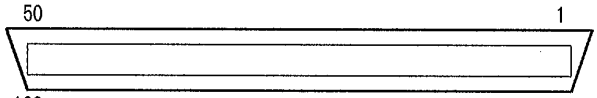


13. USER CONNECTOR AND I/O CIRCUIT

13-1. User Connector J1 Pin Arrangement

Connector Type DX10A-100S(HIROSE ELECTRIC CO,LTD)

Applicable outlets(not included in the accessories)
DX30A-100P,DX31A-100P etc.(HIROSE)



13-2. J1 Signal Table

Signals are insulated by the coupler though there are some exceptions(※).

Pin	Direction	Signal name	Description	Pin	Direction	Signal name	Description
1	input	XCWLM	X axis +(CW)direction LIMIT signal.	51	input	ZCWLM	Z axis +(CW)direction LIMIT signal.
2	input	XCCWLM	X axis -(CCW)direction LIMIT signal.	52	input	ZCCWLM	Z axis -(CCW)direction LIMIT signal.
3	input	$\overline{\text{XNORG}}$	X axis near-origin signal.	53	input	$\overline{\text{ZNORG}}$	Z axis near-origin signal.
4	input	$\overline{\text{XORG}}$	X axis origin signal. (Note1)	54	input	$\overline{\text{ZORG}}$	X axis origin signal. (Note1)
5	input	YCWLM	Y axis +(CW)direction LIMIT signal.	55	input	ACWLM	A axis +(CW)direction LIMIT signal.
6	input	YCCWLM	Y axis -(CCW)direction LIMIT signal.	56	input	ACCWLM	A axis -(CCW)direction LIMIT signal.
7	input	$\overline{\text{YNORG}}$	Y axis near-origin signal.	57	input	$\overline{\text{ANORG}}$	A axis near-origin signal.
8	input	$\overline{\text{YORG}}$	Y axis origin signal. (Note1)	58	input	$\overline{\text{AORG}}$	A axis origin signal. (Note1)
9	input	BCWLM	B axis +(CW)direction LIMIT signal.	59	input	CCWLM	C axis +(CW)direction LIMIT signal.
10	input	BCCWLM	B axis -(CCW)direction LIMIT signal.	60	input	CCCWLM	C axis -(CCW)direction LIMIT signal.
11	input	$\overline{\text{BNORG}}$	B axis near-origin signal.	61	input	$\overline{\text{CNORG}}$	C axis near-origin signal.
12	input	$\overline{\text{BORG}}$	B axis origin signal. (Note1)	62	input	$\overline{\text{CORG}}$	axis origin signal. (Note1)
13	input	$\overline{\text{ZSENSOR}}$	Z axis sensor signal for SENSOR INDEX Drive.(Note3)	63	input	$\overline{\text{ASENSOR}}$	A axis sensor signal for SENSOR INDEX Drive.(Note3)
14	—	EXTV	External power supply for coupler. (Note2)	64	—	EXTVGND	External power supply GND for coupler. (Note2)
15	—	EXTV		65	—	EXTVGND	
16	—	N.C	Using is disabled.	66	—	N.C	Using is disabled.

Note1: This origin signal is for the case where a stepping motor is used.
To use the Z phase signal of the encoder as an origin signal when a servo motor is used, never connect the above signal.

Note2: All signals are insulated by coupler HIC, so an external power supply is required.
The input voltage specification is +24V±2V and the current consumption is 270mA MAX(at +24v).
The CWLM, CCWLM and FSSTOP signals of each axis are ACTIVE OFF inputs.
Accordingly,even if all the above signals are not used, an external power supply must be connected.
For details, refer to 14-2.

Note3: For details of these $\overline{\text{SENSOR}}$ signals, refer to User's Manual [Applied Functions Part].

Pin	Direction	Signal name	Description	Pin	Direction	Signal name	Description
17	output	+COM ※	+COMMON for XCWP,XCCWP.(+5V)	67	output	+COM ※	+COMMON for ZCWP,ZCCWP.(+5V)
18	output	XCWP ※	X axis +(CW)direction positive logic output pulse.	68	output	ZCWP ※	Z axis +(CW)direction positive logic output pulse.
19	output	XCWP ※	X axis +(CW)direction negative logic output pulse.	69	output	ZCWP ※	Z axis +(CW)direction negative logic output pulse.
20	output	XCCWP ※	X axis -(CCW)direction positive logic output pulse.	70	output	ZCCWP ※	Z axis -(CCW)direction positive logic output pulse.
21	output	XCCWP ※	X axis -(CCW)direction negative logic output pulse.	71	output	ZCCWP ※	Z axis -(CCW)direction negative logic output pulse.
22	output	XDRSTCOM	+COMMON for XDRST. (+24V)	72	output	ZDRSTCOM	+COMMON for ZDRST. (+24V)
23	input	XDRST	X axis SERVO DRIVER RESET signal.	73	output	ZDRST	Z axis SERVO DRIVER RESET signal.
24	—	XDEND/ZPO	X axis positioning end signal or PO signal. (Note4)	74	input	ZDEND/ZPO	Z axis positioning end signal or PO signal. (Note4)
25	input	+XZORG	X axis encoder +Z phase signal	75	input	+ZZORG	Z axis encoder +Z phase signal
26	input	-XZORG	X axis encoder -Z phase signal	76	input	-ZZORG	Z axis encoder -Z phase signal
27	output	+COM ※	+COMMON for YCWP,YCCWP.(+5V)	77	output	+COM ※	+COMMON for ACWP,ACCWP.(+5V)
28	output	YCWP ※	Y axis +(CW)direction positive logic output pulse.	78	output	ACWP ※	A axis +(CW)direction positive logic output pulse.
29	output	YCWP ※	Y axis +(CW)direction negative logic output pulse.	79	output	ACWP ※	A axis +(CW)direction negative logic output pulse.
30	output	YCCWP ※	Y axis -(CCW)direction positive logic output pulse.	80	output	ACCWP ※	A axis -(CCW)direction positive logic output pulse.
31	output	YCCWP ※	Y axis -(CCW)direction negative logic output pulse.	81	output	ACCWP ※	A axis -(CCW)direction negative logic output pulse.
32	output	YDRSTCOM	+COMMON for YDRST. (+24V)	82	output	ADRSTCOM	+COMMON for ADRST. (+24V)
33	input	YDRST	Y axis SERVO DRIVER RESET signal.	83	output	ADRST	A axis SERVO DRIVER RESET signal.
34	—	YDEND/YPO	Y axis positioning end signal or PO signal. (Note4)	84	input	ADEND/APO	A axis positioning end signal or PO signal. (Note4)
35	input	+YZORG	Y axis encoder +Z phase signal	85	input	+AZORG	A axis encoder +Z phase signal
36	input	-YZORG	Y axis encoder -Z phase signal	86	input	-AZORG	A axis encoder -Z phase signal
37	output	+COM ※	+COMMON for BCWP,BCCWP.(+5V)	87	output	+COM ※	+COMMON for CCWP,CCCWP.(+5V)
38	output	BCWP ※	X axis +(CW)direction positive logic output pulse.	88	output	CCWP ※	C axis +(CW)direction positive logic output pulse.
39	output	BCWP ※	X axis +(CW)direction negative logic output pulse.	89	output	CCWP ※	C axis +(CW)direction negative logic output pulse.
40	output	BCCWP ※	X axis -(CCW)direction positive logic output pulse.	90	output	CCCWP ※	C axis -(CCW)direction positive logic output pulse.
41	output	BCCWP ※	X axis -(CCW)direction negative logic output pulse.	91	output	CCCWP ※	C axis -(CCW)direction negative logic output pulse.
42	output	BDRSTCOM	+COMMON for BDRST. (+24V)	92	output	CDRSTCOM	+COMMON for CDRST. (+24V)
43	input	BDRST	B axis SERVO DRIVER RESET signal.	93	output	CDRST	C axis SERVO DRIVER RESET signal.
44	—	BDEND/BPO	B axis positioning end signal or PO signal. (Note4)	94	input	CDEND/CPO	C axis positioning end signal or PO signal. (Note4)
45	input	+BZORG	B axis encoder +Z phase signal	95	input	+CZORG	C axis encoder +Z phase signal
46	input	-BZORG	B axis encoder -Z phase signal	96	input	-CZORG	C axis encoder -Z phase signal
47	input	FSSTOP	Using is disabled.	97	input	RESET	Using is disabled.
48	—	N.C	Fast Stop signal for all axis	98	—	N.C	RESET signal.
49	—	N.C	Using is disabled.	99	—	N.C	Using is disabled.
50	—	N.C	Using is disabled.	100	—	N.C	Using is disabled.

