# PISO-PS400 Getting Started Manual

(Version 2.3)

Hardware & Software & Application Using PISO-PS400 PCI BUS Motion Control Card



1

PS400 Getting Started Rev.2.3 2/12/2008

### Warranty

All products manufactured by ICPDAS Inc. are warranted against defective materials for a period of one year from the date of delivery to the original purchaser.

#### Warning

ICPDAS Inc. assumes no liability for damages consequent to the use of this product. ICPDAS Inc. reserves the right to change this manual at any time without notice. The information furnished by ICPDAS Inc. is believed to be accurate and reliable. However, no responsibility is assumed by ICPDAS Inc. for its use, or for any infringements of patents or other rights of third parties resulting from its use.

### Copyright

Copyright 1997-2005 by ICPDAS Inc., LTD. All rights reserved worldwide.

### Trademark

The names used for identification only maybe registered trademarks of their respective companies.

#### License

The user can use, modify and backup this software on a single machine. The user may not reproduce, transfer or distribute this software, or any copy, in whole or in part.

# **Contents of PISO-PS400**

1 INTRODUCTION	7
1.1 Introduction	7
1.2 Hardware Specification	8
1.2.1 Main Specification	8
1.2.2 Interpolation Function	8
1.2.3 Pulse Output	8
1.2.4 Encoder Input	
1.2.5 Position Counter	
1.2.6 FRnet	
1.2.7 Auto-Homing	
1.2.8 Servo Motor Input Signal	10
1.2.9 Limit Switch Input Signal	10
1.2.10 Other Input Signals	10
1.2.11 Emergency Stop Signal Input	10
1.2.12 General Output Signal	10
1.2.13 Integral Input Signal Filters	10
1.2.14 Software Limit	10
1.2.15 Manual Pulse Generator	11
1.2.16 Synchronous Action	11
1.2.17 Interrupt	11
1.3 Environment	12
1.4 Ordering Information	12
2 HARDWARE INSTALLATION	
2.1 Checking Deckege and Installation	13
2.1 Check Package	<b>13</b>
2.1.1 Check Tackage	13
2 1 3 SW1 Setting	14
2.1.6 SW1 Setting	14
2 2 DN-8468G Terminal Board	15
2.2 D Roard Lavout for DN-8468G	15
2.2.2 Signal Connections for DN-8468G	
2.2.3 Jumper and Switch Settings	22
2.3 Input/Output Connections	
r a company	

2.3.1 Output Pulse Signals	24
2.3.2 Connection for Limit Switch Signal	25
2.3.3 General Purpose Input Signal (nINPOS, nALARM)	26
2.3.4 Encoder Signals	26
2.3.5 Emergency Stop Signal	27
2.3.6 Manual nulse Generator Innut Signals (EXP+ EXP-)	28
2.3.7 General Purpose Output Signals(Servo On/Off)	·· 20 28
2. 0. 1 deneral rupose output organis(dervo on/orr)	·· 20
2.4 Connection Example for wrotor Driver	
<b>3 SOFTWARE DEVELOPMENT OVERVIEW</b>	30
3.1 Software Development Overview	30
3.1.1 Card Initialization	31
3.2 Safety IO Setting	31
3.2.1 Emergency Stop Signal Input	31
3.2.2 Configure the Servo ALARM Signals	31
3.2.3 Configure the Limit Switch Signals(±EL)	31
3.2.4 Configure the Software Limite (±SEL)	32
3.3 Error Checking	32
3.4 Basic Configuration of Motion	32
3.5 Manual Pulse Generator Testing	33
3.6 Home Search	33
3.6.1 Home Search Configuration	34
3.6.2 Running the Home Search	34
3.7 Basic Motion	35
3.7.1 Speed Profile of the Motion Control	35
3.7.2 Basic Motion of Single Axis	36
3.7.3 Basic Motion of Muti-Axes Interpolation	38
3.8 Synchronized Motion	38
3.9 Advanced Motion	
4 GETTING STARTED OF SOFTWARE	40
4.0 PISO-PS400 Installation	40
4.1 VC 6.0 Guideline	40
4.1.1 Confirm the Relative Files	40
4.1.2 Create a new VC++ Application Project	40
4.1.3 Add the Reference Path into VC ++ Application Project	43
4.1.4 Start the VC++ Example	45

4.1.5 Build the Project	48
4.2 VB 6.0 Guideline	49
4.2.1 Confirm the Relative Files	49
4.2.2 Create a New VB Application Project	49
4.2.3 Add The PS400.BAS into the VB Application Project	52
4.2.4 Start the VB Sample	53
4.2.5 Build the Project	56
4.3 BCB 6 Guideline	57
4.3.1 Confirm The Relative Files	57
4.3.2 Create a New BCB Application Project	57
4.3.3 Start the BCB Sample	58
4.3.4 Add The Reference Path into the BCB Application Project	61
4.3.5 Add the PS400BCB.lib into the project	61
4.3.6 Build the Project	63
5 PISO-PS400 PCEZGO(BY BASIC FUNCTION)	54

5.1 Initial Settings Dialog	66
5.1.1 Registration and Hardware Signals	66
5.1.2 Servo Input Signals	67
5.1.3 Setting SoftLimit & V-Ring Function	69
5.1.4 Interrupt Configuration	71
5.2 Command Dialog	72
5.2.1 Acc/Dec Motion	72
5.2.2 Home Search and Jog Operation	74
5.2.3 Synchronous Action	77
5.3 Interpolation Dialog	79
5.3.1 Linear Interpolation and Circular Interpolation	79
5.3.2 Two axes path plane diagram	81
5.4 FRnet DI/DODemo	82
APPENDIX-A PISO-PS400 INSTALLATION	83

## **APPENDIX-B OTHERS TERMINAL BOARDS......86**

B.1 DN-8468M Daughter Board	.86
B.1.1 Board Layout for DN-8468M	86
B.1.2 Signal Connections for DN-8468M	87
B.1.3 Jumper and Switch Settings	92

B.2 DN-8468P Daughter Board94
B.2.1 Board Layout for DN-8468P
B.2.2 Signal Connections for DN-8468P
B.2.3 Jumper and Switch Settings 100
B.3 DN-8486Y Daughter Board102
B.3.1 Board Layout for DN-8468Y 102
B.3.2 Signal Connections for DN-8468Y 103
B.3.3 Jumper and Switch Settings 107
B.4 DN-8468D Daughter Board109
B4.1 Board Layout for DN-8468D 109
B4.2 Signal Connections for DN-8468D 110
B4.3 Jumper and Switch Settings 118

## **1 INTRODUCTION**

### **1.1 Introduction**

The PISO-PS400 is a 4-axis stepping/pulse-type servo motor PCI motion card with maximum 4M PPS pulse output, and is suitable for general-purpose motion applications. This motion card contains a high-performance motion ASIC. Apart from a wide speed range, this intelligent motion controller also has a variety of motion control functions built in, as 2~3-axis linear interpolation, 2-axis circular interpolation, T/S-curve such acceleration/deceleration, various synchronous actions, automatic homing, and others. Besides, one FRnet port is contained in this motion card. The FRnet port allows this motion card to expand its fast remote I/O easily. The two-wired FRnet signal can automatically scan its 128 DI and 128 DO with a period of 0.72/2.88ms. In addition, most of the PISO-PS400 motion control functions are performed with little load on the processor. While driving the motors, the motion status, and the other I/O status on the PC can still be monitored. As a result of the low CPU loading requirements of PC, one or more motion control card may be used on a single PC (At most 16 PS400 motion card can be added in a PC). Muti-axes (4 or 8 ..... axes) motion control algorithm can be achieved in a single PC. ICPDAS has also provided a wide range of functions and examples to reduce the need for programming by users, making it a highly cost-effective solution for machine makers.

### **1.2 Hardware Specification**

### **1.2.1 Main Specification**

ASIC Chip	MCX314As
Number of controllable	4-Axes, Pulse output (stepping &
	servo motor)

■ Up to 4M PPS pulse output

### **1.2.2 Interpolation Function**

2-axes & 3-axes linear interpolation

- Interpolation range -2,147,483,646 ~ +2,147,483,646
  Vectors speed of interpolation 1 PPS ~ 4M PPS
- Precision of interpolation ± 0.5 LSB

Circular interpolation

■ Interpolation range -2,147,483,646 ~ +2,147,483,646

1 PPS ~ 4M PPS

Vectors speed of interpolation

Relative interpolation function

- Any 2-axes or 3-axes interpolation
- Fixed vectors speed
- Continuous interpolation

### 1.2.3 Pulse Output

	Output speed range	1 PPS ~ 4 MPPS
	Output precision	± 0.1%
	Jerk range of S-curve	954 ~ 62.5 x 10^6 PPS/S^2
		477 x 10^3 ~ 31.25 x 10^9 PPS/S^2
	Acceleration/deceleration range	125 ~ 1 x 10^6 PPS/S
		62.5×10^3 ~ 500 x 10^6 PPS/S
	Speed precision	1 PPS ~ 500PPS(In accordance with a
		highest speed)
	Output numbers	0 ~ 4,294,967,295 / unlimited
_	V ala site constitue a seconda s	

- Velocity profiles mode:
  - Fixed
  - Symmetrical & Asymmetrical Trapezoidal velocity profile
  - Symmetrical & Asymmetrical S-curve velocity profile

- Acceleration & Deceleration mode
  - Auto
  - By user define
- Position & Speed change on the fly
- Fixed pulse output by Trapezoidal and S-curve velocity profile
- Pulse output option: CW/CCW, PULSE/DIR
- Programmable logic level

### **1.2.4 Encoder Input**

- Encoder option: A/B phase, Up/Down (CW/CCW)
- Programmable A/B phase mode: 1, 1/2, and 1/4 A/B phase

### **1.2.5 Position Counter**

- Command counter range -2,147,483,648 ~ +2,147,483,647
- Feedback counter range
- -2,147,483,648 ~ +2,147,483,647
- Programmable ring counter
- Programmable direction of counter
- Use DI(IN3) to clear feedback counter
- Programmable read & write counter

### 1.2.6 FRnet

- DI  $\rightarrow$  max up to 128
- DO  $\rightarrow$  max up to 128

### 1.2.7 Auto-Homing

- Four Steps
  - Step 1 (High-speed to find "Near Home" sensor)
  - Step 2 (Low-speed to find "Home" sensor)
  - Step 3 (Low-speed to find Index Z sensor)
  - Step 4 (Execute offset position of high-speed)

Every step can be set to execute or not, and running direction.

### 1.2.8 Servo Motor Input Signal

- Alarm
- Choose IN2: In Position or Servo Ready signal
- Choose input signle: Enable/disable and logic level

### **1.2.9 Limit Switch Input Signal**

- Two-limit switch signal for each axis: +EL, -EL
- Programmable logic level
- Programmable action mode( slow-down stop or immediately stop)

### **1.2.10 Other Input Signals**

■ IN3 : other purpose, as a trigger of synchronal control.....

### **1.2.11 Emergency Stop Signal Input**

There is an Emergency stop signal for Each motion card

### 1.2.12 General Output Signal

The Servo-on signal (nOUT1) can be used as servo-on control or general purpose output signal for each axis.

### **1.2.13 Integral Input Signal Filters**

The motion card is equipped with an integral type filter in the input step for each input signal. User can select a digital filter with different time constant.

### 1.2.14 Software Limit

■ There are two software end-limit for each axis: +SEL & -SEL (Setting range : -2,147,483,646 ~ +2,147,483,646)

### 1.2.15 Manual Pulse Generator

- Manual pulsar mode (A/B phase pulse mode)
- Fixed Pulse Driving Mode (CW/CCW pulse mode)

Disable Mode: Disable manual pulse function

 Continuous Pulse Driving Mode (CW/CCW pulse mode)

### **1.2.16 Synchronous Action**

The synchronous action can be active from provocative (main) axis

- Provocative (Activation) Factor
  - ◆ Position ≥ COMP+ : The logical/real position counter value exceeded the value of COMP + register value.
  - Position < COMP+ : The logical/real position counter value became less than the COMP + register value.
  - Position < COMP- : The logical/real position counter value became less than the COMP - register value.
  - ◆ Position ≥ COMP- : The logical/real position counter exceeded the COMP register value.
  - Driving started.
  - Driving terminated.
  - The nIN3 signal rose from the Low to the High level.
  - The nIN3 signal fell from the High to Low level.
- Synchronous Action
  - Fixed pulse driving in the +/- direction
  - Continuous pulse driving in the +/- direction
  - Stops driving in deceleration
  - Stops driving immediately
  - Saves the current logical or real position counter value in the synchronous buffer register

### 1.2.17 Interrupt

- Interrupt of each axis can be happened from the following factors:
  - when one pulse outputs... (The interrupt will be generated at the rising edge of

pulse output for + direction driving.

- In the acceleration / deceleration driving mode, when the driving changes from the constant speed region into the decelerating region or from the accelerating region into the constant speed region.
- Once the value of logical / real position counter is larger than or equal to the value of COMP− register (Position ≥ COMP-)
- Once the value of logical/real position counter is smaller than the value of COMPregister (Position < COMP-)</li>
- Once the value of logical / real position counter is larger than or equal to the value of COMP+ register (Position ≥ COMP+)
- Once the value of logical/real position counter is smaller than the value of COMP+ register (Position < COMP+)</li>
- The automatic homing or the driving is finished.
- Each factor of interrupt can be masked to enable or disable.

### **1.3 Environment**

Operating Temp:	-20 ~ + 75°C

- Storage Temp:
- Operating Humidity:
- Storage Humidity:
- I/O optically isolated
- External Power supply( Input):

-20 ~ + 75°C -30 ~ +85°C 10 ~ 85% , non-condensing 5 ~ 90% , non-condensing 2500Vrms 24V DC (On Terminal Board)

### **1.4 Ordering Information**

- PISO-PS400
- DN-8468GB
- DN-8468MB
- DN-8468PB
- DN-8468DB
- CA-SCSI15

4 axes PCI motion card

Terminal board for general purpose usage

Terminal board for Mitsubishi Servo motor

- Terminal board for Panasonic Servo motor
- Terminal board for Detal Servo motor
- 68-pin SCSI-II cable , length:1.5 m

# **2 HARDWARE INSTALLATION**

### 2.1 Checking Package and Installation

### 2.1.1 Check Package

PISO-PS400G/S includes the following item

- PISO-PS400 4-axes PCI motion card
- DN-8468GB
- CA-SCSI15

- Terminal board for general purpose usage
- 68-pin SCSI-II cable , length:1.5 m

### 2.1.2 Installation

#### **Prepare controller**

- 1. Choose a personal PC with empty PCI slot.
- 2. Turn power off

#### Motion card Plug-in and wiring

- 1. Switch SW1 and SW2 to desired position.
- 2. Plug in the PISO-PS400 into an empty PCI slot of PC.
- 3. Connect the PISO-PS400 with DN-8468G by a CA-SCSI15 cable, as the below figure:



### 2.1.3 SW1 Setting

The Card ID of each PISO-PS400 motion card is defined by setting the on-board swith SW1 (1~4) shown in section 2.1.2. The default setting of the Card ID is 0 by setting SW1(1~4) to be OFF. If users set 1 and 2 on SW1 to be ON, the Card ID of the motion card is 3. Up to 16 motion cards in the same system can be supported by setting different Card ID (Card ID = 0~15).



