Instruction Manual
of
PPMC-312

Ver 1.2

ampère

Revision Block

REV	DETAILS	DATE	PERSON IN CHARGE	PERSON RESPONSIBLE
1.0	First printing	00.09.06	Konno	Shohji
1.1	Addition of remarks to the signal table	01.01.15	Saito	Shohji
1.2			Saito	

CONTENTS

♦ 1 Outline

PPMC-312, a “programmable stepping motor control LSI”, has been developed from the PPMC 100 series, which is provided with an increased number of functions.

One of the most important roles of a stepping motor controller is to enable accurate positioning control through practice of smooth accel/deceleration. For this purpose, therefore, it must be capable to set accel/deceleration curves suitable for load and to freely control high-speed and accurate output of driving pulses.

PPMC-312 is a control LSI that can serve as a highly effective solution to this task.

PPMC-312 can offer three kinds of accel/deceleration control methods including linear, sigmoid, and free curve accel/deceleration methods, thus realizing accurate positioning control through smooth accel/deceleration operation.

Further, it enables acceleration (deceleration) of maximum speed of 2Mpps with 16,777,215 pulses. Immediate speed change as well as speed change through acceleration/ deceleration in operation have also been realized.

Operation of an instruction, after it has been registered, can be started with the signal pin being toggled.

PPMC-312 operation is controlled by its internally stored program. With a simple instruction code and data being supplied from the host processor, control of a stepping motor is performed in a sophisticated way, sharply reducing load applied to the host processor.

♦ 2 Function Specifications

Initial Setting Function:

Accel/deceleration parameters (startup speed, high-speed rate, stopping speed, Number of acceleration pulses, Number of deceleration pulses)

Accel/deceleration methods (linear, sigmoid, free curve)

In-position, Encode, Interlock, END terminal

Operation Control Function:

Accel/deceleration, Constant-speed operation, Single step

Constant speed origin search (Constant-speed operation up to the control point.)

High-speed zero return (Accel/deceleration operation up to the control point.)

Continuous Constant Speed Operation (Constant-speed operation up to the limit.)

Continuous high-speed operation (High-speed operation up to the high-speed limit.)

Immediate speed change, Accel/deceleration speed change

Immediate stop, Deceleration stop

Excitation ON, Excitation OFF, Counter clear, Alarm clear, Zero clamp

Status Read-in

Status read-in, Current position, Control terminal read-in, Auxiliary input signal status

Auxiliary Control Functions

Current position setting, High-speed limit effective speed setting, Generalized input/output, Mask setting, Table read-in, Interlock, Version read-in, Synchronous operation monitor, Sum check, Event setting/read-in, Resetting

Pulse Output Frequency

122pps~2Mpps (Internal at 8Mhz), 0.238pps~3,906Kpps(Internal at 15.625khz)

Number of Accel/Deceleration Pulses

1~16,777,215 pulses

Maximum Number of Output Pulses

± 4,294,967,296 pulses * Infinite operation when provided with a constant speed origin search instruction or a continuous operation instruction.

Current Position Counter

0 ~ 4,294,967,296

Package

80-pin QFP

● 2-1 Basic Concept and Performance of PPMC-312

■ 2-1.1 Pulse Rate and Motor Speed

PPMC-312 adopts a numerical value called "**pulse rate**" as data to determine stepping motor speed. The relation between the pulse rate and the motor speed shall be as follows:

$$\text{Speed} = \frac{\text{Tclock}}{\text{Rate}} \text{ (PPS)} \quad \bullet\bullet\bullet\bullet \qquad \textbf{Equation 2-1}$$

Speed : Motor speed (PPS, pulses/sec.)
Tclock : Reference clock (Hz)
(Clock value having been assigned with the initial setting instruction or an external clock)
Rate : Pulse rate

■ 2-1.2 Accel/Deceleration Method

Accel/deceleration control of PPMC-312 is determined by data given by the host processor, which includes the following three accel/deceleration methods for choice:

① Linear accel/deceleration method
② Sigmoid accel/deceleration method
③ Free curve accel/deceleration method

■ 2-1.3 Linear Accel/Deceleration Method

The relation between pulse output speed and time during acceleration (deceleration) according to the linear accel/deceleration method takes "linear form" (linear equation).

$$V = V_0 + K \times t \quad \bullet\bullet\bullet\bullet \quad \textbf{Equation 2-2}$$

V : Speed
t : Time
V_0 : Initial speed
K : Acceleration

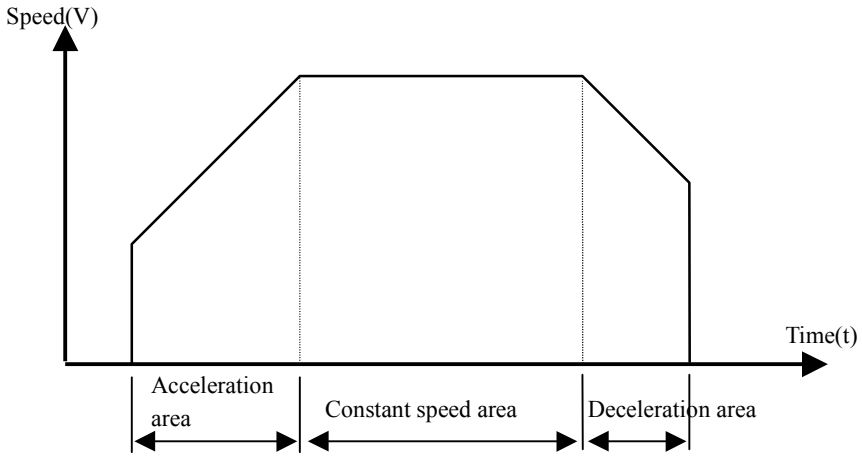


Fig. 2-1

■ 2-1.4 Sigmoid Accel/Deceleration Method

The relation between pulse output speed and time during accel/deceleration according to the sigmoid accel/deceleration method takes “sin curve”.

V= f(t) • • • • Equation 2-3

V **:** Speed
t **:** Time

The sigmoid accel/deceleration method can be expressed by f(t) of the above relative equation, where f(t) takes form of a sin function for the PPMC-312. Using this sigmoid accel/deceleration method can realize highly accurate positioning control through smooth accel/deceleration operation.

V=V0 + K1 × (1- COS(K2 × t)) • • • • Equation 2-4

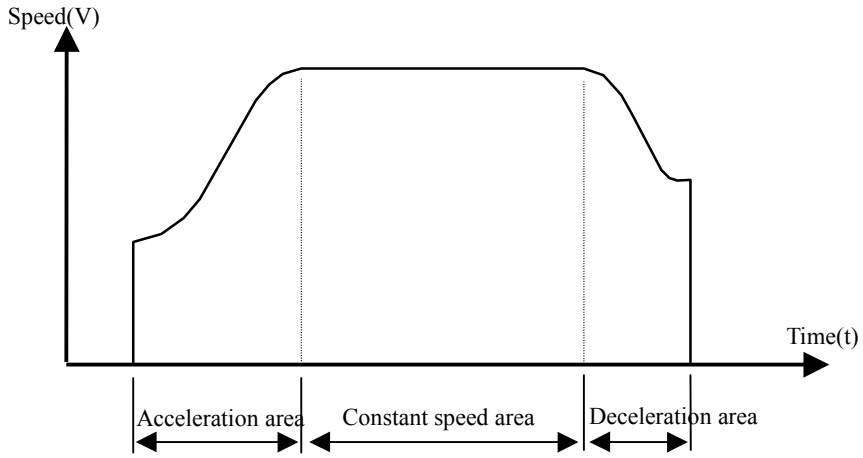


Fig. 2-2

■ 2-1.5 Free Curve Accel/Deceleration Method

The free curve accel/deceleration method makes it possible for each user to supply own accel/deceleration data from outside, thus to create his own accel/deceleration curve each independently.

■ 2-1.6 Position

The scope of positions which PPMC-312 can control includes 0~4,294,836,225 (0x0 ~ 0xffffffff).

CW indicates a direction for larger positions, while CCW for smaller positions.

■ 2-1.7 Accel/Deceleration Table

PPMC-312 performs acceleration/ deceleration based on the accel/deceleration table. The accel/deceleration table can be prepared through use of the initial setting instruction and the free curve setting instruction.

Accel/deceleration, when viewed into details, consists of speed changes in a staircase pattern as in the figure below. As speed increases, resolution of speed becomes rougher. With internal 8M operation being selected in the initial setting, speed difference between Rates 2 and 3 equals (8Mhz/2) – (8Mhz/3) = 1.333Mpps. The difference between Rates 256 and 257, however, is no more than 121.6pps. Take this fully into consideration when using a higher speed.

Although PPMC-312 can have an acceleration table and a deceleration table each independently, rates used for each stage are common to the both table. This same concept also applies to a table created with the free curve setting instruction.

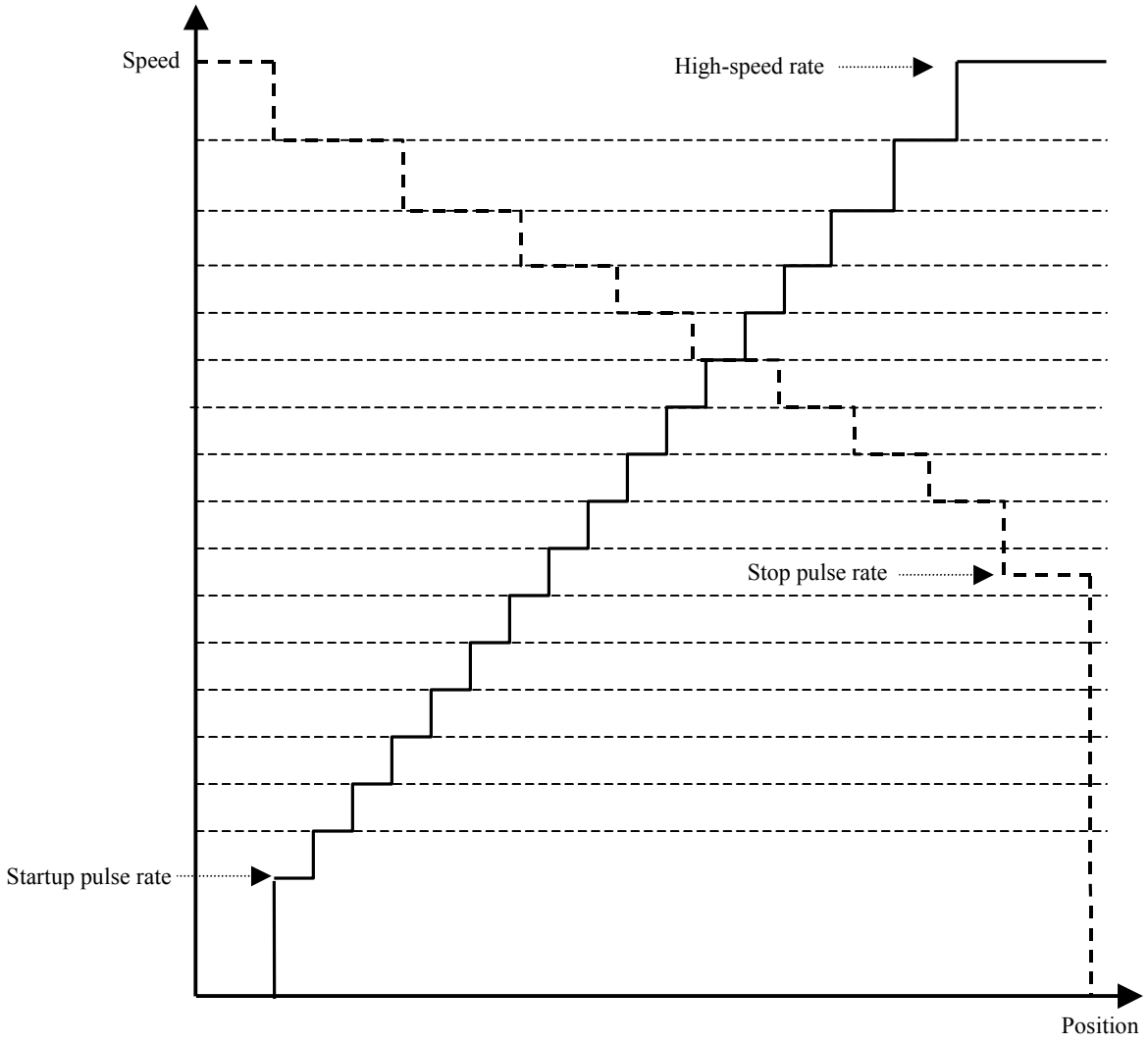


Fig. 2-3

■ 2-1.8 Status

The PPMC-312 has “Initial”, “Normal”, “Accel/Deceleration”, “Constant Speed”, “Abnormality Occurrence”, and “Waiting RUN” statuses. Each status creates different limitation regarding instruction acknowledgment.

Table 2-1

Status	Description
Initial	Status to which transition takes place on power interruption, on release of resetting, in response to the reset instruction, or on failure of the initial setting instruction. As all kinds of setting have been initialized, hardly any instructions can be accepted.
Normal	Status to which transition takes place on completion of the initial setting instruction. Or, this is a status when no pulse output exists. Almost all kinds of instructions are acceptable. The basic status of PPMC-312.
Accel/Deceleration	Status where pulse output takes place according to an accel/deceleration table created in initial setting. Because the pulse rate is not in constant speed, instructions acceptable are limited.
Constant Speed	Status where pulse output takes place at a constant pulse rate. Constant speed operation and the maximum speed operation in accel/deceleration corresponds to this status. As pulses are being output, some of the instructions are unacceptable.
Abnormality Occurrence	Status to which transition takes place when “Low” has been input in ALM signal. Hardly any instructions can be accepted. Abnormality occurrence also takes place when multiple commands and responses have occurred.
Waiting RUN	Status to hold pulse output operation in response to RUN signal. No further instructions relating to pulse output are acceptable.

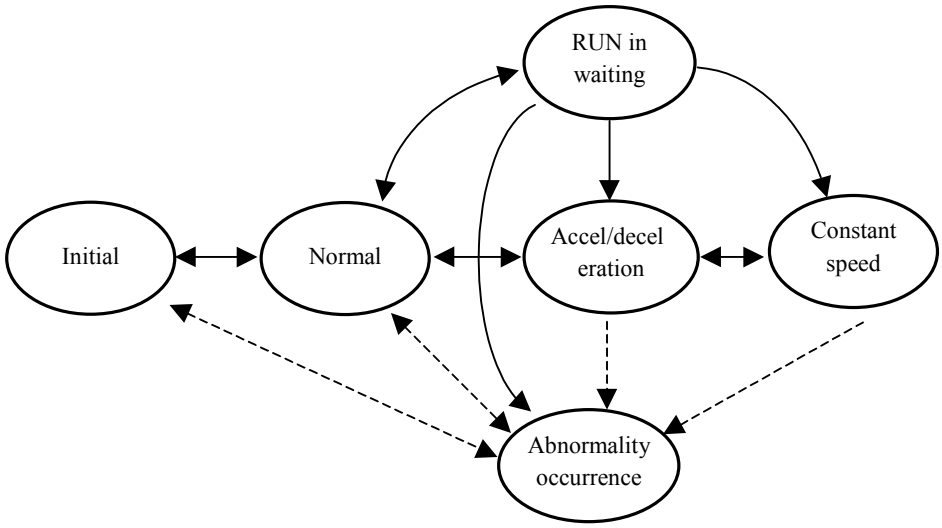


Fig. 2-4 Status Transition Diagram

Table 2-2 Instructions and Statuses

Command \ Status	Initial	Normal	Accel/Decel-eration	Constant Speed	Abnormality Occurrence	Waiting RUN
Initial Setting	○	○	×	×	×	×
Free Curve Setting	○	○	×	×	×	×
In-Position Setting	○	○	×	×	×	×
Encode Setting	○	○	×	×	×	×
Interlock Position Setting	○	○	×	×	×	×
High-Speed Limit Setting	×	○	×	×	×	×
END Terminal Setting	○	○	×	×	×	×
Immediate Setting	×	×/○	○/○	○/○	×	×
Deceleration Stop	×	×/○	○/○	○/○	×	×
Accel/Deceleration	×	○/○	×/○	×/○	×	×
Constant Speed Operation	×	○/○	×/○	×/○	×	×
Single Step	×	○	×	×	×	×
Continuous Acceleration	×	○/○	×/○	×/○	×	×
Continuous Constant Speed	×	○/○	×/○	×/○	×	×
Accel/Deceleration Speed Change	×	×/○	×/○	○/○	×	×
Immediate Speed Change	×	×/○	○/○	○/○	×	×
Constant Speed Origin search	×	○	×	×	×	×
High-Speed Zero Return	×	○	×	×	×	×
Excitation ON	○	○	×	×	×	×
Excitation OFF	○	○	×	×	×	×
Counter Clear	○	○	×	×	×	×
Zero Clamp	○	○	×	×	×	×
Alarm Clear	○	○	×	×	○	○
Generalized Input	○	○	○	○	○	○
Generalized Output	○	○	○	○	○	○
Control Terminal Readout	○	○	○	○	○	○
Mask Setting	○	○	×	×	×	×
Version Readout	○	○	○	○	○	○
Table Readout	○	○	×	×	×	×
Current Position Readout	○	○	○	○	○	○
Current Position Setting	○	○	×	×	×	×
Interlock	×	○	×	×	×	×
Synchronous Operation Monitor	×	○	×	×	×	○
Sum Check	○	○	×	×	×	×
Event Readout	×	○	×	×	×	×
Event Setting	×	○	×	×	×	×
Status Readout	○	○	○	○	○	○
Interlock Release Position Assignment	×	○	×	×	×	×
Version Readout	○	○	×	×	×	×
Error Counter Readout	○	○	×	×	×	×
Resetting	○	○	○	○	○	○

○: Instruction issuable ×: Instruction unissuable

/ : Left−Instruction issued without override registration Right−Instruction issued with override registration

♦ 3 Signals

The PPMC-312 has realized two ways of connection, via parallel bus and in serial.

After cancellation of a reset signal, mode is switched into either parallel or serial depending on MODE signal (Pin 28) status. As pin assignment largely varies between parallel and serial modes, fully pay attention to the circuit design.

During resetting, the output pins are held in input state.

They are switched into output in about 1.6mS after RESET signal becomes High.

Although some of the input pins are capable for internal pull-up, no pull-up takes place during resetting.

● 3-1 Parallel Mode Signal Terminal

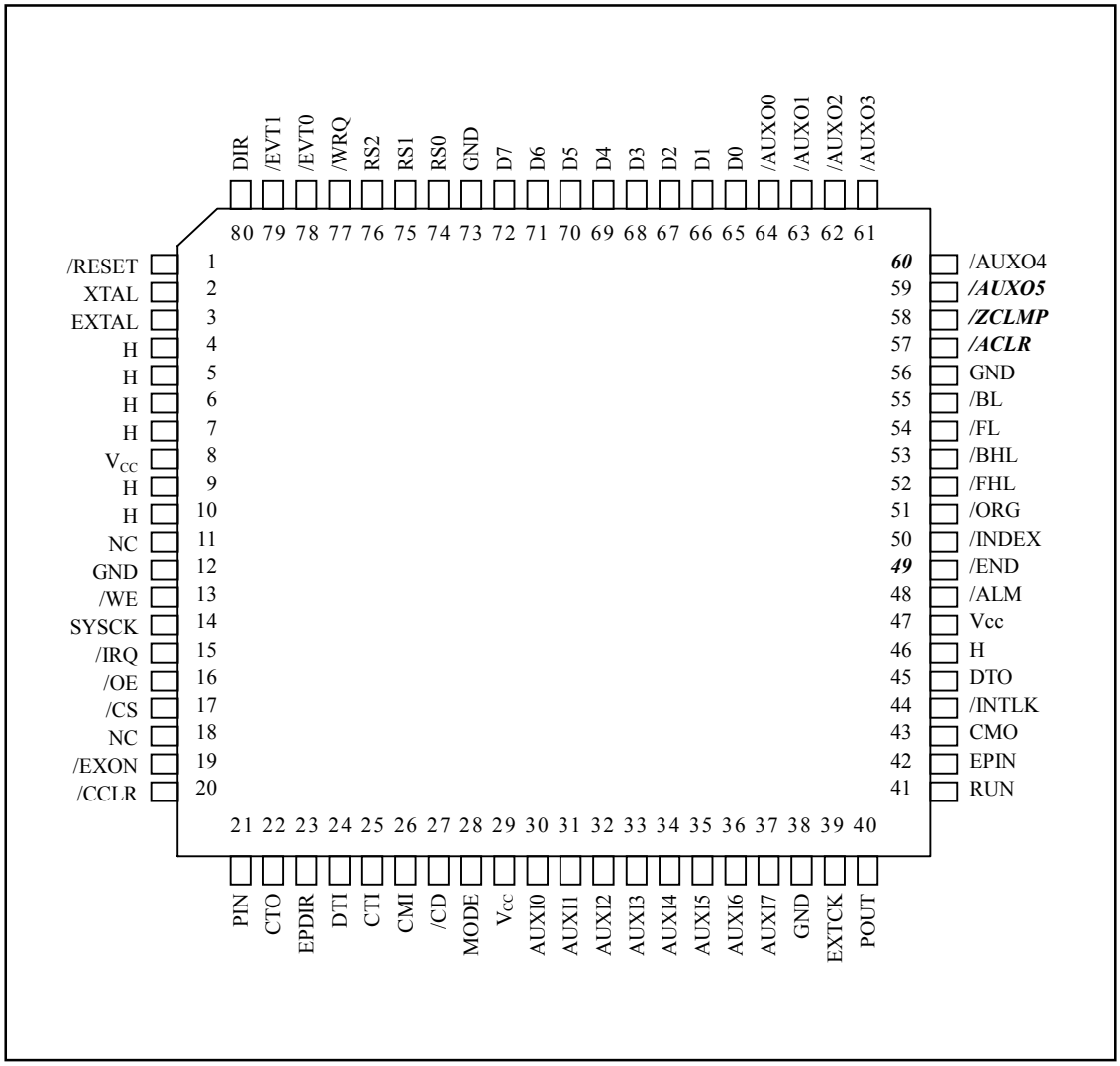


Table 3-1 Terminal Diagram (Parallel Mode)

● 3-2 Parallel Mode Signal Table

Table 3-2 Signal Table (Parallel Mode) 1/2

	Signal Name	Internal Pull-Up	I/O	Function
1	/RESET		I	Resetting
2	XTAL		I	To input crystal oscillator or oscillator clocks
3	EXTAL		I	To input crystal oscillator or oscillator clocks (Reversed phase of XTAL)
4	H		I	5V pull-up
5	H		I	5V pull-up
6	H		I	5V pull-up
7	H		I	5V pull-up
8	Vcc		PS	5V
9	H		I	5V pull-up
10	H		I	5V pull-up
11	NC		O	Unconnected
12	GND		PS	GND
13	/WE		I	Write enable
14	SYSCK		O	System clock
15	/IRQ		O	Interruption output (Open drain output)
16	/OE		I	Output enable
17	/CS		I	Chip selection
18	NC		O	Unconnected
19	/EXON		O	Excitation ON output
20	/CCLR		O	Counter clear output
21	PIN		I	Drive pulse input (Connected to POUT)
22	CTO		O	Connected to CTI
23	EPDIR		I	Encoder direction signal input
24	DTI		I	Connected to DTO.
25	CTI		I	Connected to CTO.
26	CMI		I	Connected to CMO.
27	/CD		O	Exciting current reduced output
28	MODE		I	Mode selection: For parallel mode, connected to VCC.
29	Vcc		PS	5V
30	AUXI0		I	Auxiliary input (Bit 0)
31	AUXI1		I	Auxiliary input (Bit 1)
32	AUXI2		I	Auxiliary input (Bit 2)
33	AUXI3		I	Auxiliary input (Bit 3)
34	AUXI4		I	Auxiliary input (Bit 4)
35	AUXI5		I	Auxiliary input (Bit 5)
36	AUXI6		I	Auxiliary input (Bit 6)
37	AUXI7		I	Auxiliary input (Bit 7)
38	GND		PS	GND
39	EXTCK		I	External clock input

Table 3-3 Signal Table (Parallel Mode) 2/2

	Signal Name	Internal Pull-Up	I/O	Function
40	POUT		O	Drive pulse output
41	RUN		I	Operation start enabling signal
42	EPIN		I	Encode pulse input
43	CMO		O	Connected to CMI.
44	/INTLK		O	Interlock control output
45	DTO		O	Connected to DTI.
46	H		I	5V pull-up
47	Vcc		PS	5V
48	/ALM	●	I	Alarm signal input
49	/END	●	I	Positioning end signal input
50	/INDEX	●	I	Index signal input
51	/ORG	●	I	Origin signal input
52	/FHL	●	I	CW direction high-speed limit input
53	/BHL	●	I	CCW direction high-speed limit input
54	/FL	●	I	CW direction limit input
55	/BL	●	I	CCW direction limit input
56	GND		PS	GND
57	/ACLR		O	Alarm clear output
58	/ZCLMP		O	Zero clamp output
59	/AUXO5		O	Auxiliary output (Bit 5)
60	/AUXO4		O	Auxiliary output (Bit 4)
61	/AUXO3		O	Auxiliary output (Bit 3)
62	/AUXO2		O	Auxiliary output (Bit 2)
63	/AUXO1		O	Auxiliary output (Bit 1)
64	/AUXO0		O	Auxiliary output (Bit 0)
65	D0		I/O	Data bus (Bit 0)
66	D1		I/O	Data bus (Bit 1)
67	D2		I/O	Data bus (Bit 2)
68	D3		I/O	Data bus (Bit 3)
69	D4		I/O	Data bus (Bit 4)
70	D5		I/O	Data bus (Bit 5)
71	D6		I/O	Data bus (Bit 6)
72	D7		I/O	Data bus (Bit 7)
73	GND		PS	GND
74	RS0		I	Register select (Bit 0)
75	RS1		I	Register select (Bit 1)
76	RS2		I	Register select (Bit 2)
77	/WRQ		O	Wait request
78	/EVT0		I	Event input 1
79	/EVT1		I	Event input 2
80	DIR		O	Operating direction output

Signals with "/ " indicate negative logic.