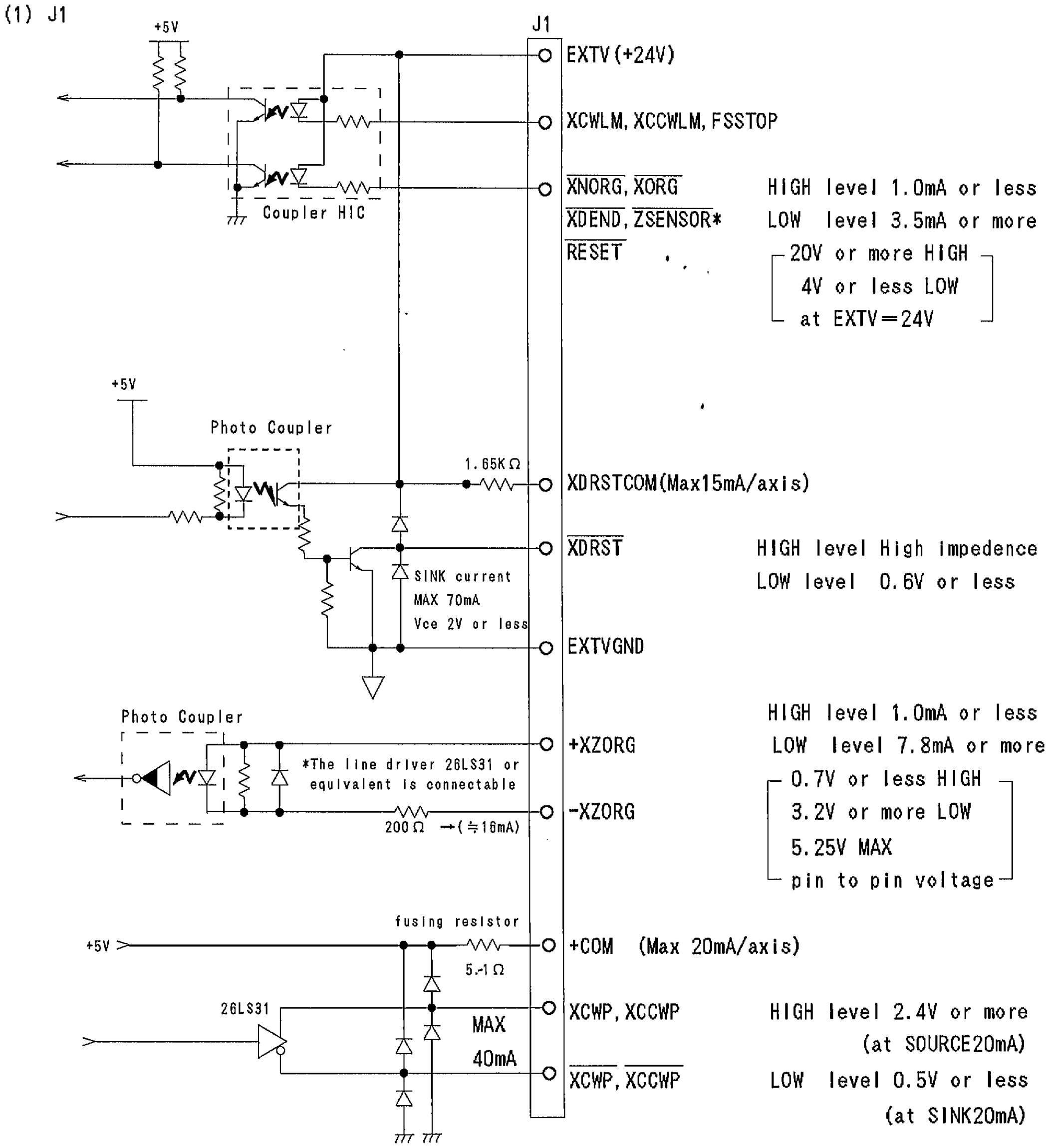
Note4: Each DEND/PO input is used as the positioning complete signal when a servo motor are selected, and they are used as the PO (excitation) signal when a stepping motor is selected.

13-3. I/O Circuit

(The following description is given about the X axis but also applicable to the another axis.)

*However, SENSOR signal is prepared for Z axis and A axis.

(1) J1



Note1: XDRSTCOM cannot use with when the XDRST is used for general-purpose output function. (It cannot be used as 24V +COM).

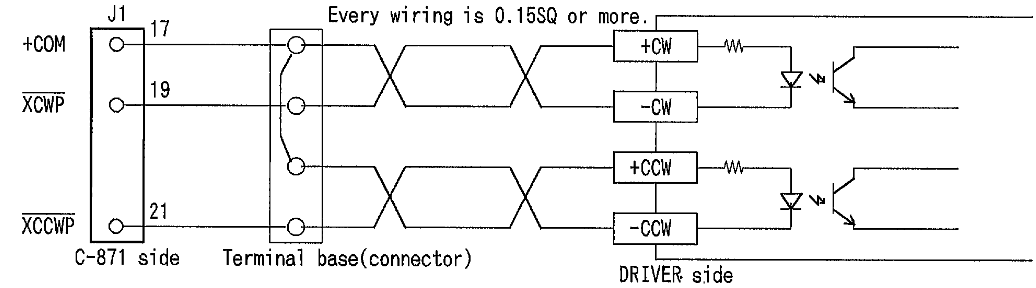
Note2: Each +COM line has a built-in fuse resistor. If it is short-circuited with the GND, the fuse will blow. A sufficient care must be taken to avoid this.

14. CONNECTIONS

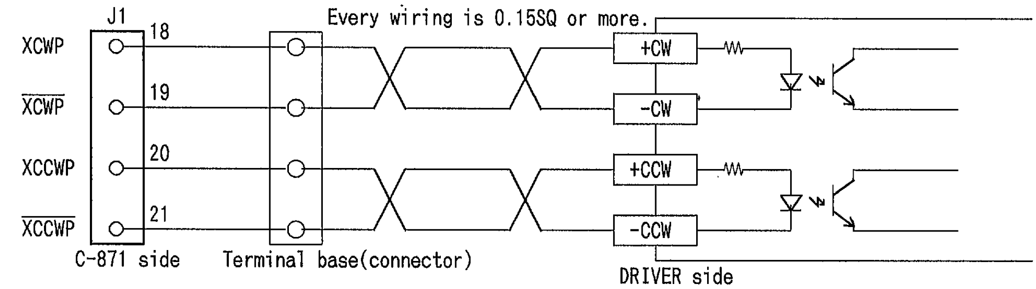
14-1. Connection to the Driver

(1) Connection example of the X axis to a negative logic pulse train CW/CCW independent input type driver.

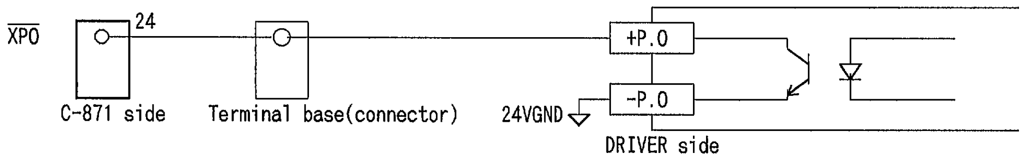
a. When using as the negative logic TTL level output:



b. When used as differential output (line driver):

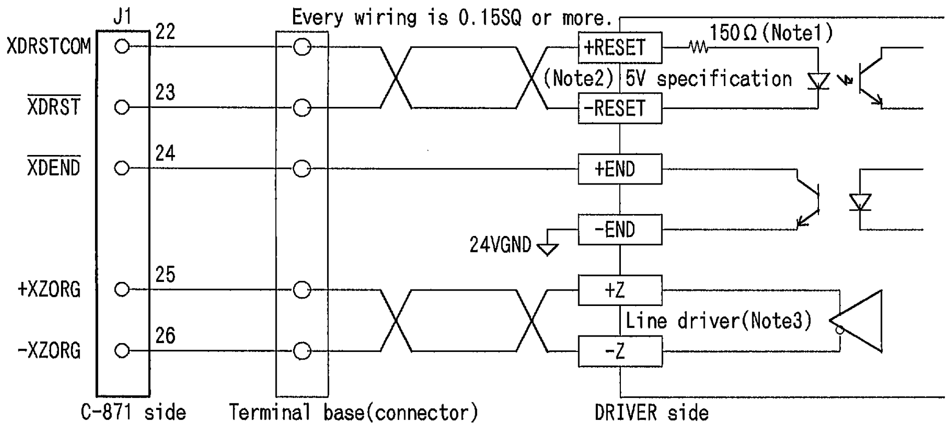


(2) Connection example of the X axis to a stepping motor.



When connecting to a stepping driver, $\overline{XDRST}$, +XZORG and -XZORG must be disconnected.
When excitation output is not used, $\overline{XPO}$ must be disconnected, too.

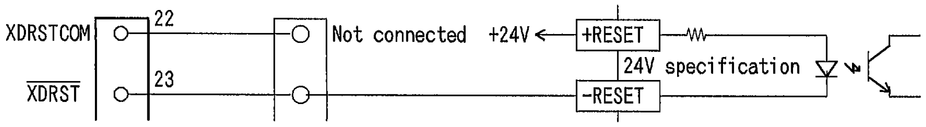
(3) Connection example of the X axis to a servo driver



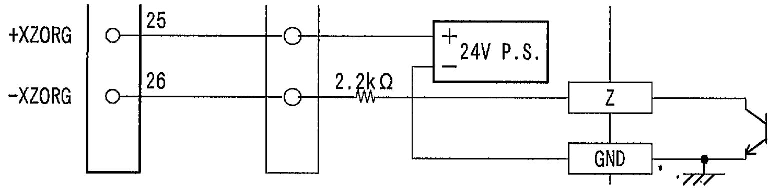
Note1: If current limiting resistor of the driver is less than 150Ω, provide an external resistor to ensure 150Ω or above.

Notes 2 and 3: See the following page.

Note2: Connection example when the servo driver counter reset input is +24V interface.

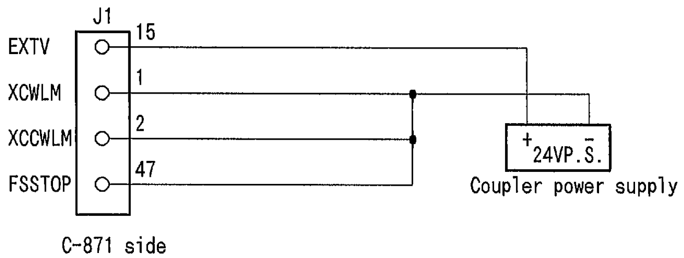


Note3: Connection example when the servo driver encoder Z phase output is open collector output.



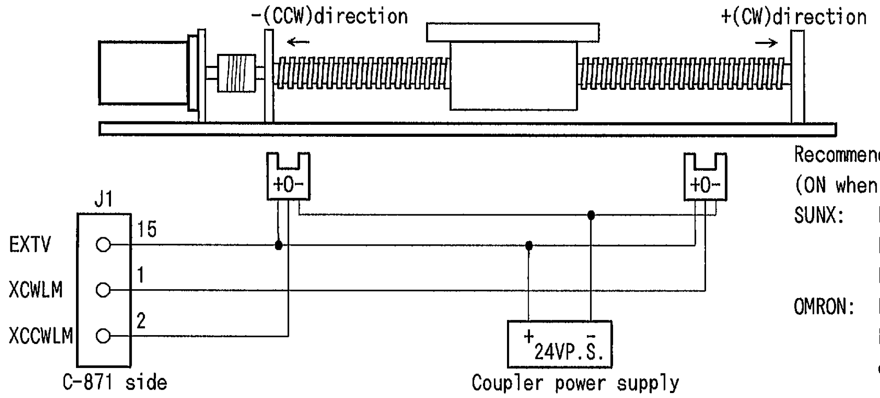
14-2. Connection Examples of a LIMIT Switch or Sensor

(1) Connection example of the X axis when LIMIT is unused



Note: The FSSTOP input signal and LIMIT input signal are an ACTIVE OFF input.
Note that when there signal are not connected,the signal becomes active and no pulse is output.