### 2.1.4 SW2 Setting

SW2(1~8) is designed for FRnet setting and is shown in section 2.1.2. Node 5 on SW2 is the transfer rate setting of FRnet. The default setting is ON for 250kHz transfer rate. If users change the node 5 to be OFF, the transfer rate will be 1MHz and the slave module must support 1MHz transfer rate to receive and send signals. Except the node 5 on SW2, the others node are for future extension setting. Do not change the default setting of SW2 except node 5. If users change the switch setting, the FRnet may not keep working.

SW2	1	2	3	4	5	6	7	8
ON								
OFF								

### 2.2 DN-8468G Terminal Board

The DN-8468 is the terminal board for general purpose amplifier usage. It has 4-axis I/O signals.

### 2.2.1 Board Layout for DN-8468G





### 2.2.2 Signal Connections for DN-8468G

Maintaining signal connections is one of the most important factors in ensuring that your application system is sending and receiving data correctly.

#### Pin Assignment for CON1

The I/O connector on the DN-8468G is a 68-pin SCSI II connector that enables you to connect to the PISO-PS400 motion card. Fig. 2.1 shows the pin assignment for the 68-pin I/O connector on the DN-8468G (or on the PISO-PS400), and refer to Table 2.1, 2.2 for description of each motion I/O signal.

		<u> </u>		
VECA	1		68	
ADOA	2		67	OF GA
XECB	- 3	-00-	66	UECB
XINPOS		юон	65	UINPOS
XALARM		юон	00	UALARM
XL.MTP		Lõõ-	04	III.MTP
YI MTM	6	പ്പ്പ	63	IT NATNA
VILLO	7	22	62	
	8		61	UIN3
XIN2	Q	-00-	60	UIN2
XIN1	10	юф	50	UIN1
XINO	10	юон	50	UINO
XEXPP		-00-	20	UF, XPP
XF. XPM	12	لمم	57	UE XPM
VOUTI	13	LĂĂ	56	UOUTI
NOOT	14		55	
APP	15		54	OPP
XPM	16	ᅮᅇᅅ	53	UPM
FR.net_A	17	юон	50	YDCC
EXPLĪN	1/	юон	51	EMGN
FR net B	18	-00-	51	XDCC
ZPM	19	പ്പ്പ	50	VDM
	20	LXX.	49	
	21	LXX-	48	
20011	22	100-	47	YOUTI
ZE XPM		-00-	46	YE XPM
ZE XPP		юон	40	YE XPP
ZINO	27	юон	4.7	YINO
ZIN1		-00-	44	YIN1
ZIN2	26	لمم	43	VIN2
7IN3	27	പ്പ്പ	42	VINI2
	28	XX	41	1111) 1111) 1111)
	29		40	
ZLMIP	30	<u>-00-</u>	30	YLMIP
ZALARM	31	မဝမ	32	YALARM
ZINPOS	21	юон	27	YINPOS
ZECB	22	юон	)/ 26	YE CB
ZECA	<u>کک</u>	Lōō-	30	YECA
(INT)	34	പ്പ്പ	35	VCC
UND		حضت		YUU
	—	CŌNI		
	-			

Fig. 2.1 I/O connector pin assignment for the CON1

Pin name	Pin number	Description	
XECA	1	Encoder A-phase signal for X axis	
YECA	36	Encoder A-phase signal for Y axis	
ZECA	33	Encoder A-phase signal for Z axis	
UECA	68	Encoder A-phase signal for U axis	
XECB	2	Encoder B-Phase signal for X axis	
YECB	37	Encoder B-Phase signal for Y axis	
ZECB	32	Encoder B-Phase signal for Z axis	
UECB	67	Encoder B-Phase signal for U axis	
XINPOS	3	In-position signal for X axis	
YINPOS	38	In-position signal for Y axis	
ZINPOS	31	In-position signal for Z axis	
UINPOS	66	In-position signal for U axis	
XALARM	4	Alarm signal for X axis	
YALARM	39	Alarm signal for Y axis	
ZALARM	30	Alarm signal for Z axis	
UALARM	65	Alarm signal for U axis	
XLMTP	5	Limit switch input signal (+) for X axis	
YLMTP	40	Limit switch input signal (+) for Y axis	
ZLMTP	29	Limit switch input signal (+) for Z axis	
ULMTP	64	Limit switch input signal (+) for U axis	
XLMTM	6	Limit switch input signal (-) for X axis	
YLMTM	41	Limit switch input signal (-) for Y axis	
ZLMTM	28	Limit switch input signal (-) for Z axis	
ULMTM	63	Limit switch input signal (-) for U axis	
XIN3	7	Input 3 signal for X axis	
YIN3	42	Input 3 signal for Y axis	
ZIN3	27	Input 3 signal for Z axis	
UIN3	62	Input 3 signal for U axis	
XIN2	8	Input 2 signal for X axis	
XIN2	43	Input 2 signal for Y axis	
XIN2	26	Input 2 signal for Z axis	
XIN2	61	Input 2 signal for U axis	
XIN1	9	Input 1 signal for X axis	
YIN1	44	Input 1 signal for Y axis	
ZIN1	25	Input 1 signal for Z axis	
UIN1	60	Input 1 signal for U axis	
XIN0	10	Input 0 signal for X axis	
YIN0	45	Input 0 signal for Y axis	
ZIN0	24	Input 0 signal for Z axis	
UIN0	59	Input 0 signal for U axis	

Table 2.1 DN-8468G I/O connector signal description (part 1)

Pin name	Pin number	Description
XEXPP	11	EXT pulsar input signal (+) for X axis
YEXPP	46	EXT pulsar input signal (+) for Y axis
ZEXPP	23	EXT pulsar input signal (+) for Z axis
UEXPP	58	EXT pulsar input signal (+) for U axis
XEXPM	12	EXT pulsar input signal (-) for X axis
YEXPM	47	EXT pulsar input signal (-) for Y axis
ZEXPM	22	EXT pulsar input signal (-) for Z axis
UEXPM	57	EXT pulsar input signal (-) for U axis
XDRIVE	13	Driver enable signal for X axis
YDRIVE	48	Driver enable signal for Y axis
ZDRIVE	21	Driver enable signal for Z axis
UDRIVE	56	Driver enable signal for U axis
XPP	14	Driving pulsar signal (+) for X axis
YPP	49	Driving pulsar signal (+) for Y axis
ZPP	20	Driving pulsar signal (+) for Z axis
UPP	55	Driving pulsar signal (+) for U axis
XPM	15	Driving pulsar signal (+) for X axis
YPM	50	Driving pulsar signal (+) for Y axis
ZPM	19	Driving pulsar signal (+) for Z axis
UPM	54	Driving pulsar signal (+) for U axis
XOUT1	16	Output 1 signal for X axis
YOUT1	48	Output 1 signal for Y axis
ZOUT1	21	Output 1 signal for Z axis
UOUT1	56	Output 1 signal for U axis
EXPLSN1	17	EXT pulse input signal for interpolation
EMGN1	52	Emergency stop input signal
FrnetA	16	FRnet port A
FrnetB	18	FRnet port B
XDCC	51	Deviation Counter Clear for X axis
YDCC	53	Deviation Counter Clear for Y axis
GND	34	Ground
VCC	35	External power (12~24V)

Table 2.2 DN-8468G I/O connector signal description (part 2)

#### (I/O connector for each AXIS) CON2 ~ CON5

The connectors CON2 ~ CON5 are 20-pin connectors that enable you to connect to the I/O signals for general purpose motor drivers. Fig. 2.2 shows the pin assignment for the 20-pin connector on the DN-8468G, and the Table 2.3 shows its I/O connector signal description.



Fig. 2.2 Pin definition for CON2 ~ CON5

Table 2.3 CON2 ~ CON5 Signal Connection		
Name	Number	Description
A+	1	Encoder A-Phase (+)
A-	2	Encoder A-Phase (-)
B+	3	Encoder B-Phase (+)
В-	4	Encoder B-Phase (-)
Z+	5	Encoder Z-Phase (+)
Z-	6	Encoder Z-Phase (-)
P+	7	Positive Direction Pulse
		Output(+)
P-	8	Positive Direction Pulse
		Output(-)
N+	9	Negative Direction Pulse
		Output(+)
N-	10	Negative Direction Pulse
		Output(-)
INP	11	Servo In Position
ALARM	12	Servo Alarm
SRV_ON	13	Servo On
LMT+	14	END Limit Signal (EL+)
LMT-	15	END Limit Signal (EL-)
IN3	16	Input Signal (IN3)
HOME	17	Home Sensor Input Signal
NHOME	18	Near Home Sensor Input
		Signal
EXP+	19	EXT Positive Direction Pulse
		(+)
EXP-	20	EXT Negative Direction Pulse
		(-)

#### CON6

The connector CON6 is 16-pin connector that enables you to connect to the signals of your motor drivers. The FRnet connectors, FR-A and FR-B, can be used to serially connect a I/O module of FRnet series, as FR-2053,FR-2057.... The more information, please refer to web-site of ICPDAS :

#### http://www.icpdas.com/products/Remote\_IO/frnet/frnet\_introduction.htm

Fig.2.3 shows the pin assignment for the 16-pin connector on the DN-8468G, and the Table 2.4 shows its I/O connector signal description.

CON6	1
1	FR-A
2	FR-B
3	X-DCC
4	Y-DCC
5	E-PLS
0	EMG-A
6	E-GND
8	X-EMG
10	Y-EMG
11	Z-EMG
12	
13	
14	7-RDY
15	U-RDY
16	E-GND

Fig. 2.3 Pin definition for CON6

Table 2.4 CON6 Signal Connection		
Description		
FRnet port A		
FRnet port B		
Deviation Counter Clear for $\boldsymbol{X}$ axis		
Deviation Counter Clear for Y axis		
EXT pulse signal		
EMG input signal for all axes		
EXT power ground		
EMG input signal for X axis		
EMG input signal for Y axis		
EMG input signal for Z axis		
EMG input signal for U axis		
Ready input signal for X axis		
Ready input signal for Y axis		
Ready input signal for Z axis		
Ready input signal for U axis		

#### Table 2.4 CON6 Signal Connection

#### ■ TB2

The connector TB2 is 5-pin connector that enables you to connect to the signals of your motor drivers. Fig.2.4 shows the pin assignment for the 5-pin connector on the DN-8468G, and the Table 2.5 shows its I/O connector signal description.



Table 2.5 TB2 Signal Connection		
Name	Description	
E-PWR	EXT power supply +24V	
E-GND	EXT power ground	
FGND	Frame ground	
	Name E-PWR E-GND FGND	

Fig. 2.4 Pin definition for TB2

▶ Note: Don't reverse connect signals with E\_PWR and E\_GND. Serious damage to your motion card and motion controller might be happen

#### ■ RJ1 (The I/O signals of the FRnet)

The connectors RJ1 is an 8-pin RJ45 connector that enable you to connect to the signals of FRnet. The FRnet connectors, FR-A and FR-B, can be used to serially connect a I/O module of FRnet series, as FR-2053,FR-2057.... The more information, please refer to web-site of ICPDAS:

#### http://www.icpdas.com/products/Remote\_IO/frnet/frnet\_introduction.htm

Fig.2.5shows the pin assignment for the 8-pin connector on the DN-8468G, and the Table 2.6 shows its I/O connector signal description.



Table 2.6 RJ1		
Pin name	Description	
FRnetA	FRnet port A	
FRnetB	FRnet port B	
NC	No connection	

Fig. 2.5 Pin definition for RJ1

Note: Don't connect NC (not connected) signals. Connecting these signals could cause permanent damage to your motion controller.

## 2.2.3 Jumper and Switch Settings

#### ■ JP7

Jumper 7 controls the EMG-A signal of the CON6 connector. The following diagram is shown the selection condition of the jumper 7.





#### ■ JP8/9, JP10/11, JP12/13, JP14/15

The Jumper8~15 are used to set the signal type of the pulse output signals. The output signal type could be differential line driver output or open collector output. The JP8 ~JP9 are set XPP XPM for X-axis(CON1), JP10 ~JP11 are for Y-axis, JP12 ~JP13 are for Z-axis and JP14 ~JP15 are for U-axis. The 2-3 Pin short is the differential line driver mode. The 1-2 Pin short is the Open Collector mode, as below example

note: Open Collector output , P+ (N+) and EXT\_5V short , offer external usage (Refer to section 2.3) .





Open Collector TTL Output

Line Drive Differential Output

Fig. 2.10 Jumper 8, 9 setting

#### EMG SW

The emergency stop signal for each servo ampilfier can be selected from EMG SW. The number 1, 2, 3, 4 on EMG SW are denoted as axis X, Y, Z, U, respectively. Fig. 2.7 is the default setting to connect the EMG singals to GND. The EMG signals from CN1 ~ CN4 will not take effect. If the switch is disconnected as shown in Fig. 2.8, the emergency stop signals can be controlled from EMG signals in CON6.



Fig. 2.7 EMG SW setting for normally GND (Default setting)



Fig. 2.8 EMG SW setting for user controlled signals.

### 2.3 Input/Output Connections

### 2.3.1 Output Pulse Signals

There are 4-axes pulse output signals on PISO-PS400, For every axis, two pairs of CW and CCW signals are used to send the pulse train. The CW and CCW signals can also be programmed as PULSE and DIR signals. Two types of the pulse output signal, Differential-Type and Open-Collector Type, can be selected from JP8/9, JP10/11, JP12/13, and JP14/15 and are described in section 2.2.3. The following wiring diagram is for the CW and CCW signals of the 4-axes.

Output to Motor Drivers in Differential Circuit



Fig. 2.9 Differential-Type pulse output circuit

Open Collector TTL Output



Fig. 2.10 Open-Collector pulse output circuit

#### • Example: wiring of pulse signal

Two types of pulse output signal, Differential-Type and Open-Collector Type, can be selected from JP8/9, JP10/11, JP12/13, and JP14/15 for each axis. The following wiring diagram is an example to select pulse type of the output signal.



#### Fig. 2.11 Output pulse example

• Pulse/Direction pulse output mode:

In Pulse/Direction pulse output mode, the PULSE signal is output only at Pulse pins (P+, P-). The driving direction is decided from the electric potential of Direction pins (N+, N-). The following diagram is example signal of Pulse/Direction pulse output mode.



• CW/CCW pulse output mode:

In CW/CCW pulse output mode, the PULSE signal is output at both CW pins (P+, P-) and CCW pins(N+, N-). At the same time, the driving direction is determined directly. The following diagram is example signal of CW/CCW pulse output mode.



## 2.3.2 Connection for Limit Switch Signal

Limit Switch Signal can prevent the over traveling appearance of the motion system. User can set the hardware limit switch signal to be normal open or normal close by the software instruction in PISO-PS400 software manual. The following figure indicates that the photo couplers are used to keep out the sensor noise of the Limit Switch.