● 3-3 Serial Mode Signal Terminals

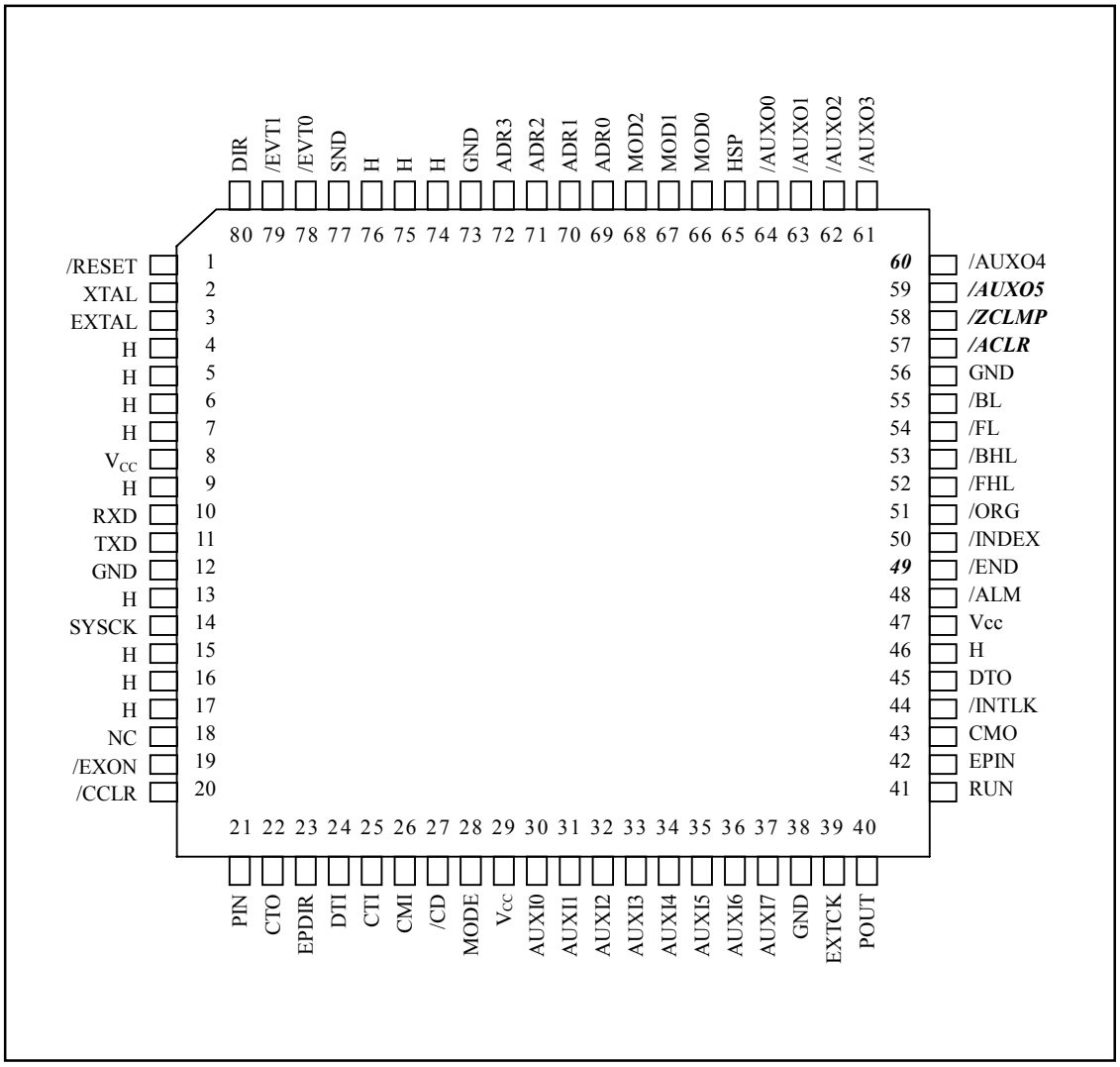


Fig. 3-1 Terminal Diagram (Serial Mode)

● *3-4 Serial Mode Signal Table*

Table 3-4 Signal Table (Serial Mode) 1/2

	Signal Name	Internal Pull-Up	I/O	Function
1	/RESET		I	Resetting
2	XTAL		I	To input crystal oscillator or oscillator clocks.
3	EXTAL		I	To input crystal oscillator or oscillator clocks. (Reversed phase of XTAL)
4	H		I	5V pull-up
5	H		I	5V pull-up
6	H		I	5V pull-up
7	H		I	5V pull-up
8	Vcc		PS	5V
9	H		I	5V pull-up
10	RXD		I	Receiving line
11	TXD		O	Transmitting line
12	GND		PS	GND
13	H		I	5V pull-up
14	SYSCK		O	System clock
15	H		I	5V pull-up
16	H		I	5V pull-up
17	H		I	5V pull-up
18	NC		O	Unconnected
19	/EXON		O	Excitation ON output
20	/CCLR		O	Counter clear output
21	PIN		I	Drive pulse input (Connected to POUT.)
22	CTO		O	Connected to CTI
23	EPDIR		I	Encoder direction signal input
24	DTI		I	Connected to DTO.
25	CTI		I	Connected to CTO.
26	CMI		I	Connected to CMO.
27	/CD		O	Exciting current reduced output
28	MODE		I	Mode selection: For serial mode, connected to GND.
29	Vcc		PS	5V
30	AUXI0		I	Auxiliary input (Bit 0)
31	AUXI1		I	Auxiliary input (Bit 1)
32	AUXI2		I	Auxiliary input (Bit 2)
33	AUXI3		I	Auxiliary input (Bit 3)
34	AUXI4		I	Auxiliary input (Bit 4)
35	AUXI5		I	Auxiliary input (Bit 5)
36	AUXI6		I	Auxiliary input (Bit 6)
37	AUXI7		I	Auxiliary input (Bit 7)
38	GND		PS	GND
39	EXTCK		I	External clock input

Table 3-5 Signal Table (Serial Mode) 2/2

	Signal Name	Internal Pull-Up	I/O	Function
40	POUT		O	Drive pulse output
41	RUN		I	Operation start enabling signal
42	EPIN		I	Encode pulse input
43	CMO		O	Connected to CMI.
44	/INTLK		O	Interlock control output
45	DTO		O	Connected to DTO.
46	H		I	Unconnected
47	Vcc		PS	5V
48	/ALM	●	I	Alarm signal input
49	/END	●	I	Positioning end signal input
50	/INDEX	●	I	Index signal input
51	/ORG	●	I	Origin signal input
52	/FHL	●	I	CW direction high-speed limit input
53	/BHL	●	I	CCW direction high-speed limit input
54	/FL	●	I	CW direction limit input
55	/BL	●	I	CCW direction limit input
56	GND		PS	GND
57	/ACLR		O	Alarm clear output
58	/ZCLMP		O	Zero clamp output
59	/AUXO5		O	Auxiliary output (Bit 5)
60	/AUXO4		O	Auxiliary output (Bit 4)
61	/AUXO3		O	Auxiliary output (Bit 3)
62	/AUXO2		O	Auxiliary output (Bit 2)
63	/AUXO1		O	Auxiliary output (Bit 1)
64	/AUXO0		O	Auxiliary output (Bit 0)
65	HSP	●	I	High-speed polling select
66	MOD0	●	I	Communication speed select
67	MOD1	●	I	Communication speed select
68	MOD2	●	I	Communication speed select
69	ADR0	●	I	Address (Bit 0)
70	ADR1	●	I	Address (Bit 1)
71	ADR2	●	I	Address (Bit 2)
72	ADR3	●	I	Address (Bit 3)
73	GND		PS	GND
74	H		I	5V pull-up
75	H		I	5V pull-up
76	H		I	5V pull-up
77	SND		O	Transmit control output
78	/EVT0		I	Event input 1
79	/EVT1		I	Event input 2
80	DIR		O	Operating direction output

Signals with "/ " indicate negative logic.

● 3-5 Basic Type Terminal

■ 3-5.1 Vcc, GND

Connect the Vcc terminal to the system power supply (+5V).

Connect the GND terminal to the system power supply (0V).

■ 3-5.2 H, NC

Pull up (around 10k~100k) the H terminal to 5V.

Leave the NC terminal unconnected.

■ 3-5.3 /RESET

This is the reset input terminal. On supply of power, hold “Low” for, at least, 20mS after power supply gets stable. To reset in operation, hold “Low” of, at least, 10 clocks.

■ 3-5.4 XTAL, EXTAL

These are the clock terminals. Connect a crystal oscillator or an oscillator of 16Mhz.

■ 3-5.5 SYSCK

Clocks of 16Mhz are output.

■ 3-5.6 MODE

Parallel and serial (MWSC) modes are switched over.

Parallel and serial modes input High and Low respectively. No internal pull-up exists.

■ 3-5.7 CMO, CMI

Be sure to connect CMO and CMI.

■ 3-5.8 DTO, DTI

Be sure to connect DTO and DTI.

■ 3-5.9 CTO, CTI

Be sure to connect CTO and CTI.

● 3-6 Parallel Mode Signal

■ 3-6.1 D0~D7

These are 8-bit two-way data buses.

■ 3-6.2 RS2~RS0

These are register select terminals for PPMC-312 inside control.

■ 3-6.3 /WRQ

This serves as the memory access wait terminal.

This is effectuated when an access has taken place to the internal register DWR and DRR at an interval lower than 9.5 clocks as having been input in XTAL/EXTAL.

You can set this with CSR.

■ 3-6.4 /WE

This is the write terminal in PPMC-312.

■ 3-6.5 /IRQ

This terminal serves to notify the host side of an access request from the PPMC-312. Open drain output is adopted. Use external pull-up.

■ 3-6.6 /OE

This is the read-in terminal from PPMC-312.

■ 3-6.7 /CS

This is the select terminal of PPMC-312.

● 3-7 Auxiliary Type Terminals

■ 3-7.1 /EVT1~0

These terminals are used for event input. When not in use, they should be held pulled-up (10k~100k).

■ 3-7.2 AUXI7~0

These are generalized input port terminals. When not in use, they should be held pulled-up (10k=100k).

■ 3-7.3 /AUXO5~0

These are generalized output port terminals. The default values on power supply are “High” for all the terminals.

● 3-8 Serial Mode Signals

■ 3-8.1 RXD, TXD

They are transmission/reception terminals for MWSC communication.

■ 3-8.2 ADR3~0

These are the input terminals for station numbers in serial mode.

■ 3-8.3 SND

This serves to control the transmission gate. Transmission starts when this terminal becomes “High”, which becomes “Low” at the end of transmission.

■ 3-8.4 HSP

This is the HSP select terminal. For normal mode operation, set this to Low and, for high speed mode operation, set this to High.

■ 3-8.5 MOD2~0

Communication Speed Select Terminal

MOD			Speed
2	1	0	
0	0	0	Invalid
0	0	1	Binary mode 31.25kbps
0	1	0	Binary mode 62.5kbps
0	1	1	Binary mode 125kbps
1	0	0	Invalid
1	0	1	ASCII mode 19.2kbps
1	1	0	ASCII mode 38.4kbps
1	1	1	ASCII mode 83.33kbps

● 3-9 Driver Control Type Terminals

■ 3-9.1 /EXON

This is the excitation output terminal. This, responding to an excitation ON/OFF instruction, performs ON/OFF operation. This is held High on supply of power. Excitation output does not influence PPMC-312 operation. Pulses are output even during “Low” state.

■ 3-9.2 /CCLR

This is the counter clear terminal. In response to a counter clear instruction, “Low” signals are output at intervals of about 24.6mS. This is held “High” on supply of power.

■ 3-9.3 /ACLR

This terminal serves to clear an alarm.

In response to an alarm clear instruction, “Low” signals are output at intervals of about 24.6mS. This is held “High” on supply of power.

■ 3-9.4 /ZCLMP

This terminal is for zero clamp output. With a zero clamp instruction, this becomes “Low”.

With start of pulse output, this becomes “High”.

This can be operated in synchronous with an /END signal under an instruction of in-position setting.

This is held “High” on supply of power.

■ 3-9.5 /CD

This terminal is for exciting current reduced output. PPMC-312 automatically controls this.

This enters “Low” state in about 100mS from end of pulse output, which becomes “High” on start of pulse output.

“High” remains if it is less than 100mS from pulse output stop until the next pulse output is started.

When the /ALM signal becomes “Low”, this automatically shifts into “Low”.

This is held “High” on supply of power.

■ 3-9.6 EPIN

This is the encode pulse input terminal. Enter pulses from the encoder in this terminal.

When not in use, this should be held pulled up (10k~100k).

■ 3-9.7 EPDIR

This terminal is for encode direction input. Connect the encode direction signal to this terminal.

Enter High for CCW direction and Low for CW direction.

When not in use, this should be held pulled up (10k~100k).

■ 3-9.8 /INDEX

This terminal serves to connect index pulses from the encoder.

Use this along with /ORG in origin detection. Use of /INDEX alone cannot perform origin detection.

When not in use, this should be held pulled up (10k~100k).

■ 3-9.9 /ALM

This terminal is for alarm input.

With this set at “Low”, abnormality occurrence status is produced regardless of state of PPMC-312.

Pulse output is immediately stopped.

When not in use, this should be held pulled up (10k~100k).

■ 3-9.10 /END

This terminal is for the positioning end signal.

On receipt of this signal after pulse output is stopped, “pulse output stop” is acknowledged.

This is also used in in-position control. When not in use, this should be held pulled up (10k~100k).

● *3-10 Pulse Type*

■ 3-10.1 POUT

This terminal is for pulse output.

■ 3-10.2 PIN

Be sure to connect this to POUT.

■ 3-10.3 DIR

This terminal is for pulse output direction and is automatically output. "High" represents CCW, while "Low" represents CW.

■ 3-10.4 EXTCK

This is for input of external clocks in pulse output. When not in use, this should be held pulled up (10k~100k).

■ 3-10.5 RUN

This terminal serves to indicate start of pulse output. Using this terminal enables synchronous operation. This terminal is checked immediately before start of pulse output. And, if it is "Low", pulse output is retained in "Hold". With "High", pulse output starts. When pulse output is already started, it is not brought to a stop even if it gets "Low". When not in use, this should be held pulled up (10k~100k).

■ 3-10.6 /INTLK

This is the terminal for interlock control output. This becomes "Low" either at the control position or with the interlock stop instruction.

■ 3-10.7 /ORG

This terminal serves to connect origin detection. Use this to detect the origin. When not in use, this should be held pulled up (10k~100k).

■ 3-10.8 /FL, /FHL

These represent limit signals in CW direction. "/FL" is the critical limit and "/FHL" the high-speed operation limit.

On detection of /LF during pulse output in CW direction, the pulse output is immediately stopped.

On detection of /FHL while in pulse output in CW direction at a rate higher than a set level, operation is brought to a deceleration stop. When not in use, this should be held pulled up (10k~100k).

■ 3-10.9 /BL, /BHL

These represent limit signals in CCW direction. "/BL" is the critical limit and "/BHL" the high-speed operation limit.

On detection of /BL during pulse output in CCW direction, the pulse output is immediately stopped.

On detection of /BHL while in pulse output in CCW direction at a rate higher than a set level, operation is brought to a deceleration stop. When not in use, this should be held pulled up (10k~100k).

♦ 4 Parallel Connection

The PPMC-312 has a parallel I/F which is different from those of the conventional PPMC's.

With the conventional models, whether it is writable (readable) or not had to be confirmed byte by byte. The PPMC-312 enables one whole block of an instruction/data to be continuously written (read) for one time confirmation.

For parallel I/F, signals including "D0~D7", "RS0~RS2", "/WRQ", "/WE", "/IRQ", "/OE", and "/CS" are used.

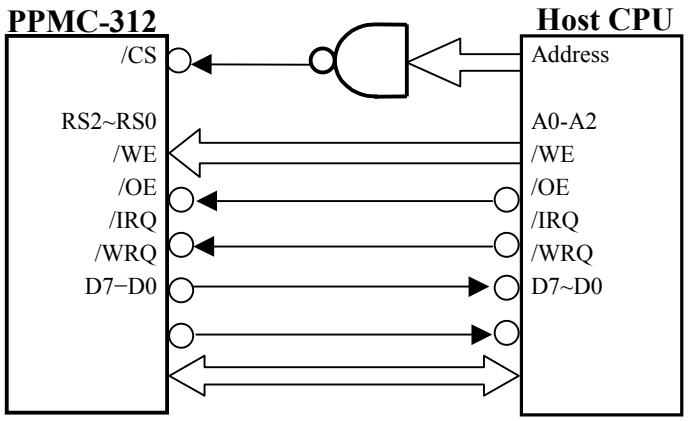


Fig. 4-1 Concept of Connection with Host CPU

● 4-1 Register

Registers used in parallel mode include the following:

Table 4-1 Registers

Address			Direction	Code	Description
RS2	RS1	RS0			
0	0	0	R/W	CSR	Control/status register
0	0	1	−	—	Inaccessible
0	1	0	R	RCR	Readout control register
0	1	1	R	RAR	Readout address register size
0	1	0	W	WCR	Write control register
0	1	1	W	WAR	Write address register size
1	0	0	−	—	Inaccessible
1	0	1	W	DWR	Instruction/data write register
1	1	0	−	—	Inaccessible
1	1	1	R	DRR	Instruction/data readout register

■ 4-1.1 CSR (Control/Status Register)

	7	6	5	4	3	2	1	0
Direction	R	R/W	R/W	R/W	R	R	R	R
Name	−	EWRQ	EWIRQ	ERIRQ	MWEF	MREF	−	−
On resetting		0	0	0				

Bit	Mnemonic	Status	Description
6	EWRQ	0	/WRQ signal not used. Initial value
		1	/WRQ signal used.
5	EWIRQ	0	Instruction writable. /IRQ signal not output. (Initial value)
		1	Instruction writable. /IRQ signal set to Low.
4	ERIRQ	0	/IRQ signal not output in response report. (Initial value)
		1	/IRQ signal set to Low in response report.
3	MWEF	0	Instruction writable.
		1	Instruction not writable.
2	MREF	0	With response data (Readout request from PPMC-312)
		1	Without response data

■ 4-1.2 RCR (Read Control Register)

This register is used for response readout. This is used for response readout procedure. Data read out do not have any meaning.

■ 4-1.3 RAR (Read Address Register)

This register is used for response readout. Size to be read is written here.

■ 4-1.4 WCR (Write Control Register)

This register is used for command writing. Be sure to write 0x20.

■ 4-1.5 WAR (Write Address Register)

This register is used for command writing.

Write the value which is equal to 0x20 (dec32) minus command size.

■ 4-1.6 DWR (Data Write Register)

Write a command.

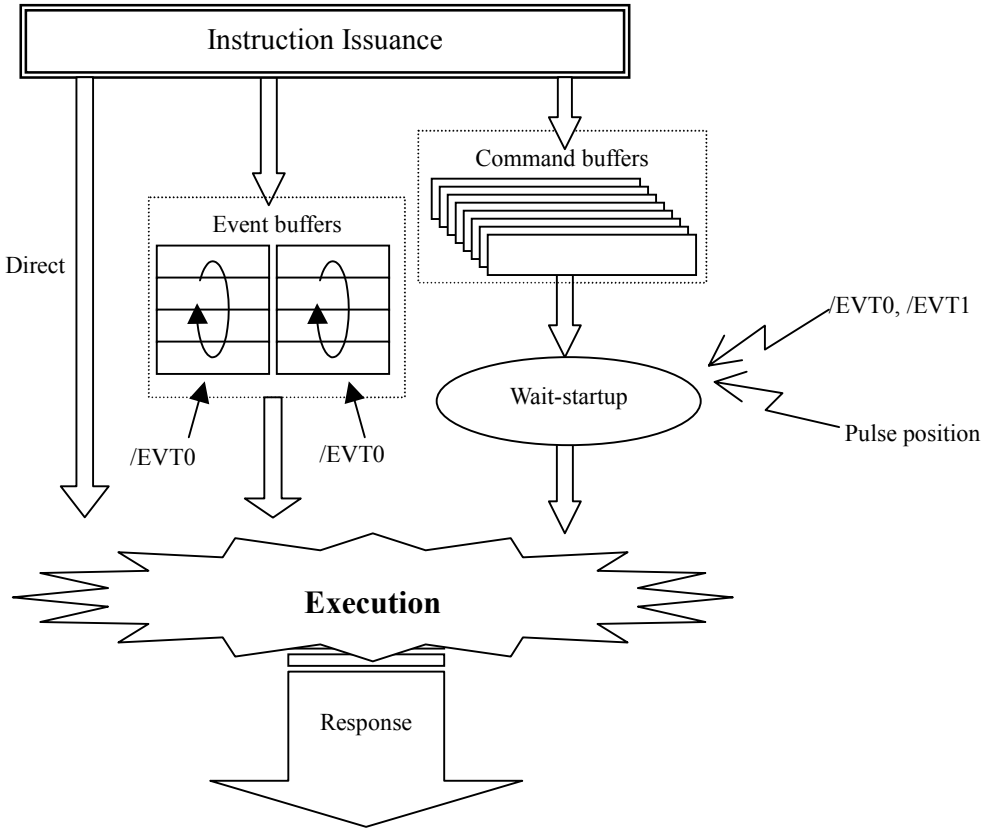
■ 4-1.7 DRR (Data Read Register)

This register is for reading a response.

♦ 5 Instruction Issuance and Responses

The PPMC-312 in parallel mode has three routes for execution of an instruction. All instructions and operations are always provided with the responses.

- An instruction directly written in are subject to immediate execution.
- Override: Command buffer instructions are executed at the execution position or with EVT0/1 signals.
- Event buffer: Instructions are first registered in the event buffer, which are then executed with EVT0/1 signals.



● 5-1 Direct

An instruction is executed in time of its issuance.

● 5-2 Override

There are eight command buffers, to which FIFO (first in first out) method is applied. The operation in which instructions stored in command buffers are executed one by one is referred to as “override”.

Execution takes place with /EVT0, /EVT1 (event signals), and pulse position serving as the trigger. Instructions, after being executed, are cancelled and the next following instruction enters into wait-startup status.

/EVT0 and /EVT1 also serve as triggers for instructions stored in the event buffers.

When Event Buffer 0 has an instruction, /EVT0 is used for Event Buffer 0 and, therefore, /EVT0 cannot be used for override. Should it be used, a response to registration failure is given.

/EVT0 and /EVT1, if not used by the event buffer, are useable for override. (It is possible that /EVT1 is used for Event Buffer 1 and /EVT0 for override.)

As override is always given priority over an instruction given by the host, ordinary instructions become more difficult to be accepted. When requests for processing take place at the same time, a command from the host is held in waiting, during which an override instruction is executed. The instruction from the host is then executed.

Through using the status read-in instruction, you can check how many instructions are currently in storage. However, you cannot see individual data. Neither can you cancel (delete) instructions thus stored.

On receipt of an abnormality response after override is executed, instructions having been registered in the override are all deleted.

● 5-3 Event

Operation in which registered instructions are executed one by one by externally given signals is referred to as the event.

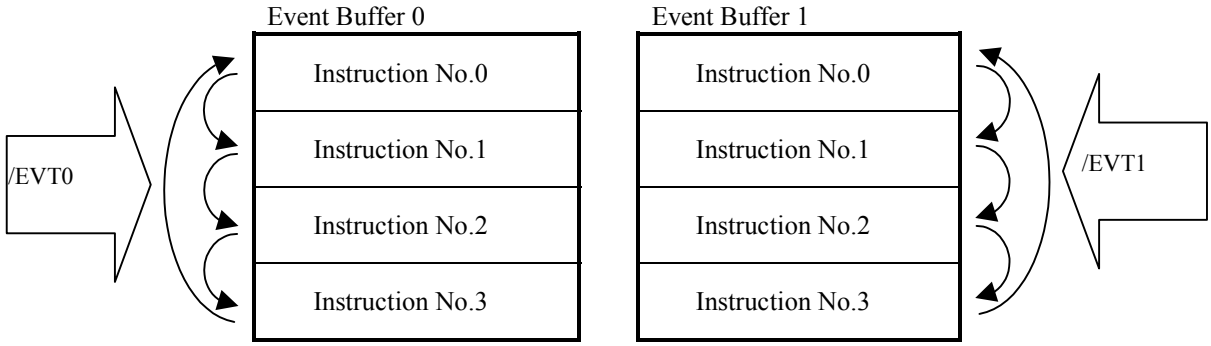
The PPMC-312 has two event buffers. Event Buffer 0 is executed with /EVT0 signal. Event Buffer 1 is executed with /EVT1 signal.

These buffer each can store four instructions. With /EVT0 or /EVT1 signals, these four instructions are executed in order.

Which of the four instructions has been executed is reported by way of a response. For event registration, instructions are written in number order, starting with No.0. They are deleted from Instruction No.3 toward Instruction No.0 in this order. On completion of registration or deletion, execution takes place, starting with Instruction No.0.

On receipt of an abnormality response after an event is executed, the instructions having been registered are deleted by one whole buffer.

Registered instructions alone can be repeated even if all four instructions are not necessarily given. However, passing or omitting any one of them is impossible.



● 5-4 Responses

Data of which the PPMC-312 wishes to notify the host are referred to as “responses”.

Responses are largely divided into the following eight items. Each item has one buffer.

1. Alarm
2. Instruction
3. End of interlock
4. Hold execution
5. Stop of pulse output
6. Change in generalized input data
7. In-position change
8. Encode change

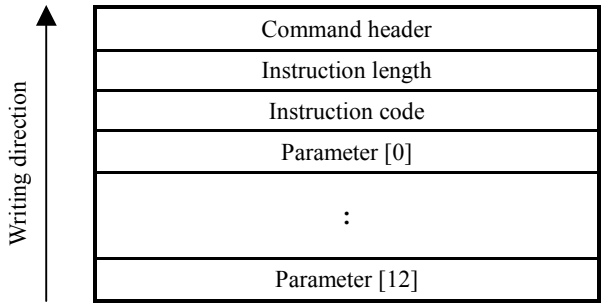
Items 1~5 each has one response buffer. When a response buffer is overwritten, an alarm (/ALM) occurs, pulse output is stopped, and the alarm occurrence response is output. For example, suppose that responses of instruction, hold execution, interlock release, and pulse output stop are output one each, which are not read by the host. No alarm occurs in this stage. When an instruction is issued here and a response for instruction issuance is output, multiple occurrence is judged and the response for the 2nd instruction is replaced by the alarm occurrence response.