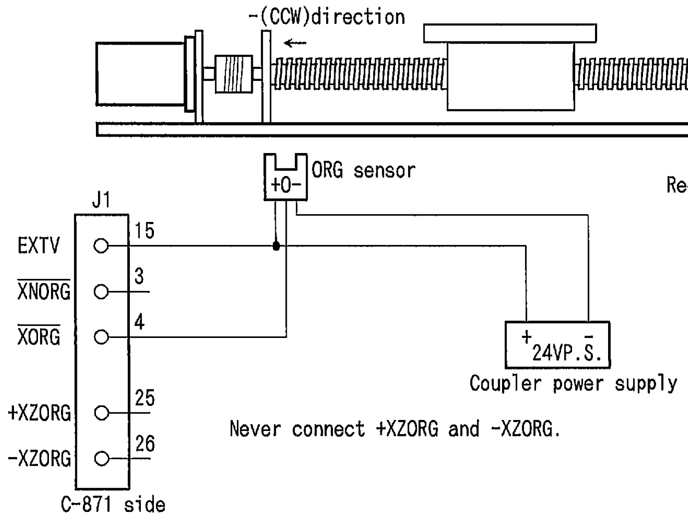
(2) Connection example of the X axis when LIMIT is used



Recommendable sensors
(ON when light enters)
SUNX: PM-K53B
PM-L53B
PM-T53B
OMRON: EE-SPX401
EE-SX670 series
etc.

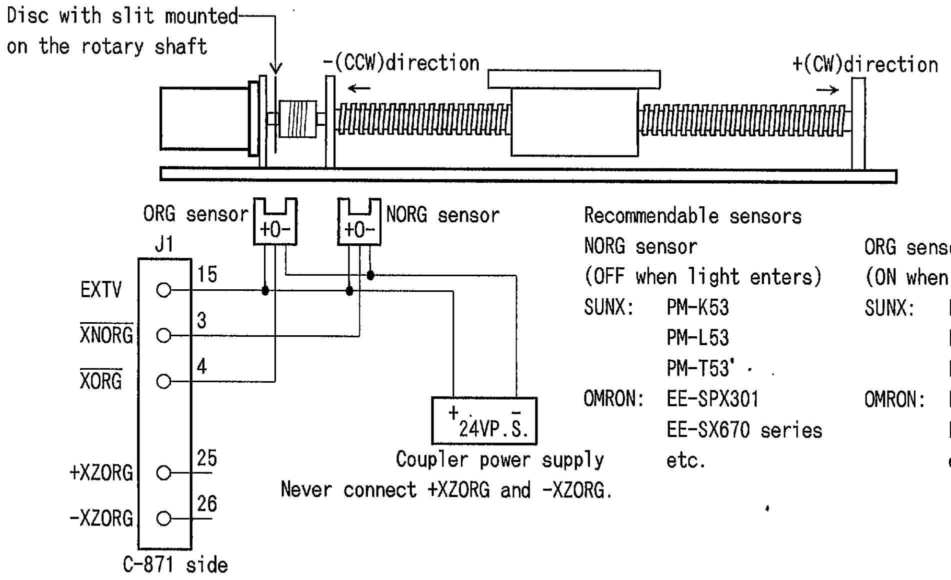
14-3. Connection Example of the Origin Sensor

(1) For ORG-0, ORG-1, ORG-2 and ORG-3 types (X axis)



Recommendable sensors(OFF when light enters)
SUNX: PM-K53
PM-L53
PM-T53
OMRON: EE-SPX301
EE-SX670 series
etc.

(2) For stepping driver ORG-4 and ORG-5 types (X axis)

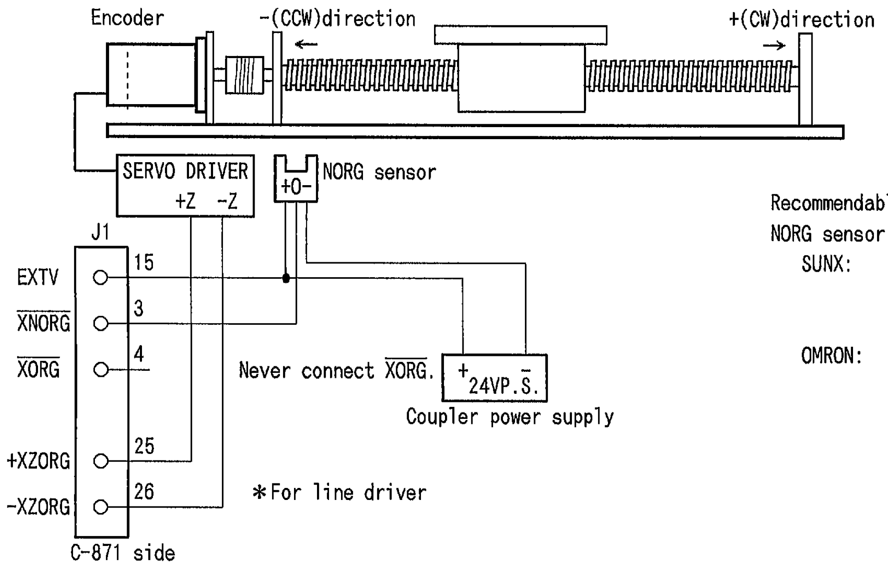


Recommendable sensors

NORG sensor
(OFF when light enters)
SUNX: PM-K53
PM-L53
PM-T53
OMRON: EE-SPX301
EE-SX670 series
etc.

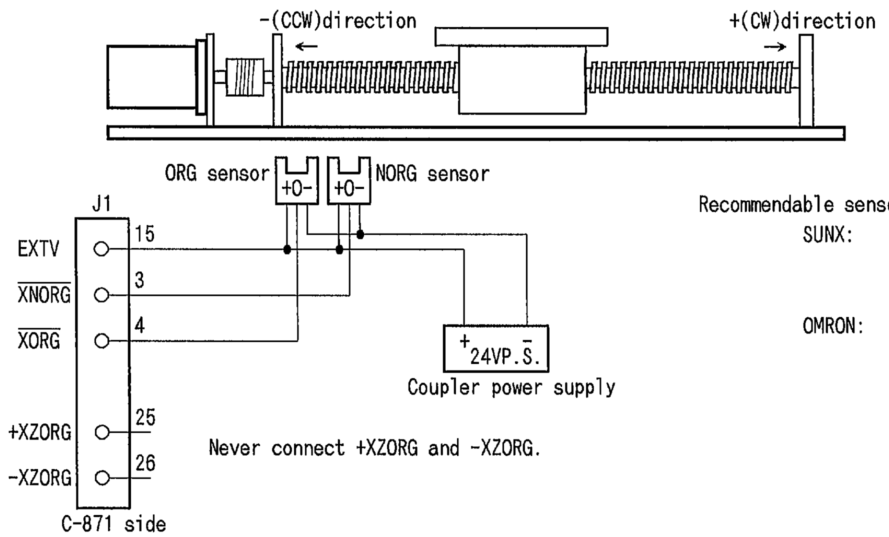
ORG sensor
(ON when light enters)
SUNX: PM-K53B
PM-L53B
PM-T53B
OMRON: EE-SPX401
EE-SX670 series
etc.

(3) For Servo driver ORG-4 and ORG-5 types (X axis)



Recommendable sensors
NORG sensor (OFF when light enters)
SUNX: PM-K53
PM-L53
PM-T53
OMRON: EE-SPX301
EE-SX670 series
etc.

(4) For ORG-10 type (X axis)



Recommendable sensors(OFF when light enters)
SUNX: PM-K53
PM-L53
PM-T53
OMRON: EE-SPX301
EE-SX670 series
etc.

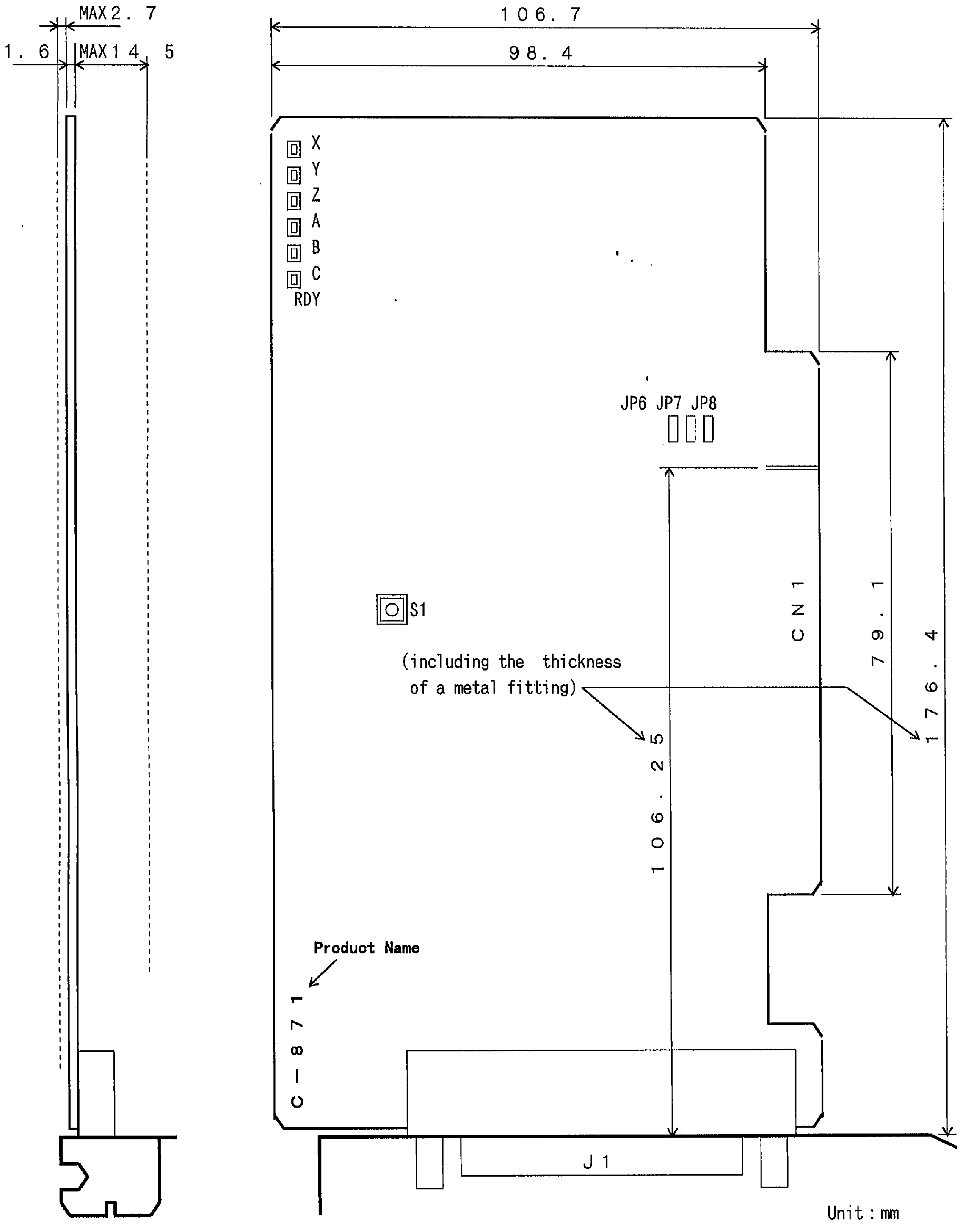
15. BOARD EDGE CONNECTOR AND BOARD SHAPE