ICPDAS Rev.2.12/12/2008 PS400 Getting Started

### 2.3.3 General Purpose Input Signal (nINPOS, nALARM)

INPOS is a digital input signal to indicate the In-Position signal of the driver. User can enable or disable the signal from the software instruction in PISO-PS400 software manual.

ALARM is a digital input signal to indicate the servo alarm signal of the driver. The output pulse will be stop if PISO-PS400 receives the ALARM signal. User can enable or disable the signal from the software instruction in PISO-PS400 software manual.



Fig. 2.12 General Digital Input circuit

### 2.3.4 Encoder Signals

The following diagram is for Differential-Type encoder signals. Connect the Phase A signal to A+ and A- pins and connect Phase B signal to B+ and B- pins. After the high speed photo coupler isolation, the isolated encoder signals are connected to motion IC.



Fig. 2.13 Encoder signal connection

### 2.3.5 Emergency Stop Signal

The following diagram is for Emergency STOP signal. If the emergency signal is occurred, the output pulse for all axes will be STOP and the error flag will be set as 1. After the photo coupler isolation, the isolated emergency signal is connected to motion IC.



Fig. 2.15 Emergency Stop Signal connection

### 2.3.6 Manual pulse Generator Input Signals (EXP+,EXP-)

The signals, EXP+ and EXP-, are used for manual pulsar signals. The following diagram is an example connection for the external inputs. User can set the signals as fixed pulse CW/CCW mode, continuous pulse CW/CCW mode, or A/B phase manual pulsar mode by using the setting in section 3.5.



Fig. 2.16 EXP+/- connection diagram

### 2.3.7 General Purpose Output Signals(Servo On/Off)

The following diagram is a digital output signal for driver Servo On/Off signal. The output signal enable or disable the driver.



Fig. 2.17 Servo On/Off signal connection diagram

### **2.4 Connection Example for Motor Driver**

The following diagram is the connection example between MITSUBISH MR-J2S AC servo driver and the extension board DN-8468G.



Fig. 2.18 The connection between MR-J2S AC servo driver and DN-8468G extension board.

## **3 SOFTWARE DEVELOPMENT OVERVIEW**

### **3.1 Software Development Overview**

Please refer to the demo\_start sample



### 3.1.1 Card Initialization

Please execute the Card Initialization procedures of each PISO-PS400 motion card before sending command, or the error message will be occurred. Please refer to *PS400\_Card\_Init()* function, the section 2.2 of PISO-PS400 user manual.

### 3.2 Safety IO Setting

There are many reasons to stop motion during driving. Some reasons are described in this subsection.

### 3.2.1 Emergency Stop Signal Input

The EMG-A input signal in CON6 is able to perform the emergency stop function immediately for all of the 4 axes during driving. The emergency stop function can prevent the critical damage occurrence from the critical accident. If user don't use this Emergency stop signal, please closing breaks between 2 and 3 of JP7 jumper. Otherwise, please closing breaks between 1 and 2 of JP7 jumper and connecting the EMG-A signal to CON6.

The EMG-X, EMG-Y, EMG-Z, and EMG-U input signals in CON6 are connected directly to the driver for each axis. These signals are able to perform the emergency stop function immediately for each driver during driving. User have to switch the EMG-SW to normal ON and connect external signal source to enable these signals.

### 3.2.2 Configure the Servo ALARM Signals

When the ALARM signals are occurred from servomotor drivers, users can be notified by these signals and determine what to do. The operating mode (Enable or Disable) and the proper trigger level of these signals can be set by user. Please refer to *PS400\_Set\_Alm()* function, the section 2.12 of PISO-PS400 user manual.

### **3.2.3 Configure the Limit Switch Signals(±EL)**

To insure the machine in safety, hardware limit switches are placed at the both ends of machine traveling range. If the machine touch the hardware limit switch sensors, PISO-PS400 will stop immediately. The operating mode (Enable or Disable) and the proper trigger level of these signals can be set by user. Please refer to *PISO-PS400\_SET\_LIMIT()* function, the section 2.7 of PISO-PS400 user manual.

### 3.2.4 Configure the Software Limite (±SEL)

To insure the machine in safety, hardware limit switches are placed at the both ends of machine traveling range. In addition, user can set the software limits to avoid the happening of the over range before the hardware limit takes effect. If the machine reach the software limits condition, PISO-PS400 will stop immediately. The operating mode (Enable or Disable) and the proper trigger condition of these signals can be set by user. Please refer to *PISO-PS400\_SET\_SoftLIMIT()* function, the section 2.9 of PISO-PS400 user manual.

### **3.3 Error Checking**

Check whether there is any error. If there are something wrongs, please use the *PS400\_Get\_Error\_Code()* function to get the error-code, then check the reason and remove it. Please refer to *PS400\_Get\_Error\_Code()* function, the section 3.6 of PISO-PS400 manual.

User also can use *PS400\_Get\_DI\_Stutus()* function to check the all of DI status. Please refer to *PS400\_Get\_DI\_Stutus()* function, the section 3.5 of PISO-PS400 user manual.

### **3.4 Basic Configuration of Motion**

The basic motion configuration must be set for necessarily general settings and are described below:

1. Pulse output mode setting: Pulse/Dir 
CW/CCW...

✓ Relative function: *PS400\_Set\_PulseMode()* 

(Please refer to the section 2.5 of PISO-PS400 user manual)

- 2. Max. speed limitation setting for each axis
  - ✓ Relative function: PS400\_Set\_MaxSpeed()

(Please refer to the section 2.6 of PISO-PS400 user manual)

- 3. Encoder input setting
  - ✓ Relative function: *PS400\_Set\_EncoderMode()*

(Please refer to the section 2.10 of PISO-PS400 user manual)

- 4. DI noise filter setting (If necessary)
  - ✓ Relative function: PS400\_Set\_Filter()

(Please refer to the section 2.14 of PISO-PS400 user manual )

5. Circular motion declaration (Ring counter) (If necessary)

✓ Relative function: *PS400\_Set\_Vring()* 

(Please refer to the section 2.15 of PISO-PS400 user manual)

### 3.5 Manual Pulse Generator Testing

User can use the manual pulse generator function directly to drive motion forward or backward. For further wiring and parameter tuning, user have to check the correction of the DI signals and the moving direction.

The manual pulse generator can be achieved from three driving methods described below:

1. A/B phase Manual Pulse Generator:

Use the A/B phase manual pulse signals for forward/backward moving.



2. Fixed-pulse driving Manual Pulse Generator:

User have to preset fixed driving pulses. After setting, user can push the forward or backward button to drive fixed pulses for each direction.

3. Continuous- pulse driving Manual Pulse Generator:

User can preset output-pulse frequency. After setting, user can push the forward or backward button to drive fixed velocity for each direction. If user release the button, the motion will be stop immediately.

ALL the above three methods and the operation mode (enable or disable) from external pulse input can be chosen and set from the function, *PS400\_Set\_ManualPulsar()* (Please refer to the section 2.18 of PISO-PS400 user manual )

### 3.6 Home Search

PS400 provides the automatic homing function. After proper settings, the homing function can be operated automatically. Four homing steps are provided for setting and are described bellow:

- Near-home sensor searching under high-speed motion.
- Home sensor searching under low-speed motion.

- Servomotor Z-phase searching under low-speed motion.
- Offset movement to the origin of the working area under high-speed motion.

User can select which steps are ignored when setting for the actual operation. It performs automatically that economize the CPU resource and program code reducing. Although there are four home search steps, but user can create more than 10 types of different home search mode by vary with the software functions. It is attributed to the configurable home search direction and perform it or not of each step.

### 3.6.1 Home Search Configuration

1. Logic level setting for Near home sensor and Home sensor

PS400\_Set\_Home () (Please refer to section 2.8 of PISO-PS400 user manual)

2. Home-speed setting

PS400\_Set\_HomeSpeed () (Please refer to section 6.1 of PISO-PS400 user manual)

3. Homing mode setting

PS400\_Set\_HomeMode ()(Please refer to section 6.3 of PISO-PS400 user manual)

### **3.6.2 Running the Home Search**

1. Start homing

PS400\_Home \_Start () (Please refer to section 6.4 of PISO-PS400 user manual)

2. Waiting for homing completion

PS400\_Home\_Done () (Please refer to section 6.5 of PISO-PS400 user manual)

### **3.7 Basic Motion**

### 3.7.1 Speed Profile of the Motion Control



1 Symmetrical T-profile of motion velocity





2 Asymmetrical T-profile of motion velocity





### 3.7.2 Basic Motion of Single Axis

- 1. Fixed-pulse driving output: There are five speed profiles
  - Constant speed (V)
    - ✓ Relative function: PS400\_Const\_Move()
  - Symmetrical T-Profile (SV \ V \ A \ AO)
  - ✓ Relative function: *PS400\_T\_Move()* 
    - Symmetrical S-curve (SV \ V \ K \ AO) ✓ Relative function: *PS400\_S\_Move()*
  - Asymmetrical T-profile (SV \ V \ A \ D \ AO)

ICPDAS Rev.2.12/12/2008 36

PS400 Getting Started
- ✓ Relative function: *PS400\_T\_As\_Move()*
- Asymmetrical S-curve (SV · V · K · L · AO)
  - ✓ Relative function: *PS400\_S\_As\_Move()*
- (Please refer to section 7.1~7.3 of PISO-PS400 user manual)
- 2. Continuous-pulse driving output: Perform continuous pulse output.
  - ✓ Relative function: *PS400\_Conti\_Move()*
  - (Please refer to section 7.4 of PISO-PS400 user manual)
- 3. Waiting for motion done: Waiting for the axis driving accomplished.
  - Relative function: *PS400\_Motion\_Done()*(Please refer to section 7.9.5 of PISO-PS400 user manual)

### 3.7.3 Basic Motion of Muti-Axes Interpolation

- 1 2-axes linear interpolation: Perform 2-axes linear interpolation.
   ✓ Relative function: *PS400\_Line2\_Move()*, *PS400\_Line2\_As\_Move()* ( Please refer to section 7.5 of PISO-PS400 user manual)
- 2 3-axes linear interpolation: Perform 3-axes linear interpolation.
  - ✓ Relative function: PS400\_Line3\_Move(), PS400\_Line3\_As\_Move()
  - (Please refer to section 7.5 of PISO-PS400 user manual)
- 3 2-axes ARC interpolation: Perform 2-axes ARC interpolation.
  - ✓ Relative function: *PS400\_Arc2\_Move ()*
  - (Please refer to section 7.6 of PISO-PS400 user manual)

#### **3.8 Synchronized Motion**

- Synchronized motion for Muti-axes (more than 2 axes) can be set in the same card by using the following function.
  - ✓ Relative function: *PS400\_Set\_SyncMotion()*

(Please refer to section 7.7 of PISO-PS400 user manual)

- Position latch functions for single axis or multi-axes are set in the same card
  - ✓ Relative function: *PS400\_Set\_Latch()*, *PS400\_Get\_Latch()*

(Please refer to section 7.7 of PISO-PS400 user manual)

- Preset the synchronous condition and data for multi-axes.
  - ✓ Relative function: *PS400\_Sync\_Preset()*, *PS400\_Preset\_Data()*

(Please refer to section 7.7 of PISO-PS400 user manual)

### 3.9 Advanced Motion

- 1. 2-axes continuous rectanglar interpolation:
- Perform the 2-axes rectanglar continuous interpolation.
  - ✓ Relative function: *PS400\_Rectangle()*
  - (Please refer to section 7.8.1 of PS400 manual)
- 2. 2-axis linear interpolation with symmetric T-curve:
- Initial setting of the 2-axis linear interpolation with symmetric T-curve
  - ✓ Relative function: *PS400\_Set\_Line2()*
- Start 2-axis linear interpolation motion
  - ✓ Relative function: *PS400\_Line2\_Start()*
  - (Please refer to section 7.8.2 of PISO-PS400 user manual)
- 3. 3-axis linear interpolation
- Initial setting of the 3-axis linear interpolation with symmetric T-curve
  - ✓ Relative function: *PS400\_Set\_Line3()*
- Start 3-axis linear interpolation motion
  - ✓ Relative function: *PS400\_Line3\_Start()*
  - (Please refer to section 7.8.3 of PISO-PS400 user manual)
- 4. 3-axis helical interpolation
- 3-axis helical interpolation with symmetric T-curve
  - ✓ Relative function: *PS400\_Helix3\_Move()*
  - (Please refer to section 7.8.6 of PISO-PS400 user manual)
- 5. 2-axis ratio movement
- The initial setting for 2-axis ratio movement with symmetric T-curve
  - ✓ Relative function: *PS400\_Set\_Ratio2()*
- Start the 2-axis ratio movement
  - ✓ Relative function: *PS400\_Ratio2\_Start()*
  - (Please refer to section 7.8.6 of PISO-PS400 user manual)

# **4 GETTING STARTED OF SOFTWARE**

#### 4.0 PISO-PS400 Installation

Please refer to Appendix A:

#### 4.1 VC 6.0 Guideline

#### 4.1.1 Confirm the Relative Files

Please confirm you have the following relevance files in the relative path:

(C:\ICPDAS\PISO-PS400\Include):

• PS400.h

(C:\ICPDAS\PISO-PS400\Lib):

• PS400DLL.lib

If you don't have, please look for CD or download the latest edition from ICPDAS's website <a href="http://www.icpdas.com/">http://www.icpdas.com/</a>

#### 4.1.2 Create a new VC++ Application Project

Please execute the Microsoft Visual C++ 6.0. Then click "File" -> "New" to create a new application project. In the "Projects" property page, choose "MFC AppWizard (exe)" option and specify the project name "PS400VCDemo", then key in the disk path in the "Location" field, then select the "Win 32" in CPU list. If necessary, please also select others options together. And then click "OK"

New						? 🛛
Files Files ATT Clu Clu Clu Clu Clu Clu Clu Clu Clu Clu	Projects L COM AppWi ister Resourc stom AppWiz tabase Project vStudio Add-i tended Storec VStudio Add-i tended Storec vStudio Add-i tended Storec vStudio Add-i tended Storec vStudio Add-i tended Storec vStudio Add-i tended Storec vStudio Add-i vStudio Ad	Workspaces izard er Type Wizard ard st n Wizard I Proc Wizard Wizard ntrolWizard [dtt] [cxc] Wizard on Application -Link Library orary	Other Documen	nts   	Project <u>mame</u> : PS400VCDemo Lo <u>cation</u> : D:\PS400\PS400VCDem Create new workspan Add to current works Dependency of: Platforms: Platforms:	ce pace
					ОК	Cancel

MFC AppWizard - Step 1	
▲ Application ○K Caseel ● Single do ● Multiple of ● Dialog ba ■ Documen What language 中文 白灣	oplication would you like to create? cument locuments sed tylew architecture support? would you like your resources in?
< Back Ne	xt > <u>F</u> inish Cancel

Choose " Dialog based " and click "NEXT"

#### Click "NEXT" following

MFC AppWizard - Step 2 of 4		? 🗙
Application Move Close About App Cancel Editing Control: Record Cancel Cancel Radio Button Radio Button	What features would you like to include? About box Context-sensitive Help 3D controls What other support would you like to include? Automation Automation ActiveX Controls Would you like to include WOSA support? Windows Sockets Please enter a title for your dialog: PS400VCDemo	
<	Back Next≻ Finish Car	ncel

Click "Finish".