“Change in generalized input data (AUXI0~7)” of Item 6 notifies of a change in the generalized input signal. The generalized input signal is not always synchronized with the host. When the host has failed to deal with all processing, generalized input is changed and the buffer is overwritten with new data. This change is notified.

As for Items 7 and 8, if a set value for each instruction is small and each change is to be reported, responses can occur too often. Changes in in-position and encode is reported as being either within-range or outside-range. For example, reading of within-range responses by two consecutive times can create a strange state. The PPMC-312 keeps the last read response in storage and, on occurrence of a different status, it sends out a response.

● 5-5 Instruction Write Format

Writing of an instruction consists of a series of blocks including a header, instruction length, instruction code, and parameters [0]~[12]. They can be written in consecutively.



■ 5-5.1 Command Header

Header		
0x00	Parameter only	With continued part
0x01	Parameter only	Instruction executed
0x02	Instruction code + Parameter	With continued part
0x03	Instruction code + Parameter	Instruction executed

Attributes of data in storage are shown. As the parameter is not more than 13 bytes in many cases, “0x03” is set to the header.

However, the free curve setting instruction requires many parameters. It is necessary in this case that a block consisting only of parameters is set without an instruction code being inserted. The parameter, in some cases, has a continued part. The header, in this case, becomes “0x02, 0x00, 0x00 … 0x00, 0x01”.

■ 5-5.2 Instruction Code

This is a proper code for each instruction. See “Instruction” column.

■ 5-5.3 Instruction Length

The total number of bytes of the instruction code and all the parameters is set here.

■ 5-5.4 Parameters 0~12

These are the data subordinating to an instruction (Operation pulse number for accel/deceleration, etc.) For example, “0x12345678” becomes “0x12, 0x34, 0x56, 0x78”.

For an instruction which requires many parameters like a free curve setting instruction, it cannot be written at one time and has to be written divided into several times. So long as the numbers of the header, instruction length, and parameters are matching, no problem takes place.

● 5-6 Instruction Issuance Procedure

① Set the CSR. This serves as initialization operation for instruction writing. With this performed only once on power supply, the setting is retained until power is cut or resetting occurs.

② Confirm that CSR MWEF bit is 0. Writing is not enabled when it is 1. There is no guarantee of proper operation after writing.

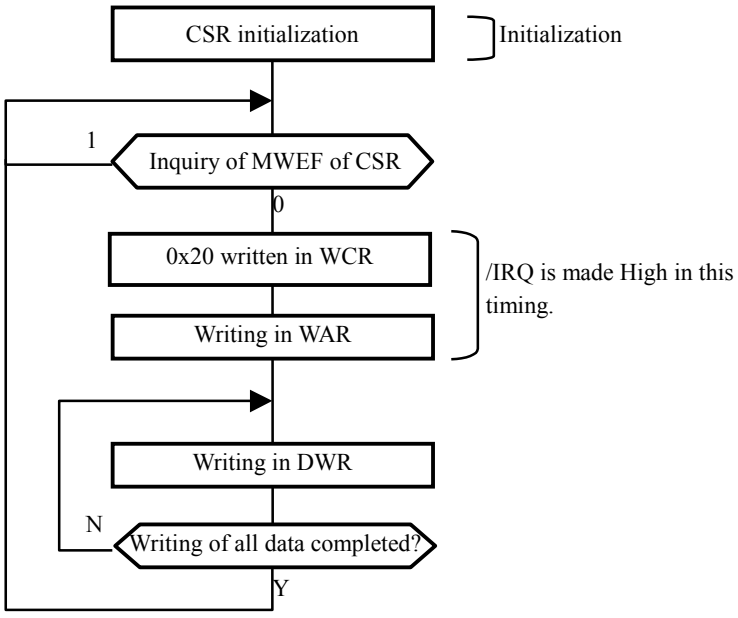
③ Write "0x20" into WCR. (An + ything except 0x20 is invalid.)

④ Write "0x20 – (write data size)" in WAR.
Write data size equals "header + instruction length".
Supposing that it is an accel/deceleration instruction, "header (1) + instruction length (instruction code(1) + parameter(4) =6" becomes the write data size. It is, therefore, necessary to write "0x20 – 0x06 =0x1A".

⑤ Write data in DWR, starting with the lowest byte of the instruction write format.
For example, when the operation number in CCW direction for an accel/deceleration instruction is 0x12345678;

Parameter	0x78
Parameter	0x56
Parameter	0x34
Parameter	0x12
Parameter	0x40
Instruction code	0x82
Instruction length	0x06
Command header	0x03

With completion of header writing, instruction issuance is brought to an end. With this timing, MWEF bit of CSR becomes 1.

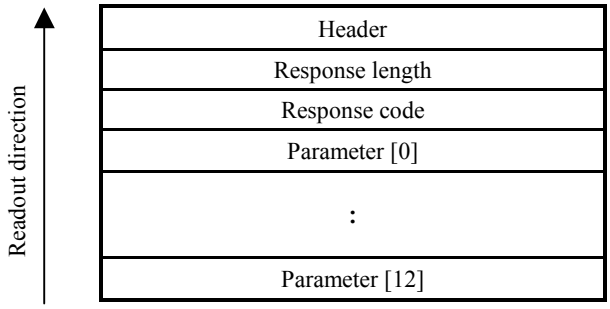


Note 1.
When 1 is set in EWIRQ of CSR while MWEF bit of CSR is 0, /IRQ signal is immediately output.

Note 2.
As /IRQ signal is shared by Read and Write, interruption contention can take place. Preventive measures such as monitoring of EWIRQ bit and MWEF bit are necessary.

● 5-7 Response Readout

Reading of a response consists of a series of blocks including a header, response length, response code, and parameters [0]~[12]. They can be read consecutively. Reading takes place, starting with the lowest place of parameters.



■ 5-7.1 Response Header

Attributes of data in storage are shown. Although the response parameter is not more than 13 bytes in many cases, the table read instruction is capable to read many parameters. At this time, blocks with parameters only can be read. Also, the parameter may have any continued part.

A response may not always be directly from a parallel bus, which may be sent from an event buffer or a command buffer. This is judged by Bits 3/4.

Other than responses to instructions, responses may appear at odd intervals on operation ending or on occurrence of abnormal status. Judge them by Bit 5.

	7	6	5	4	3	2	1	0
Name	0	0	CMR	RESP		0	PAL	NEXT

Bit	Mnemonic	Status	Description
5	CMR	0	Response to an instruction
		1	Response to a matter other than instruction
4,3	RESP	00	Response from a parallel bus
		01	Response from an event
		10	Impossible
		11	Response from override
1	PAL	0	Parameter only
		1	Containing response length/ response code/ parameters.
0	NEXT	0	Response provided with a continued part. Keep reading out the response.
		1	Response is all finished.

● 5-8 Response Access Procedure

① Set the CSR. This serves as initialization operation for instruction writing. With this performed only once on power supply, the setting is retained until power is cut or resetting occurs.

② Wait until MREF bit of CSR becomes 0. With CSR ERIRQ held at 1, MREF bit becomes 0 and /IRQ signal is made Low.

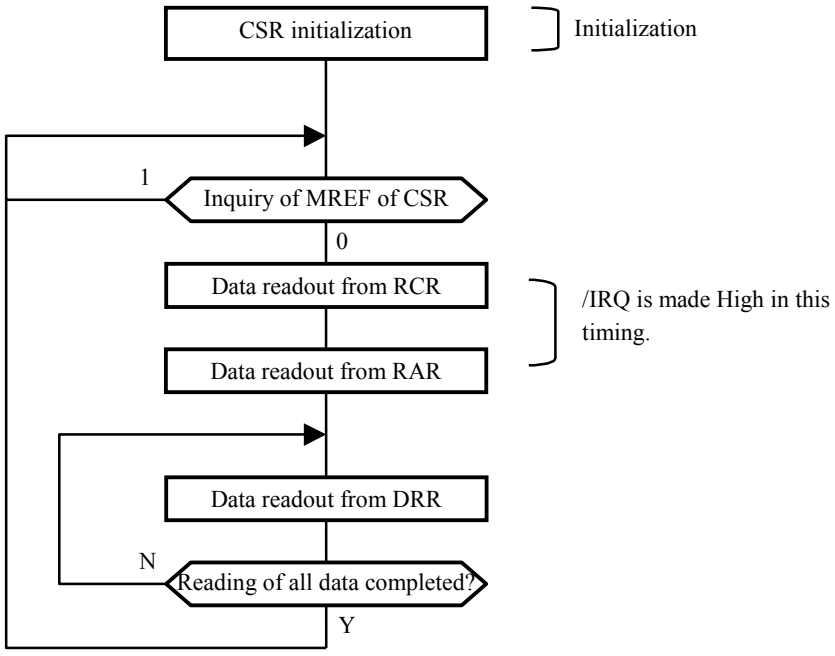
③ False read RCR. (Data here do not have meaning.)

④ Reading RAR, calculate the data size in storage with "0x40 – RAR value".

⑤ Data are read out by the data size obtained in ④.
For example, the response for current position readout can be read starting with the lower parameter.

Parameter [4]	0x78	Lower order
Parameter [3]	0x56	Medium lower order
Parameter [2]	0x34	Medium upper order
Parameter [1]	0x12	Upper order
Parameter [0]	0x00	Current position
Response code	0x42	Current position readout response code
Response length	0x06	
Header	0x03	

⑥ With all data having been read, MREF bit of CSR becomes 1.



Note 1.

As /IRQ signal is shared by Read and Write, interruption contention can take place. Preventive measures such as monitoring of EWRDY bit and MWEF bit are necessary.

Note 2.

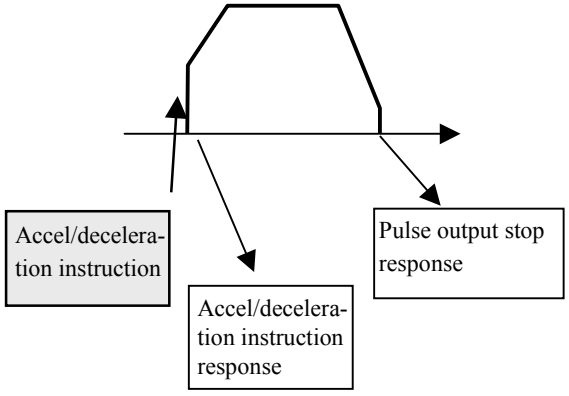
Pay attention to Bits 0/1 of the header. When a long parameter such as table readout response exists, the response is delivered by these bits. When Bit 1 is 0, no response code exists. The response code realm becomes Parameter [0], which is moved up one by one. As a result, Responses [0]~[12] become Responses [0]~[13]

Note 3.

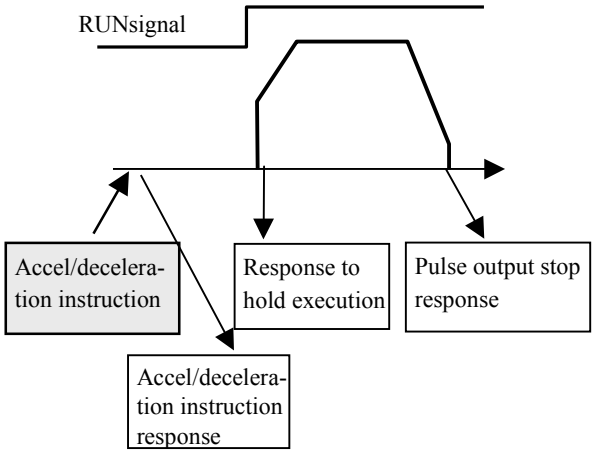
When reading data from DRR byte by byte, read it in 3.625ns per byte.

♦ 6 Instruction and Operation/Response

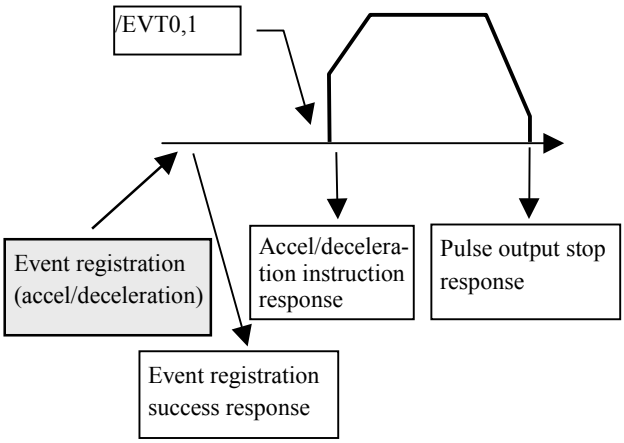
The relations between operation and instruction/response are illustrated as accel/deceleration being taken as an example.



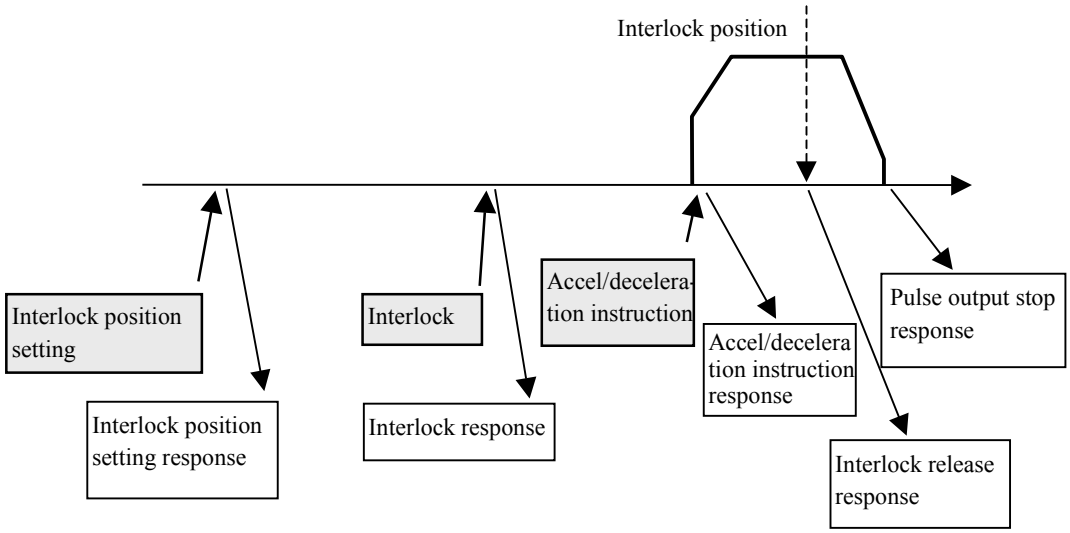
Accel/Deceleration



When Execution Is Held by RUN Signal



Execution by Event



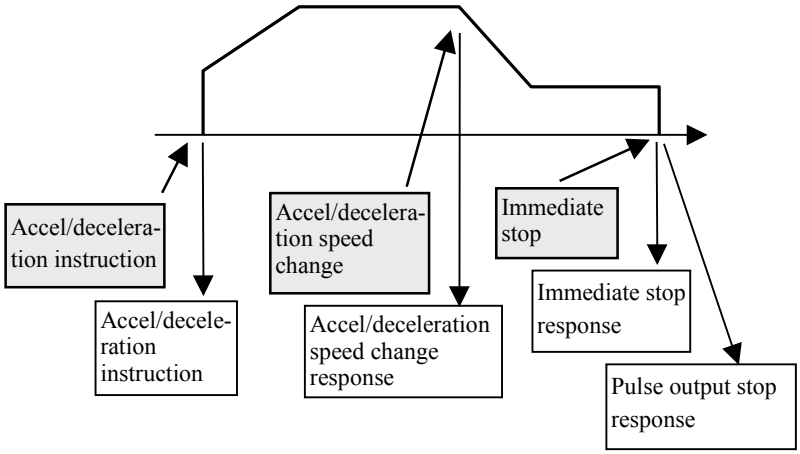
When Interlock Position Is Set

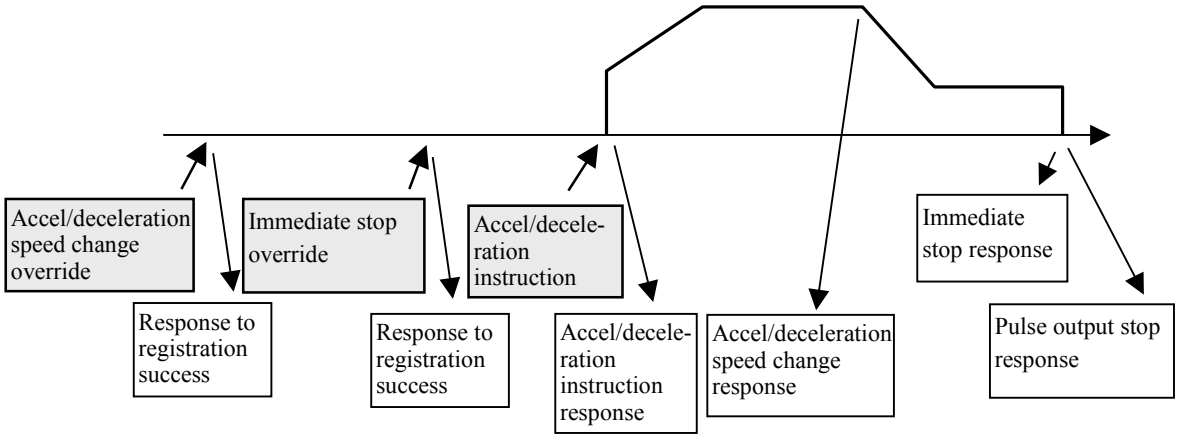
Examples of speed change and immediate stop are shown below.

An accel/deceleration change instruction and an immediate instruction are executed as soon as the instruction is issued. Using an instruction in override, speed change and stop can be performed at a predetermined position.

Please keep in mind that override is performed in the order of registration. Registration is performed in the order of speed change → stop. Should it be performed in the reversed order, that is, stop→ speed change, stopping takes place at the immediate stop position where no change in speed occurs at the speed change place. Speed change remains in the command buffer, which is executed when the next override operation is not performed or when it comes to the same place once again.

The command buffer clears the buffer with initial setting instruction/ override abnormality/ resetting.





♦ 7 Operation Setting Instructions

This instruction is necessary for operating the PPMC-312. There are "initial setting instruction", "free curve setting instruction", "in-position setting instruction", "encode setting instruction", "interlock position setting instruction", "high-speed limit setting instruction", and "END terminal setting".

● 7-1 Initial Setting Instruction

This instruction serves to prepare an accel/deceleration table. On supply of power or after release of resetting, this must be issued first so that pulse output is enabled. (This is not necessary when an free curve setting instruction has been issued.)

Unlike the conventional PPMC, tables for acceleration and deceleration can be prepared separately. To combine the acceleration and deceleration tables, set LE bit to 0. The stop pulse rate and the number of deceleration pulses are not necessary.

Output speed is calculated by "CLK bit status ÷ pulse rate". For example, with pulse rate 0x1000 being set at CLK=000 (8Mhz), 8Mhz ÷ 0x1000=1953pps is obtained.

Following writing of an instruction, acceptance of an operation instruction is enabled in 115mS for linear accel/deceleration combined, 200mS for linear accel/deceleration separate, or 175mS for sigmoid.

Restriction

- High-speed pulse rate ≥ 0x02
- Startup pulse rate ≥ high-speed pulse rate + 0x80
- and stop pulse rate ≥ high-speed pulse rate + 0x80

- Startup pulse rate ≥2 and startup/stop pulse rate ≥ 2
- (and, 2Mpps should not be exceeded.)

- No. of acceleration pulses ≥ 0x100 and no. of deceleration pulses ≥ 0x100

- When stop speed is 62.5kpps and above (when CLK=8Mhz, pulse rate of 0x80 or less), extra several tens of pulses are output in stop rate. Either increase the stop rate or confirm the position with the current position read instruction. (The current position counter serves to count all the output pulses.)

After instruction issuance, the high-speed pulse rate becomes the initial value for the high-speed limit setting instruction.

Instructions inside event and override buffers are all deleted.

Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x0e/0x09								Instruction length
0x00								Instruction code
×	×	×	LE	FO	CLK			
Startup pulse rate (upper order)								Setting range includes 0x11~0xffff.
Startup pulse rate (lower order)								
Stop pulse rate (upper order)								Setting range includes 0x11~0xffff. Not required when LE=0.
Stop pulse rate (lower order)								
High-speed pulse rate (upper order)								Setting range includes 0x2~0xffef.
High-pulse rate (lower order)								
No. of acceleration pulses (upper order)								Setting range includes 0x10~0xfffff.
No of acceleration pulses (medium order)								
No. of acceleration pulses (lower order)								
No. of deceleration pulses (upper order)								Setting range includes 0x10~0xfffff. Not required when LE=0.
No. of deceleration pulses (medium order)								
No. of deceleration pulses (upper order)								

Mnemonic	Status	Description
CLK	000	8Mhz
	001	1Mhz
	010	500khz
	011	250khz
	100	62.5kz
	101	15.625kz
	110	Leading edge of EXTCK signal
	111	Trailing edge of EXTCK signal
FO	0	Linear accel/deceleration method
	1	Sigmoid accel/deceleration emthod
LE	0	Accel/deceleration tables combined
	1	Accel/deceleration tables separately

Response at normal time:

This is a response to the initial setting instruction. When this instruction is completed, status shifts into “normal”. When a normal table is available, the initial setting instruction is newly issued. On occurrence of abnormal end, the normal table is destroyed and status moves into the initial setting. In this case, the initial setting instruction has to be ended properly once again.

Whether the accel/deceleration tables are combined or made separately, responses given show no difference.

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x01								Response length
0x00								Response code

● 7-2 Free Curve Setting Instruction

This instruction serves to freely set the accel/deceleration table. Following processing, a set value for high-speed limit becomes the high-speed pulse rate. Because many parameters are required in table preparation, one time of block writing is not enough. Writing needs to be performed by being divided into a few times.

- The number of pulses per stage is less than 0x02 ~ 0xffffff
- The number of pulses per stage x pulse rate ≥ 256
- The total number of pulses in acceleration (deceleration) (The total number of pulses per stage) is less than 0xffffff.

Issuance Procedure of Free Curve Setting Instruction:

1. Write instruction code blocks.
2. Write a pulse rate parameter for each stage by the necessary number of stages. A parameter for each stage consists of 2 bytes. Write it, starting with the stage for higher speed (smaller pulse rate).
3. Write the pulse number for each stage by the necessary number of stages. A parameter for each stage consists of 3 bytes. Write it, starting with the stage for higher speed (smaller pulse rate). When the deceleration side is not necessary, the last block has a different header value.
4. With LE=1, write the pulse number of the deceleration side by the necessary number of stages. A parameter for each stage consists of 3 bytes. Write it, starting with the stage for higher speed (smaller pulse rate). The last block has a different header value.

Instruction (Instruction code block)

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	0	Header
0x03/0x04								Instruction length
0x01								Instruction code
×	×	×	LE	×	CLK			
No. of acceleration stages								Setting range includes 0x2~0x7f
No. of deceleration stages								Setting range includes 0x2~0x7f

Mnemonic	Status	Description
CLK	000	8Mhz
	001	1Mhz
	010	500khz
	011	250khz
	100	62.5kz
	101	15.625kz
	110	Leading edge of EXTCK signal
	111	Trailing edge of EXTCK signal
LE	0	Accel/deceleration tables combined
	1	Accel/deceleration tables separately

Instruction (Parameter block with continued part)

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	0	0	Header
Undefined								Instruction length
Data								

Instruction (Parameter block: Instruction executed in the last block)

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	0	1	Header
Undefined								Instruction length
Data								

Responses on instruction issuance

A response is given on writing of each block.

The response while in writing of a pulse rate includes 1 byte which indicates a code to write a pulse rate in Status and the stage number necessary to be written. The same applies to the acceleration and deceleration pulse numbers.

On completion of writing of all the pulse rates and pulse numbers, Status becomes 0x0f and the numbers of acceleration pulses (3bytes) and deceleration pulses (3bytes) are reported.

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x03/0x08								Response length
0x01								Response code
Status								
No. of remaining stages/ no. of acceleration pulses (upper order)								
No. of acceleration pulses (medium order)								
No. of acceleration pulses (lower order)								
No. of deceleration pulses (upper order)								
No. of deceleration pulses (medium order)								
No. of deceleration pulses (lower order)								

Status	Description
0x00	Pulse rate No. of remaining stages
0x01	No. of acceleration pulses No. of remaining stages
0x02	No. of deceleration pulses No. of remaining stages
0x0f	Writing completion/ acceleration pulse number/ deceleration pulse number are reported.

● 7-3 In-position Setting Instruction

In-position control concerns operation after pulses are output.

On completion of pulse output, it is monitored if the difference between the number of output pulses and the encode pulse count value falls within the in-position range. On detection of irruption into or deviation from the set range, a response is reported. On occurrence of deviation from the in-position range, the response/IST bit in the status read instruction is set to 1. When it shifts from "outside the in-position range" to "inside", IST bit becomes 0.

Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x04								Instruction length
0x02								Instruction code
EN	RES	×	×	FO	×	×	×	
In-position range (upper order)								Setting range includes 0x1~0xffff
In-position range (lower order)								

Mnemonic	Status	Description
FO	0	Without ZCLMP signal control
	1	When the difference between the encode pulse counter value and the number of output pulses lowers the in-position rage, ZCLMP signal becomes L. Or, when END signal becomes Low, ACLMP signal is made into L.
RES	0	On changing in status due to deviation/irruption, reporting is made only once. Once a report is made, "in-position operation enable" is automatically inhibited (EN=0).
	1	A report is made every time there is a change in status of deviation/irruption.
EN	0	In-position operation disabled (initial status)
	1	In-position operation enabled

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x01								Response length
0x02								Response code

Response to change in in-position range status

Irruption into (or deviation from) the set in-position range is reported.