15-1. Board Edge Connector Signal Table

No.	Name	No.	Name	No.	Name	No.	Name
A1	~~TRST#~~	A32	AD[16]	B1	~~-12V~~	B32	AD[17]
A2	~~+12V~~	A33	+3.3V	B2	~~TCK~~	B33	C/BE[2]#
A3	~~TMS~~	A34	FRAME#	B3	GND	B34	GND
A4	TDI	A35	GND	B4	TDO	B35	IRDY#
A5	+5V	A36	TRDY#	B5	+5V	B36	+3.3V
A6	INTA#	A37	GND	B6	+5V	B37	DEVSEL#
A7	~~INTC#~~	A38	STOP#	B7	~~INTB#~~	B38	GND
A8	+5V	A39	+3.3V	B8	~~INTD#~~	B39	LOCK#
A9	~~reserved~~	A40	~~SDONE~~	B9	PRSNT1#	B40	PERR#
A10	+5V(I/O)	A41	~~SBO#~~	B10	~~reserved~~	B41	+3.3V
A11	~~reserved~~	A42	GND	B11	PRSNT2#	B42	SERR#
A12	GND	A43	PAR	B12	GND	B43	+3.3V
A13	GND	A44	AD[15]	B13	GND	B44	C/BE[1]#
A14	~~reserved~~	A45	+3.3V	B14	~~reserved~~	B45	AD[14]
A15	RST#	A46	AD[13]	B15	GND	B46	GND
A16	+5V(I/O)	A47	AD[11]	B16	CLK	B47	AD[12]
A17	~~GNT#~~	A48	GND	B17	GND	B48	AD[10]
A18	GND	A49	AD[09]	B18	~~REQ#~~	B49	GND
A19	~~reserved~~	A50	Key	B19	+5V(I/O)	B50	Key
A20	AD[30]	A51	Key	B20	AD[31]	B51	Key
A21	+3.3V	A52	C/BE[0]#	B21	AD[29]	B52	AD[08]
A22	AD[28]	A53	+3.3V	B22	GND	B53	AD[07]
A23	AD[26]	A54	AD[06]	B23	AD[27]	B54	+3.3V
A24	GND	A55	AD[04]	B24	AD[25]	B55	AD[05]
A25	AD[24]	A56	GND	B25	+3.3V	B56	AD[03]
A26	IDSEL	A57	AD[02]	B26	C/BE[3]#	B57	GND
A27	+3.3V	A58	AD[00]	B27	AD[23]	B58	AD[01]
A28	AD[22]	A59	+5V(I/O)	B28	GND	B59	+5V(I/O)
A29	AD[20]	A60	~~REQ64#~~	B29	AD[21]	B60	~~ACK64#~~
A30	GND	A61	+5V	B30	AD[19]	B61	+5V
A31	AD[18]	A62	+5V	B31	+3.3V	B62	+5V

Note: Signals marked by "═══" are not connected on this board.
+3.3V and +5V(I/O) is not used, but connected to the decoupling condenser.

15-2. Board Shape and Dimensions



S1 ························· Board identifier rotary switch
JP6,JP7,JP8 ·············· Interrupt setting jumper
RDY X,Y,Z,A,B,C ········ RDY LED(Light on during ready)

* Jumpers and connectors except above have been reserved. So don't manipulate these things.

16. CONTROL PROGRAM EXAMPLES

This chapter offers an example of the user program (coded in C language conformed to the ANSI standard) used for controlling the C-871.

In the examples, base address of the C-871 I/O address have been asumed to be set at valiable "iobase".

```
/******************************/
/*        DEFINITION         */
/******************************/
#define UC      unsigned char
#define UL      unsigned long
#define US      unsigned short

#define XMCCCOM iobase+0x0              /* X-AXIS MCC05 COMMAND PORT    */
#define XMCCDT1 iobase+0x1              /* X-AXIS MCC05 DATA1 PORT      */
#define XMCCDT2 iobase+0x2              /* X-AXIS MCC05 DATA2 PORT      */
#define XMCCDT3 iobase+0x3              /* X-AXIS MCC05 DATA3 PORT      */
#define XCNTCOM iobase+0x4              /* X-AXIS COUNTER COMMAND PORT  */
#define XCNTDT1 iobase+0x5              /* X-AXIS COUNTER DATA1 PORT    */
#define XCNTDT2 iobase+0x6              /* X-AXIS COUNTER DATA2 PORT    */
#define XCNTDT3 iobase+0x7              /* X-AXIS COUNTER DATA3 PORT    */
#define XMCCST1 iobase+0x0              /* X-AXIS MCC05 STATUS1 PORT    */
#define XMCCST2 iobase+0x4              /* X-AXIS MCC05 STATUS2 PORT    */
#define XMCCST3 iobase+0x5              /* X-AXIS MCC05 STATUS3 PORT    */
#define XMCCST4 iobase+0x6              /* X-AXIS MCC05 STATUS4 PORT    */
#define XMCCST5 iobase+0x7              /* X-AXIS MCC05 STATUS5 PORT    */

#define YMCCCOM iobase+0x10             /* Y-AXIS MCC05 COMMAND PORT    */
#define YMCCDT1 iobase+0x11             /* Y-AXIS MCC05 DATA1 PORT      */
#define YMCCDT2 iobase+0x12             /* Y-AXIS MCC05 DATA2 PORT      */
#define YMCCDT3 iobase+0x13             /* Y-AXIS MCC05 DATA3 PORT      */
#define YCNTCOM iobase+0x14             /* Y-AXIS COUNTER COMMAND PORT  */
#define YCNTDT1 iobase+0x15             /* Y-AXIS COUNTER DATA1 PORT    */
#define YCNTDT2 iobase+0x16             /* Y-AXIS COUNTER DATA2 PORT    */
#define YCNTDT3 iobase+0x17             /* Y-AXIS COUNTER DATA3 PORT    */
#define YMCCST1 iobase+0x10             /* Y-AXIS MCC05 STATUS1 PORT    */
#define YMCCST2 iobase+0x14             /* Y-AXIS MCC05 STATUS2 PORT    */
#define YMCCST3 iobase+0x15             /* Y-AXIS MCC05 STATUS3 PORT    */
#define YMCCST4 iobase+0x16             /* Y-AXIS MCC05 STATUS4 PORT    */
#define YMCCST5 iobase+0x17             /* Y-AXIS MCC05 STATUS5 PORT    */

#define ZMCCCOM iobase+0x20             /* Z-AXIS MCC05 COMMAND PORT    */
#define ZMCCDT1 iobase+0x21             /* Z-AXIS MCC05 DATA1 PORT      */
#define ZMCCDT2 iobase+0x22             /* Z-AXIS MCC05 DATA2 PORT      */
#define ZMCCDT3 iobase+0x23             /* Z-AXIS MCC05 DATA3 PORT      */
#define ZCNTCOM iobase+0x24             /* Z-AXIS COUNTER COMMAND PORT  */
#define ZCNTDT1 iobase+0x25             /* Z-AXIS COUNTER DATA1 PORT    */
#define ZCNTDT2 iobase+0x26             /* Z-AXIS COUNTER DATA2 PORT    */
#define ZCNTDT3 iobase+0x27             /* Z-AXIS COUNTER DATA3 PORT    */
#define ZMCCST1 iobase+0x20             /* Z-AXIS MCC05 STATUS1 PORT    */
#define ZMCCST2 iobase+0x24             /* Z-AXIS MCC05 STATUS2 PORT    */
#define ZMCCST3 iobase+0x25             /* Z-AXIS MCC05 STATUS3 PORT    */
#define ZMCCST4 iobase+0x26             /* Z-AXIS MCC05 STATUS4 PORT    */
#define ZMCCST5 iobase+0x27             /* Z-AXIS MCC05 STATUS5 PORT    */
```

```
#define AMCCCOM iobase+0x30                    /* A-AXIS MCC05 COMMAND PORT    */
#define AMCCDT1 iobase+0x31                    /* A-AXIS MCC05 DATA1 PORT      */
#define AMCCDT2 iobase+0x32                    /* A-AXIS MCC05 DATA2 PORT      */
#define AMCCDT3 iobase+0x33                    /* A-AXIS MCC05 DATA3 PORT      */
#define ACNTCOM iobase+0x34                    /* A-AXIS COUNTER COMMAND PORT  */
#define ACNTDT1 iobase+0x35                    /* A-AXIS COUNTER DATA1 PORT    */
#define ACNTDT2 iobase+0x36                    /* A-AXIS COUNTER DATA2 PORT    */
#define ACNTDT3 iobase+0x37                    /* A-AXIS COUNTER DATA3 PORT    */
#define AMCCST1 iobase+0x30                    /* A-AXIS MCC05 STATUS1 PORT    */
#define AMCCST2 iobase+0x34                    /* A-AXIS MCC05 STATUS2 PORT    */
#define AMCCST3 iobase+0x35                    /* A-AXIS MCC05 STATUS3 PORT    */
#define AMCCST4 iobase+0x36                    /* A-AXIS MCC05 STATUS4 PORT    */
#define AMCCST5 iobase+0x37                    /* A-AXIS MCC05 STATUS5 PORT    */

#define BMCCCOM iobase+0x40                    /* B-AXIS MCC05 COMMAND PORT    */
#define BMCCDT1 iobase+0x41                    /* B-AXIS MCC05 DATA1 PORT      */
#define BMCCDT2 iobase+0x42                    /* B-AXIS MCC05 DATA2 PORT      */
#define BMCCDT3 iobase+0x43                    /* B-AXIS MCC05 DATA3 PORT      */
#define BCNTCOM iobase+0x44                    /* B-AXIS COUNTER COMMAND PORT  */
#define BCNTDT1 iobase+0x45                    /* B-AXIS COUNTER DATA1 PORT    */
#define BCNTDT2 iobase+0x46                    /* B-AXIS COUNTER DATA2 PORT    */
#define BCNTDT3 iobase+0x47                    /* B-AXIS COUNTER DATA3 PORT    */
#define BMCCST1 iobase+0x40                    /* B-AXIS MCC05 STATUS1 PORT    */
#define BMCCST2 iobase+0x44                    /* B-AXIS MCC05 STATUS2 PORT    */
#define BMCCST3 iobase+0x45                    /* B-AXIS MCC05 STATUS3 PORT    */
#define BMCCST4 iobase+0x46                    /* B-AXIS MCC05 STATUS4 PORT    */
#define BMCCST5 iobase+0x47                    /* B-AXIS MCC05 STATUS5 PORT    */

#define CMCCCOM iobase+0x50                    /* C-AXIS MCC05 COMMAND PORT    */
#define CMCCDT1 iobase+0x51                    /* C-AXIS MCC05 DATA1 PORT      */
#define CMCCDT2 iobase+0x52                    /* C-AXIS MCC05 DATA2 PORT      */
#define CMCCDT3 iobase+0x53                    /* C-AXIS MCC05 DATA3 PORT      */
#define CCNTCOM iobase+0x54                    /* C-AXIS COUNTER COMMAND PORT  */
#define CCNTDT1 iobase+0x55                    /* C-AXIS COUNTER DATA1 PORT    */
#define CCNTDT2 iobase+0x56                    /* C-AXIS COUNTER DATA2 PORT    */
#define CCNTDT3 iobase+0x57                    /* C-AXIS COUNTER DATA3 PORT    */
#define CMCCST1 iobase+0x50                    /* C-AXIS MCC05 STATUS1 PORT    */
#define CMCCST2 iobase+0x54                    /* C-AXIS MCC05 STATUS2 PORT    */
#define CMCCST3 iobase+0x55                    /* C-AXIS MCC05 STATUS3 PORT    */
#define CMCCST4 iobase+0x56                    /* C-AXIS MCC05 STATUS4 PORT    */
#define CMCCST5 iobase+0x57                    /* C-AXIS MCC05 STATUS5 PORT    */

void    xmcc05inz(void);
void    xjog(void);
void    xscan(void);
void    xabsindex(void);
void    xorg(void);
```