🥙 PS400VCDemo - Microsoft Visual C++ - [PS40	DVCDemo.rc - IDD_PS400VCDEMO_DIALOG (Dialog)]		
] 🔄 File Edit View Insert Project Build Layout To	ols <u>Wi</u> ndow <u>H</u> elp		_ @ ×
CPS400VCDemoDlg / (All class members	] ▼   ♦ CPS400VCDemoDlg   ▼ 🔃 ▼ 🗍 🗇 🕮 🔏 🗜 🖳 🖑		
] 🎦   😅 🖬 🕼   X 📭 🖻   그 - C -   🛙	] 🗖 😤 🙀 _PS400DLL_EXPORT_ 💽 🍾		
PS400VCDemo resources Dialog Con String Table Version			
Class ) 📾 Resou ) 🗈 FileView			
■ = = = = = = = = = = = = = = = = = = =			
Build ( Debug ) Find in Files 1 )	Fin( •		×
Ready	2 m	) 🕄 🗘 0, 0	1 320 × 200 READ

Finish the new project establishment.

#### 4.1.3 Add the Reference Path into VC ++ Application Project

A. Open the "Options" dialog in "Tools" menu and select the "Directories" item.

- B. Add in the path of including files. Choose the "Include files" in "Show directories" item and double click at the empty line below the "Directories" (the nearest spare rectangular area as the following figure shown). Key in the path include the files where they installed. (C:\ICPDAS\PISO-PS400\Include).
- C. Add in the path of library files. Choose the "Library files" in "Show directories" item and double click at the empty line below the "Directories" (the nearest spare rectangular area as the following figure shown). Key in the path include the files where they installed. (C:\ICPDAS\PISO-PS400\Lib).

Options				? ×				
Editor Tabs D	ebug   Compat	ibility Build	Directories	30				
<u>P</u> latform:		Show directori	ies for:					
Win32	<b>_</b>	Include files						
<u>D</u> irectories:			🖄 🗙	<b>+ </b>				
C:\Program Files\Mi C:\Program Files\Mi C:\Program Files\Mi C:\PROGRAM FILE C:\ICPDAS\PISO-P	C:\Program Files\Microsoft Visual Studio\VC98\INCLUDE C:\Program Files\Microsoft Visual Studio\VC98\MFC\INCLUDE C:\Program Files\Microsoft Visual Studio\VC98\ATL\INCLUDE C:\PROGRAM FILES\MICROSOFT PLATFORM SDK FOR WINDOWS XP S							
			<u></u>					
			UK	Cancel				
Options				? 🗙				
<mark>Options</mark> Editor   Tabs   D	ebug   Compat	ibility   Build	Directories	<b>? X</b>				
Options Editor   Tabs   D <u>P</u> latform:	ebug   Compat	ibility   Build <u>S</u> how directori	Directories	<b>? X</b>				
Options Editor Tabs D <u>P</u> latform: Win32	ebug   Compat	ibility   Build <u>S</u> how directori Library files	Directories	<b>?</b> ×				
Options Editor Tabs D Platform: Win32 Directories:	ebug   Compat	ibility   Build <u>S</u> how directori Library files	Directories ies for:	? × } •				
Options Editor Tabs D Platform: Win32 Directories: C:\Program Files\Mi C:\Program Files\Mi C:\ICPDAS\PISO-P	ebug Compat	ibility Build Show directori Library files dio\VC98\LIB dio\VC98\MFC\I DK for Windows	Directories les for:	?× ↓ ↓ ★ ↓				
Options Editor Tabs D Platform: Win32 Directories: C:\Program Files\Mi C:\Program Files\Mi C:\Program Files\Mi C:\Program Files\Mi	ebug Compat	ibility Build Show directori Library files dio\VC98\LIB dio\VC98\MFC\I DK for Windows	Directories ies for: IIB XP SP2\Lib	? ×				
Options Editor Tabs D Platform: Win32 Directories: C:\Program Files\Mi C:\Program Files\Mi C:\Program Files\Mi	ebug Compat	ibility Build <u>Show directori</u> Library files dio\VC98\LIB dio\VC98\MFC\I DK for Windows	Directories ies for: LIB XP SP2\Lib	? ×				
Options Editor Tabs D Platform: Win32 Directories: C:\Program Files\Mi C:\Program Files\Mi C:\ICPDAS\PISO-P	ebug Compat	ibility Build Show directori Library files dio\VC98\LIB dio\VC98\MFC\I DK for Windows	Directories les for:	?× ↓ ↓ ★ ↓				
Options Editor Tabs D Platform: Win32 Directories: C:\Program Files\Mi C:\Program Files\Mi C:\Program Files\Mi C:\Program Files\Mi	ebug Compat	ibility Build Show directori Library files dio\VC98\LIB dio\VC98\MFC\ DK for Windows	Directories ies for:	? × ↓ ↓				

## 4.1.4 Start the VC++ Example

Add a BUTTON in the Dialog ,as shown below.:

PS400YCDemo - Microsoft Visual C++ - [PS400YCDemo.rc - IDD_PS400YCDEMO_DIALOG (Dialog)]	I I X
Ele Edit Yiew Inzert Project Build Leyout Iools Window Help	_ 8 ×
CPS400VCDemoDlg VIDC_BUTTON1 VBN_CLICKED VC 🛛 🗷 V	
📔 🥔 🖬 🕼 🐍 🕰 - 🖂 - 🖂 👦 😤 🙀 _PS400DLL_EXPORT_ 🔄 🦕	
■ PS400VCDemo resources + ● Dialog ● Icon ● String Table ● Version TODO: 在此處放置對話方現控制項。 取消	
■ Class  ■ Resou  ■ FileView	
Build / Debug > Find in Files 1 > Find 4	• •
Ready 71, 66 🛱 80 × 28	READ

DoubleClick on the BUTTON and create a sub-routine in the code editor.

Add one line "#include "PS400.h" and declare other variables at the beginning of this file as shown in the following figure

🗴 PS400VCDemo - Microsoft Visual C++ - [PS400VCDemoDlg.cpp]
EIE Edit Yiew Insert Project Build Tools Window Help
CPS400VCDemoDlg 🗸 (All class members) 🚽 💊 OnButton1 📃 🕱 👻 🕼 🖄 🖽 🚣 🗜 🗒
🖹 😂 🖬 🕼 🐇 🖻 🛍 🕰 ד 🗠 ד 📴 🗖 😤 🎇 PS400DLL_EXPORT_ 💽 🉀
#include "PS400VCDemoDlg.h"
□ □ PS400VCDemo resources     #include "PS400.h"
String Table     Static char THIS_FILE[] =FILE;
Version Version
bool Driver_Open = false;
BYTE CardID[16];
// CAboutDlg dialog used for App About
Class Resou FileView
× PS400VCDemo.exe - 0 error(s), 0 warning(s)
Build / Debug / Find in Files 1 / Find
Ready Ln 0, Col 0 REC COL OVR READ

Call the specific function from PS400.DLL library in the OnButton1 sub-routine as shown in following figure:

अ PS400∀CDemo - Microsoft ¥isual C++ - [PS-	400¥CDemoDlg.cpp]					
Eile Edit View Insert Project Build Iools	<u>N</u> indow <u>H</u> elp		_ & ×			
CPS400VCDemoDig - (All class membe	rs] 🗾 💊 OnButton1	- 🗷 -	🕸 🖽 🗶 ! 🗉 🕀			
12   <b>2 - 3 - 4</b>   % 🖻 🖻   <b>2 -</b> 2 - 1	🖪 🗖 😤 🉀 _PS400DLL_EXP	ORT_ 💽 🐪				
PS400VCDemo resources Dialog IDD_ABOUTBOX IDD_PS400VCDEMO_DIA Con String Table Version	<pre>void CPS400VCDemoDlg::OnE {     // TODO: Add your cor     short wRet;     //===== Step 1 Driver     if (!Driver_Open)     {         wRet = PS400_Carc         if (wRet != SUCCE         {</pre>	utton1() trol notificat init Linit (); SS_NO_ERROR ) (TEXT("PS400 C	ion handler code   ard init() Failed			
Class 📓 Resou 🖹 FileView	return; 7		- ···			
PS400UCDemo.exe - 0 error(s), 0 warning(s)						
Ready		Ln 182, Col 5	REC COL OVR READ			

#### Detailed code as follows:

```
//===='Step 1 Driver init
 short wRet;
 if (!Driver_Open)
 {
       wRet = PS400_Card_Init ();
       if (wRet != SUCCESS_NO_ERROR )
       {
            AfxMessageBox(TEXT("PS400_Card_init() Failed!") );
            return;
       }
       else
       {
            short card_num = PS400_Total_Card();
            for (short i = 0; i < card_num; i++)
            {
                  CardID[i] = PS400_Get_CardNo((BYTE)i);
            }
            CardNo = CardID[0]; // pick up the 1st motion card
            Driver_Open = true;
       }
```

```
}
   //===='Step 2 CONFIG IO
     PS400 Reset Card(CardNo);
     PS400_Set_PulseMode(CardNo, AXIS_XYZU, 2); //set the pulse output mode
     PS400_Set_Alm(CardNo, AXIS_XYZU, 0, 0); //disable the SERVO ALARM Input
     PS400 Set EncoderMode(CardNo, AXIS XYZU, 0); //set the encoder input type
     PS400_Set_MaxSpeed(CardNo, AXIS_XYZU, 16000); //set the max speed for XYZU
     PS400 T Move(CardNo, AXIS XYZU, 500, 10000, 5000, 0, 50000 ); // Starting volecity = 500, Maximum
volecity = 10000, Acceleration = 5000, Offset Pulse = 0, Pulse Command = 50000
     PS400_Set_Servo_ON(CardNo, AXIS_XYZU, 1);
                                                              //set the Servo_ON to servo motors
     //====='Step 3 Check ERROR
     while (PS400_Motion_Done(CardNo, AXIS_XYZU) == NO)
     {
          Sleep(1000);
          //wait for axis to stop
     }
```

Please refers to the example "First\_demo" for detail.

After you finished that, please choose the menu, "Project"->"Setting". The "Project Settings" dialog is appeared as shown below. Then, select the "Link" item and key in "PS400DLL.lib" into the "Object/library modules" field, and click "OK".

Project Settings	? 🛛
Settings For: Win32 Debug ▼	General Debug C/C++ Link Resources M () Category: General Reset Output file name:
	Debug/PS400VCDemo.exe Object/library modules: PS400DLL.lib
	<ul> <li>✓ Link incrementally</li> <li>☐ Generate mapfile</li> <li>☐ Enable profiling</li> <li>Project Options:</li> </ul>
	/nologo /subsystem:windows /incremental:yes /pdb:"Debug/PS400VCDemo.pdb" /debug /machine:1386 /out:"Debug/PS400VCDemo.exe"

# 4.1.5 Build the Project

Please choose "Build"->"Build All" in the menu, then you will be finished this simple program if there is nothing wrong.

🚧 PS400VCDemo - Microsoft Vi	sual C++ - [PS400¥CDemoDlg.c	[gu			
Eile Edit View Insert Project	<u>Build</u> <u>T</u> ools <u>W</u> indow <u>H</u> elp			_ 8 ×	
CPS400VCDemoDlg 🔪 (All o	<ul> <li>Compile PS400VCDemoDlg.cpp</li> <li>Build PS400VCDemo.exe</li> <li>Rebuild All</li> </ul>	Ctd+F7 F7	▼ 🗷 ▼ 🔰 🔹	> 🔛 👗 📜 地	
B - S400VCDemo resou B - Salog B - Salog B - B - B - B - B - B - B - B - B - B -	Batch B <u>u</u> ild Clean Start <u>D</u> ebug		g::OnButton1() ur control notification	handler code	
IDD_ABOUTBO IDD_PS400VCD ⊕ ⊡ Icon ⊕ ⊡ String Table	Debugger Remote Connection Execute PS400VCDemo.exe	Ctrl+F5	Driver init en)		
🗄 🦳 Version	Set Active C <u>o</u> nfiguration Con <u>f</u> igurations <u>P</u> rofile	Ē	ð_Card_Init (); SUCCESS NO ERROR )		
		AfxMessa return;	ageBox(TEXT("PS400_Card	_init() Failed	
📲 Class ) 🏙 Resou ) 🔳	FileView			•	
PS400VCDemo.exe - 0 er	rror(s), 0 warning(s)				
∭ <b>. I ▶   Build / Debug </b> } Fi:	nd in Files 1 $\lambda$ Find $  $				
Rebuilds Active Project and a	all project dependencies		Ln 182, Col 5 REC	COL OVR READ	

#### 4.2 VB 6.0 Guideline

#### 4.2.1 Confirm the Relative Files

Please confirm you have the following relevance files in the relative path:

(C:\ICPDAS\PISO-PS400\Include) PS400.bas

If you don't have, please look for CD or download the latest edition from ICPDAS's website <u>http://www.icpdas.com/</u>.

#### 4.2.2 Create a New VB Application Project

Please execute the Microsoft Visual Basic 6.0. Then click "File" -> "New Project" to create a new application. In the "New Projects" dialog, choose "Standard EXE" template, and click "OK" in the end.

Ne	w Project					? 🗙
		Micro	sual	Bas		$\leq$
]	New Existin	ng   Recent				
(	Standard EXE	ActiveX EXE	ActiveX DLL	ActiveX Control	VB Applicati	
	VB Wizard Manager	ActiveX Document Dll	Activex Document Exe	Addin	Data Project	
					開啓(0 取消 説明(H	
Γ	Don't show thi	is dialog in the fy	iture			

Select the "File"→"Save Project" in pull-down menu.

🍓 Project1 - Microsoft Visual	Basic [desig	n]				
File Edit <u>V</u> iew Project Format	Debug <u>R</u> un	Query	Diagram	<u>T</u> ools	<u>A</u> dd-Ins	W
<u>N</u> ew Project ② Open Project	Ctrl+N Ctrl+O	N I		11		7
A <u>d</u> d Project <u>R</u> emove Project		)				
Save Project Save Project As	>					
<u>Save Form1</u> Save Form1 <u>A</u> s Save Selection Save Change Script	Ctrl+2					
🞒 Print 🛄 Print Setup	Ctrl+P					
Ma <u>k</u> e Project1.exe Make Project <u>G</u> roup		· · · · · ·				
1\DEMO\VB6\First_Demo\Pro 2 D:\Motion\PS400\PS400\PS400\Pb yb	oject1.vbp			:::::	:::::	Ļ

Specify the file name of the form as "PS400VBDemo", then click "Save".

Save File As	? 🛛
儲存於①: 🗁 PS400	▼ 🕂 🖻 🕈
檔案名稱(U): PS400VBDemo	儲存⑤
存檔類型(I): Form Files (*.frm)	▼ 取消



Specify the file name of the Project as "PS400VBDemo" then click "Save".

Choose "Yes" or "No" to determine adding the source code into the SourceCodeControl.