7	6	5	4	3	2	1	0	
0	0	1	0	0	0	1	1	Header
0x02								Response length
0xe4								Response code
0x00: inside range 0x01: outside range								Status

● 7-4 Encode Setting Instruction

A motor may not always follow pulse output of the PPMC-312. The stepping motor may bring about a step out. The encode setting instruction serves to constantly monitor the difference between the number of output pulses and the encode pulse input. On detection of a change in status caused by deviation from/irruption into the encode range, a response is issued.

On occurrence of deviation from the encode range, the response/EST bit in the status read instruction is set to 1. On changing from "outside the encode range" to "inside", EST bit becomes 0.

Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x04								Instruction length
0x03								Instruction code
EN	RES	×	×	×	×	×	×	
Encode range (upper order)								Setting range includes 0x1~0xffff
Encode range (lower order)								

Mnemonic	Status	Description
RES	0	Single. On changing in status due to deviation/irruption, the change is reported once only. Once a report is made, "in-position operation enable" is automatically inhibited.
	1	Auto. A report is made every time there is a change in status of deviation/irruption.
EN	0	Encode operation disabled (initial status)
	1	Encode operation enabled

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x01								Response length
0x03								Response code

Response to change in encode range status

Irruption into (or deviation from) the set in-position range is reported.

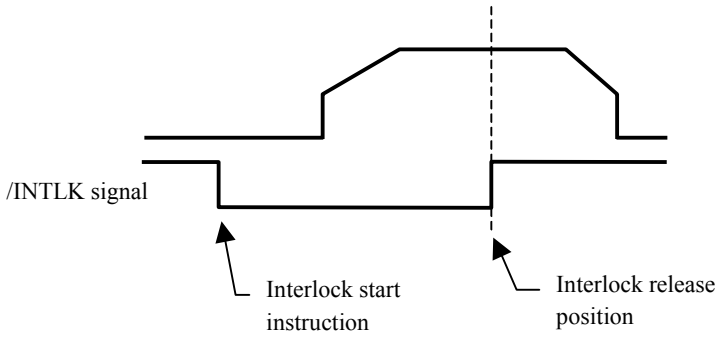
7	6	5	4	3	2	1	0	
0	0	1	0	0	0	1	1	Header
0x02								Response length
0xe3								Response code
0x00: inside range 0x01: outside range								

● 7-5 Interlock Position Setting Instruction

On reaching the set pulse position (interlock position) while in motor rotation, trigger signal (/INTLK) is output. By this signal being connected to RUN signal of any other PPMC series, the other PPMC to which /INTLK is connected can wait for start of operation until it reaches the set position (until RUN becomes High).

On supply of power or after release of resetting, 0x0 is set. Unless the interlock instruction is issued, /INTLK signal does not become Low.

The pulse position is designated in absolute position (0x0~0xffffffff). The position, once being set, remains valid until it is subject to resetting or is newly set. With toggle operation, interlock position can be cancelled.



Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x05								Instruction length
0x04								Instruction code
Interlock position (upper order)								Setting range includes 0x0~0xffffffff
Interlock position (medium upper order)								
Interlock position (medium lower order)								
Interlock position (lower order)								

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x01								Response length
0x04								Response code

Interlock completion response

7	6	5	4	3	2	1	0	
0	0	1	0	0	0	1	1	Header
0x01								Response length
0xe1								Response code

● 7-6 High-Speed Limit Setting Instruction

This instruction serves to set a speed range which makes high-speed limit (/BHL, /FHL) signals effective. Execution is enabled only while pulse output is held stopped.

The initial value of high-speed limit effective speed is equal to the pulse rate value at high speed which has been set at the time of initial setting instruction.

On detection of a high-speed limit signal which corresponds to rotation direction while pulses are output at a rate equal to or below the level set in this instruction, deceleration stop takes place even if it is during pulse output based any instruction.

On the contrary, designated pulse output continues by ignoring a high-speed limit signal which corresponds to rotation direction while pulses are output at a rate below the speed set in this instruction.

On detection of /FL,/BL during operation at a level higher than self activation frequency, the PPMC-312 immediately stops pulses. When the stepping motor stops rotating, however, offset can occur due to step out.

This high-speed limit setting instruction serves to prevent such offset caused by stepping motor step out from occurring.

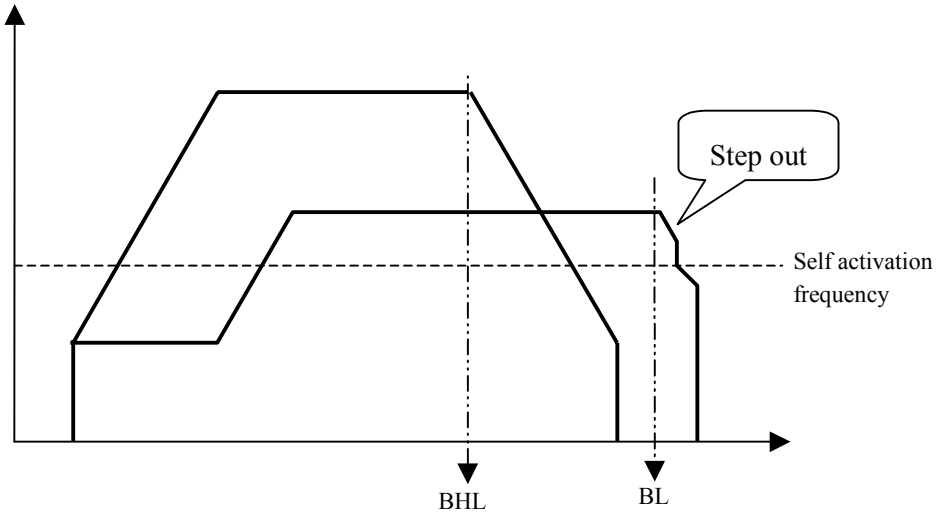


Fig. 7-1 Without High-Speed Limit Setting

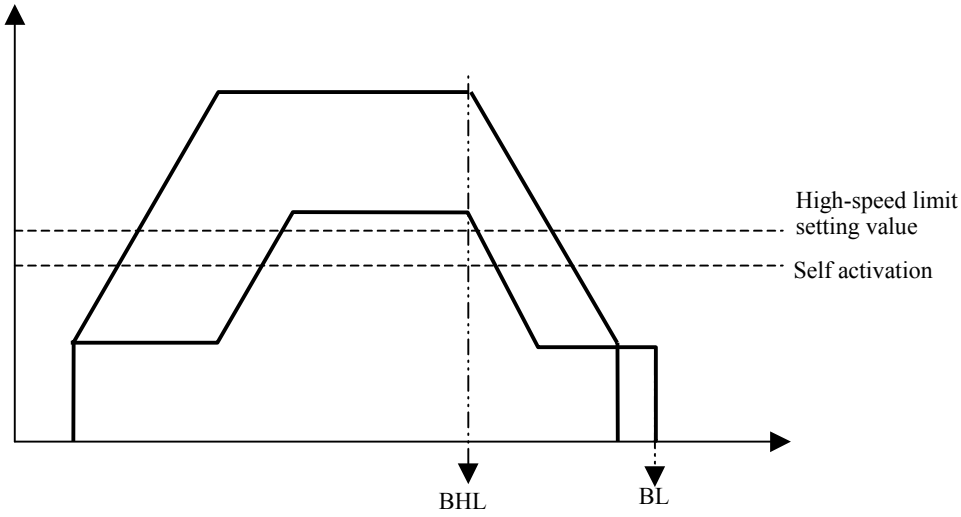


Fig. 7-2 With High-Speed Limit Setting

With the high-speed limit signal (/FHL, /BHL) being input while pulses are being output at a rate higher than the set level, deceleration stop takes place.

On power supply or following reset release, 0xffff is set. After an instruction of initial setting/free curve setting is issued, the high-speed pulse rate is stored as the pulse rate initial value.

On deceleration stopping, the pulse output stop response is given.

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x03								Instruction length
0x05								Instruction code
Pulse rate (upper order)								Setting range includes 0x2~0xffff
Pulse rate (lower order)								

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x01								Response length
0x05								Response code

● 7-7 END Terminal Setting Instruction

Whether or not END signal monitoring is available is set here. A motor may not always follow pulse output of the PPMC-312. When, at the time of large load, etc., a motor remains operating even after the PPMC-312 finishes with pulse output, inconvenience may occur with the following operation instruction being issued.

By the positioning end signal which is output by the motor driver being connected with END signal, completion of motor operation can accurately be acknowledged.

With END signal set to "with monitor", the pulse output stop response is not issued until END signal becomes Low. Even with pulse output stopped, the PPMC-312 remains in "in pulse output" status. To escape from this status, issue the END terminal setting instruction of "without monitor". Then, the pulse output stop response is issued.

With END signal set to "without monitor", the pulse output stop response is issued on completion of pulse output. On supply of power or following reset release, END signal remains in "without monitor" status.

By changing setting from "with monitor" into "without monitor" while pulse output is held stopped, you can issue the pulse output stop response.

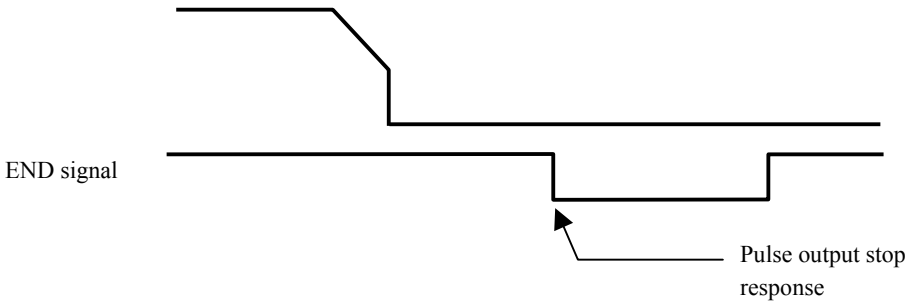


Fig. 7-3 END Signal Output

Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x02								Instruction length
0x06								Instruction code
0x00: without monitor 0x01: with monitor								

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x02								Response length
0x06								Response code
0x00: without monitor 0x01: with monitor								

♦ 8 Operation Instructions

These instructions serve to operate pulse output. There are 11 types including “immediate stop”, “deceleration stop”, “accel/deceleration”, “constant speed”, “single step”, “continuous high speed”, “continuous constant speed”, “accel/deceleration change”, “immediate speed change”, “high-speed zero return”, and “constant speed origin search”.

The pulse output instruction, after writing the last instruction block, starts to output pulses within 500μS.

On occurrence of the pulse output stop response, the next following instruction becomes ready to be received.

● 8-1 Immediate Stop Instruction

This instruction serves to immediately stop pulses in output. With this being set without override control, immediate stop takes place. Setting of override position data is not needed here.

With setting of “with override control”, it is stored in the command buffer. Immediate stop takes place due to factors set in TRG. Even with immediate stop of “with control + position” being issued, immediate stop cannot occur until the pulse reaches the set position. This instruction remains in the command buffer and immediate stop takes place when it reaches the set position by the next operation instruction. An instruction left behind is not executed until the immediate stop position is satisfied. Execution starts when the pulse reaches the set position by the operation instruction and immediate stop takes place. Please keep in mind that cancellation (deletion) of the command buffer cannot be performed.

When writing of the last instruction block is completed, pulse output is brought to a stop in 350μS. This, however, varies by pulse speed. When pulses are output at 2kpps, for example, stopping takes place in 1/2 kpps=0.5mS.

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x06/0x02								Instruction length
0x80								Instruction code
×	×	×	×	×	OV	TRG		
Stop position (upper order)								Setting range includes 0x0~0xffffffff
Stop position (lower order)								
Stop position (medium lower order)								
Stop position (lower order)								

Mnemonic	Status	Description
OV	0	Without override control
	1	With override control
TRG	00	When pulse position has reached override position.
	01	When event input 1 becomes L.
	10	When pulse position has reached override position.
	11	When event input 2 becomes L.

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x01								Response length
0x80								Response code

● 8-2 Deceleration Stop Instruction

This instruction serves to decelerate and stop pulses in output.

With this set to “without override control”, deceleration stop takes place. Override position data need not be set here.

With setting of “with override control”, the instruction is stored in the command buffer. With factors set in TRG, deceleration starts and stopping takes place.

With deceleration stop of “with control + position” being issued, deceleration stop cannot occur until the pulse reaches the set position. The instruction left behind is not executed until it reaches the deceleration start position. Execution starts when the pulse reaches the set position by the operation instruction and deceleration stop takes place. Please keep in mind that cancellation (deletion) of the command buffer cannot be performed.

The high-speed limit (/BHL, /FHL) and deceleration stop instructions, in view of deceleration stop of pulse output, both perform the same processing. Once deceleration stop processing is started by either one of them, the other is ignored until either pulse output is stopped or a pulse rate is changed by the immediate speed change instruction. There is no response to the ignored processing. Example: Even with /BHL being input while in deceleration by the deceleration stop instruction, there is no response to deceleration stop by /BHL.

Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x06/0x02								Instruction length
0x81								Instruction code
×	×	×	×	×	OV	TRG		
Deceleration start position (upper order)								Setting range includes 0x0~0xffffffff
Deceleration start position (medium upper order)								
Deceleration start position (medium lower order)								
Deceleration start position (lower order)								

Mnemonic	Status	Description
OV	0	Without override control
	1	With override control
TRG	00	When pulse position has reached override position.
	01	When event input 1 becomes L.
	10	When pulse position has reached override position.
	11	When event input 2 becomes L.

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x01								Response length
0x81								Response code

● 8-3 Accel/Deceleration Instruction

This instruction performs accel/deceleration according to the accel/deceleration table prepared by the initial setting instruction.

On receipt of this instruction, pulse output starts at a startup speed as designated by the initial setting instruction, which is accelerated with the designated number of accel/deceleration pulses up to the high speed rate. It is then subject to high speed operation and, on reaching to the deceleration start point, deceleration takes place with the number of accel/deceleration (deceleration) pulses until reaching the startup (deceleration) speed, finally ending pulse output.

With setting of "with override control", this instruction is stored in the command buffer, which can be started with an event input.

Actual operation varies by the number of operation pulses, startup pulse rate, and stop pulse rate. Differences in operation caused by different startup/stop pulse rates are explained below.

After writing of all parameters is completed till pulse output starts, the following time periods are required:

	Triangular/Right-Angled Triangular Operation	Trapezoidal Operation
Without rate assignment	700μS	100μS
With rate assignment	1400μS	900μS

■ 8-3.1 Trapezoidal Operation

When the number of operation pulses set in the accel/deceleration instruction is larger than the sum of the numbers of acceleration and deceleration pulses set in the initial setting instruction, trapezoidal operation is performed.

No. of operation pulses ≥ no. of acceleration pulses + no. of deceleration pulses

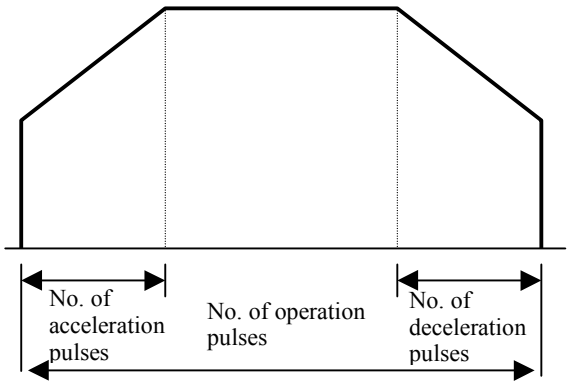


Fig. 8-1 Trapezoidal Operation

■ 8-3.2 Triangular Operation

Operation without constant speed operation at a high speed is referred to as triangular operation. With a startup pulse rate being given, startup → acceleration → deceleration → stopping with a stop pulse rate take place.

The PPMC-312 can set the startup pulse rate and the deceleration pulse rate each differently through using the initial setting instruction. The pulse difference between the two rates is referred to as "rate difference pulse number".

When the number of operation pulses set in the accel/deceleration instruction is equal to or smaller than the sum of the numbers of acceleration and deceleration pulses set in the initial setting instruction, triangular operation is performed.

The ratio between the acceleration pulse number and deceleration pulse number set in the initial setting instruction is also applied to the numbers of acceleration and deceleration pulses in actual operation.

No. of operation pulses ≤ no. of acceleration pulses + no. of deceleration pulses, and

No. of operation pulses > rate difference pulse number

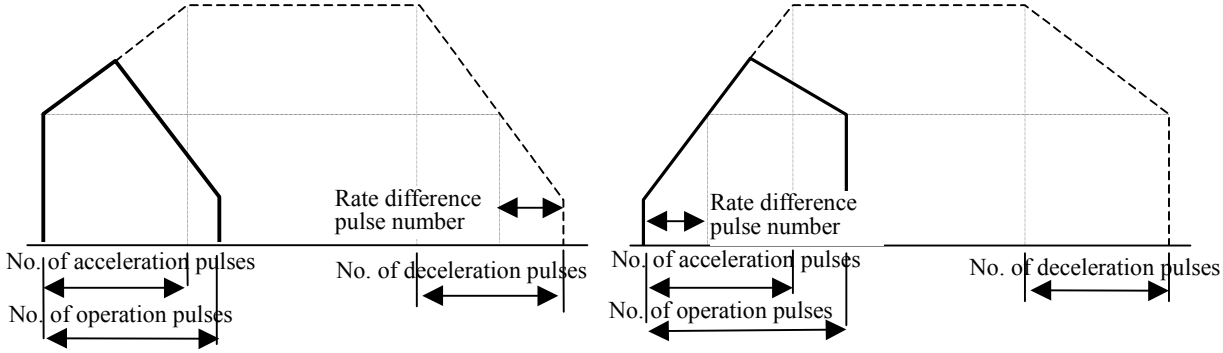


Fig. 8-2 Triangular Operation

■ 8-3.3 Right-Angled Triangular Operation

Deceleration stop without acceleration or immediate stop during acceleration is referred to as "right-angled triangular operation".

The PPMC-312 can set the startup pulse rate and the deceleration pulse rate each differently through using the initial setting instruction. The pulse difference between the two rates is referred to as "rate difference pulse number".

When the startup pulse rate and the deceleration pulse rate differ, the number of operation pulses is smaller than the rate difference pulse number. Operation, therefore, consists of either startup → deceleration → stop or startup → acceleration → stop.

No. of operation pulses ≤ rate difference pulse number

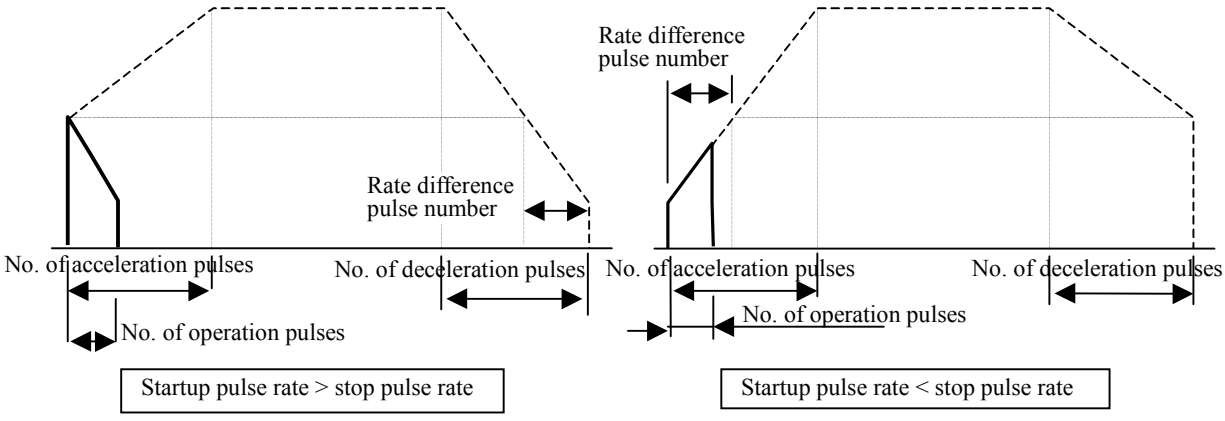


Fig. 8-3 Right-Angled Triangular Operation

With setting of a high-speed pulse rate which is equal to or higher than that assigned in the initial setting instruction, operation is performed at the set high speed pulse rate. To operate at the high-speed pulse rate of the initial setting, set 0x00 for the high-speed rate.

Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x08								Instruction length
0x82								Instruction code
×	DIR	×	×	×	OV	TRG		
High-speed pulse rate (upper order)								
High-speed pulse rate (lower order)								
No. of operation pulses (upper order)								Setting range includes 0x0~0xffffffff
No. of operation pulses (medium upper order)								
No. of operation pulses (medium lower order)								
No. of operation pulses (lower order)								

Mnemonic	Status	Description
OV	0	Without override control *
	1	With override control
TRG	00	Disabled setting (parameter abnormality)
	01	When event input 1 becomes L.
	10	Disabled setting (parameter abnormality)
	11	When event input 2 becomes L.
DIR	0	CW direction
	1	CCW direction

* Where OV=0, TRG setting is ignored. Normally, set 00.

No. of Operation Pulses		Driving System
No. of accel. pulses + No. of decel. pulses ≤ No. of op. pulses		Trapezoidal
Startup pulse rate =decel. pulse rate	No. of accel. pulses + no. of decel. pulses > no. of op. pulses	Triangular
Startup pulse rate ≠ decel. pulse rate	Rate difference pulse number ≤ no. of op. pulses	Triangular
	Rate difference pulse number > no. of op. pulses	Right-angled triangular

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x02								Response length
0x82								Response code
Status								Status

Status	Trapezoidal driving	Triangular driving	Right-angled triangular (decel.)	Right-angled triangular (accel.)
Pulse Output Start	0x00	0x10	0x20	0x40
Pulse Output Hold	0x01	0x11	0x21	0x41

These responses are given against the accel/deceleration instruction. A response is sent when pulse output is ready to start. The driving system is reported in the response.

When a designated high-speed rate is smaller than the value in the 2nd stage (immediately above the startup pulse rate) on the table prepared by the initial setting instruction, acceleration cannot take place and a response of right-angled triangular operation is sent out.

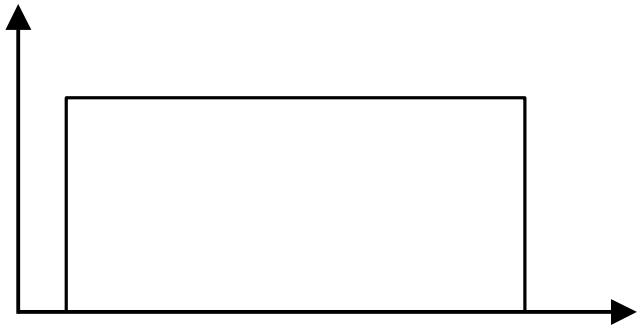
The accel/deceleration operation instruction can be registered in the command buffer. When it is registered in the command buffer, a response either of registration success or of registration failure is given.

● 8-4 Constant Speed Instruction

This instruction serves to output pulses of the set rate by the set number. The pulse rate value needs to be within the rate range assigned by the initial setting instruction (free curve setting instruction). To operate at the high-speed pulse rate of initial setting, be sure to set 0x00 to the high-speed rate.

With setting of "with override control", it is stored in the command buffer.

From completion of writing of all the parameters to output of pulses, it requires 400μS.



Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x08								Instruction length
0x83								Instruction code
×	DIR	×	×	×	OV	TRG		
Pulse rate (upper order)								
Pulse rate (lower order)								
No. of pulses (upper order)								Setting range includes 0x0~0xffffffff
No. of pulses (medium upper order)								
No. of pulses (medium lower order)								
No. of pulses (lower order)								

Mnemonic	Status	Description
OV	0	Without override control *
	1	With override control
TRG	00	Disabled setting (parameter abnormality)
	01	When event input 1 becomes L.
	10	Disabled setting (parameter abnormality)
	11	When event input 2 becomes L.
DIR	0	CW direction
	1	CCW direction

* Where OV=0, TRG setting is ignored. Normally, set 00.

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x02								Response length
0x83								Response code
0x00: Output start 0x01: Output standby								Status

● 8-5 Single Step Instruction

Output of one pulse is performed in an assigned direction. Pulses are output within the startup pulse rate width assigned by the initial setting instruction. For repeated operation, a rate higher than the startup pulse rate cannot be applied.

Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x02								Instruction length
0x84								Instruction code
×	DIR	×	×	×	×	×	×	

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x01								Response length
0x84								Response code

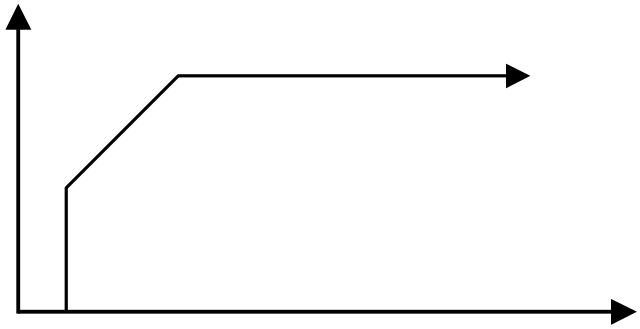
● 8-6 Continuous High Speed Instruction

Acceleration takes place according to the accel/deceleration table prepared by the initial setting instruction. Following acceleration to the set pulse rate, pulse output is continued.

To operate at the high-speed pulse rate of initial setting, be sure to set 0x00 to the high-speed rate. Use it within the rate range assigned by the initial setting instruction.

With setting of "with override control", it is stored in the command buffer.

From completion of writing of all the parameters to output of pulses, 400μS is required when it is "with rate assignment", and 60μS when it is "without rate assignment".



Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x04								Instruction length
0x85								Instruction code
×	DIR	×	×	×	OV	TRG		
Rate (upper order)								
Rate (lower order)								

Mnemonic	Status	Description
OV	0	Without override control *
	1	With override control
TRG	00	Disabled setting (parameter abnormality)
	01	When event input 1 becomes L.
	10	Disabled setting (parameter abnormality)
	11	When event input 2 becomes L.
DIR	0	CW direction
	1	CCW direction

* Where OV=0, TRG setting is ignored. Normally, set 00.