Frequently used MCC05v2 RDY check is coded as macroinstruction so that the program may be simplified.

```
#define xmccrdy() while(inp(XMCCST1) & 0x01)     /* X-AXIS MCC05v2 READY WAIT    */
#define ymccrdy() while(inp(YMCCST1) & 0x01)     /* Y-AXIS MCC05v2 READY WAIT    */
#define zmccrdy() while(inp(ZMCCST1) & 0x01)     /* Z-AXIS MCC05v2 READY WAIT    */
#define amccrdy() while(inp(AMCCST1) & 0x01)     /* A-AXIS MCC05v2 READY WAIT    */
#define bmccrdy() while(inp(BMCCST1) & 0x01)     /* B-AXIS MCC05v2 READY WAIT    */
#define cmccrdy() while(inp(CMCCST1) & 0x01)     /* C-AXIS MCC05v2 READY WAIT    */
```

Although the X axis is taken as an example in the following, the same applies to the Y, Z, A, B and C axes, too.
RAM area used in the program is defined as follows.

```
/******************************/
/*        RAM AREA            */
/******************************/
US      iobase;                         /* C-871 I/O BASE ADDRESS */
UC      urate;                          /* UP RATE No.          */
UC      drate;                          /* DOWN RATE No.        */
UL      lspd;                           /* LOW SPEED DATA       */
UL      hspd;                           /* HIGH SPEED DATA      */
UL      cspd;                           /* CONSTANT SPEED DATA  */
long    absdt;                          /* OBJECT ADDRESS DATA FOR INDEX DRIVE */
UC      orgno;                          /* ORG TYPE No.         */
UC      offset;                         /* OFFSET PULSE DATA    */
UC      ldelay;                         /* LIMIT DELAY TIME     */
UC      sdelay;                         /* SCAN DELAY TIME      */
UC      jdelay;                         /* JOG DELAY TIME       */
```

The program offered in this manual is for your reference only, thus you may not strictly it.

16-1. INITIALIZE Program Example

Execute this program at POWER ON/RESET as needed.
This program is developed based on the following specifications.

(1) DRIVE Specification

DRIVE TYPE=L, LIMIT STOP TYPE=fast stop, MOTOR TYPE=Stepping motor and RDYINT=Not output in any case shall be specified.

(2) PULSE COUNTER and Comparator Specification

The PULSE COUNTER is supposed to operate with drive pulse from the MCC05v2, and the COMPARE REGISTER1 output on the conferred matching is to output to the CNTINT. Address of the COMPARE REGISTER1 detection shall be 10000(2710_H), and the deceleration-to-stop shall be selected for the COMP STOP TYPE.

(3) Address Specification

Motor current address shall be defined as 1000($3E8_H$), and 1000($3E8_H$) shall also be preset for the pulse counter.

```
/*------------------------------------*/
/*     X-AXIS MCC05 INITIALIZE        */
/*------------------------------------*/
void    xmcc05inz( void )
{
        /** SPEC INITIALIZE1 COMMAND **/
        xmccrdy();                          /* X-AXIS MCC05 RDY WAIT */
        outp(XMCCDT1 ,0x28);                /* DRIVE SPEC DATA OUT */
        outp(XMCCCOM ,0x01);                /* SPEC INITIALIZE1 COMMAND OUT */

        /** PULSE COUNTER INITIALIZE COMMAND **/
        xmccrdy();                          /* X-AXIS MCC05 RDY WAIT */
        outp(XMCCDT1 ,0x01);                /* COUNTER SPEC DATA1 OUT */
        outp(XMCCDT2 ,0x20);                /* COUNTER SPEC DATA2 OUT */
        outp(XMCCDT3 ,0x00);                /* COUNTER SPEC DATA3 OUT */
        outp(XMCCCOM ,0x02);                /* PULSE COUNTER INITIALIZE COMMAND OUT */

        /** ADDRESS INITIALIZE COMMAND **/
        xmccrdy();                          /* X-AXIS MCC05 RDY WAIT */
        outp(XMCCDT1 ,0x00);                /* ADDRESS MSB OUT */
        outp(XMCCDT2 ,0x03);
        outp(XMCCDT3 ,0xe8);                /* ADDRESS LSB OUT */
        outp(XMCCCOM ,0x03);                /* ADDRESS INITIALIZE COMMAND OUT */

        /** COUNTER PRESET COMMAND **/
        outp(XCNTDT1 ,0x00);                /* COUNTER MSB OUT */
        outp(XCNTDT2 ,0x03);
        outp(XCNTDT3 ,0xe8);                /* COUNTER LSB OUT */
        outp(XCNTCOM ,0x00);                /* COUNTER PRESET COMMAND OUT */

        /** COUNTER REGISTER1 SET COMMAND **/
        outp(XCNTDT1 ,0x00);                /* COMPARE REGISTER1 MSB OUT */
        outp(XCNTDT2 ,0x27);
        outp(XCNTDT3 ,0x10);                /* COMPARE REGISTER1 LSB OUT */
        outp(XCNTCOM ,0x01);                /* COUNTER REGISTER1 SET COMMAND OUT */
}
```

Note: At POWER ON/RESET, above settings are all initialized their specific specification.
Thus, the above processing shall be done only when a modification is needed.
For details of the initial specifications, see Chapter 10.

16-2. JOG DRIVE Program Example

The JOG DRIVE does not need specific data, so you can directly turn on using the JOG DRIVE command.

```
/*----------------------------------*/
/*      X-AXIS +JOG DRIVE           */
/*----------------------------------*/
void    xjog( void )
{
        xmccrdy();                          /* X-AXIS MCC05v2 RDY WAIT */
        outp(XMCCCOM ,0x10);                /* +JOG COMMAND OUT */
}
```

16-3. SCAN DRIVE Program Example

The SCAN DRIVE requires URATE, DRATE, LSPD and HSPD data. You must set these data prior to the drive. Once set, these rate and speed data remain valid until a change is needed of them.

```
/*----------------------------------*/
/*      X-AXIS SCAN DRIVE           */
/*----------------------------------*/
void    xscan( void )
{
        /** RATE SET COMMAND **/
        xmccrdy();                          /* X-AXIS MCC05 RDY WAIT */
        outp(XMCCDT2 ,urate);               /* UP RATE No. OUT */
        outp(XMCCDT3 ,drate);               /* DOWN RATE No. OUT */
        outp(XMCCCOM ,0x06);                /* RATE SET COMMAND OUT */

        /** LSPD SET COMMAND **/
        xmccrdy();                          /* X-AXIS MCC05 RDY WAIT */
        outp(XMCCDT1 ,*((UC *)&lspd + 2));  /* LOW SPEED DATA MSB SET */
        outp(XMCCDT2 ,*((UC *)&lspd + 1));
        outp(XMCCDT3 ,*((UC *)&lspd ));     /* LOW SPEED DATA LSB SET */
        outp(XMCCCOM ,0x07);                /* LSPD SET COMMAND OUT */

        /** HSPD SET COMMAND **/
        xmccrdy();                          /* X-AXIS MCC05 RDY WAIT */
        outp(XMCCDT1 ,*((UC *)&hspd + 2));  /* HIGH SPEED DATA MSB SET */
        outp(XMCCDT2 ,*((UC *)&hspd + 1));
        outp(XMCCDT3 ,*((UC *)&hspd ));     /* HIGH SPEED DATA LSB SET */
        outp(XMCCCOM ,0x08);                /* HSPD SET COMMAND OUT */

        /** SCAN DRIVE COMMAND **/
        xmccrdy();                          /* X-AXIS MCC05 RDY WAIT */
        outp(XMCCCOM ,0x12);                /* +SCAN DRIVE COMMAND OUT */
}
```

Note: The above program is developed on the assumption that the RAM AREA URATE and DRATE contain the RATE DATA TABLE No., and the LSPD and HSPD contain the speed data set in Hz.