A new project establishment is finished.

## 4.2.3 Add The PS400.BAS into the VB Application Project

Append the "PS400.BAS" to the project as below: Select the " Project"→"Add Module" in pull-down menu.



In "Add Module" popup dialog, choose "Existing" page and found the file, "PS400.BAS" in the specified directory.

(C:\ICPDAS\PISO-PS400\Include)

Add Module			? 🔀
New Existing			
Look in: 🗀 Include	•	] 🗢 🗈 👩	* 💷 -
<b>*</b> PS400.BAS			
File <u>n</u> ame:			<u>O</u> pen
Files of type: Basic Files (*.	bas)	<u> </u>	Cancel <u>H</u> elp
Don't show this dialog in the	f <u>u</u> ture		

## 4.2.4 Start the VB Sample

Add a BUTTON in the Dialog, as shown below:



Double click the "BUTTON" and create a sub-routine in the code editor.

Add the declarations

Dim cardNo As Byte

Dim Driver\_Open As Boolean

Dim CardID(16) As Byte

at the beginning of this file. Call the specific function of PS400DLL library in the "command1\_Click" sub-routine as shown in the following figure:



Detailed code are presented below:

Dim wRet As Integer Dim card\_num As Integer Dim i As Integer

#### 'Step 1 Driver init

```
If Driver_Open = False Then
```

wRet = PS400\_Card\_Init()

If wRet <> SUCCESS\_NO\_ERROR Then

MsgBox "PS400\_Card\_init() Failed!"

End

Else

Card\_Init = True

card\_num = PS400\_Total\_Card()

```
For i = 0 To card_num Step 1
CardID(i) = PS400_Get_CardNo(i)
Next
```

```
cardNo = CardID(0) 'pick up the 1st motion card
Driver_Open = True
```

End If

End If

'Step 2 Configure the Motion Card

```
PS400_Reset_Card (cardNo)

wRet = PS400_Set_PulseMode(cardNo, AXIS_XYZU, 2) 'set the pulse output mode

wRet = PS400_Set_Alm(cardNo, AXIS_XYZU, 0, 0) 'disable the SERVO ALARM Input

wRet = PS400_Set_EncoderMode(cardNo, AXIS_XYZU, 0) 'set the encoder input type

wRet = PS400_Set_MaxSpeed(cardNo, AXIS_XYZU, 16000) 'set the max speed for XYZU

wRet = PS400_T_Move(cardNo, AXIS_XYZU, 500, 10000, 5000, 0, 50000) 'Starting volecity = 500, Maximum

volecity = 10000, Acceleration = 5000, Offset Pulse = 0, Pulse Command = 50000
```

wRet = PS400\_Set\_Servo\_ON(cardNo, AXIS\_XYZU, 1) 'set the Servo\_ON to servo motors

'Step 3 Waiting for Motion done

Do While PS400\_Motion\_Done(cardNo, AXIS\_XYZU) <> NO Sleep (1000)

Please refer to the example "First\_demo" for further details

## 4.2.5 Build the Project

🍖 Project1 - Microsoft Visual H	lasic [desig	n]
<u>File</u> Edit <u>V</u> iew Project Format I	Debug <u>R</u> un	Query Diagram <u>T</u> ools <u>A</u> dd-Ins
<u>N</u> ew Project	Ctrl+N	🗠 🗠 🕨 🔳 📓 📓
🚰 Open Project	Ctrl+O	
A <u>d</u> d Project		
<u>R</u> emove Project		0
层 Sa <u>v</u> e Project		lode)
Sav <u>e</u> Project As		- (
<u>S</u> ave PS400VBDemo.frm	Ctrl+S	= O To card_num Step 1
Save PS400VBDemo.frm <u>A</u> s		dID(i) = PS400_Get_CardNc t
Save Selection		n link(0) i ish ul
Save Change Script		_Open = True
Print	Ctrl+P	
Print Setup		
Make PS400VBDemo.exe	>	
Make Project <u>G</u> roup		ure the Motion Card rd (cardNo)
<u>1</u> D:\P\$400\P\$400\BDemo.vbp		et_PulseMode(cardNo, AXI)
2\DEMO\VB6\First_Demo\Proj	ect1.vbp	et_EncoderMode(cardNo, A)
3 D:\Motion\PS400\PS400\PS400\b.vbp	12.ubn	et_MaxSpeed(cardNo, AXIS_ Move(cardNo, AXIS XYZU,

Select the "File"→" Make PS400VBDemo.exe" in pull-down menu.

After pressing the button "OK", this simple program is finished, if there is no error message happened.

Make Project	? 🛛
儲存於①: 🔂 PS400 🔽	← 🗈 💣 🎟+
檔案名稱(N): PS400VBDemo	ОК
	取消
	說明( <u>H</u> )
	Options

### 4.3 BCB 6 Guideline

#### 4.3.1 Confirm The Relative Files

Please confirm you have the following relevance files in the relative path:

(C:\ICPDAS\PISO-PS400\Include):

• PS400BCB.h

(C:\ICPDAS\PISO-PS400\Lib):

• PS400BCB.lib

If you don't have, please look for CD or download the latest edition from ICPDAS's website <u>http://www.icpdas.com/</u>.

### **4.3.2 Create a New BCB Application Project**

Please execute the Borland C++ Builder 6.0, then click "File"  $\rightarrow$  "New"  $\rightarrow$  "Application" to create a new project.



Select the "File"→"Save Project As..." in pull-down menu and save this project.



#### 4.3.3 Start the BCB Sample

Drag an UI component "Tbutton" from VCL ToolBox, and draw it on the FORM as shown below:

Form1	
Button1	
· · · · · · · · · · · · · · · · · · ·	

DoubleClick "button1" on the Form1 to popup the code editor window as shown below:

Double Click "button1" on the "Form1" dialog. The code editor window will be popup as shown

below:

Double click the "BUTTON" and create a sub-routine in the code editor.

Add the declarations

Dim cardNo As Byte

Dim Driver\_Open As Boolean

Dim CardID(16) As Byte

at the beginning of this file. Call the specific function of PS400DLL library in the "command1\_Click" sub-routine as shown in the following figure:



Add the declarations

#include "PS400BCB.h"
bool Driver\_Open;
BYTE CardNo=0;
BYTE CardID[16];

at the beginning of this file.

🗎 Unit1.cpp		
→ <u>×</u> → Classes	Unit1.cpp	$\leftarrow \cdot \rightarrow \cdot$
	//	^
	<pre>#include <vcl.h></vcl.h></pre>	
	#pragma hdrstop	
	#include "Unit1.h"	_
	#include "PS400BCB.h" hool Driver Onen:	
	BYTE dardNo=0;	
	BYTE CardID[16];	
	#pragma package(smart init)	
	#pragma resource "*.dfm"	
	TForm1 *Form1:	
	//	
	fastcall TForm1::TForm1(TComponent* Owner)	~
		>
XII.	3: 7 Modified Insert <u>Unit1.cpp</u> /Unit1.h/Ulagram/	
[C++ Error] Unit1.cpp(46): E24 [C++ Error] Unit1.cpp(53): E24	IS1 Undefined symbol 'CardNo' IS1 Undefined symbol 'CardNo'	
Build/		

Coding in the "Button1Click" function section :

```
short wRet;
//==== Step 1 Driver init
if (!Driver_Open)
{
    wRet = PS400_Card_Init ();
    if (wRet != SUCCESS_NO_ERROR )
    {
       Application->MessageBox( "PS400_Card_init() Failed!", "ERROR", MB_OK );
       return;
    }
    else
    {
       short card_num = PS400_Total_Card();
       for (short i = 0; i < card_num; i++)
       {
            CardID[i] = PS400_Get_CardNo((BYTE)i);
       }
       CardNo = CardID[0]; // pick up the 1st motion card
       Driver_Open = true;
    }
}
//===='Step 2 Configure the Motion Card
PS400_Reset_Card(CardNo);
PS400_Set_PulseMode(CardNo, AXIS_XYZU, 2);
                                                   //set the pulse output mode
                                                 60
```

```
PS400_Set_Alm(CardNo, AXIS_XYZU, 0, 0); //disable the SERVO ALARM Input

PS400_Set_EncoderMode(CardNo, AXIS_XYZU, 0); //set the encoder input type

PS400_Set_MaxSpeed(CardNo, AXIS_XYZU, 16000); //set the max speed for XYZU

PS400_T_Move(CardNo, AXIS_XYZU, 500, 10000, 5000, 0, 50000 ); // Starting volecity = 500, Maximum

volecity = 10000, Acceleration = 5000, Offset Pulse = 0, Pulse Command = 50000

PS400_Set_Servo_ON(CardNo, AXIS_XYZU, 1); //set the Servo_ON to servo motors

//====='Step 3 Waiting for Motion done

while (PS400_Motion_Done(CardNo, AXIS_XYZU) == NO)

{

Sleep(1);

//wait for axis to stop

}
```

Please refer to the example "First\_demo" for further details

#### 4.3.4 Add The Reference Path into the BCB Application Project

Please selecting the "Directories/Conditional" Tab in the menu "Project"  $\rightarrow$  "Options" to add the installation path (C:\ICPDAS\PISO-PS400\Include, C:\ICPDAS\PISO-PS400\Lib) into the Include path and Library path.

#### 4.3.5 Add the PS400BCB.lib into the project

Select the "Project" → "Add to Project..." in pull-down menu.



Select "Library File (\*.lib)" from the file-type-filter ComboBox. Choose the file "PS400BCB.lib" and click "Open".

Add to project		? 🗙
查詢①:	🔁 PS400BCBDemo 💌 🖛 🗈 📸 🕶	
我最近的文件	PS400BCB	
<b></b> 身面		
<b>夏</b> 夏 我的電腦		
網路上的芳鄰		
(	檔名(M): PS400BCB ▼	開啓(0)
	檔案類型(I): Library file (*.lib)	取消

### 4.3.6 Build the Project

Please choose "Project"  $\rightarrow$  "Build Project1" in the menu. This simple program is finished, if there is no error message happened.



# 5 PISO-PS400 PCEzGo(by Basic Function)

The initial frame of PISO-PS400 PCEzGo is shown in the following figure. Four categories of test function are displayed in the initial frame.

![](_page_63_Picture_2.jpeg)

1. Initial Settings: (in section 5.1.1 ~ 5.1.4)

Four sub-items, "<u>Registration and Hardware Signals</u>", <u>"Servo Input /Output</u> <u>Signals</u>", <u>"Compare Register Counters</u>", and <u>"Interrupt Configuration</u>", are individually presented in "Initial Settings" Dialog. Please refer to the illustrations of each section for further detail.

2. Command: (in section 5.2.1 ~ 5.2.3)

This category is set the Common Motion commands and the advanced commands. Three sub-items, "<u>Acc/Dec Motion</u>" (Constant, T-profile, S-curve), "<u>Automatic Home Search</u>", and "<u>Jog Operation</u>", are individually presented in "Command" Dialog. Please refer to the illustrations of each section for further detail.

- Interpolation: (in section 5.3.1 ~ 5.3.2)
   This category is set the Interpolation commands. Two sub-items, "<u>Linear and Circular</u> interpolation" and "<u>Offline profiling</u>", are individually presented in "Interpolation" dialog. Please refer to the illustrations of each section for further detail.
- 4. FRnet Demo: (In section 5.4)

The FRnet DI/DO dialog is for the FRnet remote I/O data transmission test. The FRnet can scan the dynamic data directly at a constant period without CPU communicating. Therefore, the FRnet DI/DO transmission can be regarded as a background daemon to provide the DI/DO data for motion control. And also the FRnet DI/DO module in this motion card provides an interrupt signal with different timer base. Please refer to the illustrations of each

section for further detail.

## **5.1 Initial Settings Dialog**

There are four sub-items, "<u>Registration and Hardware Signals</u>", <u>"Servo Input /Output</u> <u>Signals</u>", <u>"Compare Register Counters</u>", and <u>"Interrupt Configuration</u>", in "Initial Settings" Dialog. The initial settings must be finished to continue the following motion operations if the PCEzGo is executed again. Please refer to the illustrations of each section for further detail.

## **5.1.1 Registration and Hardware Signals**

Initia	l Setting					×
Re	egistration and	Hardware Sigr	als   Servo Inp	outs Signals   9	oftLimit & V-Rin	ig Interru
	Output Pulse	Mode				
	CW/CCV PULSE/E	V: 0~1 )IR:2~5	0 💌			
	-Hardware Sig	nal Settings—				
		X-Axis	Y-Axis	Z-Axis	U-Axis	
	LIMIT+/-	Low 💌	Low	Low 🔻	Low 🔻	
	HOME (IN1)	Low	Low	Low	Low	
	NEAR HOME (IN0)	Low	Low	Low	Low	
	INDEX (IN2)	Low	Low	Low	Low	
	SERVO	ON/OFF	ON/OFF	ON/OFF	ON/OFF	
	Card Configurat Card Index	ion 5	<ul> <li>En /</li> </ul>		Save	Exit
				Johng 7	Conng	

#### Group definition and brief description

- 1. Card Configuration:
  - Press the "InitCard" button to search all PISO-PS400 motion cards plugged in the PC. The searched cards are listed its ID number in order. Choose the ID number of the controlled motion card and press "Apply" to finish the Card Configuration.
    - ✓ Relative Functions: PS400\_Card\_Init(), PS400\_Toatal\_Card(), PS400\_Get\_CardNo() ∘
- 2. Output Pulse Mode:
  - The types of pulse output are classified into 6 modes: 0, 1 is CW/CCW dual channel mode, 2~5 is PULSE/DIR single channel mode. Finish the operation choice of pulse output mode after determining the mode ID and clicking the button, "Apply".

- ✓ Relative Function: PS400\_Set\_PulseMode() ∘
- 3. Hardware Signals Settings:
  - The logical trigger level (Hi/Low) of the hardware signals, front/rear traveling limits(LIMIT+/-), home limit sensor(HOME), near home limit sensor(NEAR HOME), servo motor Z-phase signal(INDEX), and servo on/off (SERVO ON/OFF), are set in this sub-item. The logical trigger level (Hi/Low) of each signals for each axis can be set individually. Finish the hardware signals settings after determining the logical trigger level and clicking the button, "Apply".
    - ✓ Relative Function: PS400\_Set\_Limit(), PS400\_Set\_Home(), PS400\_Set\_Servo\_ON()
- 4. Function of Buttons
  - Apply: designate the card number and apply all of the setting values of this dialog.
  - Load: Load the preserve setting values which are stored previously.
  - SaveConfig: Reserve settings of the hardware signals into the ini file.
  - Exit: Exit this dialog window.