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x02								Response length
0x85								Response code
0x00: Output start 0x01: Output standby								Status

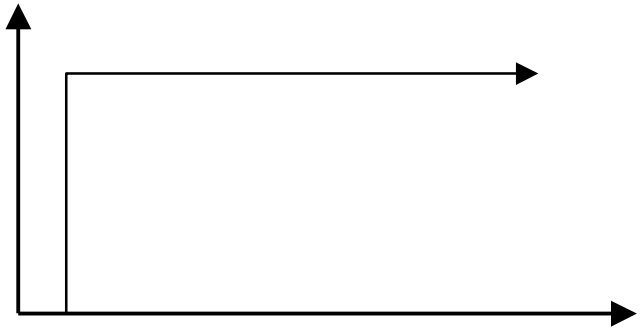
● 8-7 Continuous Constant Speed Instruction

This instruction serves to keep outputting pulses at the set rate.

The rate needs to be within the rate range assigned by the initial setting instruction (free curve setting instruction). To operate at the high-speed pulse rate of initial setting, be sure to set 0x00 to the high-speed rate.

With setting of "with override control", it is stored in the command buffer.

From completion of writing of all the parameters to output of pulses, it requires 400μS.



Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x04								Instruction length
0x86								Instruction code
×	DIR	×	×	×	OV	TRG		
Pulse rate (upper order)								Setting range includes 0xa~0xffff
Pulse rate (lower order)								

Mnemonic	Status	Description
OV	0	Without override control *
	1	With override control
TRG	00	Disabled setting (parameter abnormality)
	01	When event input 1 becomes L.
	10	Disabled setting (parameter abnormality)
	11	When event input 2 becomes L.
DIR	0	CW direction
	1	CCW direction

* Where OV=0, TRG setting is ignored. Normally, set 00.

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x02								Response length
0x86								Response code
0x00: Output start 0x01: Output standby								Status

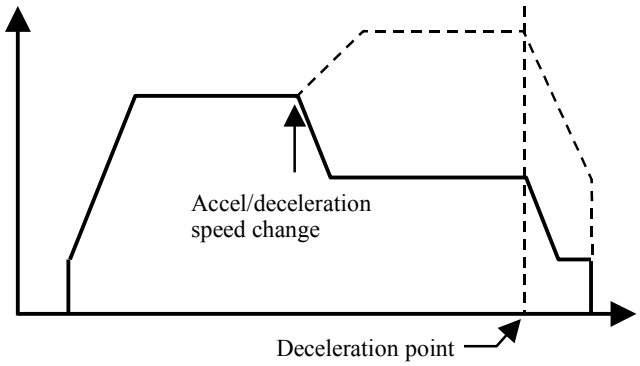
● 8-8 Accel/Deceleration Change Instruction

This instruction changes a pulse rate in output into the set rate through accel/deceleration. The set rate needs to be within the rate range assigned by the initial setting instruction (free curve setting instruction).

With setting of "with override control", it is stored in the command buffer, enabling speed change at a predetermined place. With the instruction length of 0x08, set the override position.

To immediately perform speed change, override position need not be set. (At this time, the instruction length of 0x04 is set.) To change speed at the pulse position, set the override position.

When speed is changed in operation of an instruction whose number of operation pulses has been set, no change occurs in the set number of operation pulses. As the deceleration points are the same, however, stopping can take place at a speed higher than the stop pulse rate or constant speed operation at a stop pulse rate can take place.



Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x04/0x08								Instruction length
0x87								Instruction code
×	×	×	×	×	OV	TRG		
Rate (upper order)								Setting range includes 0xa~0xffff
Rate (lower order)								
Override position (upper order)								Setting range includes 0xa~0xffffffff
Override position (medium upper order)								
Override position (medium lower order)								
Override position (lower order)								

Mnemonic	Status	Description
OV	0	Without override control *
	1	With override control
TRG	00	When pulse position has reached override position.
	01	When event input 1 becomes L.
	10	When pulse position has reached override position.
	11	When event input 2 becomes L.

* Where OV=0, TRG setting is ignored. Normally, set 00.

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x01								Response length
0x87								Response code

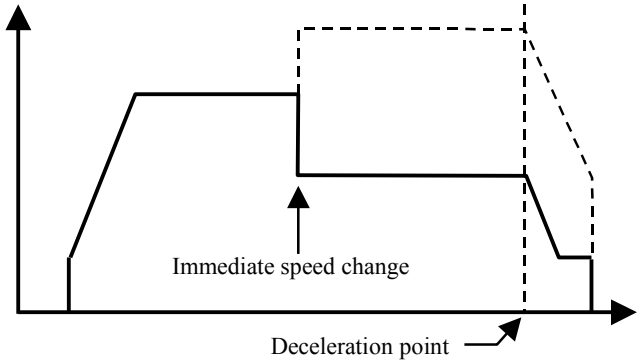
● 8-9 Immediate Speed Change Instruction

This instruction serves to immediately change a pulse rate in output to the set rate. The set rate needs to be within the rate range assigned by the initial setting instruction (free curve setting instruction).

With setting of "with override control", it is stored in the command buffer, enabling speed change at a predetermined place. With the instruction length of 0x08, set the override position.

To immediately perform speed change, override position need not be set. (At this time, the instruction length of 0x04 is set.)

When speed is changed in operation of an instruction whose number of operation pulses has been set, no change occurs in the set number of operation pulses. As the deceleration points are the same, however, stopping can take place at a speed higher than the stop pulse rate or constant speed operation at a stop pulse rate can take place.



Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x04/0x08								Instruction length
0x88								Instruction code
×	×	×	×	×	OV	TRG		
Pulse rate (upper order)								Setting range includes 0xa~0xffff
Pulse rate (lower order)								
Override position (upper order)								Setting range includes 0xa~0xffffffff
Override position (medium upper order)								
Override position (medium lower order)								
Override position (lower order)								

Mnemonic	Status	Description
OV	0	Without override control *
	1	With override control
TRG	00	When pulse position has reached override position.
	01	When event input 1 becomes L.
	10	When pulse position has reached override position.
	11	When event input 2 becomes L.

* Where OV=0, TRG setting is ignored. Normally, set 00.

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x01								Response length
0x88								Response code

● 8-10 Constant Speed Origin Search Instruction

This instruction serves to search the /ORG signal. Pulse output is started at an assigned pulse rate and in an assigned direction and, on detection of the /ORG signal, it is brought to an immediate stop.

The pulse rate available for setting should be within the pulse rate range (between the higher of the startup and stop pulse rates and the high-speed pulse rate) assigned by the initial setting instruction (free curve setting instruction).

With IDX bit=1, origin search is performed with use of AND of /ORG signal Low and /INDEX signal Low. Detection of only one of them does not represent the origin. Pulse output continues in an assigned direction until the both are simultaneously detected. Each signal required pulse width of 200μS and above.

Fig. 8-4 Origin Recognition (ORG and INDEX)

Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x04								Instruction length
0x10								Instruction code
×	DIR	×	×	×	×	×	IDX	
Rate (upper order)								Setting range includes 0x2~0xffff
Rate (lower order)								

Mnemonic	Status	Description
IDX	0	Without /INDEX detection
	1	With /INDEX detection
DIR	0	CW direction
	1	CCW direction

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x02								Response length
0x10								Response code
0x00: Output start 0x01: Output standby								Status

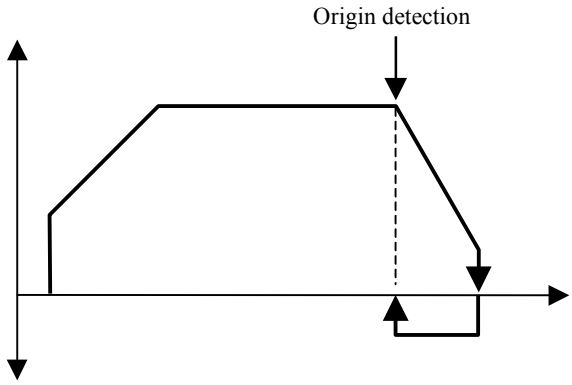
● 8-11 High-Speed Zero Return Instruction

Acceleration operation starts at a startup pulse rate in an assigned direction. On detection of the origin, deceleration and, then, stopping at the stop pulse rate takes place. This is immediately followed by constant speed operation in the reverse direction at the startup pulse rate, being brought to immediate stop when the origin is found.

When an assigned rate is 0x0000, a high-speed pulse rate set by the initial setting instruction is used. The rate assigned here has to be used within the rate having been set by the initial setting instruction.

In the same way as the constant speed origin search instruction, with IDX=1, origin search can be performed together with INDEX signal. In this case, combined use of INDEX signal is necessary also on deceleration start and on immediate stop.

From completion of writing of all the parameters to output of pulses, 400μS is required when it is "with rate assignment", and 60μS when it is "without rate assignment".



Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x04								Instruction length
0x11								Instruction code
×	DIR	×	×	×	×	×	IDX	
Rate (upper order)								Setting range includes 0x2~0xffff
Rate (lower order)								

Mnemonic	Status	Description
IDX	0	Without /INDEX detection
	1	With /INDEX detection
DIR	0	CW direction
	1	CCW direction

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x02								Response length
0x11								Response code
0x00: Output start 0x01: Output standby								Status

♦ 9 Terminal Control Instruction

● 9-1 Excitation ON Instruction

With this instruction, /EXON signal is made into Low.

Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x01								Instruction length
0x20								Instruction code

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x01								Response length
0x20								Response code

● 9-2 Excitation OFF Instruction

With this instruction, /EXON signal is made into High. Pulse output is performed regardless of issuance of the excitation OFF instruction.

Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x01								Instruction length
0x21								Instruction code

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x01								Response length
0x21								Response code

● 9-3 Counter Clear instruction

This instruction serves to hold the /CCLR signal in Low for about 20mS.

Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x01								Instruction length
0x22								Instruction code

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x01								Response length
0x22								Response code

● 9-4 Zero Clamp Instruction

With this instruction, ZCLMP signal is made into Low. The ZCLMP signal, on start of pulse output, is automatically set to High. This is in sync with switchover of the DIR signal.

Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x01								Instruction length
0x23								Instruction code

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x01								Response length
0x23								Response code

● 9-5 Alarm Clear Instruction

With this instruction, /ACLR signal is held in Low for about 20mS, where abnormal status of the PPMC-312 is cleared.

Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x01								Instruction length
0x24								Instruction code

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x01								Response length
0x24								Response code

● 9-6 Generalized Input Instruction

This instruction serves to read data from the generalized input port. With AUXI terminal in Low, it is “02. With the terminal in High, it is “1”.

Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x01								Instruction length
0x25								Instruction code

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x02								Response length
0x25								Response code
AUXI7	AUXI6	AUXI5	AUXI4	AUXI3	AUXI2	AUXI1	AUXI0	

● 9-7 Generalized Output Instruction

With this instruction, assigned data are output from the generalized output terminal.

Bits 0~5 correspond to the generalized output terminals /AUXO0~5. A terminal corresponding the bit set in "1" becomes Low. With the bit reset to "0", it becomes High.

On supply of power or following /RESET signal release, all the bits are "0", therefore, holding the generalized output terminals in High. Instruction, when issued, is reflected on the output terminal in 300μS.

Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x02								Instruction length
0x26								Instruction code
×	×	/AUXO 5	/AUXO 4	/AUXO 3	/AUXO 2	/AUXO 1	/AUXO 0	

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x01								Response length
0x26								Response code

● 9-8 Control Terminal Read Instruction

This instruction serves to read status of the control input terminal. "Low" of each terminal is read as "0", and "High" as "1".

Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x01								Instruction length
0x27								Instruction code

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x03								Response length
0x27								Response code
/ALM	/END	/INDEX	/ORG	/FHL	/BHL	/FL	/BL	
EVT1	EVT0	RUN	×	×	×	×	×	

● 9-9 Mask Setting Instruction

The generalized input terminal, by detecting changes of High/Low, can send a response. (See "Response to Generalized Input Change".)

With this instruction, a generalized input terminal whose change is to be detected is set. The bit assigned with "1" is subject to sampling and, every time the input signal status changes, a response to generalized input change is reported.

On supply of power and following resetting release, it is set to "0".

As sampling of the generalized input terminal is an operation with lower priority for the PPMC-312, it is performed for max. 1 second and at intervals of min. 50μ.

Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x03								Instruction length
0x28								Instruction code
×	RES	×	×	×	×	×	×	
Mask data								

Mnemonic	Status	Description
RES	0	Single. When a generalized input terminal for which "1" has been set by the mask setting instruction changes, the input change response is given once only.
	1	Auto. Every time a generalized input terminal for which "1" has been set by the mask setting instruction changes, the input change response is given.

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x01								Response length
0x28								Response code

♦ 10 Control Aiding Instructions

● 10-1 Table Read Instruction

This instruction serves to read an accel/deceleration table.

When neither the initial setting instruction nor the free curve setting instruction has been issued, the abnormality response is given.

The following can be read out from the first response block:
No. of stages of pulse rate (1 byte)
No. of stages of acceleration-side pulse number (1 byte)
No. of stages of deceleration-side pulse number (1 byte)
Total no. of acceleration-side pulses (3 bytes)
total no. of deceleration-side pulses (3 bytes)

Next, information of all the stages is continuously output in each of the corresponding items. The information is output, starting with a stage with the highest speed toward ones with lower speed.

As different items never coexist in one block, always confirm the response length.
Pulse rate of each stage (2 bytes/stage)
No. of acceleration-side pulses (3 bytes/stage)
No. of deceleration-side pulses (3 bytes/stage)

Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x01								Instruction length
0x41								Instruction code

Responses on instruction issuance

7	6	5	4	3	2	1	0
0x02							
0x0A							
0x41							
No. of stages of pulse rate							
No. of stages of acceleration-side pulse number							
No. of stages of deceleration-side pulse number							
Total no. of acceleration-side pulses (upper order)							
Total no. of acceleration-side pulses (medium order)							
Total no. of acceleration-side pulses (lower order)							
Total no. of deceleration-side pulses (upper order)							
Total no. of deceleration-side pulses (medium order)							
Total no. of deceleration-side pulses (lower order)							

7	6	5	4	3	2	1	0
0x00							
Variable							
0x01							
127th stage rate upper order							
127th stage rate lower order							
126th stage rate upper order							
126th stage rate lower order							
125th stage rate upper order							
125th stage rate lower order							
124th stage rate upper order							
124th stage rate lower order							
...							

Pulse rates numbers

...

7	6	5	4	3	2	1	0
0x00							
Variable							
0x02							
127th stage pulse number upper order							
127th stage pulse number medium order							
127th stage pulse number lower order							
126th stage pulse number upper order							
126th stage pulse number medium order							
126th stage pulse number lower order							
125th stage pulse number upper order							
125th stage pulse number medium order							
125th stage pulse number lower order							
...							

Acceleration pulse number

7	6	5	4	3	2	1	0
0x00							
Variable							
0x03							
127th stage pulse number upper order							
127th stage pulse number medium order							
127th stage pulse number lower order							
126th stage pulse number upper order							
126th stage pulse number medium order							
126th stage pulse number lower order							
125th stage pulse number upper order							
125th stage pulse number medium order							
125th stage pulse number lower order							
...							

Deceleration pulse number

Note 1
The last data header is 0x01.

● *10-2 Current Position Read Instruction*

Current position is read out. There are two positions including the current position by pulse output and the encode position by pulse input. You have to specify which of them is to be read.

Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x02								Instruction length
0x42								Instruction code
0x00: current position 0x01: encode position								

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x06								Response length
0x42								Response code
0x00: current position 0x01: encode position								
Position (upper order)								
Position (medium upper order)								
Position (medium lower order)								
Position (lower order)								

● *10-3 Current Position Setting Instruction*

This instruction serves to set the current position.

This instruction is capable to set the current position on a side of pulse output and the position from the encode.

Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x06								Instruction length
0x43								Instruction code
0x00: current position 0x01: encode position								
Position (upper order)								Setting range includes 0x0~0xffffffff
Position (medium upper order)								
Position (medium lower order)								
Position (lower order)								

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x01								Response length
0x43								Response code

● *10-4 Interlock Instruction*

This instruction serves to start interlock control. With control start, the INTLK signal becomes Low. With control stop, the INTLK signal becomes High.

Interlock in-control status can be confirmed with a response of the status instruction.

When the pulse position reaches the value set by the interlock position setting instruction, interlock is automatically brought to a control stop and he INTLK signal becomes High.

Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x01								Instruction length
0x44								Instruction code

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x02								Response length
0x44								Response code
0x00: control stop 0x01: control start								

● *10-5 Version Read Instruction*

This instruction serves to read the program version of the PPMC-312.

Use this to confirm your program when version upgrading, etc. of PPMC-312 has been performed.

Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x01								Instruction length
0x40								Instruction code

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x02								Response length
0x40								Response code
Version								

● *10-6 Synchronous Operation Monitor Instruction*

With this instruction, whether or not RUN signal is monitored is set.

In default setting, RUN signal is subject to monitoring. When RUN signal is Low in pulse output, the pulse output is held. With synchronous operation set to "without monitor", pulse output is performed regardless of RUN signal.

Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x01								Instruction length
0x45								Instruction code

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x02								Response length
0x45								Response code
0x00: in monitor 0x01 no monitor								

● *10-7 Event Setting Instruction*

This instruction serves to register an event. As for details of events, see “Events”.

When all the four buffers are occupied, the buffer-full response is given.

“Data” section must be fully entered including the header of each instruction.

Instruction length consists of the number of bytes which is smaller than that of the instruction code. Therefore, it is equal to “Data number + 0x02”.

For event deletion, Data are not required.

Deletion starts with the one with the largest instruction number registered in descending order.

The instruction numbers are subject to deletion one at a time.

Instruction (on registration)

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
Instruction length								Instruction length
0x49								Instruction code
0	0	0	REG	0	0	1	No	
Data								Write all including, also, the header.
▪ ▪ ▪								
Data								

Mnemonic	Status	Description
No	0	Event buffer No.0
	1	Event buffer No.1
REG	0	Event registration
	1	Event deletion

Example: An accel/deceleration instruction is registered in No.1 event buffer:

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x0c								Instruction length
0x49								Instruction code
0	0	0	0	0	0	1	1	
0	0	0	0	0	0	1	1	Header of accel/deceleration instruction
0x08								Instruction length of accel/deceleration instruction
0x82								Instruction code of accel/deceleration instruction
×	DIR	×	×	×	OV	TRG		
High-speed pulse rate (upper order)								
High-speed pulse rate (lower order)								
No. of operation pulses (upper order)								
No. of operation pulses (medium upper order)								
No. of operation pulses (medium lower order)								
No. of operation pulses (lower order)								

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x02								Response length
0x49								Response code
Status								

Status	
0x00	Registration has succeeded.
0x01	Registration has failed as the trigger is used by override.
0x02	All the buffers are occupied.
0x03	Deletion has succeeded.
0x04	After deletion, buffers are empty.
0x05	Deletion is not possible as nothing has been registered.

● *10-8 Event Read Instruction*

Event information having been registered is read out.

Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x01								Instruction length
0x47								Instruction code
Event No.(0 or 1)								

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x04 ~ 0x07								Response length
0x47								Response code
No. of instructions registered								
Instruction No. to be executed next								
Instruction code of Instruction No.1 registered								
Instruction code of Instruction No.2 registered								
Instruction code of Instruction No.3 registered								
Instruction code of Instruction No.4 registered								

● *10-9 Status Read Instruction*

With this instruction, status data inside the PPMC-312 are read.

Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x01								Instruction length
0x4d								Instruction code

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x04								Response length
0x4d								Response code
CWS	ILS	×	ENC	INP	END	EST	IST	Operation status
×	×	RUN	ALM	FIX	ACC	NOR	INT	Condition status
EV1E	EV0E	EV1O	EV0O	CBCNT				Buffer status

Operation status

Mnemonic	Status	Description
CWS	0	RUN signal is monitored on start of pulse output.
	1	RUN signal is ignored on start of pulse output.
ILS	0	Normal status
	1	Interlocked On interlock release or following control, back to 0.
ENC	0	Encoder monitor control is inhibited.
	1	Encoder monitoring is under control.
INP	0	In-position monitor control is inhibited.
	1	In-position monitor control is in operation.
END	0	Without END signal control
	1	With END signal control
EST	0	Within encode range. Undefined when encode control is inhibited.
	1	Outside encode range. Undefined when encode control is inhibited.
IST	0	Within in-position range. Undefined when in-position control is inhibited.
	1	Outside in-position. Undefined when in-position control is inhibited.

Condition status

Each bit, when in "1", indicates the specified condition. For details, see "Status".

Mnemonic	Status	Description
RUN	1	Waiting RUN
ALM	1	Alarm status
FIX	1	Pulses in output at constant speed
ACC	1	During acceleration or deceleration
NOR	1	Normal status
INI	1	Initial status

Buffer status

Mnemonic	Status	Description
EV1E	0	No registered command exists in Event Buffer 1.
	1	Event command has been registered in Event Buffer 1.
EV0E	0	No registered command exists in Event Buffer 0.
	1	Event command has been registered in Event Buffer 0.
EV1O	0	No registered command exists in Event Buffer 1.
	1	Override command has been registered in Event Buffer 1.
EV0O	0	No registered command exists in Event Buffer 0.
	1	Override command has been registered in Event Buffer 0.
CBCNT	The number of override commands having been registered in command buffers. With registration, +1. With execution, - 1.	

● *10-10* Resetting

With this instruction, software resetting is performed. The same status as that of power cutoff or following resetting release is created. On completion of resetting processing, a response is given.

Instruction

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	1	Header
0x01								Instruction length
0xff								Instruction code

Responses on instruction issuance

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x01								Response length
0xff								Response code

♦ 11 Responses

● 11-1 Response to Pulse Output Stop

A response is given when pulse output has been stopped.

7	6	5	4	3	2	1	0	
0	0	1	0	0	0	1	1	Header
0x06								Response length
0xe0								Response code
ALS	NOS	ORG	LOW	DEC	STP	DEL	STL	
Pulse stop position (upper order)								
Pulse stop position (medium upper order)								
Pulse stop position (medium lower order)								
Pulse stop position (lower order)								

Each bit, when in "1", indicates the specified condition.

Mnemonic	Status	Description
ALS	1	Stop by the alarm signal.
NOS	1	Normal stop by the output stop position.
ORG	1	Stop by the origin input.
LOW	1	Immediate stop by the deceleration start as the current pulse rate is lower than the stop pulse rate.
DEC	1	Stop by the deceleration stop instruction.
STP	1	Stop by the immediate stop instruction.
DEL	1	Deceleration stop by the limit.
STL	1	Immediate stop by the limit.

● 11-2 Response to Hold Execution

The operation instruction having been held by /RUN signal notifies of start of execution.

7	6	5	4	3	2	1	0	
0	0	1	0	0	0	1	1	Header
0x01								Response length
0xe2								Response code

● *11-3* Response to Registration Success

It is reported that the instruction has been properly registered in the command buffer.

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x01								Response length
0xe8								Response code

● *11-4* Response to Registration Failure

It is reported that the instruction registration in the command buffer has failed.

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x02								Response length
0xf7								Response code
Status								

Status	Cause
0x00	Command buffer full
0x01	Assigned /EVT0, /EVT1 are now used in the event buffer.

● *11-5* Size Abnormality Response

A response is given when an instruction length of 31 bytes or above has been written.

Status transition is not performed. Attached parameters include two matters which are the number of commands and command data.

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x02								Response length
0xf6								Response code
No. of instructions								← This occurred as the number here is 31 or above.

● *11-6 Parameter Abnormality Response*

It is reported that abnormality is observed in an instruction parameter.

The instruction and parameter having been judged as abnormal by error status are sent back.

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
Undefined								Response length
0xf0								Response code
Error status								
Instruction								
▪ ▪ ▪								
Instruction								

Error status

No	Details
0x00	Initial setting Data length is incorrect.
0x01	Initial setting Pulse rate data are out of range.
0x02	初 Initial setting Difference between the larger of the startup pulse rate and the stop pulse rate, and the high-speed pulse rate is 127 or below.
0x03	Initial setting The startup pulse rate or the stop pulse rate is equal to or below the high-speed pulse rate.
0x04	Initial setting As the startup pulse or the stop pulse rate is close to the high-speed pulse rate, an accel/deceleration table cannot be prepared.
0x05	Initial setting Operation is disabled because a stage whose number of operation pulses is 2 or below has been made in the accel/deceleration table. Increase the number of acceleration (deceleration) pulses and re-issue the initial setting instruction.
0x10	Free curve The command header is incorrect.
0x11	Free curve The number of stages is outside the range of 2~127.
0x15	Free curve A value of 31(dec) or above has been set in the instruction length.
0x16	Free curve Pulse rate setting data are too big.
0x17	Free curve The command header is incorrect.
0x18	Free curve A pulse rate which is equal to or below 1 exists among the accel/deceleration stages.
0x19	Free curve A pulse number which is equal to or below 1 exists among the accel/deceleration stages.
0x1a	Free curve The sum of the numbers of pulses of each stage has exceeded 0xffffff.
0x20	Pulse rate setting is outside the range.
0x21	An instruction impossible for pulse position assignment has been used in override. For example, the accel/deceleration instruction has been started in override.
0x23	The number of operation pulses is incorrect.
0x24	A number which does not exist in event registration has been assigned in event reading.
0x25	An error exists in instruction length of the event setting instruction.
0x26	The in-position range is 0.
0x27	The encode range is 0.

● *11-7 Status Abnormality Response*

On issuance of an instruction, it may not be issued properly depending on the PPMC-312 conditions. Also, when a header used in instruction reading is incorrect, this status abnormality response is reported.

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x04								Response length
0xf1								Response code
Status								Status Information 1
×	×	RUN	ALM	FIX	ACC	NOR	INT	Status Information 2
Instruction code								1byte

Status	
0x00	Status abnormality
0x01	Information for the instruction header is incorrect. Data has been accepted in the timing of instruction acceptance.
0x02	Information for the instruction header is incorrect. An instruction has been accepted in the timing of data acceptance.

Each bit, when in "1", indicate the specified condition:

Mnemonic	Status	Description
RUN	1	Waiting RUN
ALM	1	Alarm status
FIX	1	Pulses in output at constant speed
ACC	1	During acceleration or deceleration
NOR	1	Normal status
INI	1	Initial status

● *11-8 Limit Abnormality Response*

FL/ BL/ FHL/ BHL signal status is investigated on start of pulse output and failure of pulse output, if any, is reported.