16-4. Example of INDEX DRIVE Program Specified in Absolute Value

The INDEX DRIVE specified in absolute value requires URATE, DRATE, LSPD and HSPD data. You must set these data prior to the drive. Once set, these rate and speed data remain valid until a change is needed of them. And, the target address of the drive must be set before turning on the INDEX DRIVE.
Whenever starting the drive, the address data must be specified.

```
/*-----------------------------------*/
/*      X-AXIS ABSOLUTE INDEX DRIVE  */
/*-----------------------------------*/
void    xabsindex( void )
{
        /** RATE SET COMMAND **/
        xmccrdy();                          /* X-AXIS MCC05 RDY WAIT */
        outp(XMCCDT2 ,urate);               /* UP RATE No. OUT */
        outp(XMCCDT3 ,drate);               /* DOWN RATE No. OUT */
        outp(XMCCCOM ,0x06);                /* RATE SET COMMAND OUT */

        /** LSPD SET COMMAND **/
        xmccrdy();                          /* X-AXIS MCC05 RDY WAIT */
        outp(XMCCDT1 ,*((UC *)&lspd + 2));  /* LOW SPEED DATA MSB SET */
        outp(XMCCDT2 ,*((UC *)&lspd + 1));
        outp(XMCCDT3 ,*((UC *)&lspd ));     /* LOW SPEED DATA LSB SET */
        outp(XMCCCOM ,0x07);                /* LSPD SET COMMAND OUT */

        /** HSPD SET COMMAND **/
        xmccrdy();                          /* X-AXIS MCC05 RDY WAIT */
        outp(XMCCDT1 ,*((UC *)&hspd + 2));  /* HIGH SPEED DATA MSB SET */
        outp(XMCCDT2 ,*((UC *)&hspd + 1));
        outp(XMCCDT3 ,*((UC *)&hspd ));     /* HIGH SPEED DATA LSB SET */
        outp(XMCCCOM ,0x08);                /* HSPD SET COMMAND OUT */

        /** ABSOLUTE INDEX DRIVE COMMAND **/
        xmccrdy();                          /* X-AXIS MCC05 RDY WAIT */
        outp(XMCCDT1 ,*((UC *)&absdt + 2)); /* ABS INDEX DATA MSB SET */
        outp(XMCCDT2 ,*((UC *)&absdt + 1));
        outp(XMCCDT3 ,*((UC *)&absdt ));    /* ABS INDEX DATA LSB SET */
        outp(XMCCCOM ,0x15);                /* ABS INDEX DRIVE COMMAND OUT */
}
```

Note: The above program is developed on the assumption that the RAM AREA URATE and DRATE contain the RATE DATA TABLE No., and the LSPD and HSPD contain the speed data set in Hz.
And, the absdt is supposed to contain the target address.

16-5. ORIGIN DRIVE Program Example

The ORIGIN DRIVE requires data on URATE, DRATE, LSPD, HSPD, CSPD, OFFSET PULSE, LDELAY, SDELAY and JDELAY. You must set these data prior to the drive. Once set, these data remain valid until a change is needed of them.

Also, you must specify the machine origin detect type before turning on the ORIGIN DRIVE.

This data is needed whenever starting the drive.

```
/*-----------------------------------*/
/*      X-AXIS ORIGIN DRIVE          */
/*-----------------------------------*/
void    xorg( void )
{
        /** RATE SET COMMAND **/
        xmccrdy();                              /* X-AXIS MCC05 RDY WAIT */
        outp(XMCCDT2 ,urate);                   /* UP RATE No. OUT */
        outp(XMCCDT3 ,drate);                   /* DOWN RATE No. OUT */
        outp(XMCCCOM ,0x06);                    /* RATE SET COMMAND OUT */

        /** LSPD SET COMMAND **/
        xmccrdy();                              /* X-AXIS MCC05 RDY WAIT */
        outp(XMCCDT1 ,*((UC *)&lspd + 2));      /* LOW SPEED DATA MSB SET */
        outp(XMCCDT2 ,*((UC *)&lspd + 1));
        outp(XMCCDT3 ,*((UC *)&lspd ));         /* LOW SPEED DATA LSB SET */
        outp(XMCCCOM ,0x07);                    /* LSPD SET COMMAND OUT */

        /** HSPD SET COMMAND **/
        xmccrdy();                              /* X-AXIS MCC05 RDY WAIT */
        outp(XMCCDT1 ,*((UC *)&hspd + 2));      /* HIGH SPEED DATA MSB SET */
        outp(XMCCDT2 ,*((UC *)&hspd + 1));
        outp(XMCCDT3 ,*((UC *)&hspd ));         /* HIGH SPEED DATA LSB SET */
        outp(XMCCCOM ,0x08);                    /* HSPD SET COMMAND OUT */

        /** CSPD SET COMMAND **/
        xmccrdy();                              /* X-AXIS MCC05 RDY WAIT */
        outp(XMCCDT1 ,*((UC *)&cspd + 2));      /* CONSTANT SPEED DATA MSB SET */
        outp(XMCCDT2 ,*((UC *)&cspd + 1));
        outp(XMCCDT3 ,*((UC *)&cspd ));         /* CONSTANT SPEED DATA LSB SET */
        outp(XMCCCOM ,0x1a);                    /* CSPD SET COMMAND OUT */

        /** OFFSET PULSE SET COMMAND **/
        xmccrdy();                              /* X-AXIS MCC05 RDY WAIT */
        outp(XMCCDT3 ,offset);                  /* OFFSET PULSE DATA OUT */
        outp(XMCCCOM ,0x1b);                    /* OFFSET PULSE SET COMMAND OUT */

        /** ORG DELAY SET COMMAND **/
        xmccrdy();                              /* X-AXIS MCC05 RDY WAIT */
        outp(XMCCDT1 ,ldelay);                  /* LIMIT DELAY TIME OUT */
        outp(XMCCDT2 ,sdelay);                  /* SCAN DELAY TIME OUT */
        outp(XMCCDT3 ,jdelay);                  /* JOG DELAY TIME OUT */
        outp(XMCCCOM ,0x1c);                    /* ORG DELAY SET COMMAND OUT */

        /** ORIGIN DRIVE COMMAND **/
        xmccrdy();                              /* X-AXIS MCC05 RDY WAIT */
        outp(XMCCDT1 ,orgno);                   /* ORIGIN TYPE No. OUT */
        outp(XMCCCOM ,0x1e);                    /* ORIGIN DRIVE COMMAND OUT */
}
```

Note: The above program is developed on the assumption that the RAM AREA URATE and DRATE contain the RATE DATA TABLE No., the LSPD and HSPD contain the speed data set in Hz and the OFFSET contains the offset pulse number.

Likewise, the LDELAY, SDELAY and JDELAY are supposed to contain respective delay time, and the ORGNO is supposed to contain the machine origin detection type.

16-6. PULSE COUNTER Read Program Example

The following offers an example of the function that utilizes the PULSE COUNTER counting as the return value.

```
/*-----------------------------------*/
/*      COUNTER READ                 */
/*-----------------------------------*/
long    xcntred( void )
{
        long    a;

        outp(XMCCCOM ,0xfc);                    /* PULSE COUNTER PORT SELECT COMMAND OUT */

        *( (UC *)&a + 2 ) = inp(XMCCDT1);       /* COUNTER MSB IN */
        *( (UC *)&a + 1 ) = inp(XMCCDT2);
        *( (UC *)&a ) = inp(XMCCDT3);           /* COUNTER LSB IN */
        if( (*( (UC *)&a + 2 ) & 0x80 ) != 0 )  /* SIGN BIT ON ? */
        {
                *( (UC *)&a + 3) = 0xff;
        } else {
                *( (UC *)&a + 3) = 0x00;
        }
        return( a );
}
```

Note: The PULSE COUNTER PORT SELECT is valid only when other counter ports or the speed port is selected.
It is not needed when the PULSE COUNTER PORT is already selected.

17. TROUBLESHOOTING

This chapter describes probable types of trouble and check points for them in using the C-871.

	Symptom	Check point
1	*BUSY BIT in STATUS1 never becomes 0. Or BUSY BIT in STATUS1 does not become 1 after a command is written.	*Isn't LOW LEVEL input to the RESET signal? *Is I/O ADDRESS(S1 and S2) setting is correct?
2	*Access seems to have been made normally. However, even if a pulse output command is written, pulse output is not perform. At this time, both DRIVE BIT and BUSY BIT in STATUS1 are 0.	*Isn't the output pulse 0 INDEX DRIVE? (e.g., when the specified absolute address is the current position) *Check each of ERROR, LSEND and FSEND BITS in STATUS1. For details, refer to 4-8.
3	*Access seems to have been made normally. However, when a pulse output command is written, pulse output is not performed. At this time, both DRIVE BIT and BUSY BIT in STATUS1 are 1.	*Refer to the check point in 2. *Isn't HIGH LEVEL input to the DEND signal by SERVO specification?
4	*Pulse output was started but never terminated.	*Isn't SCAN, SPECIAL SCAN or ORG DRIVE performed? *In case of INDEX DRIVE: When INCREMENTAL is specified The specified number of pulses is large. When ABSOLUTE is specified The specified address is remote. The above causes are probable. In this case, DRIVE will stop before long.
5	*Pulse output was terminated. However, BUSY BIT in STATUS1 does not become 0.	*Is SERVO MOTOR set? Isn't the DEND signal at HIGH LEVEL? BUSY BIT in STATUS1 becomes 0 by causing this signal to go to the low level.
6	*Machine origin detection (ORG DRIVE) cannot be performed normally. Or machine origin detection is never terminated.	*Is the sensor logic (ON when light enters or OFF when light enters) correct? *Is sensor connection (specially GND line) correct? *Isn't edge 1 formed in the CCWLM area because the shield plate is too long in case of the ORG1 or ORG3 type? *In case of the ORG2,ORG3,ORG4 or ORG5 type , be careful about an effect of mechanical vibration. If any vibration exists, use ORG0 or ORG1, or make LD,SD and JD longer by the ORIGIN DELAY SET command. *When SERVO MOTOR is set, check the DEND signal for each process. Accordingly, if DEND is not returned, it will stop at a process on the way. *When ORG3 or ORG5 is selected to complete ORG DRIVE in the ORG sensor, the sensor may be turned off by slight mechanical vibration because it enters only 1 pulse from the sensor edge a into the sensor area when ORG DRIVE is completed. In this case,make a correction so that INDEX DRIVE may be performed several pulses in the +(CW)direction after ORG DRIVE is completed with the result that it can enter surely into the sensor area.