### 5.1.2 Servo Input Signals

Initial Setting		X
Registration and Hardware	Signals Servo Inputs Sign	als   SoftLimit & V-Ring   Interru
InPosition Signal	-Axis	-Pulse Input Signal
🗖 Enable/Disable	🔽 X-Axis 🔽 Y-Axis	① 1/1 ΔB Dhase     ③
Active Level Low 💌	🔽 Z-Axis 🔽 U-Axis	© 1/2 AB Phase
Servo Alarm Signal	N3 Signal	O 1/4 AB Phase
🗖 Enable/Disable	🗖 Enable/Disable	○ cw/ccw
Active Level	Active Level Low 💌	
_Input Signal Filter		Stop Mode
Input Sig Delay Time	EMG, LMT, ORG, SD INDEX	€ Emg.Stop
None Selected 💌	INP, ALM	O Slowdown Stop
	IN3	
Card Configuration		
Card Index 5	Load Config	Save Exit

#### Group definition and brief description

- 1. Servo InPosition Signal
  - Configurable feature enable/disable and logical trigger level of the Servo In-position signal.
    - ✓ Relative Function: PS400\_Set\_Inp().
- 2. Axis Setting

- To assign the axes to set properties of the input signals.
- 3. Servo Alarm Signals
  - Configurable feature enable/disable and logical trigger level of the Servo Alarm signal.
    - ✓ Relative Function: PS400\_Set\_Alm().
- 4. IN3 Signal
  - Configurable feature enable/disable and logical trigger level of the IN3 signal.
    - ✓ Relative Function: PS400\_Set\_Input().
- 5. Pulse Input Signal
  - Configure the encoder input mode as AB phase or CW/CCW (Up/Down count). Specify the frequency division at AB phase mode.( 0=1:1 , 1=1:2 , 2=1:4 )
    - ✓ Relative Function: PS400\_Set\_EncoderMode().
- 6. Digital filters of the input signals
  - Setting the delay time of each input signal filter: The suitable delay time and the related removable maximum noise width are listed in the following table:

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

- Setting the input signals with digital filter:
  There are five check box (FE0 ~ FE4) to set the input signals to use digital filter.
  FE0 is for Emg. Signal (EMGN), +/- limits (LMT±), Home limit(IN1), and Near Home limit(IN0)
  FE1 is for Encoder Z phase signal (IN2)
  FE2 is for Servo In-position signal (INP) and Servo alarm signal (ALM).
  FE3 is for +/- external pulse input(EXP+/EXP-).
  FE4 is for IN3 signal.
  ✓ Relative Function: PS400\_Set\_Filter() ∘
- 7. Function of Buttons
  - LoadConfig: Load the preserve setting values which are stored previously.
  - SaveConfig: Reserve settings of the hardware signals into the ini file.

#### 5.1.3 Setting SoftLimit & V-Ring Function

Initial Setting	×
Servo Inputs Signals SoftLimit & V-Ring Interrupt Configuration	••
SoftLimit & VRing Settings	
O Disable SLimit or VRing	
• Enable Software Limit	
Compare Mode Logic Position 💌	
Compare 🔽 X-Axis 🔽 Y-Axis 🔽 Z-Axis 🔽 U-Axis Source:	
SL+ 1000000 1000000 1000000 1000000	
SL1000000 -1000000 -1000000 -1000000	
© Enable Variable Ring	
Image: Counter     Image: Counte	
Card Configuration Card Index 5	kit

#### Group definition and brief description

- 1. Disable SLimit or VRing:
  - SLMT+/-: Disable the software limit.

- ✓ Relative function: *PS400\_Disable\_SoftLimit()*,
- VRing: Disable the VRing.
  - ✓ Relative function: PS400\_Set\_Vring();
- 2. Enable Software Limit
  - Counter Mode : Choose the compare register source: Logic Position or Real Position.
  - To set the SLMT+/- value of 4-axes to the compare register counter.
  - Relative function: *PS400\_Disable\_SoftLimit()*,,PS400\_Set\_Compare().
- 3. Enable Variable Ring:
  - To set the Variable Ring counter value of 4-axes.
  - Relative Function: PS400\_Set\_Vring()
- 4. Function of Buttons
  - LoadConfig: Load the particular Files(.ini) which are stored by user previously..
  - SaveConfig: Save the relative hardware signal setting to the .ini file.

## 5.1.4 Interrupt Configuration

ompare Register Cou	nters I	nterrupt	: Configu	uration	<u>•</u>
Motion INT Factor Se	ettings X-Axis	Y-Axis	Z-Axis	U-Axis	Card Info Address
Pulse-Up	Г	Г	Г		Dxaa61c8
Position Counter >= Comp- Register	Г	Г	Г	Г	IRQ Level
Position Counter < Comp- Register	Г	Г		Г	10
Position Counter < Comp+ Register	Г	Г		Г	
Position Counter >= Comp+ Register	Г	Г	Γ		
End of Constant Speed Drive	Г	Г	Г	Г	
Start of Constant Speed Drive	Г	Г	Γ	Г	
Drive Finished	Г	Г		Г	
Home Termination	Г	Г	Г	Г	
				-	

#### Group definition and brief description

1. Interrupt Factor Setting

Ten kinds of interrupt event settings are provided in PISO-PS400 motion card.

- 1. **Pulse-Up:** the interrupt is produced as the first pulse happened.
- 2. **Position Counter >= Comp- Counter:** Position counter is greater than or equal to the negative direction compared register.
- 3. **Position Counter < Comp- Counter:** Position counter is less than the negative direction compared register.
- 4. **Position Counter >= Comp+ Counter:** Position counter is greater than or equal to the positive direction compared register.
- 5. **Position Counter < Comp+ Counter:** Position counter is less than the positive direction compared register.
- 6. End of Constant Speed Drive,
- 7. Start of Constant Speed Drive,
- 8. Drive Finished,
- 9. Home Termination,
- 10. Interrupt event in Synchronous action.

User can exercise the multi-thread concept to program the interrupt service routine in threads and co-operate with the called interrupt function.

✓ Relative Function: *PS400\_Set\_INT\_Factor()*, *PS400\_Enable\_INT()*,

PS400 Getting Started

PS400\_Disable\_INT(), PS400\_Is\_INT\_Active(), PS400\_Get\_INT\_Flag(), PS400\_Clear\_INT\_Flag().

#### **5.2 Command Dialog**

### 5.2.1 Acc/Dec Motion

Motion Command			
Acc/dec Motion (C/T/S) Home Search and Jog Operation Synchronous Action			
Acc Mode Axis		Driving Mode	
C Const 🔽	X-Axis 🔽 Y-Axis	F-DRV+	
© T-Curve	Z-Axis 🔲 U-Axis	C F-DRV-	
		C-DRV+	
Parameters		C C-DRV-	
Start Velocity (SV)	5000 PPS		
Drive Velocity (V)	80000 PPS		
Acceleration (A)	20000 PPS/Sec	Acc/Dec Sym	
Deceleration (D)	20000 PPS/Sec		
Jerk (K)	5000 PPS/Sec^2	C Asym	
Decelerating Rate (L)	1 PPS/Sec^2		<u>C</u> lear
Output Pulse (P)	50000		Hold
Offset Pulse (AO)	0		<u></u>
Position and Velocity Counters			
LP EP E	Err CV LMT	- SLMT- ORG SD SLMT+ LMT+ DRV RDY ALM EMG	GetErr
X: 0 0	0 0		South South
Y: 0 0	0 0		Reset
Z: 0 0	0 0 •		
	uj u 🖝		Exit

#### Group definition and brief description

1. Acc Mode

PISO-PS400 provides 3 basic modes of the Acc./Dec. profile: Const. speed, T-Curve, and S-Curve.

- 2. Axis
  - To assign the axes to set the desired command.
- 3. Driving Mode
  - PISO-PS400 provides 4 driving modes: fixed-amount driving in positive direction, fixed-amount driving in negative direction, continuous driving in positive direction, and
continuous driving in negative direction.

- ✓ Relative Function: PS400\_Const\_Move(), PS400\_T\_Move(), PS400\_T\_Move(), PS400\_T\_As\_Move(), PS400\_S\_Move(), PS400\_S\_As\_Move(), PS400\_Conti\_Move().
- 4. Acc/Dec Sym
  - The default Acc./Dec. profile is symmetrical. User can specify the relative parameters, deceleration (D) and deceleration rate (L), to configure as asymmetrical (Asym) profile.
    - ✓ Relative Function: *PS400\_T\_As\_Move()*, *PS400\_S\_As\_Move()*.
- 5. Position and Velocity Counters
  - The dialog is separated into two sub-pages: "position and velocity counters" and "I/O Signal Status".
  - The position/velocity counter sub-page displays the motion information for each axis, such as the logical position counter (LP), physical position counter (EP), position error (Error), and current velocity (Current Speed).
  - The I/O Signal Status sub-page displays the I/O signals of motion, such as the hardware limit switch, Home, NRHome, IN3, Alarm, EMG....etc..
- 6. Function of Buttons
  - Go: Perform the motion operation after click "Apply".
  - Stop: Stop motion immediately.
  - ClearStop: Clear the flag of the Stop command.
    - After you click "Stop", you must click "ClearStop" for performing motion later.
  - Clear: Clear logical and actual register of the 4 axes.
  - Hold: Press "Hold" before click "Go". This will suspend motion commands until the "Release" button is pressed.
  - Release: Release the "Hold" operation.
  - GetError: Click it to check the ErrorCode displayed in the 3rd column of StatusBar on the screen below.
  - Reset: Restore PS400 to the initial state
  - Exit: Exit this dialog window.

#### Motion Command x Acc/dec Motion (C/T/S) Home Search and Jog Operation Synchronous Action Axis **Atomatic Home Search** Homing Mode 🔽 X-Axis 0: Dir-, NearHome, Lmt--🔽 Y-Axis Start Velocity (SV) 500 pps Drive Velocity (V) 80000 PPS 🔽 Z-Axis <u>G</u>o Acceleration (A) 50000 pps/sec 🔽 U-Axis 5000 PPS/Sec Home Speed (HV) Stop 1000 Offset Pulse (AO) <u>C</u>learStop Jog Operation Clear • Fix-Pulse Mode O Contious Mode C Manual Pulsar Mode O Disable Hold Drive Velocity (V) 50000 PPS Step Pulse (P) 1000 Release Position and Velocity Counters LΡ EΡ Err CV LMT-SLMT-ORG SD SLMT+LMT+DRV RDY ALM EMG GetErr x: [ 0 Ο σΓ 0 Y: [ σΓ σ σ Reset z: [ Ο 0 ο 0 U: [ σ σ Ο 0 Exit

## 5.2.2 Home Search and Jog Operation

#### Group definition and brief description

- 1. Automatic Home Search
  - PISO-PS400 provides several Homing modes, user can compose these 4 steps by himself. PCEzGo only provides 12 modes for application and testing.
  - 12 Home search modes are descript as the below table:

mode 0/1	Hardware signal	Enable/Disable	Direction
step1	Near Home (IN0)	Yes	-/+
step2	Home (IN1)	No	-/+
step 3	Z-phase Index	No	-/+
step 4	Offset Pulse	Yes	User define
mode 2/3	Hardware signal	Enable/Disable	Direction

step 1	Near Home (IN0)	No	-/+
step 2	Home (IN1)	Yes	-/+
step 3	Z-phase Index	No	-/+
step 4	Offset Pulse	Yes	User define
mode 4/5	Hardware signal	Enable/Disable	Direction
step 1	Near Home (IN0)	Yes	-/+
step 2	Home (IN1)	Yes	-/+
step 3	Z-phase Index	No	-/+
step 4	Offset Pulse	Yes	User define
mode 6/7	Hardware signal	Enable/Disable	Direction
step 1	Near Home (IN0)	No	-/+
step 2	Home (IN1)	Yes	-/+
step 3	Z-phase Index	Yes	-/+
step 4	Offset Pulse	Yes	User define
mode 8/9	Hardware signal	Enable/Disable	Direction
step 1	Near Home (IN0)	Yes	-/+
step 2	Home (IN1)	No	-/+
step 3	Z-phase Index	Yes	-/+
step 4	Offset Pulse	Yes	User define
mode10/11	Hardware signal	Enable/Disable	Direction
step 1	Near Home (IN0)	Yes	-/+
step 2	Home (IN1)	Yes	-/+
step 3	Z-phase Index	Yes	-/+
step 4	Offset Pulse	Yes	User define

- ✓ Relative Function: PS400\_Set\_HomeSpeed(), PS400\_Set\_HomeMode(), PS400\_Home\_Start().
- 2. Axis
  - Specify the motion axes for home search and jog operation.
- 3. Jog Operation
  - The operation in PCEzGo can be operated from the external signals (EXP+ / EXP-) input which can be produced from an external hardware, such as the Manual Pulse Generator or switch buttons. Four driving modes are provided in this jog operation: JOG Disable, Fix-Pulse Mode, Continuous Mode, and Manual Pulse Mode.
    - Relative Function: PS400\_Set\_ManualPalsar().

✓

- 4. Parameter settings
  - Homing parameters includes starting velocity (SV), driving velocity (V), home speed (HV), acceleration(A), and offset pulse (AO), are set in "Automatic Home Search" sub-page. For Jog operation, only driving velocity (V) at constant velocity and step pulse (P) are needed to be written in the "Jog Operation" sub-page.
- 5. I/O Signal Status
  - PS400 provides the LED indication of HW/SW signals and Servo status of each axis. The HW/SW signals includes the hardware limit switch sensors (LMT+/LMT-), software limit (SLMT+/SLMT-), Home limit switch sensors (Home) and Near Home limit switch sensor (NRHome). Servo status signals includes the pulse-driving (DRV), servo in position (INP), servo alarm (ALARM), and Emergency stop (EMG).
    - ✓ Relative Function: *PS400\_Get\_DI\_Status()*.
- 6. Function of Buttons
  - The same as the illustration in section 5.2.1.

## **5.2.3 Synchronous Action**

Motion Command	×
Acc/dec Motion (C/T/S) Home Search and Jog Operation Synchronous Action	
Provocative Factors	
P>=Comp+ □ P>=Comp- □ IN3 (L->H)	
P <comp+ (h-="" d-sta="" in3="">L) F-DRV-</comp+>	
□ P <comp- d-end<="" td="" □=""><td></td></comp->	
C C-DRV-	
• F-DRV+ O C-DRV+ O SSTOP O LPSAV	
X Axis	Stop
Parameters Active	
Start Velocity (SV) 500 PPS Y Axis	
Drive Velocity (V) 80000 PPS	Clear
Acceleration (A) 15000 PPS/Sec	
	Hold
Compare Value(+/-) 25000	<u>R</u> elease
Position and Velocity Counters	
LP EP Err CV LMT-SLMT-ORG SD SLMT+ LMT+ DRV RDY ALM EMG	db GotErr
	Gecch
	<u>R</u> eset
	Exit

#### Group definition and brief description

- 1. Provocative Factors
  - Eight provocative factors can induce the synchronous action. These factors are described as:

P≥COMP+: Pulse counter is greater than or equal to COMP+,

P<COMP+: Pulse counter is smaller than COMP+,

P<COMP-: Pulse counter is smaller than COMP-,

P≥COMP-: Pulse counter is greater than or equal to COMP-,

D-STA: Constant speed starts,

D-END: Deceleration is ended,

- **IN3 (L\rightarrowH):** nIN3 is in rising edge,
- IN3 (H $\rightarrow$ L): nIN3 is in falling edge

- 2. Provocative Mode
  - There are four driving modes of the provocative axis. They are
    - F-DRV+: Fixed pulse movement in positive direction,
    - F-DRV-: Fixed pulse movement in negative direction,
    - **C-DRV+:** Positive continuous movement,
    - C-DRV-: Negative continuous movement.
- 3. Action Factors
  - Specify the action of the active axis as the provocative factors happened. Six actions are acted in synchronous mode. They are
    - F-DRV+: Fixed pulse movement in positive direction,
    - F-DRV-: Fixed pulse movement in negative direction,
    - C-DRV+: Positive continuous movement,
    - C-DRV-: Negative continuous movement,
    - **SSTOP:** Deceleration stop,
    - ISTOP: Immediate stop.