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x01								Response length
0xf2								Response code

● *11-9 Alarm Occurrence Response*

With this response given, pulse output stops and alarm status is created.

7	6	5	4	3	2	1	0	
0	0	1	0	0	0	1	1	Header
0x02								Response length
0xf4								Response code
Status								

Status	Causes
0x00	An alarm signal has been input.
0x01	Multiple write of instructions has occurred. The response to instruction issuance may not have been taken by the upper host.
0x02	Multiple occurrence of responses has occurred. The responses other than that to instruction issuance may not have been taken by the upper host.
0x03	Multiple occurrence of pulse output stop responses has occurred. The pulse output stop response many not have been taken by the upper host.

● *11-10 Command Abnormality Response*

It is reported that an unsupported instruction code has been received.

The instruction code which has been judged as abnormal is stored in the parameter.

7	6	5	4	3	2	1	0	
0	0	0	RESP		0	1	1	Header
0x02								Response length
0xf5								Response code
Instruction code judged as abnormal								

● *11-11 Override Interruption Response*

It is reported that an instruction subject to override registration has been interrupted due to abnormal end.

The response of an executed (overridden) command ceases to exist and all the instructions subject to override registration are destroyed.

7	6	5	4	3	2	1	0	
0	0	0	1	1	0	1	1	Header
0x02								Response length
0xf8								Response code
Status								

Status	Cause
0x00	Parameter abnormality
0x01	Status abnormality (An instruction was to be issued when not issuable.)
0x02	Operation cannot take place as the limit signal has been input.
0x05	Execution of an instruction code not existing has been attempted.
0x06	A value larger than 0x20 is written in the instruction length.

● *11-12 Event Interruption Response*

It is reported that an instruction subject to event registration has been interrupted due to abnormal end.

The response of an executed command ceases to exist and all the instructions subject to event registration are destroyed.

7	6	5	4	3	2	1	0	
0	0	0	0	1	0	1	1	Header
0x02								Response length
0xf9								Response code
Status								

Status	Cause
0x00	Parameter abnormality
0x01	Status abnormality (An instruction was to be issued when not issuable.)
0x02	Operation cannot take place as the limit signal has been input.
0x05	Execution of an instruction code not existing has been attempted.
0x06	A value larger than 0x20 is written in the instruction length.

● *11-13* Response to Generalized Input Change

A response is given when change takes place in a generalized input terminal having been set by the mask setting instruction.

Sampling of the generalized input terminal starts by the mask setting instruction.

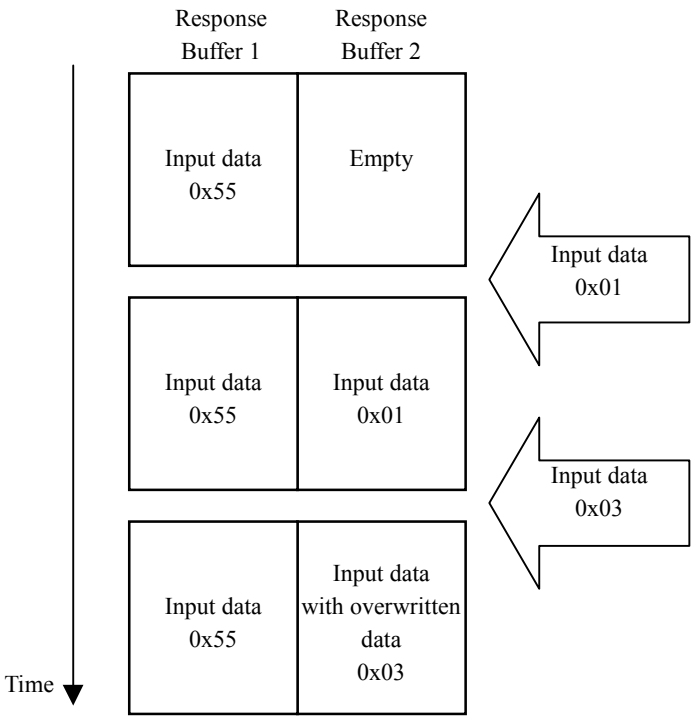
A report is prepared in which generalized input status at the last sampling and the current generalized input status are used as the EXCLUSIVE-OR operation (EXOR), and the product with the mask data of the mask setting instruction are used as input data.

(Former generalized input value	EXOR	new generalized input value)	AND	mask dada	=	input data
0x32		0x66		0x24		0x04

For the first generalized input change, the generalized input value on issuance of the mask setting instruction is temporarily handled as the former generalized input value.

There are two buffers for the input change response. If the 3rd input change takes place before the responses are taken, the 2nd buffer is subject to update to the newest input data and a report is prepared as having overwritten data.

7	6	5	4	3	2	1	0	
0	0	1	0	0	0	1	1	Header
0x03								Response length
0xe9								Response code
0x00: without 0x01: with								With/without overwritten data
Input data								



PPMC-312
Serial

♦ 12 Serial Outline

In addition to bus connection, the PPMC-312 can exchange instructions and responses by means of serial communication.

Serial communication can afford connection of one master and 16 slaves (PPMC, etc.) The PPMC-312 functions as a slave. Electrical connection adopts RS-485 and four-wire half-duplex.

The serial communication format has two kinds of protocol including MWSC mode and ASCII mode. The MWSC mode, through using our MWSC-101, can perform communication of max. 125kbps.

The ASCII mode, through carrying out RS-232C/RS-485, enables control from a personal computer at a rate of 19.2kbps or above.

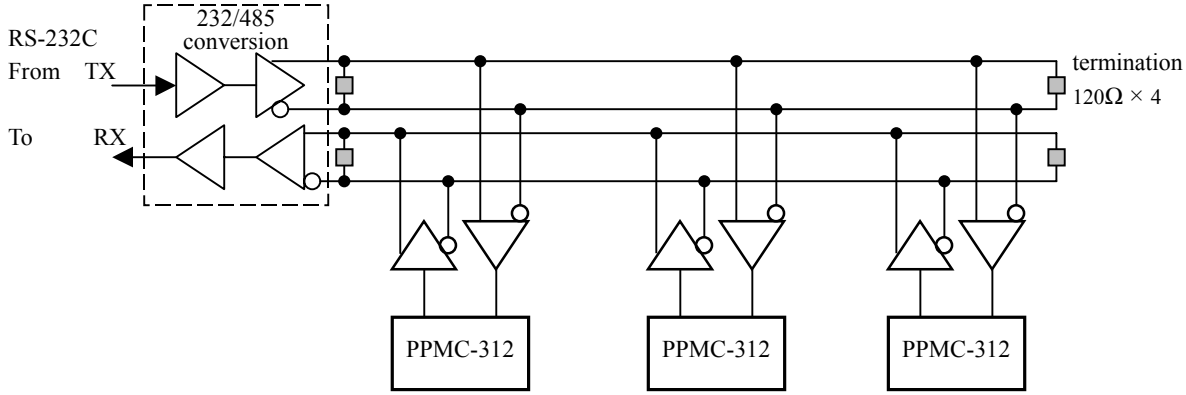


Fig. 12-1 Conceptual Diagram of Connection in ASCII Mode

♦ 13 ASCII Mode

Max. 16 slaves can be subject to control through use of RS-232C ports of a personal computer, etc.

19.2kbps 8bit 1stop Nonparity: It is necessary in ASCII mode that protocol is taken into account in programming.

● 13-1 Protocol

In case of bus connection, status change can be reported to the host CPU. For serial communication where many slaves are connected, however, spontaneous transmission by slaves can cause collision of the communication line. In order to eliminate this risk, procedures need to be obeyed, in which the master (MWSC/PC) always starts communication and, on receiving transmission of the master, the slave performs transmission.

The protocol consists of the following three parts:

- Control code
- Frame (instruction/data)
- Checksum

● 13-2 Control Code

The control code contains, in one byte, the information on frame kind (instruction/data) and on data identification as to which slave they are to(from).

As Bit 7 is always held in "1", please note that it does not belong to the ASCII code characters.

Bit position	Status	Master transmission (Slave receiving)	Master receiving (Slave transmission)
7	1	Always "1".	
6	0	Always "0"	
5,4	00	Polling	Busy
	01	Instruction data	Ready
	10	Undefined	Data reply
	11	Undefined	特 Specific data reply
3,2,1,0	−	Slave address	

■ 13-2.1 Polling

Communication performed regularly by the master to the slave is called "polling". This serves as means to acknowledge pulse output status of the slave, in which the master sends a polling frame. Either busy or ready is sent back from the slave.

■ 13-2.2 Instruction Data

This indicates that an instruction and data are sent to the slave.

■ 13-2.3 Busy

This is a reply to polling, which indicates that the slave is in pulse output.

■ 13-2.4 Ready

This is a reply to polling, which indicates that the slave is not in pulse output. Or, it indicates that instruction data have been accepted.

■ 13-2.5 Data Return

This indicates that data have been send back from the slave concerning an instruction such as the current position acquisition and the generalized input pin state.

■ 13-2.6 Specific Data Return

This is a reply to polling, which indicates any specific report such as slave abnormality, the cause of a stop, and change in status.

● *13-3 Instruction Data*

The instruction data include the instruction code and data to the slave.

The instruction data correspond to the instruction code and parameters of the PPMC-312 parallel mode, after the instruction header and the instruction length being deleted.

Through use of the alphabet lowercase letters and numerals of ASCII code, one-byte Hex is sent in two ASCII codes. After data are converted into ASCII codes, the upper four bits of ASCII code and the lower four bits of ASCII code are sent in this order.

Example 0x0f → 0x30,0x66(" 0 " , " f ") 0x3d → 0x33,0x64(" 3 "," d ")

● *13-4 Check Sum*

In order to prepare check sum, all items from the control code to the end of the instruction data are added in binary in Hex stage prior to conversion into ASCII, which are then reversed totally and Bit 7(MSB) is set to "0".

Example: For control code of 0x01 and instruction data of 0x02, 0x03, 0x04, 0x05, 0x06,
0x01 + 0x02 + 0x03 + 0x04 + 0x05 + 0x06=0x15
0x15 is subject to total bit reversion, 0xea
MSB is set to "0", 0x6a

Example: To PPMC-312 with address 0x01, when initial setting instruction of CLK8M, linear accel/deceleration startup pulse rate of 0x1122, high-speed pulse rate of 0.5566, and accel/deceleration pulse number of 0xabcdef are given:

0x91	"001011225566abcdef"	0x4d
Control code	Instruction data	Check sum

● *13-5 Communication Error*

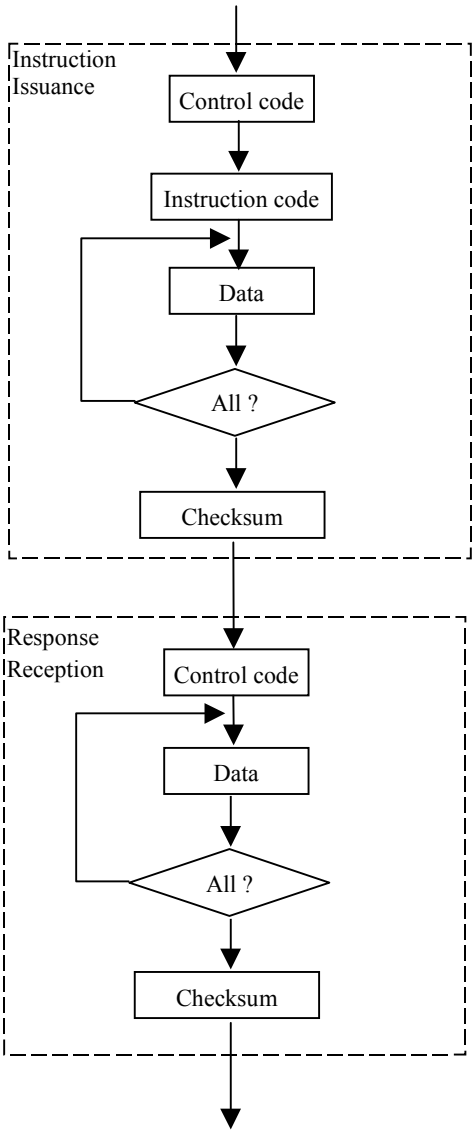
Every frame is provided with check sum. If check sum is incorrect, the PPMC-312 sends back a checksum error.

A framing error, overrun error, etc. can take place in control codes. This can also lower reliability of the slave address of the control code. When this occurs, the PPMC-312 internally stores the error occurrence and does not perform reply to the master. The master, through performing error processing such as time-out, issues the error code read instruction to acknowledge these errors.

♦ 14 Procedure

The procedures for instruction issuance and response reception are illustrated below. At the time of polling, a control code and check sum are issued. (No data are attached.)

An instruction always has its corresponding response. A polling and a response, also, correspond each other. Special attention is required for multiaxial control. The PPMC-312 decides that the right of use of the serial circuit after an instruction (polling) is received till the response is sent back belongs to the PPMC. Should instructions alone be continuously issued to different axes, the responses can overlap, making normal data judgment impossible. In order to prevent this, be sure to receive the response each time an instruction is issued.



- The control/ instruction/ data/ checksum are written in the transmission buffer of the serial port.
- The control code of 1 byte is written in the transmission buffer as it is.
- For the instruction/ data/ checksum, 1-byte Hex is converted into two ASCII codes and written in the transmission buffer.

- The control/ data/ checksum are read from the receiving buffer of the serial port.
- The control code of 1 byte is read from the receiving buffer as it is.
- For the data/ checksum, the two ASCII codes read from the receiving buffer are converted into 1-byte Hex.

♦ 15 Operation Setting Instructions

● 15-1 Initial Setting Instruction

The startup pulse rate and the deceleration pulse rate are in common with each other.

The number of acceleration pulses and the number of deceleration pulses are in common with each other.

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0	0	0	0	FO	CLK			
Startup pulse rate (upper order)								Setting range includes 0x11~0xffff
Startup pulse rate (lower order)								
High-speed pulse rate (upper order)								Setting range includes 0x2~0xffef
High-speed pulse rate (lower order)								
No. of accel/deceleration pulses (upper order)								Setting range includes 0x2~0xffef
No. of accel/deceleration pulses (medium order)								
No. of accel/deceleration pulses (lower order)								
Checksum								

Mnemonic	Status	Description
CLK	000	8Mhz
	001	1Mhz
	010	500khz
	011	250khz
	100	62.5kz
	101	15.625kz
	110	EXTCK signal leading edge
	111	EXTCK signal trailing edge
FO	0	Linear accel/deceleration method
	1	Sigmoid accel/deceleration method

Instruction reception response

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

● *15-2 In-position Setting Instruction*

Every time an irruption/deviation occurs into/from the in-position rage, a response to polling changes. If polling timing is slower than that of the irruption/deviation, responses may be affected by occurrence of deviation, deviation (irruption, irruption).

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
1	1	FO	EN	0	1	1	0	Instruction code
In-position range (upper order)								Setting range includes 0x1~0xffff
In-position range (lower order)								
Checksum								

Mnemonic	Status	Description
FO	0	Without ZCLMP signal control
	1	When difference between the encode pulse count value and the number of output pulses lower the in-position range, ZCLMP signal becomes L. Or, when END signal becomes Low, ZCLMP signal shifts into L.
EN	0	In-position operation inhibited (initial status)
	1	In-position operation enabled

Instruction reception response

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

Response to polling (in-position irruption)

7	6	5	4	3	2	1	0	
0xb0+Address								Control code
0x25								Response code
Checksum								

Response to polling (in-position deviation)

7	6	5	4	3	2	1	0	
0xb0+Address								Control code
0x27								Response code
Checksum								

● *15-3 Encode Setting Instruction*

Every time an irruption/deviation occurs into/from the encode rage, a response to polling changes. If polling timing is slower than that of the irruption/deviation, responses may be affected by occurrence of deviation, deviation (irruption, irruption).

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
1	1	0	EN	0	1	1	1	Instruction code
Encode range (upper order)								Setting range includes 0x1~0xffff
Encode range (lower order)								
Checksum								

Mnemonic	Status	Description
EN	0	Encode operation inhibited (initial status)
	1	Encode operation enabled

Instruction reception response

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

Response to polling (in-position irruption)

7	6	5	4	3	2	1	0	
0xb0+Address								Control code
0x24								Response code
Checksum								

Response to polling (in-position deviation)

7	6	5	4	3	2	1	0	
0xb0+Address								Control code
0x26								Response code
Checksum								

● *15-4 Interlock Position Setting Instruction*

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0xc8								Instruction code
Interlock position (upper order)								Setting range includes 0x0~0xffffffff
Interlock position (medium upper order)								
Interlock position (medium lower order)								
Interlock position (lower order)								
Checksum								

Instruction reception response

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

Response to polling (interlock release)

7	6	5	4	3	2	1	0	
0xb0+Address								Control code
0x20								Response code
Checksum								

● *15-5 High-Speed Limit Setting Instruction*

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0x47								Instruction code
Pulse rate (upper order)								Setting range includes 0x2~0xffff
Pulse rate (lower order)								
Checksum								

Response to instruction acceptance

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

● *15-6 END Terminal Setting Instruction*

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0xc9								Instruction code
0x00: Without monitor 0x01: With monitor								
Checksum								

Response to instruction acceptance

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

♦ 16 Operation Instructions

● 16-1 Immediate Stop Instruction

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0x80								Instruction code
Checksum								

Instruction (position assignment in override)

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0x8b								Instruction code
Stop position (upper order)								Setting range includes 0x0~ 0xffffffff
Stop position (medium upper order)								
Stop position (medium lower order)								
Stop position (lower order)								
Checksum								

Instruction (Started with Event Input 0)

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0x9b								Instruction code
Checksum								

Instruction (Started with Event Input 1)

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0xbb								Instruction code
Checksum								

Response to instruction reception

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

● *16-2 Deceleration Stop Instruction*

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0x81								Instruction code
Checksum								

Instruction (position assignment in override)

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0x8c								Instruction code
Deceleration start position (upper order)								Setting range includes 0x0~ 0xffffffff
Deceleration start position (medium upper order)								
Deceleration start position (medium lower order)								
Deceleration start position (lower order)								
Checksum								

Instruction (Started with Event Input 0)

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0x9c								Instruction code
Checksum								

Instruction (Started with Event Input 1)

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0xbc								Instruction code
Checksum								

Response to instruction reception

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

● *16-3 Accel/Deceleration Instruction*

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
1	0	DIR	0	0	0	1	1	Instruction code
High-speed pulse rate (upper order)								
High-speed pulse rate (lower order)								
No. of operation pulses (upper order)								Setting range includes 0xa~0xffffffff
No. of operation pulses (medium upper order)								
No. of operation pulses (medium lower order)								
No. of operation pulses (lower order)								
Checksum								

Mnemonic	Status	Description
DIR	0	CW direction
	1	CCW direction

Response to instruction reception

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

● *16-4 Constant Speed Instruction*

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
1	0	DIR	0	0	1	0	0	Instruction code
Pulse rate (upper order)								
Pulse rate (lower order)								
No. of pulses (upper order)								Setting range includes 0xa~0xffffffff
No. of pulses (medium upper order)								
No. of pulses (medium lower order)								
No. of pulses (lower order)								
Checksum								

Mnemonic	Status	Description
DIR	0	CW direction
	1	CCW direction

Response to instruction reception

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

● *16-5 Single Step Instruction*

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
1	0	DIR	0	0	0	1	0	Instruction code
Checksum								

Mnemonic	Status	Description
DIR	0	CW direction
	1	CCW direction

Response to instruction reception

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

● *16-6 Continuous High Speed Instruction*

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
1	0	DIR	0	0	1	1	0	Instruction code
Pulse rate (upper order)								
Pulse rate (lower order)								
Checksum								

Mnemonic	Status	Description
DIR	0	CW direction
	1	CCW direction

Response to instruction reception

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

● *16-7 Continuous Constant Speed Instruction*

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
1	0	DIR	0	0	1	0	1	Instruction code
Pulse rate (upper order)								
Pulse rate (lower order)								
Checksum								

Mnemonic	Status	Description
DIR	0	CW direction
	1	CCW direction

Response to instruction reception

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

● *16-8 Accel/Deceleration Change Instruction*

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0x89								Instruction code
Pulse rate (upper order)								
Pulse rate (lower order)								
Checksum								

Instruction (position assignment in override)

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0x8e								Instruction code
Pulse rate (upper order)								
Pulse rate (lower order)								
No. of pulses (upper order)								Setting range includes 0xa~0xffffffff
No. of pulses (medium upper order)								
No. of pulses (medium lower order)								
No. of pulses (lower order)								
Checksum								

Instruction (Started with Event Input 0)

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0x9e								Instruction code
Pulse rate (upper order)								
Pulse rate (lower order)								
Checksum								

Instruction (Started with Event Input 0)

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0xbe								Instruction code
Pulse rate (upper order)								
Pulse rate (lower order)								
Checksum								

Response to instruction reception

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

● *16-9 Immediate Speed Change Instruction*

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0x88								Instruction code
Pulse rate (upper order))								
Pulse rate (lower order)								
Checksum								

Instruction (position assignment in override)

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0x8d								Instruction code
Pulse rate (upper order)								
Pulse rate (lower order)								
No. of pulses (upper order)								Setting range includes 0xa~0xffffffff
No. of pulses (medium upper order)								
No. of pulses (medium lower order)								
No. of pulses (lower order)								
Checksum								

Instruction (Started with Event Input 0))

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0x9d								Instruction code
Pulse rate (upper order)								
Pulse rate (lower order)								
Checksum								

Instruction (Started with Event Input 0)

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0xbd								Instruction code
Pulse rate (upper order)								
Pulse rate (lower order)								
Checksum								

Response to instruction reception

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

● *16-10* Constant Speed Origin Search Instruction

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
1	0	DIR	IDX	0	1	1	1	Instruction code
Pulse rate (upper order)								
Pulse rate (lower order)								
Checksum								

Mnemonic	Status	Description
IDX	0	Without /INDEX detection
	1	With /INDEX detection
DIR	0	CW direction
	1	CCW direction

Response to instruction reception

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

● *16-11* High-Speed Zero Return Instruction

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
1	0	DIR	IDX	0	1	1	1	Instruction code
Pulse rate (upper order)								
Pulse rate (lower order)								
Checksum								

Mnemonic	Status	Description
IDX	0	Without /INDEX detection
	1	With /INDEX detection
DIR	0	CW direction
	1	CCW direction

Response to instruction reception

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

♦ 17 Terminal Control Instruction

● 17-1 Excitation ON Instruction

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0xc0								Instruction code
Checksum								

Response to instruction reception

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

● 17-2 Excitation OFF Instruction

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0xc1								Instruction code
Checksum								

Response to instruction reception

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

● 17-3 Counter Clear Instruction

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0xc2								Instruction code
Checksum								

Response to instruction reception

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

● *17-4 Zero Clamp Instruction*

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0xc3								Instruction code
Checksum								

Response to instruction reception

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

● *17-5 Alarm Clear Instruction*

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0xc4								Instruction code
Checksum								

Response to instruction reception

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

● *17-6 Generalized Input Instruction*

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0x44								Instruction code
Checksum								

Response to instruction reception

7	6	5	4	3	2	1	0	
0xa0+Address								Control code
AUXI 7	AUXI 6	AUXI 5	AUXI 4	AUXI 3	AUXI 2	AUXI 1	AUXI 0	
Checksum								

● *17-7 Generalized Output Instruction*

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0x45								Instruction code
×	×	/AUXO 5	/AUXO 4	/AUXO 3	/AUXO 2	/AUXO 1	/AUXO 0	
Checksum								

Response to instruction reception

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

● *17-8 Control Terminal Read Instruction*

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0x46								Instruction code
Checksum								

Response to instruction reception

7	6	5	4	3	2	1	0	
0xa0+Address								Control code
/ALM	/END	/INDEX	/ORG	/FHL	/BHL	/FL	/BL	
EVT1	EVT0	RUN	×	×	×	×	×	
Checksum								

● *17-9 Mask Setting Instruction*

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0xc5								Instruction code
Mask data								
Checksum								

Response to instruction reception

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

♦ 18 Control Aiding Instructions

● 18-1 Current Position Read Instruction

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0	1	P	0	0	0	1	0	Instruction code
Checksum								

Mnemonic	Status	Description
P	0	Pulse position
	1	Encode position

Response to instruction reception

7	6	5	4	3	2	1	0	
0xa0+Address								Control code
0x00: current position 0x01: encode position								
Position (upper order)								
Position (medium upper order)								
Position (medium lower order)								
Position (lower order)								
Checksum								

● 18-2 Current Position Setting Instruction

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0	1	P	0	0	0	1	1	Instruction code
Position (upper order)								
Position (medium upper order)								
Position (medium lower order)								
Position (lower order)								
Checksum								

Mnemonic	Status	Description
P	0	Pulse position
	1	Encode position

Response to instruction reception

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

● *18-3 Interlock Instruction*

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0x48								Instruction code
Checksum								

Response to instruction reception

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

● *18-4 Version Read Instruction*

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0x4a								Instruction code
Checksum								

Response to instruction reception

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Version								
Checksum								

● *18-5 Synchronous Operation Monitor Instruction*

Confirm, by using the status read instruction, whether or not synchronous operation is currently monitored.

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0x4d								Instruction code
Checksum								

Response to instruction reception

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

● *18-6 End Status Read Instruction*

The cause of stopping pulse output is read out.

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0x40								Instruction code
Checksum								

Response to instruction reception

7	6	5	4	3	2	1	0	
0xa0+Address								Control code
END code								
Checksum								

END code	Cause
0x31	Stopped due to input of an alarm.
0x32	Stopped due to input of the origin.
0x33	Stopped due to detection of a limit.
0x34	Decelerated and stopped due to detection of the high-speed limit.
0x36	Stopped due to the deceleration stop instruction.
0x37	Stopped due to the immediate stop instruction.
0x38	Normal stop (Stopped at the assigned position)
0x39	As the rate is the same as the stop (=startup) rate, deceleration cannot take place. Therefore, immediate stop has occurred.

● *18-7 Event Setting Instruction*

The number of data is fixed at 9 bytes.

The data section has to be set in parallel mode format.