	Symptom	Check point
7	*When the counter values of the pulse counter are always read out, some counter values seemed to be wrong.	*Are the counter values read in the order of high-order bytes ($2^{23}\sim2^{16}$) to low-order bytes ($2^{7}\sim2^{0}$) ? Unless the PULSE COUNTER is read out starting from high-order bytes, the counter value may become wrong. *In order to implement an optimized compilation, some compilers may not compile data in the sequence provided in the source list. In such case, the optimum compilation function shall be canceled. When C language is used, see Chapter 16, too.
8	*Sometimes incorrect speed data is indicated.	*Are the speed data read in the order of high-order bytes ($2^{23}\sim2^{16}$) to low-order bytes ($2^{7}\sim2^{0}$) ? Unless the speed data is read out starting from high-order bytes, the counter value may become wrong. *Aren't you trying to read an extremely low speed whose data length exceeds 3 bytes? Note that an extra low speed at or less than approximately 9.5 Hz cannot be read.
9	*The CNTINT interrupt seems to be generated at a counter value different from the set value.	*Does the counter value of the PULSE COUNTER overflow because there is any PLS COMPARE REGISTER where data is not set yet? PLS COMPARE REGISTER is initialized to the overflow value of 800000_H at POWER ON/RESET, so the CNTINT signal is generated at the overflow value if there is any PLS COMPARE REGISTER where data is not set yet. Put the unused COMP INT of COMPARE REGISTER into a disabled state by the PULSE COUNTER INITIALIZE command.
10	*Output pulse speed deviates from the specified value.	*In the high speed area, there can be a conflict between the actual and specified speeds. For details, see Section 5-16.
11	*The acceleration/deceleration constant seems to be different from the set URATE/DRATE value.	*Is DRIVE TYPE different from the contents of the specified data? Note that the contents of the data set in RATE differ depending on the DRIVE TYPE selected by the SPEC INITIALIZE1.

18. DATA TABLES

18-1. L-TYPE RATE DATA TABLE

No.	ms/1000Hz
0	1000
1	800
2	600
3	500
4	400
5	300
6	200
7	150
8	125
9	100
10	75
11	50
12	30
13	20
14	15
15	10
16	7.5
17	5.0
18	4.0
19	2.0
20	1.5
21	1.0

18-2. M-TYPE RATE DATA TABLE

No.	ms/1000Hz
0	50
1	20
2	15
3	10
4	7.5
5	5.0
6	3.0
7	1.5
8	1.0
·9	0.5
10	0.3
11	0.2
12	0.1
13	0.075
14	0.05

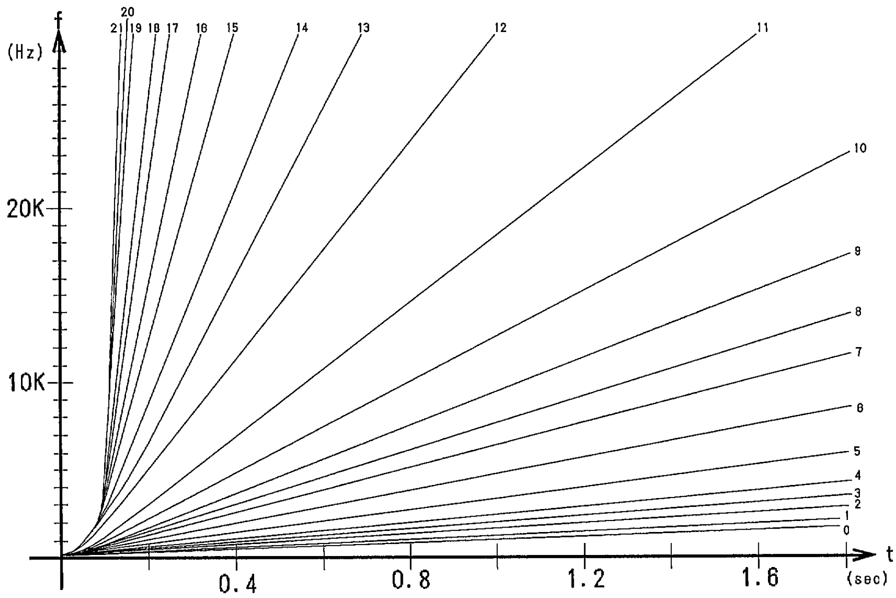
18-3. H-TYPE RATE DATA TABLE

No.	ms/1000Hz
0	5.0
1	2.0
2	1.5
3	1.0
4	0.75
5	0.50
6	0.30
7	0.15
8	0.10
9	0.05
10	0.03
11	0.02
12	0.01
13	0.0075
14	0.005

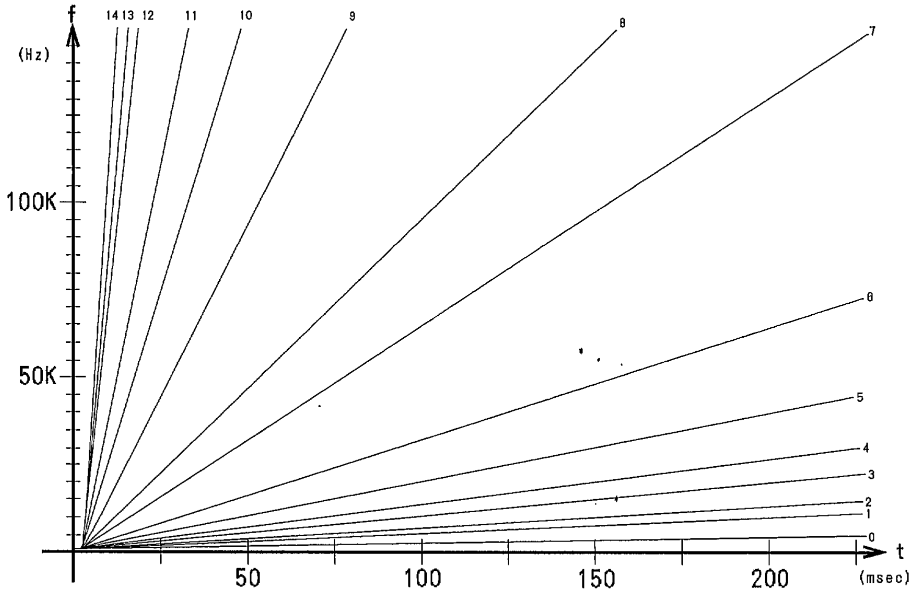
Note: "ms/1000Hz" is the mean time required for the acceleration or deceleration of 1000Hz.

18-4. RATE CURVE GRAPH

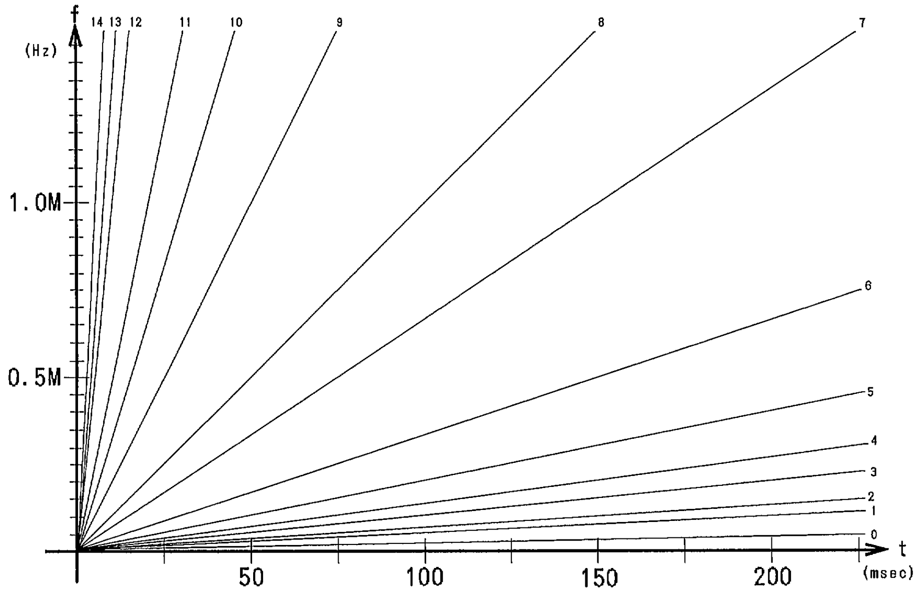
(1) L－TYPE



(2) M－TYPE



(3) H－TYPE



19. SAFETY DESIGN PRECAUTIONS

In order to ensure safety of the user system employing the C-871, the users are advised to develop suitable safety measures considering vulnerability of the given system and reliability of the following possible actions.

The C-871 or the system employing the C-871 **may fail to stop pulse output** due so a reason or another (so called run-away).
The following table rates reliability of each possible measure in preventing the pulse output in such case.

1	Cutting off the drive system power supply	The most safe and sure approach.
2	Entry of the $\overline{\text{RESET}}$ signal.	Entering the $\overline{\text{RESET}}$ to the C-871 alone can save your system. This operation initializes the C-871. The function for stopping the pulse output can also be adversely affected by failure on a circuit connected to the system.
3	Entry of the FSSTOP signal.	Entering the FSSTOP saves data stored on the MCC05v2. The function for stopping the pulse output can also be adversely affected by failure on a circuit connected to the system.
4	Entry of the LIMIT signal	This input signal also stops the pulse output. Its reliability, however, is lower than the above two measures.

Considering circuit configurations of the C-871 and MCC05v2, reliability of each input signal can be rated as shown below:

$\overline{\text{RESET}}$>FSSTOP>LIMIT

When the above trouble on the system can involve personal injuries, the drive system power must be turned off (the measure "1" in the table) immediately.
Measures 3 and 4 should be employed only when protection of the user system is the key issue.
In this case, too, you should consider replacing them with measures 2 or 1 if magnitude of actual damages greater than anticipated.