In latch mode, another two actions can also work. They are

- **LPSAV:** Store the current logic position  $\, \cdot \,$  [Command  $\rightarrow$  BR]
- **EPSAV:** Store the feedback position [Position  $\rightarrow$  BR]
- ✓ Relative Function: PS400\_Set\_SyncMotion(), PS400\_Set\_Latch(), PS400\_Get\_Latch() ∘
- 4. Axis
  - Specify the provocative (main) axis and the active (synchronous) axes.

#### 5. Parameters

- The parameters, Starting Velocity (SV), Driving Velocity (V), Acceleration(A), Output Pulse (P), and the Compare Value are set in this sub-page.
- 6. Function of Buttons
  - The same as the illustration in section 5.2.1.

## **5.3 Interpolation Dialog**

## 5.3.1 Linear Interpolation and Circular Interpolation

Interpolation			×
Linear and Circular Interpo	lation Offline Profile		
Interpolation Mode	-Acc Mode	Axis Dispostion	
C Linear	🔿 Const	Axis1 X-Axis 🔻	
O Circular	T-Curve		
	C S-Curve	Axis2 Y-Axis 💌	
Parameters		1	
Start Velocity (SV)	500 PPS	Axis3 None 💌	
Drive Velocity (V)	80000 PPS	AreMede	
Acceleration (A)	50000 PPS/Sec	© CW	
Deceleration (D)	5000 PPS/Sec	O CCW	
Jerk (K)	20000 PPS/Sec^2	And Date Date	***
Decelerating Rate (L)	10000 PPS/Sec^2	Sym	B→ <u>G</u> o
Offset Pulse (AO)	0	C Asym	
Finish Points and Cente	r Points		
50000	FP2 50000	FP3 0	
CP1 0	CP2 50000	СРЗ	
	. 1		
Position and Velocity Coun	ters		
LP EP	Err CV LMT-	SLMT-ORG SD SLMT+ LMT+ DRV RDY	ALM EMG
x:   0  0	o  o •		
Y: 0 0	0 0		eset
Z: 0 0	0 0		
, , ,			
	U] U 🛡		

#### Group definition and brief description

- 1. Interpolation Mode Setting
  - PISO-PS400 PCEzGo offers two interpolation modes: Linear and Circular interpolation.
  - ✓ Relative Function: PS400\_Line2\_Move(), PS400\_Line2\_As\_Move(), PS400\_Line3\_Move(), PS400\_Line3\_As\_Move(), PS400\_Arc2\_Move() ∘
- 2. Acc. Mode Setting
  - Three Acc. Modes can be chosen for interpolation: Const. Mode, T-Curve Model, and S-Curve Mode.
- 3. Axis Configuration
  - Specify the motion axes for interpolation.

- 4. Direction setting of Arc Mode
  - Direction of Arc interpolation can be configured into CW or CCW.
- 5. Position and Velocity Counters
  - There are 2 tab-pages of the dialog which monitors position, velocity counters and the I/O status.
  - The position/velocity counters tab-page displays the information of logic position, actual position, position error and current speed of each axis.
    - ✓ Relative Function: PS400\_Get\_Cpmmand(), PS400\_Get\_Position(), PS400\_Get\_Speed() ∘
- 6. Function of Buttons
  - Go: Perform the motion operation after click "Apply".
  - Stop: Stop motion immediately.
  - ClearStop: Clear the flag of the Stop command. After you click "Stop", you must click "ClearStop" for performing motion later.
  - Clear: Clear logical and actual register of the 4 axes.
  - GetError: Click it to check the ErrorCode displayed in the 3rd column of StatusBar on the screen below.
  - Reset: Restore PS400 to the initial state
  - Exit: Exit this dialog window.

Interpolation		0.00	pue (ile il					2
Linear and Circ	ular Interpolati	on Offline	Profile				_	
1000	00							
5000	)							
0								<b>4</b> 8
-500	0							<u>Stop</u>
-100	00						ZoomIn	<u>C</u> learStop
	-10000	-5000	0	5000	10000			<u>C</u> lear
Position and Ve	locity Counters	= ]						
LP	EP	Err	CV LI	MT-SLMT-OR	S SD SLMT+	LMT+ DRV RD	DY ALM EMG	
X: 0	0	0	0		• •			Get Error
Y: 0	0	0	0		• •			<u>R</u> eset
Z: 0	0	0	0		• •			and the second
U: 0	0	0	0		• •	• • •		Exit

## 5.3.2 Two axes path plane diagram

1. Function of Buttons

- Profile: Draw the logical position of interpolation.
- Zoom In: Enlarge view of the axes path.
- Zoom Out: Shrink view of the axes path.

## 5.4 FRnet DI/DODemo

FRnet (F	Remote	DIO) De	mo					X	I
SAn	(Digit	al Out	put) S	et —				FRnet Setting	
0	1	2	3	4	5	6	7	SAn Node (SA0~SA7) 1	
8	9			12	13	14	15	RAn Node 9	
- RAn	(Digit	al Inp	ut) Se	t					
0	1	2	З	4	5	6	7		
•	•	0	•	•	0	•	•		
8	9	10	11	12	13	14	15		
0	•	•	0	•	•	0	•	Recover <b>E</b> xit	

#### Group definition and brief description

- 1. SAn (Digital Output) Set
  - FR-net Communication feature: 128 (16×8) digital input totally.
    - ✓ Relative Function: PS400\_Write\_FRnet() ∘

#### 2. RAn (Digital Input) Set

- FR-net Communication feature: 128 (16×8) digital output totally.
  - ✓ Relative Function: PS400\_Read\_FRnet() ∘

#### 3. FRnet Settings

- Active Cardno is displayed in this tab-page.
- User can select the RAn node (RA8~RA15) and the SAn node (SA0~SA7) for DI/DO.

# **APPENDIX-A PISO-PS400 Installation**

Please execute "setup.exe" from the directory "//PISO-PS400/Setup\_2K\_XP/" in CD and click "Next".



Key-in User Name and Company Name, then click "Next".

PISO-PS400 Setup		×
Customer Information Please enter your information.		
	Please enter your name and the name of the company for which you work.	
	User Name: Jeff Chen	
	 Company Name: ICP DAS Inc.	
InstallShield	<pre>&lt; Back Next &gt; Cancel</pre>	

Specify the desired path for installation, then click the "Next" button. ICPDAS 83 Rev.2.12/12/2008

PS400 Getting Started

PISO-PS400 Setup		×
Choose Destination Location Select folder where setup will install files.		
	Install PISO-PS400 to: C:\ICPDAS\PISO-PS400	Change
<b>Install</b> Shield	< <u>B</u> ack	Cancel

Click "Install" to begin the installation.

PISO-PS400 Setup		×
Ready to Install the Program The wizard is ready to begin inst	allation.	
	Click Install to begin the installation.	
	If you want to review or change any of your installation settings, click Back. Click Cancel to exit the wizard.	
InstallShield	< Back Install Cancel	



Please click the "Finish" button for reboot after installation accomplished

#### File tree after installation:

C:\ICPDAS\PISO-PS400	
Include	Header Files
├LIB	Library Files
Drivers	PISO-PS400 drivers and registration information file
Win2K	The WDM for Windows 2000
└──WinXP	The WDM for Windows XP
Manuals	HW/SW manuals
Samples	Sample Code
└──VC6	VC6 sample code
First_Demo.dsw	detailed sample code Workspace
PS400_VC6.dsw	VC 6.0 Dialog basedsampleWorkspace
PS400_INT_Console.	dsw VC 6.0 Console interrupt sample Workspace
	VB6 sample code
First_demo	Development and testing demo code
BCB6	BCB6 sample code
First_Demo	BCB 6.0 first program sample code
└──Ultility	PCEzGo.exe

## **APPENDIX-B Others Terminal Boards**

## **B.1 DN-8468M Daughter Board**

The DN-8468M is the daughter board for Mitsubishi J2 Series Amplifier. It has 4-axis I/O signals.

## **B.1.1 Board Layout for DN-8468M**



Fig. 1-1 Board layout for the DN-8468M

## **B.1.2 Signal Connections for DN-8468M**

Maintaining signal connections is one of the most important factors in ensuring that your application system is sending and receiving data correctly.

#### Pin Assignment for CON1

The I/O connector on the DN-8468M is a 68-pin SCSI II connector that enables you to connect to the PISO-PS400 motion card. Please refer to the section 2.2.1( page 15).

#### ■ TB1

The connector TB1 is 7-pin connector that enables you to connect to the signals of your motor drivers. Fig.1-3 shows the pin assignment for the 7-pin connector on the DN-8468M, and the Table 1-4 shows its I/O connector signal description.



Table 1-4 TB1 Signal Connection

Name	Description
FR-A	FRnet port A
FR-B	FRnet port B
X-DCC	Deviation Counter Clear for X axis
Y-DCC	Deviation Counter Clear for Y axis
E-PLS	EXT pulse signal
EMG-A	EMG input signal for all axes
E-GND	EXT power ground

Fig. 1-3 Pin definition for TB1

#### ■ TB2

The connector TB2 is 5-pin connector that enables you to connect to the signals of your motor drivers. Fig.1-4 shows the pin assignment for the 5-pin connector on the DN-8468M, and the Table 1-5 shows its I/O connector signal description.

#### Table 1-5 TB2 Signal Connection



Pin name	Description
E-PWR	EXT power supply +24V
E-GND	EXT power ground
FGND	Frame ground

Fig. 1-4 Pin definition for TB2

▶ Note: Don't reverse connect signals with E\_PWR and E\_GND. Serious damage to your motion card and motion controller might be happened.

#### CN-XA, CN-YA, CN-ZA, CN-UA (CNA connector for each AXIS)

The connectors CN-XA, CN-YA, CN-ZA, and CN-UA are 20-pin connectors that enable you to connect to the CNA connector of Mitsubishi motor drivers. Fig.1-5 shows the pin assignment for the 20-pin connector on the DN-8468M, and the Table 1-6 shows its I/O connector signal description.



Fig. 1-5 Pin definition for CN-XA, CN-YA, CN-ZA, CN-UA

#### Table 1-6 CNA Signal Connection

Name	Number	Description
A+	6	Encoder A-Phase (+)
A-	16	Encoder A-Phase (-)
B+	7	Encoder B-Phase (+)
B-	17	Encoder B-Phase (-)
Z+	5	Encoder Z-Phase (+)
Z-	15	Encoder Z-Phase (-)
P+	3	Positive Direction Pulse Output(+)
P-	13	Positive Direction Pulse Output(-)
N+	2	Negative Direction Pulse Output(+)
N-	12	Negative Direction Pulse Output(-)
INP	18	Servo In Position
RDY	19	Servo Ready
E-PWR	9	EXT power +24∨
E-GND	1, 10, 20	EXT power ground
NC	4,8,11,14	No connection

Note 1: There are two sets encoder signals for X and Y axes. In X axis, one is from CN-XA and the other is from CN5. In Y axis, one is from CN-YA and the other is from CN6. Users can select encoder signals from JP1 and JP2, respectively.
 Note 2: In Z and U axes, only one set of encoder signals is used for each axis. In Z axis, do not connect CN-ZA and CN7 at the same time. In U axis, do not connect CN-UA and CN8 at the same time.
 Note 3 : Don't connect NC (not connected) signals. Connecting these signals could cause permanent damage to your motion controller.

#### CN-XB, CN-YB, CN-ZB, CN-UB (CNB connector for each AXIS)

The connectors CN-XB, CN-YB, CN-ZB, and CN-UB are 20-pin connectors that enable you to connect to the CNB connector of your motor drivers. Fig.1-6 shows the pin assignment for the 20-pin connector on the DN-8468M, and the Table 1-7 shows its I/O connector signal description.

E-GND NC NC NC NC E-PWR С 14 NC RESET O 15 SVON EMG б 16 NC E-GND E-GND NC 8 18 ALARM NC 9 19 NC NC 1020 E-GND E-GND

		4
Pin	Pin	Description
SVON	5	Servo On
RESET	14	Servo Reset
EMG	15	Emergent Stop
ALARM	18	Servo Alarm
E-PWR	13	EXT power +24V
E-GND	1, 10,	EXT power ground
	16,17, 20	
NC	2, 3, 4, 6,	No connection
	7, 8, 9, 11,	
	12, 19	

Table 1-7 CNB Signal Connection

Fig. 1-6 Pin definition for CN-XB, CN-YB CN-ZB, CN-UB

Note: Don't connect NC (not connected) signals. Connecting these signals could cause permanent damage to your motion controller.

### ■ CN1~CN4 (The I/O signals of the X, Y, Z, U AXIS )

The connectors CN1~CN4 are 11-pin connectors that enable you to connect to the signals of your motor drivers. Fig.1-7 shows the pin assignment for the 20-pin connector on the DN-8468M, and the Table 1-8 shows its I/O connector signal description.

Table 1-8 CN1~4 Signal Connection



Pin name	Description
E-PWR	EXT power supply +24V
EMG	EMG input signal
LMT+	Limit Switch Input Signal (+)
LMT-	Limit Switch Input Signal (-)
INPUT3	Input Signal (IN3)
NRHOME	Near Home Sensor Input Signal
HOME	Home Sensor Input Signal
RESET	Reset input signal
EXP+	EXT Positive Direction Pulse (+)
EXP-	EXT Negative Direction Pulse (-)
E-GND	EXT power ground

Fig. 1-7 Pin definition for CN1~CN4

The connectors CN5~CN8 are 15-pin connectors that enable users to connect the signals to external motor drivers. Fig.1-8 shows the pin assignment for the 15-pin connector on the DN-8468M, and the Table 1-9 shows its I/O connector signal description.



Table	1-9	CN5~	-8
			-

Name	No.	Description
A+	9	Encoder A-Phase (+)
A-	4	Encoder A-Phase (-)
B+	10	Encoder B-Phase (+)
B-	5	Encoder B-Phase (-)
Z+	11	Encoder Z-Phase (+)
Z-	12	Encoder Z-Phase (-)
P+	1	Positive Direction Pulse Output(+)
P-	7	Positive Direction Pulse Output(-)
N+	2	Negative Direction Pulse Output(+)
N-	8	Negative Direction Pulse Output(-)
E-PWR	15	EXT power +24V
E-GND	3	EXT power ground
EXT-5V	6	EXT power +5V
NC	13,	No connection
	14	

Fig. 1-8 Pin definition for CN5~CN8

►	Note 1: There are two sets encoder signals for X and Y axes. In X axis, one is from CNX
	and the other is from CN5. In Y axis, one is from CNY and the other is from CN6.
	Users can select encoder signals from JP1 and JP2, respectively.
►	Note 2: In Z and U axes, only one set of encoder signals is used for each axis. In Z axis, do
	not connect CNZ and CN7 at the same time. In U axis, do not connect CNU and
	CN8 at the same time.
•	Note 3 : Don't connect NC (not connected) signals. Connecting these signals could cause

permanent damage to your motion controller.