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0	1	REG	No.	1	0	0	1	Instruction code
Data								No. of data fixed at 9 bytes.
▪ ▪ ▪ ▪								
Data								
Checksum								

Mnemonic	Status	Description
No	0	Event Buffer No. 0
	1	Event Buffer No. 1
REG	0	Event registration
	1	Event deletion

Response to instruction reception

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

Response to registration failure

7	6	5	4	3	2	1	0	
0xb0+Address								Control code
0x48								Instruction code
Checksum								

Example: The accel/deceleration instruction is registered in the event buffer.

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0	1	REG	No.	1	0	0	1	Instruction code
0	0	0	0	0	0	1	1	Header of accel/deceleration inst.
0x08								Instruction length of accel/deceleration inst.
0x82								Instruction code of accel/deceleration inst.
×	DIR	×	×	×	OV	TRG		
High-speed pulse rate (upper order)								
High-speed pulse rate (lower order)								
No. of operation pulses (upper order)								
No. of operation pulses (medium upper order)								
No. of operation pulses (medium lower order)								
No. of operation pulses (lower order)								
Checksum								

● *18-8 Event Read Instruction*

Unavailable

● *18-9 Status Read Instruction*

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0x4e								Instruction code
Checksum								

Response to instruction reception

7	6	5	4	3	2	1	0	
0xa0+Address								Control code
CWS	ILS	×	ENC	INP	END	EST	IST	
×	×	RUN	ALM	FIX	ACC	NOR	INT	
EVT1E	EVT0E	EV1O	EV0O	CBCNT				
Checksum								

Operation status

Mnemonic	Status	Description
CWS	0	RUN signal monitored on start of pulse output.
	1	RUN signal ignored on start of pulse output.
ILS	0	Normal status
	1	In interlock. Returns to 0 on release of interlock or following control.
ENC	0	Encoder monitor control inhibited.
	1	Encoder monitor in control.
INP	0	In-position monitor control inhibited.
	1	In-position monitor in control.
END	0	Without END signal control.
	1	With END signal control.
EST	0	Within encode range. Undefined when encode control is inhibited.
	1	Outside encode range. Undefined when encode control is inhibited.
IST	0	Within in-position rage. Undefined when in-position control is inhibited.
	1	Outside in-position range. Undefined when in-position control is inhibited.

Condition status

Each bit, when it is "1", indicates the specified condition. For details, see "Status".

Mnemonic	Status	Description
RUN	1	Waiting RUN
ALM	1	Alarm status
FIX	1	Pulses in output at constant speed
ACC	1	During acceleration or deceleration
NOR	1	Normal status
INI	1	Initial status

Buffer status

Mnemonic	Status	Description
EV1E	0	No registered command exists in Event Buffer 1.
	1	Event command has been registered in Event Buffer 1.
EV0E	0	No registered command exists in Event Buffer 0.
	1	Event command has been registered in Event Buffer 0.
EV1O	0	No registered command exists in Event Buffer 1.
	1	Override command has been registered in Event Buffer 1.
EV0O	0	No registered command exists in Event Buffer 0.
	1	Override command has been registered in Event Buffer 0.
CBCNT	The number of override commands having been registered in command buffers. With registration, +1. With execution, - 1.	

● *18-10 Error Code Readout*

With this instruction, the cause of occurrence of a parameter error or a status error is read out.

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0x41								Instruction code
Checksum								

Response to instruction reception

7	6	5	4	3	2	1	0	
0x90+Address								Control code
ErrCode								
Checksum								

Code	Details
0x42	Parameter error
0x43	Status abnormality
0x44	Limit abnormality
0x47	Undefined instruction
0x48	Size abnormality
0x49	Registration failure

● *18-11 Error Count Readout*

This instruction is exclusive for serial mode. This concerns a framing error, a overrun error, and a checksum error.

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0x4c								Instruction code
Checksum								

Response to instruction reception

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Error count (upper order)								
Error count (lower order)								
Checksum								

● *18-12 Resetting*

On reception of the resetting instruction, reset processing is immediately performed. On completion of reset processing, a response is sent to the first polling.

Instruction

7	6	5	4	3	2	1	0	
0x90+Address								Control code
0xff								Instruction code
Checksum								

Response to instruction reception

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

♦ 19 Responses

● 19-1 Pulse Output in-Stop Response

It is reported that no pulse is output.

7	6	5	4	3	2	1	0	
0x90+Address								Control code
Checksum								

● 19-2 Pulse in-Output Response

It is reported that pulse output is in progress.

7	6	5	4	3	2	1	0	
0x80+Address								Control code
Checksum								

● 19-3 Response to Hold Execution

It is reported that the operation instruction held by /RUN signal has started execution.

7	6	5	4	3	2	1	0	
0xb0+Address								Control code
0x23								
Checksum								

● 19-4 Parameter Abnormality Response

It is reported that abnormality exists in an instruction parameter.

7	6	5	4	3	2	1	0	
0xb0+Address								Control code
0x42								
Checksum								

● 19-5 Status Abnormality Response

On instruction issuance, an instruction may not be issuable depending on the PPMC-312 status. This cause can be obtained by the status read instruction.

7	6	5	4	3	2	1	0	
0xb0+Address								Control code
0x43								
Checksum								

● 19-6 Limit Abnormality Response

On start of pulse output, FL/ BL/ FHL/ BHL signal statuses are investigated and failure of pulse output, if any, is reported.

7	6	5	4	3	2	1	0	
0xb0+Address								Control code
0x44								
Checksum								

● 19-7 Alarm Occurrence Response

An alarm signal has been input. All operations are suspended.

7	6	5	4	3	2	1	0	
0xb0+Address								Control code
0x46								
Checksum								

● 19-8 Command Abnormality Response

An error exists in the instruction.

7	6	5	4	3	2	1	0	
0xb0+Address								Control code
0x47								
Checksum								

● *19-9 Override Execution Response*

An instruction registered in override has been executed.

7	6	5	4	3	2	1	0	
0xb0+Address								Control code
0x21								
Checksum								

● *19-10 Override Interruption Response*

As an error exists in the instruction which is registered in override, execution has been suspended. Registered instructions are all deleted.

7	6	5	4	3	2	1	0	
0xb0+Address								Control code
0x49								
Checksum								

● *19-11 Event Execution Response*

An instruction registered in the event has been executed.

7	6	5	4	3	2	1	0	
0xb0+Address								Control code
0x22								
Checksum								

● *19-12 Event Interruption Response*

As an error exists in the instruction registered in the event, execution has been suspended.

Registered instructions are all deleted.

7	6	5	4	3	2	1	0	
0xb0+Address								Control code
0x4a								
Checksum								

● *19-13* Response to Generalized Input Change

A response is issued on occurrence of change in the generalized input terminal having been set by the mask setting instruction.

7	6	5	4	3	2	1	0	
0xb0+Address								Control code
0x28								
Checksum								

● *19-14* Response to Pulse Output Stop

7	6	5	4	3	2	1	0	
0xa0+Address								Control code
END code								
Checksum								

END code	Cause
0x31	Stopped due to input of an alarm.
0x32	Stopped due to input of the origin.
0x33	Stopped due to detection of a limit.
0x34	Decelerated and stopped due to detection of the high-speed limit.
0x36	Stopped due to the deceleration stop instruction.
0x37	Stopped due to the immediate stop instruction.
0x38	Normal stop (Stopped at the assigned position)
0x39	As the rate is the same as the stop (=startup) rate, deceleration cannot take place. Therefore, immediate stop has occurred.

♦ 20 Oscillator

There are two ways to provide PPMC-312 with clock, i.e. the way to use a quartz oscillator and the way to use a oscillator.

● 20-1 How to Connect Crystal Oscillator

■ 20-1.1 Circuit Structure

An example of crystal oscillator connections is shown in Fig. 20-1. Be sure to use an AT cut parallel resonance type of crystal oscillator.

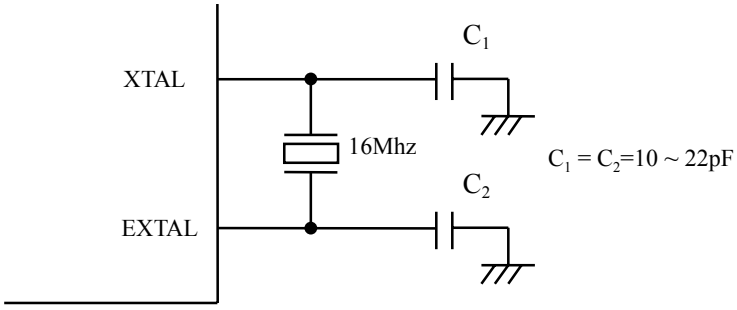


Fig. 20-1 Example of Crystal Oscillator Connections

■ 20-1.2 Crystal Oscillator

Fig. 20-1 shows an equivalent network of the crystal oscillator. Use the crystal oscillator of characteristics shown in Table 20-1:

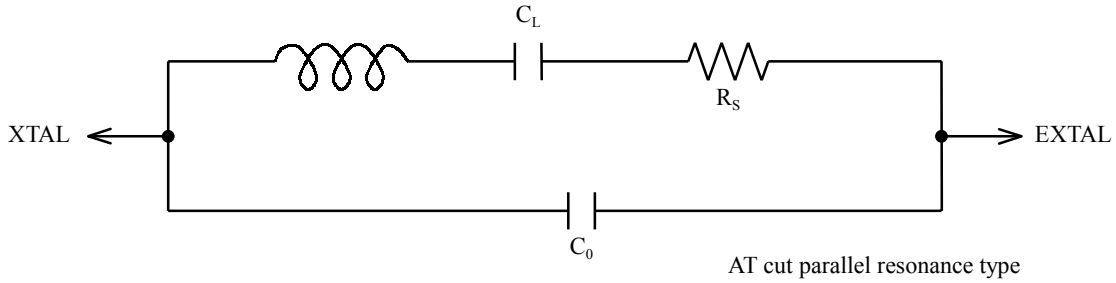


Fig. 20-2 Parameter of Crystal Oscillator

Table 20-1 Parameter of Crystal Oscillator

Frequency(MHz)	16
Rs max(Ω)	50
C_0(pF)max	7

■ 20-1.3 Cautions in Board Designing

In oscillation with a crystal oscillator being connected, pay attention to the following:

Do not pass a signal conductor close to the oscillator circuit section. It may hinder proper oscillation because of induction. (Fig. 20-1)

Further, in board designing, arrange a crystal oscillator and load carrying capacity as close as the XTAL/ EXTAL terminals.

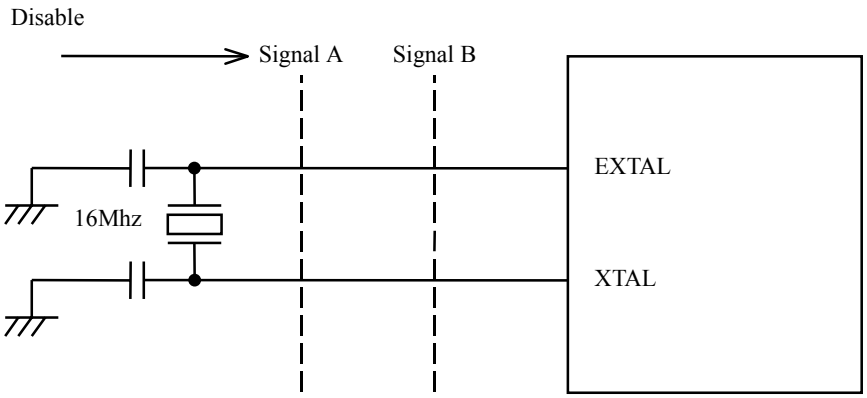


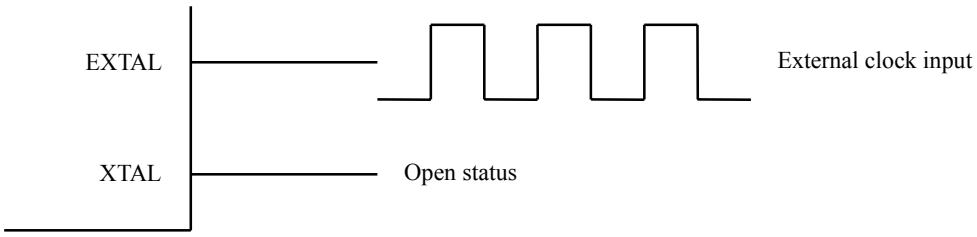
Fig. 20-3 Cautions in Board Designing of Oscillator Circuit Section

● *20-2 How to Connect Oscillator*

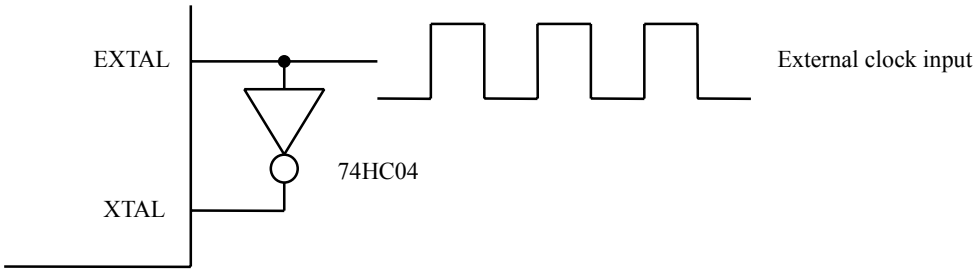
■ 20-2.1 Circuit Structure

An example of crystal oscillator connections is shown in Fig. 20-1.

To place the XTAL terminal in open status, set the parasitic capacitance to 10pF or below.



(a) Example of Connection for XTAL Terminal Open



(b) Example of Connection to Input Opposite-Phase Clock in XTAL Terminal

Fig. 20-4 Example of Connection to Input Oscillator Clock

■ 20-2.2 Oscillator Clock

The oscillator clock timing is shown here (Table 20-2, Fig. 20-5)

Table 20-2 Oscillator Clock Timing

Item	Signal	Min	Max	Unit
Oscillator clock input Pulse width of Low level	t_{EXL}	20	−	ns
Oscillator clock input Pulse width of High level	t_{EXH}	20	−	ns
Oscillator clock Leading time	t_{EXr}	−	5	ns
Oscillator clock Trailing time	t_{EXr}	−	5	ns
Clock pulse width Low level	t_{CL}	0.3	0.7	t_{cyc}
Clock pulse width High level	t_{CH}	0.3	0.7	t_{cyc}

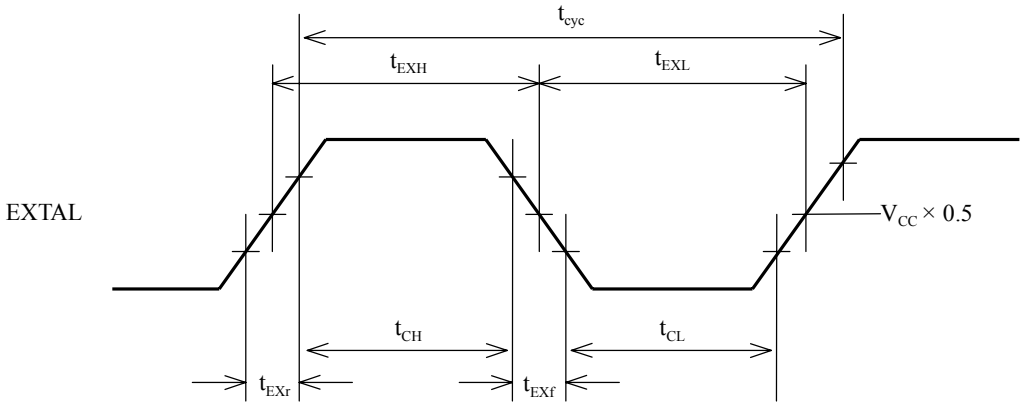


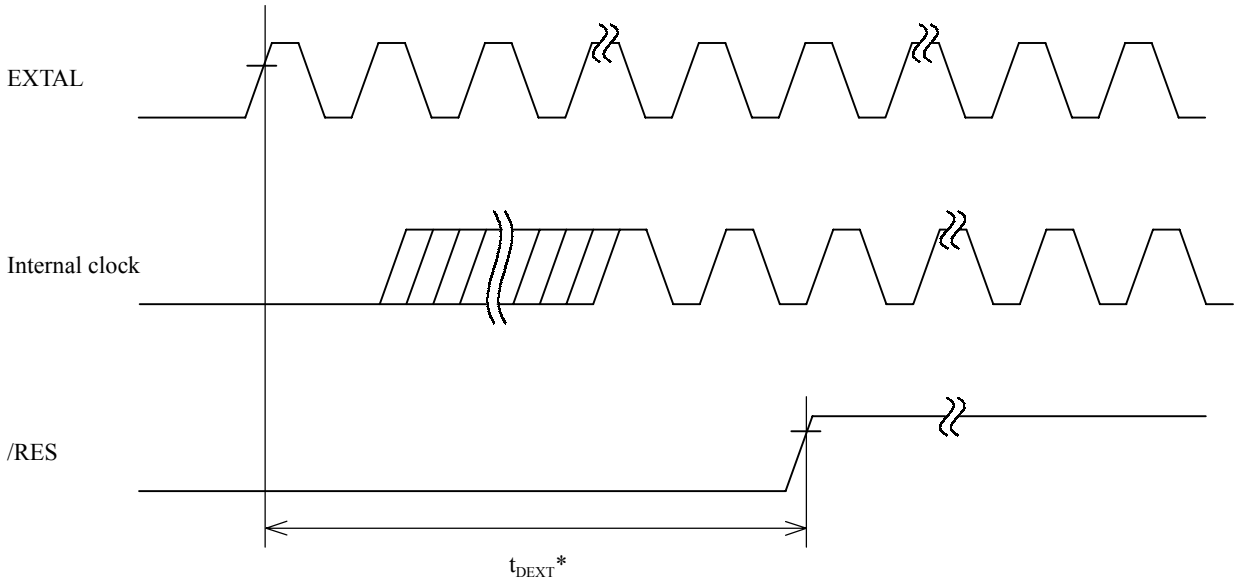
Fig. 20-5 Oscillator Clock Input timing

Table 20-3 Oscillator Clock Output Stabilizing Delay Time

Item	Symbol	min	max	Unit
Oscillator clock output stabilizing delay time	t_{DEXT}*	500	−	µs

[Note] * t_{DEXT} includes $10t_{cyc}$ of /RES pulse width (t_{RESW}).

Fig. 20-6 shows timing of the stabilizing delay time of oscillator clock output. The internal clock signal is defined when the oscillator inside the PPMC-312 and the duty correction circuit have undergone the oscillator clock output stabilizing delay time (t_{DEXT}). As the internal clock signal output is not defined during t_{DEXT} period, keep the reset signal Low to retain reset status.



[Note] * t_{DEXT} includes $10t_{cyc}$ of /RES pulse width (t_{RESW}).

Table 20-6 Timing of Oscillator Clock Output Stabilizing Delay Time

♦ 21 Electrical Characteristics

● 21-1 Absolute Maximum Rating

Table 21-1 Absolute Maximum Rating

Item	Symbol	Rated value	Unit
Supply voltage	V_{CC}	−0.3 ~ + 7.0	V
Input voltage	V_{in}	−0.3 ~ V_{CC} + 0.3	V
Operating temperature	T_{opr}	−20 ~ + 75	℃
Storage temperature	T_{stg}	−55 ~ + 125	℃

[Precautions for Use]

Use of the PPMC-312 beyond the absolute maximum rating can result in permanent damage.

● 21-2 DC Characteristics

Table 21-2 DC Characteristics

Item		Symbol	min	typ	max	Unit	Measurement condition
Schmidt trigger input voltage	RS0, RS1, RS2 /EVT0, /EVT1 EXTCK, DIR /WE, /OE, /CS EPDIR, DTI, CTI, CMI EPIN, RUN, MODE	V_T^-	1.0	−	−	V	
		V_T^+	−	−	V_{CC}×0.7	V	
		$V_T^+ - V_T^-$	0.4	−	−	V	
Input High level voltage	/RESET,EXTAL	V_{IH}	V_{CC}−0.7	−	V_{CC} + 0.3	V	
	Input terminals other than the above		2.0	−	V_{CC} + 0.3	V	
Input Low level voltage	/RESET	V_{IL}	−0.3	−	0.5	V	
	Input terminals other than the above		−0.3	−	0.8	V	
Output High level voltage	All output terminals	V_{OH}	V_{CC}−0.5	−	−	V	I_{OH}=−200μA
			3.5	−	−	V	I_{OH}=−1.0mA
Output Low level voltage	All output terminals	V_{OL}	−	−	0.4	V	I_{OL}=1.6mA
	/ACLR, /ZCLMP, /AUXO0 ~ 5		−	−	1.0	V	I_{OL}=10.0mA
Pull-up current	Pull-up pin	$-I_P$	30	−	250	μA	V_{in}=0V
Input leak current	/RESET	$\lvert I_{in} \rvert$	−	−	10.0	μA	V_{in}=0.5V ~ V_{CC}−0.5V
	Input terminals other than the above		−	−	1.0	μA	V_{in}=0.5V ~ AV_{CC}−0.5V
Free-state leak current (OFF status)	D0 ~ 7	$\lvert I_{TSI} \rvert$	−	−	1.0	μA	V_{in}=0.5V ~ V_{CC}−0.5V
Input capacity	/RESET	C_{in}	−	−	60	pF	V_{in}=0V
	Input terminals other than the above		−	−	15	pF	T_a=25°C
Current consumption	In normal operation	I_{CC}	−	36	60	mA	f=16MHz

[Condition: V_{cc}=5.0V ± 10%, T_a= - 20°C ~+75°C]

Table 21-3 Output Allowable Current

Item		Symbol	min	typ	max	Unit
Output Low level allowable current (per terminal)	/ACLR, /ZCLMP, /AUXO0 ~ 7	I_{OL}	–	–	10	mA
	Input terminals other than the above		–	–	2	
Output Low level allowable current (total)	Sum of /ACLR, /ZCLMP, /AUXO0~7 terminals	$\sum I_{OL}$	–	–	80	
	Input terminals other than the above		–	–	120	
Output High level allowable current (per terminal)	Input terminals other than the above	$-I_{OH}$	–	–	2	
Output High level allowable current (total)	Input terminals other than the above	$\sum -I_{OH}$	–	–	40	

[T_a=−20°C ~ + 75°C]

[Precautions for Use]

To maintain reliability, pay attention so that the output current value does not exceed the values of Table 21-3. Especially, in direct drive of a Darlington pair or LED, be sure to insert a current-limiting resistance for output. (See Figs. 21-1 and 21-2.)