20. C−871 ALL COMMAND TABLES

20-1. DRIVE Command Table

The mark ＊denotes a command accompanied by pulse output.

The reference pages shown by ■ are found in the User's Manual [Applied Functions Part].

	D7D6D5D4D3D2D1D0	HEX CODE	COMMAND NAME	Reference page
	0 0 0 0 0 0 0 0	0 0	NO OPERATION	2 8
	0 0 0 0 0 0 0 1	0 1	SPEC INITIALIZE1	2 9
	0 0 0 0 0 0 1 0	0 2	PULSE COUNTER INITIALIZE	3 0
	0 0 0 0 0 0 1 1	0 3	ADDRESS INITIALIZE	3 3
	0 0 0 0 0 1 0 0	0 4	ADDRESS READ	3 3
	0 0 0 0 0 1 0 1	0 5	SERVO RESET	3 4
	0 0 0 0 0 1 1 0	0 6	RATE SET	3 4, 1 9
	0 0 0 0 0 1 1 1	0 7	LSPD SET	3 4, 2 1
	0 0 0 0 1 0 0 0	0 8	HSPD SET	3 5, 2 1
	0 0 0 0 1 0 0 1	0 9	DFL COUNTER INITIALIZE	3 6, 3 7
	0 0 0 0 1 0 1 0	0 A	SET DATA READ	3 8
	0 0 0 0 1 0 1 1	0 B	CW SOFT LIMIT SET	3 9
	0 0 0 0 1 1 0 0	0 C	CCW SOFT LIMIT SET	3 9
	0 0 0 0 1 1 0 1	0 D	Setting is disabled.	--------
	0 0 0 0 1 1 1 0	0 E	DFL DIVISION DATA SET	4 0
	0 0 0 0 1 1 1 1	0 F	SENSOR INDEX3 DATA SET	4 0
＊	0 0 0 1 0 0 0 0	1 0	+JOG	3 9
＊	0 0 0 1 0 0 0 1	1 1	-JOG	3 9
＊	0 0 0 1 0 0 1 0	1 2	+SCAN	3 9
＊	0 0 0 1 0 0 1 1	1 3	-SCAN	3 9
＊	0 0 0 1 0 1 0 0	1 4	INCREMENTAL INDEX	3 9
＊	0 0 0 1 0 1 0 1	1 5	ABSOLUTE INDEX	4 0
		1 6～1 7	Setting is disabled.	--------
	0 0 0 1 1 0 0 0	1 8	END PULSE SET	4 1
	0 0 0 1 1 0 0 1	1 9	ESPD SET	4 1
	0 0 0 1 1 0 1 0	1 A	CSPD SET	4 0, 2 1
	0 0 0 1 1 0 1 1	1 B	OFFSET PULSE SET	4 1
	0 0 0 1 1 1 0 0	1 C	ORIGIN DELAY SET	4 1
	0 0 0 1 1 1 0 1	1 D	ORIGIN FLAG RESET	4 1
＊	0 0 0 1 1 1 1 0	1 E	ORIGIN	4 2
	0 0 0 1 1 1 1 1	1 F	Setting is disabled.	--------
	0 0 1 0 0 0 0 0	2 0	SPEC INITIALIZE3	4 2
	0 0 1 0 0 0 0 1	2 1	Setting is disabled.	--------
	0 0 1 0 0 0 1 0	2 2	RESOLUTION SET	4 3
	0 0 1 0 0 0 1 1	2 3	PART HSPD BUFFER SET	4 4
	0 0 1 0 0 1 0 0	2 4	PART HSPD SET	4 4
	0 0 1 0 0 1 0 1	2 5	INCREMENTAL DATA SET	4 5
	0 0 1 0 0 1 1 0	2 6	ABSOLUTE DATA SET	4 5
	0 0 1 0 0 1 1 1	2 7	PART PULSE SET	4 6
	0 0 1 0 1 0 0 0	2 8	SERIAL INDEX CHECK	4 7
	0 0 1 0 1 0 0 1	2 9	PART RATE SET	4 8
	0 0 1 0 1 0 1 0	2 A	SPECIAL SERIAL INDEX CHECK	4 8
	0 0 1 0 1 0 1 1	2 B	MARGIN TIME SET	4 9
	0 0 1 0 1 1 0 0	2 C	PEAK PULSE SET	4 9
	0 0 1 0 1 1 0 1	2 D	SEND PULSE SET	5 0
	0 0 1 0 1 1 1 0	2 E	SESPD SET	5 0
	0 0 1 0 1 1 1 1	2 F	SPEC INITIALIZE4	5 1
＊	0 0 1 1 0 0 0 0	3 0	+SPECIAL SCAN1	5 2
＊	0 0 1 1 0 0 0 1	3 1	-SPECIAL SCAN1	5 2
＊	0 0 1 1 0 0 1 0	3 2	+SPECIAL SCAN2	5 2
＊	0 0 1 1 0 0 1 1	3 3	-SPECIAL SCAN2	5 2
＊	0 0 1 1 0 1 0 0	3 4	SPECIAL INCREMENTAL INDEX1	5 3
＊	0 0 1 1 0 1 0 1	3 5	SPECIAL ABSOLUTE INDEX1	5 3

	$D^7D^6D^5D^4D^3D^2D^1D^0$	HEX CODE	COMMAND NAME	Reference page
*	0 0 1 1 0 1 1 0	3 6	SPECIAL INCREMENTAL INDEX2	5 4
*	0 0 1 1 0 1 1 1	3 7	SPECIAL ABSOLUTE INDEX2	5 4
*	0 0 1 1 1 0 0 0	3 8	+SERIAL INDEX	5 5
*	0 0 1 1 1 0 0 1	3 9	-SERIAL INDEX	5 5
*	0 0 1 1 1 0 1 0	3 A	+SPECIAL SERIAL INDEX	5 5
*	0 0 1 1 1 0 1 1	3 B	-SPECIAL SERIAL INDEX	5 5
*	0 0 1 1 1 1 0 0	3 C	SENSOR INDEX1	5 5
*	0 0 1 1 1 1 0 1	3 D	SENSOR INDEX2	5 6
*	0 0 1 1 1 1 1 0	3 E	SENSOR INDEX3	5 6
		3 F	Setting is disabled.	--------
	0 1 0 0 0 0 0 0	4 0	+SENSOR SCAN1	5 7
	0 1 0 0 0 0 0 1	4 1	-SENSOR SCAN1	5 7
		4 2～4 F	Setting is disabled.	--------
			Setting is disabled.	--------
	0 1 0 1 0 0 0 0	5 0	DEND TIME SET	5 7
	0 1 0 1 0 0 0 1	5 1	EXTEND ORIGIN SPEC SET	5 8
	0 1 0 1 0 0 1 0	5 2	CONSTANT SCAN MAX PULSE SET	5 8
	0 1 0 1 0 0 1 1	5 3	CHANGE POINT SET	5 9
	0 1 0 1 0 1 0 0	5 4	CHANGE DATA SET	5 9
	0 1 0 1 0 1 0 1	5 5	AUTO CHANGE SET	6 0
		5 6～5 E	Setting is disabled.	--------
	0 1 0 1 1 1 1 1	5 F	SPEC INITIALIZE5	6 1
	0 1 1 0 0 0 0 0	6 0	SRATE SET	4 2, 2 1
	0 1 1 0 0 0 0 1	6 1	SLSPD SET	4 3, 2 1
	0 1 1 0 0 0 1 0	6 2	SHSPD SET	4 3, 2 1
	0 1 1 0 0 0 1 1	6 3	SSRATE ADJUST	4 4, 2 1
	0 1 1 0 0 1 0 0	6 4	SERATE ADJUST	4 4, 2 1
	0 1 1 0 0 1 0 1	6 5	SCSPD1 ADJUST	4 5, 2 1
	0 1 1 0 0 1 1 0	6 6	SCSPD2 ADJUST	4 5, 2 1
		6 7～6 E	Setting is disabled.	--------
	0 1 1 0 1 1 1 1	6 F	SRATE DOWN POINT SET	6 2
*	0 1 1 1 0 0 0 0	7 0	+ S-RATE SCAN	4 6
*	0 1 1 1 0 0 0 1	7 1	- S-RATE SCAN	4 6
*	0 1 1 1 0 0 1 0	7 2	S-RATE INCREMENTAL INDEX	4 6
*	0 1 1 1 0 0 1 1	7 3	S-RATE ABSOLUTE INDEX	4 6
		7 4～C F	Setting is disabled.	--------
	1 1 0 1 0 0 0 0	D 0	DRIVE CALCULATE	6 2
	1 1 0 1 0 0 0 1	D 1	SRATE DRIVE CALCULATE	6 3
		D 2～E 1	Setting is disabled.	--------
	1 1 1 0 0 0 1 0	E 2	ERROR STATUS READ	4 5

20-2. Special Command Table

Special commands can always be executed.

$D^7D^6D^5D^4D^3D^2D^1D^0$	HEX CODE	COMMAND NAME	Reference page
1 1 1 1 0 0 1 1	F 3	SIGNAL OUT	6 4
1 1 1 1 0 1 0 0	F 4	INDEX CHANGE	6 4
1 1 1 1 0 1 0 1	F 5	RATE CHANGE	6 5
1 1 1 1 0 1 1 0	F 6	DRST OUT	6 5
1 1 1 1 0 1 1 1	F 7	SPEED CHANGE	4 7
1 1 1 1 1 0 0 0	F 8	INT MASK	4 8
1 1 1 1 1 0 0 1	F 9	ADDRESS COUNTER PORT SELECT	4 9
1 1 1 1 1 0 1 0	F A	DFL COUNTER PORT SELECT	4 9
1 1 1 1 1 1 0 0	F C	PULSE COUNTER PORT SELECT	4 9
1 1 1 1 1 1 0 1	F D	SPEED PORT SELECT	4 9
1 1 1 1 1 1 1 0	F E	SLOW STOP	4 9
1 1 1 1 1 1 1 1	F F	FAST STOP	4 9

Technical Service

TEL.(042)664-5382　FAX.(042)666-5664
E-mail　s-support@melec-inc.com

Sales and Service

TEL.(042)664-5384　FAX.(042)666-2031
URL:http://www.melec-inc.com

Melec Inc. Control equipment marketing department
516-10,Higashiasakawa-cho,Hachioji-shi,Tokyo 193-0834,Japan