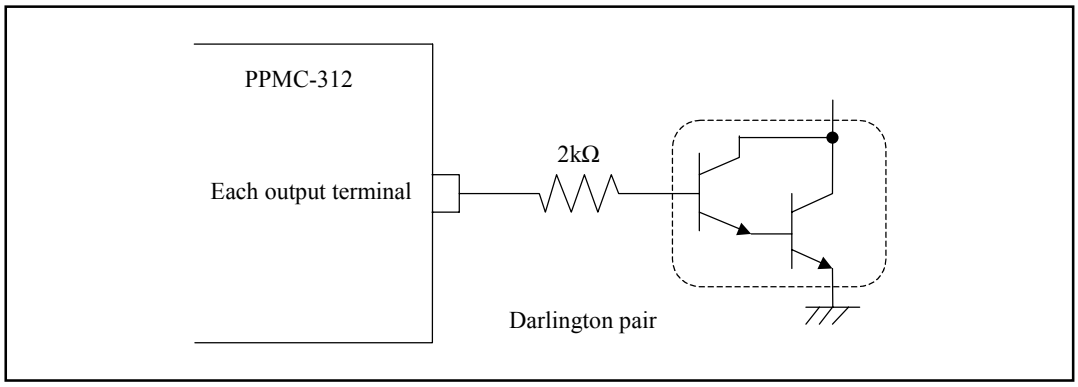


Fig. 21-1 Darlington Pair Driving Circuit

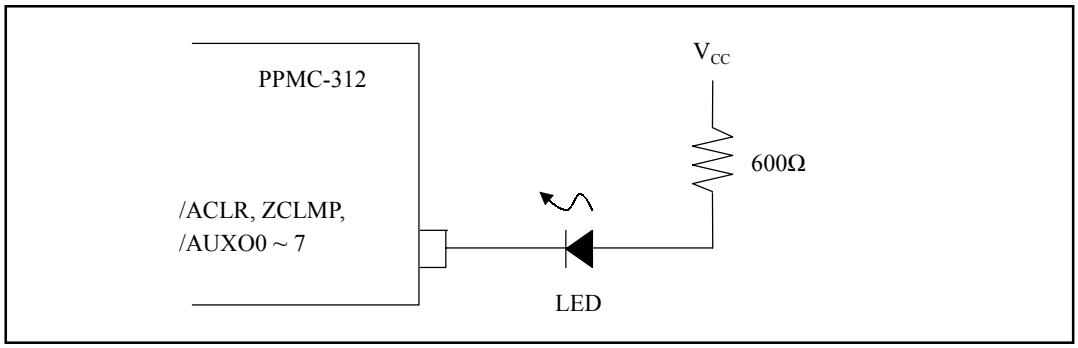
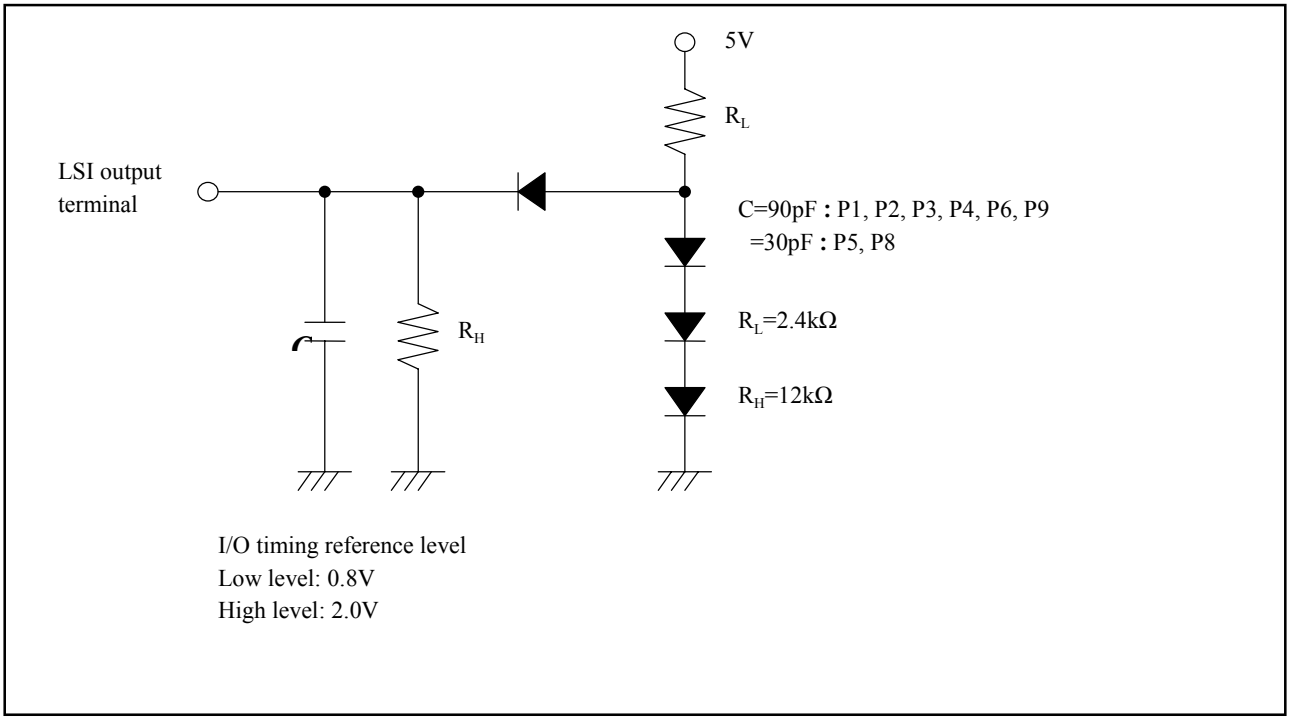


Fig. 21-2 LED Driving Circuit

● *21-3 AC Characteristics*

■ 21-3.1 AC Characteristics Measurement Conditions

Fig. 21-3 Output Load Circuit



■ 21-3.2 Oscillation Stability Time Timing

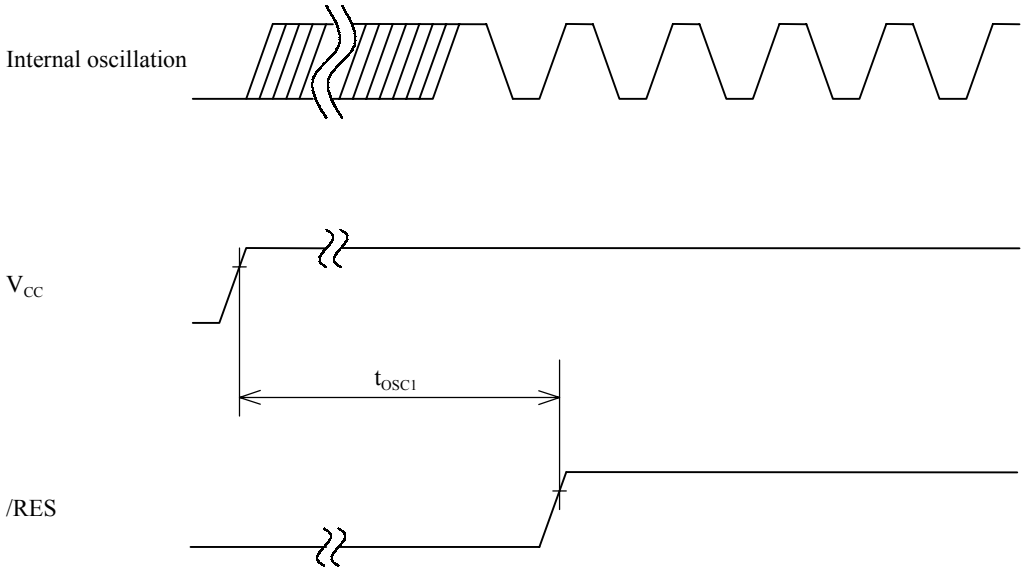


Fig. 21-4 Oscillation Stability Time Timing

Table 21-4 Oscillation Stability time Timing

Item	Symbol	min	max	Unit
Reset oscillation stability time (crystal)	t_{OSC1}	20	–	ms

● 21-4 Timing

■ 21-4.1 Parallel Mode Write Cycle

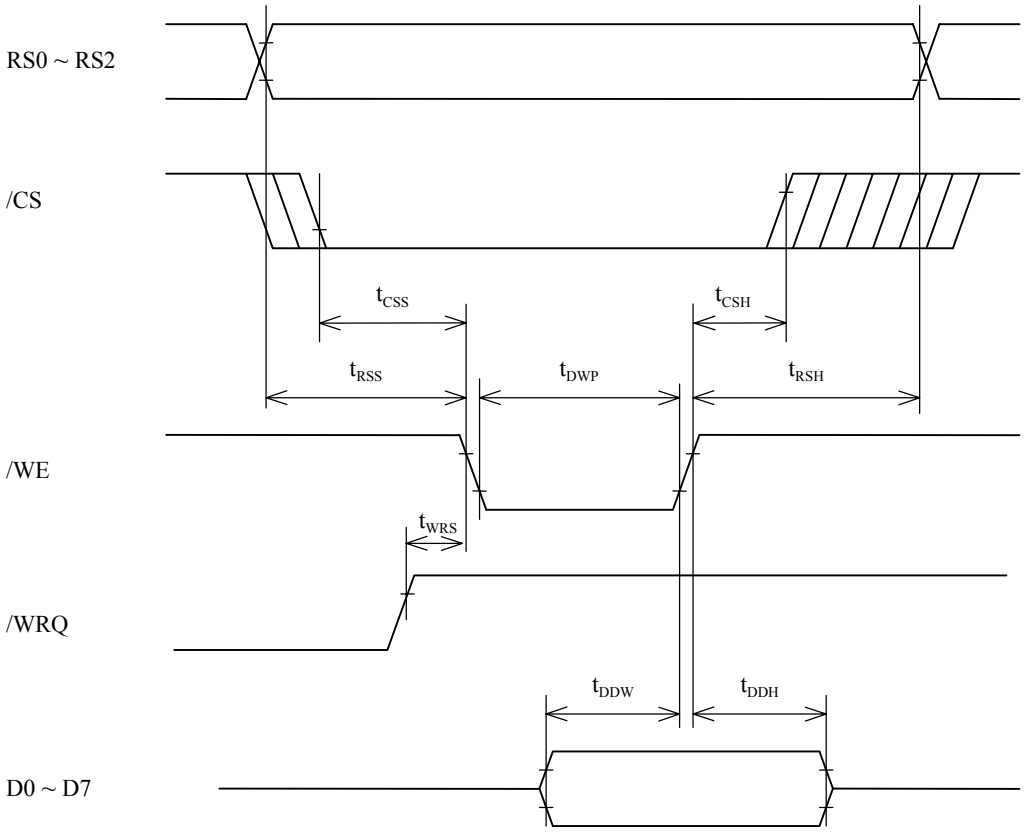


Fig. 21-5 Write Timing

Item	Symbol	min	max	Unit
Address hold time	t_{RSH}	10		ns
Chip select hold time	t_{CSH}	10		ns
Address setup time	t_{RSS}	10		ns
Chip select setup time	t_{CSS}	10		ns
Write pulse width	t_{DWP}	65		ns
Write data setup time	t_{DDW}	35		ns
Write data hold time	t_{DDH}	20		ns
Write access time	t_{WRS}	0		ns

■ 21-4.2 Parallel Mode Read Cycle

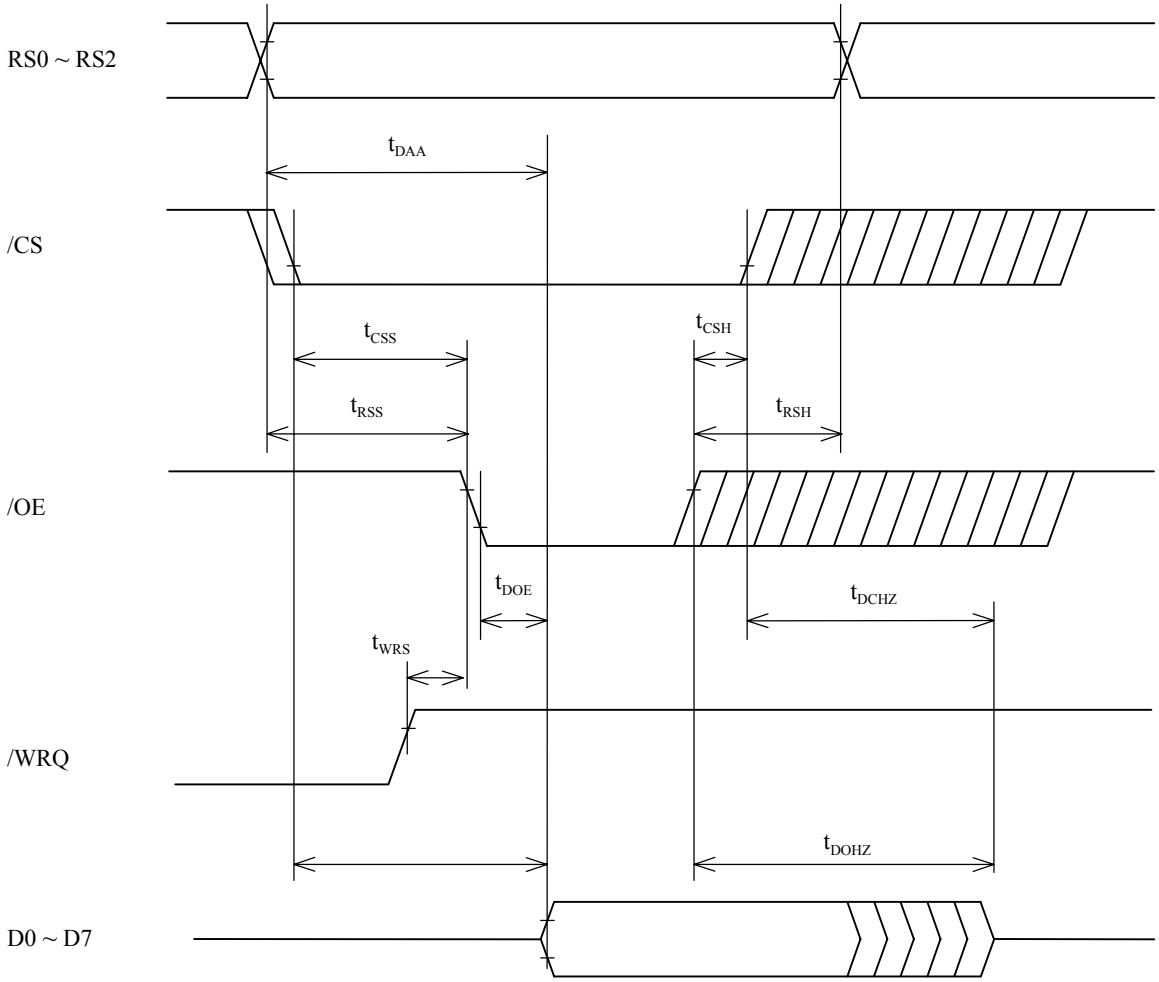
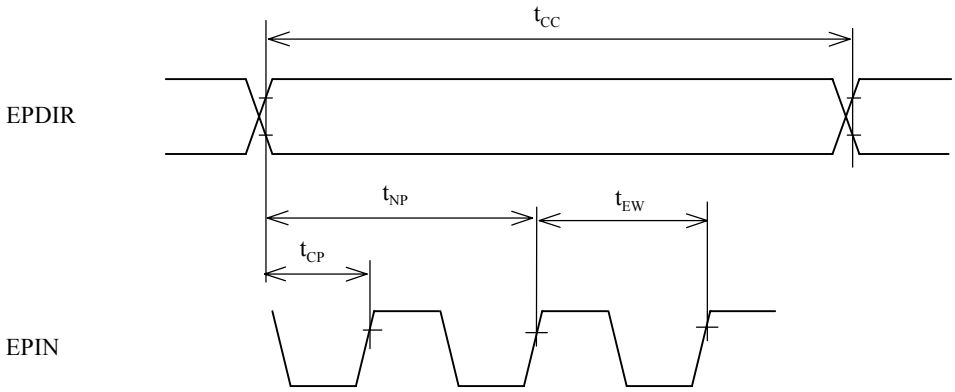


Fig. 21-6 Read Timing

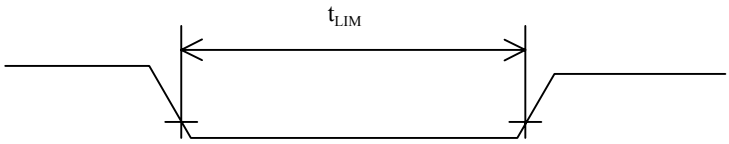
Item	Symbol	min	max	Unit
Address hold time	t_{RSH}	10		ns
Chip select hold time	t_{CSH}	10		ns
Address setup time	t_{RSS}	10		ns
Chip select setup time	t_{CSS}	10		ns
Access time	t_{DAA}		85	ns
Read data delay time	t_{DOF}		85	ns
Chip select access time	t_{DACS}		85	ns
/CS output floating time	t_{DCHZ}	0	50	ns
/OE output floating time	t_{DOHZ}	0	50	ns
Read access time	t_{WRS}	0		ns

■ 21-4.3 Encoder Timing



Item	Symbol	min	max	Unit
EPDIR width	t_{CC}	500		µS
EPIN width	t_{EW}	500		nS
Time to 1st EPIN in direction switching	t_{CP}		5	µS
Time to 2nd EPIN in direction switching	t_{NP}	25		µS

■ 21-4.4 Limit Timing



Item	min	max	Unit
/EVT0,/EVT1	25		mS
/ALM,/END,/INDEX,/ORG /FHL,/BHL,/FL,/BL	200*		µS

* No monitoring is performed while an instruction is being processed.

The initial setting instruction cannot be subject to monitoring for 800mS.

No monitoring is available for 400mS when the startup pulse rate and stop pulse rate of the accel/deceleration instruction are set equally, and for 800mS when they are set differently.

No monitoring is available for 300µS in high-speed rate assignment of the high-speed zero return instruction or the continuous high speed instruction.

For other instructions, monitoring is not available for 100µS.

♦ 22 Dimensional Outline Drawing

The dimensional outline drawing is shown in Fig. 22-1:

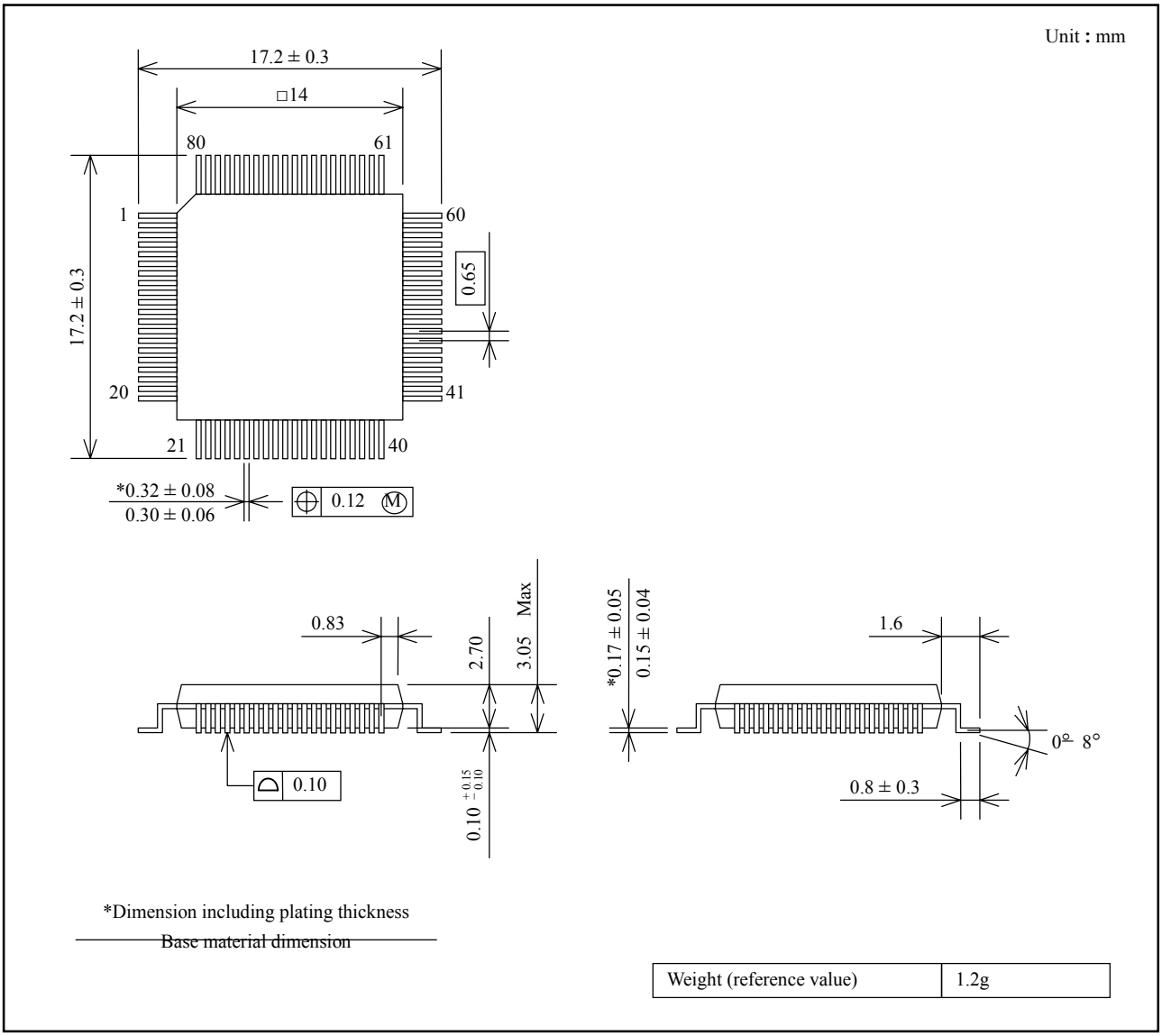


Fig. 22-1 Dimensional Outline Drawing

♦ 23 Explanatory Notes on Packaging

● 23-1 Soldering Temperature Profile Setting

■ 23-1.1 Basic Concept of Temperature Profile Setting

The basic concept of temperature profile setting includes the following:

(1) To set temperature conditions for appropriate soldering.

(2) To set temperature conditions to prevent thermal damages of parts.

■ 23-1.2 Key Points Applied in Actual Temperature Profile Setting

Key points in setting the temperature profile are:
Peak temperature
Solder melting time
Preheating temperature/time
Temperature gradient

- Peak Temperature

  Pay attention to the following and set the optimal conditions:

  (a) The surface temperature of mounted parts should be held within heat resisting temperature range.

  (b) The lead part temperature should be above the solder fusing point.

  Normally, it should be equal to the solder fusing point +30°C.

  (Example) For Sn/Pb=63/37 eutectic solder,
  Solder fusing point: 183°C
  Soldered part temperature: 210~220°C (183°C + approx. 30°C)

  Depending on part size, temperature difference is produced between a lead part and a package surface. While lead part temperature tends to be lower than package surface temperature for smallerOFP1420 size, lead part temperature tends to be higher than package surface temperature for larger QFP2828.

- Solder Melting Time

  Solder melting time, if being too short, can result in poor wetting/spread over footprint or lead. If it is too long, on the other hand, "Ag scab" of Ag/ Ag-Pd electrode parts can take place, resulting in soldering strength degradation. Set the conditions, therefore, taking these into consideration.

- Preheating

  Roles of preheating are enhanced as high-density packaging advances. The major roles include:

  (a) To prevent substrate warping or to minimize it,

  (b) To volatilize solvent in solder paste,

  (c) To prevent wicking and Manhattan phenomena.

  Although not affecting thermal damages of parts, preheating is an essential process for good "soldering".

  Preheating, if lasting long hours at a high temperature, can cause oxidation of solder paste and substrate surface, resulting in occurrence of soldering balls and poor wetting/spread. On the other hand, low temperature and short hours can increase curvature amount, increasing temperature differences of substrates, IC leads, and package surface. Also, this tends to develop wicking and Manhattan phenomena.

  Taking account of the above, you are requested to set the preheating conditions by targeting 140~160°C which is lower than the solder fusing point (Sn/Pb eutectic crystal 183°C) and approximately 1 minute.

- Temperature Gradient

  There may be a worry about heat damage of parts caused by too large a temperature gradient (temperature increase rate or temperature decrease rate). This should not be worried about so long as the range we recommend (1~5°C / sec.) is observed. On the other hand, by the cooling temperature gradient being increased (for faster cooling), luster of "soldered part" surfaces is normally improved.

  However, the cooling gradient, if being increased, is more likely to bring about substrate warping. Pay attention, therefore, to condition setting.

● 23-2 Cleaning

If, following reflow soldering, flux residue containing corrosive substance remains on a substrate, reliability of the parts and the substrate wiring can be seriously affected. It is, therefore, necessary to remove residue by cleaning or to select solder paste containing non-cleaning type flux with less residue content. In judging "cleaning" or "non-cleaning", the following matters need to be discussed:

Reliability level of a product
Operating environment of a product
External appearance level required
Characteristics of flux used
With/without an in-circuit test

■ 23-2.1 Selection of Cleaning Liquid

Selection of cleaning liquid is largely affected by whether the flux used is of rosin type or water soluble. Cleaning liquid needs to be selected to suit specific nature of flux residue.

To use rosin-type flux:
(a) Terpenoid solvent: liquid containing elements extracted from orange peel
(b) Petroleum solvent: mixed liquor of petroleum solvent and surface active agent
(c) Alcohols solvent: ethanol, methyl alcohol, etc.
(d) Alkaline thinner

To use water soluble flux:
(a) Water (including, also, warm water)
(b) Water + alkali neutralizing liquid

Cleaning liquid conforming to fluorine regulation/ flux

■ 23-2.2 Cleaning Conditions

Ultrasonic cleaning, for example, is performed according to the following conditions. In addition to them, cares should be taken to prevent device breakdown so that impressed frequency, electric power (especially, peak power), time, and device do not resonate.

Frequency:	28kHz ~ 29kHz (The device must not resonate.)
Ultrasonic output:	15W/litre (1 time)
Time:	30 seconds or less
Others:	Neither a device nor a printed wiring board directly touches a vibration source.

Especially, as the ceramic package QFN(LCC), QFP(ceramic), etc. are cavity package, ultrasonic cleaning, if conducted, can bring about connecting wire resonance, causing disconnection.

■ 23-2.3 Criteria of Cleanliness

Judge cleanliness based on the "Substrate Cleanliness Following Component Mounting".

■ 23-2.4 Other Remarks

As marks may be deleted from long hours of cleaning, be sure to refer to actual service conditions before starting using it.

Cleaning with Organic Solvent:
(a) When using terpenoid solvent, alcohols, or petroleum solvent, it is necessary to use an explosion-proof device in view of inflammability.
(b) When using cleaning liquid which involves washing in water, it is necessary to fully discuss the drainage treatment.

Washing in Water:

It is necessary to fully discuss drainage treatment in conformity with related laws and regulations.

● 23-3 Cautions in Packaging

■ 23-3.1 Electrostatic Discharge Damage

As, in general, semiconductors are more easily broken under electrostatic discharge, mounting on substrates should be performed with the greatest care.

- Working Environment
  As relative humidity becomes lower, electrostatic discharge tends to occur more easily. In storage, the surface mount type packages need to be held in dry environment to avoid moisture absorption. During handling and packaging on substrates, however, they are recommended to be placed in relative humidity of 45~75% from antistatic point of view.

- Antistatic Measures During Work
  At a packaging work site, avoid using insulating material susceptive to electrostatic discharge as much as possible. Even if a semiconductor or a finished substrate is not directly touched by any charged object, just being close to it alone can bring about induction electrification. Some of preventive measures include, for example, use of electrostatic work clothes, electrically conductive carrier boxes, and aeroionization blowers.

  To prevent electrostatic accumulation, it is necessary to earth the measuring instrument, conveyors, work stands, floor mats, tools, and soldering irons. Earth each work stand and floor by using a conductive mat (approx. 10_9~$10_{11}\Omega$).

  Use an armlet and an anklet for grounding a human body. For shock prevention, however, connect resistances of 1MΩ or larger in series. When using a soldering iron, it should be exclusively for semiconductors (low voltage of 12V~24V), whose tip has to be grounded.

- Measures for Preventing Discharge from Semiconductors
  Should a package or a chip be charged, the semiconductor is not destroyed by electrification alone. However, if a lead frame touches metal when it is electrified, discharge occurs, destroying the semiconductor. In this circumstance, grounding metal cannot give any effect.

  The preventive measures are shown below:

  Avoid contact or friction of a semiconductor with/against insulating material which tends to be easily electrically charged.

  Avoid handling a semiconductor on a metal plate. Use a high-resistance conductive mat which is grounded. When electrification of a semiconductor is suspected, be careful not to bring it into direct contact with metal.

- Cautions in Mounting on Substrate

  When mounting a semiconductor on a substrate, use a high-resistance mat, etc. and be sure to complete earthing beforehand. Especially, cares should be taken following a burn-in of substrates as electric charge may have been accumulated in a capacitor.

  Substrates, also, are subject to electrification through touching, friction, induction, etc. It is necessary, therefore, to use an antistatic bag or to isolate substrates in transportation in a carrier box, etc. so that discharge does not occur by substrate contact.

● 23-4 Baking

You are requested to provide products with baking when any of the following cases applies (125 ± 5°C, 16~24 hours, For repetition, less than 96 hours in accumulating total):

(a) Products provided with a humidity indicator card:
When, on unsealing of the vaporproof packaging, the 30%-spot of the card is pink in color.
When the indicator card and the silica gel differ in color, judge by color of the indicator card.

(b) Products not provided with a humidity indicator card:
When, on unsealing of the vaporproof packaging, color change of the silica gel is observed.

(c) When specified storage conditions (5°C ~30°C, 60% or less, within 168 hours after unsealing) have overrun since the vaporproof packaging is opened.