#### ■ RJ1 (The I/O signals of the FRnet)

The connectors RJ1 is an 8-pin RJ45 connector that enable you to connect to the signals of FRnet. Fig.1-9 shows the pin assignment for the 8-pin connector on the DN-8468M, and the Table 1-10 shows its I/O connector signal description.



Fig. 1-9 Pin definition for RJ1

Note: Don't connect NC (not connected) signals. Connecting these signals could cause permanent damage to your motion controller.

## **B.1.3 Jumper and Switch Settings**

#### ■ JP5

Jumper 5 controls the EMG-A signal of the TB1 connector. The following diagram is shown the selection condition of the jumper 5.



#### ■ JP1, JP2

The encoder signals of axis X and axis Y can be chosen from servo driver encoder or external encoder. Fig. 1-11 shows that the encoder signals are selected from servo driver encoder. In meantime, Fig. 1-12 shows that the encoder signals are selected from external encoder.





Fig. 1-12 External encoder signals setting

#### ■ EMG SW

The emergency stop signal for each servo ampilfier can be selected from EMG SW. The number 1, 2, 3, 4 on EMG SW are denoted as axis X, Y, Z, U, respectively. Fig. 1-13 is the default setting to connect the EMG singals to GND. The EMG signals from CN1 ~ CN4 will not take effect. If the switch is disconnected as shown in Fig. 1-14, the emergency stop signals can be controlled from EMG signals in CN1 ~ CN4.



Fig. 1-13 EMG SW setting for normally GND (Default setting)



Fig. 1-14 EMG SW setting for user controlled signals.

## **B.2 DN-8468P Daughter Board**

The DN-8468P is the daughter board for Panasonic A4 Series Ampilifier. It has 4-axis I/O signals.

## **B.2.1 Board Layout for DN-8468P**



Fig. B2-1 Board layout for the DN-8468P

## **B.2.2 Signal Connections for DN-8468P**

Maintaining signal connections is one of the most important factors in ensuring that your application system is sending and receiving data correctly.

#### Pin Assignment for CON1

The I/O connector on the DN-8468P is a 68-pin SCSI II connector that enables you to connect to the PISO-PS400 motion card. Please refer to the section 2.2.1( page 15).

#### ■ TB1

The connector TB1 is 7-pin connector that enables you to connect to the signals of your motor drivers. Fig.1-3 shows the pin assignment for the 7-pin connector on the DN-8468P, and the Table 1-4 shows its I/O connector signal description.



Fig. 3-3 Pin definition for TB1

la	ole 3-4	IB1	Signal	Connection	

Name	Description
FR-A	FRnet port A
FR-B	FRnet port B
X-DCC	Deviation Counter Clear for X axis
Y-DCC	Deviation Counter Clear for Y axis
E-PLS	EXT pulse signal
EMG-A	EMG input signal for all axes
E-GND	EXT power ground

The connector TB2 is 5-pin connector that enables you to connect to the signals of your motor drivers. Fig.1-4 shows the pin assignment for the 5-pin connector on the DN-8468P, and the Table 1-5 shows its I/O connector signal description.



Table 1-5 TB2 Signal Connection		
Pin name	Description	
E-PWR	EXT power supply +24V	
E-GND	EXT power ground	
FGND	Frame ground	

Fig. 1-4 Pin definition for TB2

▶ Note: Don't reverse connect signals with E-PWR and E-GND. Serious damage to your motion card and motion controller might be happened.

#### ■ CNX, CNY, CNZ, CNU (CN X5 connector for each AXIS in Driver)

The connectors CNX, CNY, CNZ, and CNU are 50-pin connectors that enable you to connect to the CN X5 connector of Panasonic motor drivers. Fig.1-5 shows the pin assignment for the 50-pin connector on the DN-8468P, and the Table 1-6 shows its I/O connector signal description.



Fig. 1-5 Pin definition for CNX, CNY, CNZ, CNU

Table 1-6 CN X5 Signal Connection			
Name	Number	Description	
A+	21	Encoder A-Phase (+)	
A-	22	Encoder A-Phase (-)	
B+	48	Encoder B-Phase (+)	
B-	49	Encoder B-Phase (-)	
Z+	23	Encoder Z-Phase (+)	
Z-	24	Encoder Z-Phase (-)	
P+	4	Positive Direction Pulse Output(+)	
P-	3	Positive Direction Pulse Output(-)	
N+	6	Negative Direction Pulse	
N-	5	Negative Direction Pulse Output(-)	
INP	39	Servo In Position	
RDY	35	Servo Ready	
SVON	29	Servo On	
A-CLR	31	Alarm Clear	
ALARM	37	Servo Alarm	
E-PWR	7	EXT power +24V	
E-GND	8, 9, 13, 15,17, 25, 33,34, 36, 38,41	EXT power ground	
NC	1,2,10,11, 12,14,16, 18,19,20, 26,27,28, 30,32,40, 42,43,44, 45,46,47, 50	No connection	

Note 1: There are two sets encoder signals for X and Y axes. In X axis, one is from CNX and the other is from CN5. In Y axis, one is from CNY and the other is from CN6. Users can select encoder signals from JP1 and JP2, respectively.

Note 2: In Z and U axes, only one set of encoder signals is used for each axis. In Z axis, do not connect CNZ and CN7 at the same time. In U axis, do not connect CNU and CN8 at the same time.

Note 3 : Don't connect NC (not connected) signals. Connecting these signals could cause permanent damage to your motion controller. The connectors CN1~CN4 are 11-pin connectors that enable you to connect to the signals of your motor drivers. Fig.1-7 shows the pin assignment for the 20-pin connector on the DN-8468P, and the Table 1-8 shows its I/O connector signal description.



Fig. 1-7 Pin definition for CN1 ~ CN4

Table 1-8 CN1~4 Signal Connection		
Pin name	Description	
E-PWR	EXT power supply +24V	
EMG	EMG input signal	
LMT+	Limit Switch Input Signal (+)	
LMT-	Limit Switch Input Signal (-)	
INPUT3	Input Signal (IN3)	
NRHOME	Near Home Sensor Input Signal	
HOME	Home Sensor Input Signal	
RESET	Reset input signal	
EXP+	EXT Positive Direction Pulse (+)	
EXP-	EXT Negative Direction Pulse (-)	
E-GND	EXT power ground	

#### ■ CN5~CN8 (The I/O signals of the X, Y, Z, U AXIS )

The connectors CN5~CN8 are 15-pin connectors that enable users to connect the signals to external motor drivers. Fig.1-8 shows the pin assignment for the 15-pin connector on the DN-8468P, and the Table 1-9 shows its I/O connector signal description.



Table 1-9 CN5~8		
Name	No.	Description
A+	9	Encoder A-Phase (+)
A-	4	Encoder A-Phase (-)
B+	10	Encoder B-Phase (+)
B-	5	Encoder B-Phase (-)
Z+	11	Encoder Z-Phase (+)
Z-	12	Encoder Z-Phase (-)
P+	1	Positive Direction Pulse Output(+)
P-	7	Positive Direction Pulse Output(-)
N+	2	Negative Direction Pulse Output(+)
N-	8	Negative Direction Pulse Output(-)
E-PWR	15	EXT power +24V
E-GND	3	EXT power ground
EXT-5V	6	EXT power +5V
NC	13, 14	No connection

- Note 1: There are two sets encoder signals for X and Y axes. In X axis, one is from CNX and the other is from CN5. In Y axis, one is from CNY and the other is from CN6. Users can select encoder signals from JP1 and JP2, respectively.
- Note 2: In Z and U axes, only one set of encoder signals is used for each axis. In Z axis, do not connect CNZ and CN7 at the same time. In U axis, do not connect CNU and CN8 at the same time.
- Note 3 : Don't connect NC (not connected) signals. Connecting these signals could cause permanent damage to your motion controller.

#### ■ RJ1 (The I/O signals of the FRnet)

The connectors RJ1 is an 8-pin RJ45 connector that enable you to connect to the signals of FRnet. Fig.1-9 shows the pin assignment for the 8-pin connector on the DN-8468P, and the Table 1-10 shows its I/O connector signal description.



Table 1-10 RJ1		
Pin name	Description	
FRnetA	FRnet port A	
FRnetB	FRnet port B	
NC	No connection	

Fig. 1-9 Pin definition for RJ

Note: Don't connect NC (not connected) signals. Connecting these signals could cause permanent damage to your motion controller.

## **B.2.3 Jumper and Switch Settings**

#### ■ JP5

Jumper 5 controls the EMG-A signal of the TB1 connector. The following diagram is shown the selection condition of the jumper 5.



Fig. 1-10 Jumper 5 setting

#### ■ JP1, JP2

The encoder signals of axis X and axis Y can be chosen from servo driver encoder or external encoder. Fig. 1-11 shows that the encoder signals are selected from servo driver encoder. In meantime, Fig. 1-12 shows that the encoder signals are selected from external encoder.



Fig. 1-11 Primary encoder signals setting



Fig. 1-12 External encoder signals setting

The emergency stop signal for each servo ampilfier can be selected from EMG SW. The number 1, 2, 3, 4 on EMG SW are denoted as axis X, Y, Z, U, respectively. Fig. 1-13 is the default setting to connect the EMG singals to GND. The EMG signals from CN1 ~ CN4 will not take effect. If the switch is disconnected as shown in Fig. 1-14, the emergency stop signals can be controlled from EMG signals in CN1 ~ CN4.



Fig. 1-13 EMG SW setting for normally GND (Default setting)



Fig. 1-14 EMG SW setting for user controlled signals.

## **B.3 DN-8486Y Daughter Board**

The DN-8468Y is the daughter board for Yaskawa Ampilifier. It has 4-axis I/O signals.

## **B.3.1 Board Layout for DN-8468Y**



Fig. 3-1 Board layout for the DN-8468Y

## **B.3.2 Signal Connections for DN-8468Y**

Maintaining signal connections is one of the most important factors in ensuring that your application system is sending and receiving data correctly.

#### ■ Pin Assignment for CON1

The I/O connector on the DN-8468Y is a 68-pin SCSI II connector that enables you to connect to the PISO-PS400 motion card. Please refer to the section 2.2.1( page 15).

#### ■ TB1

The connector TB1 is 7-pin connector that enables you to connect to the signals of your motor drivers. Fig.3-3 shows the pin assignment for the 7-pin connector on the DN-8468Y, and the Table 3-4 shows its I/O connector signal description.



#### TB2

The connector TB2 is 5-pin connector that enables you to connect to the signals of your motor drivers. Fig.3-4 shows the pin assignment for the 5-pin connector on the DN-8468Y, and the Table 3-5 shows its I/O connector signal description.



Fig. 3-4 Pin definition for TB2

Note: Don't reverse connect signals with E-PWR and E-GND. Serious damage to your motion card and motion controller might be happened.

The connectors CNX, CNY, CNZ, and CNU are 50-pin connectors that enable you to connect to the CN X5 connector of Panasonic motor drivers. Fig.3-5 shows the pin assignment for the 50-pin connector on the DN-8468Y, and the Table 3-6 shows its I/O connector signal description.

	Name	Number	Description
	A+	21	Encoder A-Phase (+)
NC 2 00 27 NC	A-	22	Encoder A-Phase (-)
P- 3 00 28 NC	B+	48	Encoder B-Phase (+)
P+ 4 0 0 19 SVON	В-	49	Encoder B-Phase (-)
N- 6 00 31 ACB	Z+	23	Encoder Z-Phase (+)
E-PWR - 00 - 12 NG	Z-	24	Encoder Z-Phase (-)
E-GND 9 00 33 E-GND	P+	4	Positive Direction Pulse Output(+)
NC 10 00 35 ROY	p.	3	Positive Direction Pulse Output(-)
NC 11 00 36 E-GND	N+	6	Negative Direction Pulse
NC 12 00 37 ALARM	N-	5	Negative Direction Pulse Output(-)
NC 14 00 39 INP	INP	39	Servo in Position
E-GND 15 00 40 NC	RDY	55	Servo Ready
NC 17 00 41 E-GND	SVON	29	Servo On
NC 18 0 0 43 NC	A-CLR	31	Alarm Clear
NC 19 00 44 NC	ALARM	37	Servo Alarm
NC 21 00 46 NC	E-PWR	7	EXT power +24V
A 22 00 47 NG	E-GND	8, 9, 13,	EXT power ground
Z+ 23 00 48 B+		15,17, 25,	
Z: 25 00 10 B		33,34, 36,	
E-GND - COOF- NC		38,41	
	NC	1,2,10,11,	No connection
Fig. 3-5 Pin definition for CNX,		12,14,16,	
CNY, CNZ, CNU		18,19,20, 26,27,28	
		30,32,40	
		42,43,44	
		45.46.47.	

Table 3-6 CN X5 Signal Connection

Note 1: There are two sets encoder signals for X and Y axes. In X axis, one is from CNX and the other is from CN5. In Y axis, one is from CNY and the other is from CN6. Users can select encoder signals from JP1 and JP2, respectively.

50

- Note 2: In Z and U axes, only one set of encoder signals is used for each axis. In Z axis, do not connect CNZ and CN7 at the same time. In U axis, do not connect CNU and CN8 at the same time.
- Note 3 : Don't connect NC (not connected) signals. Connecting these signals could cause permanent damage to your motion controller.

The connectors CN1~CN4 are 11-pin connectors that enable you to connect to the signals of your motor drivers. Fig.3-7 shows the pin assignment for the 20-pin connector on the DN-8468Y, and the Table 3-8 shows its I/O connector signal description.



	HOME
	RESET
1 22 242 2 20 20 14	

Fig. 3-7 Pin definition for CN1 ~ CN4

Table 3-8 CN1~4 Signal Connection		
Pin name	Description	
E-PWR	EXT power supply +24V	
EMG	EMG input signal	
LMT+	Limit Switch Input Signal (+)	
LMT-	Limit Switch Input Signal (-)	
INPUT3	Input Signal (IN3)	
NRHOME	Near Home Sensor Input Signal	
HOME	Home Sensor Input Signal	
RESET	Reset input signal	
EXP+	EXT Positive Direction Pulse (+)	
EXP-	EXT Negative Direction Pulse (-)	
E-GND	EXT power ground	

#### ■ CN5~CN8 (The I/O signals of the X, Y, Z, U AXIS )

The connectors CN5~CN8 are 15-pin connectors that enable users to connect the signals to external motor drivers. Fig.3-8 shows the pin assignment for the 15-pin connector on the DN-8468Y, and the Table 3-9 shows its I/O connector signal description.



- Note 1: There are two sets encoder signals for X and Y axes. In X axis, one is from CNX and the other is from CN5. In Y axis, one is from CNY and the other is from CN6. Users can select encoder signals from JP1 and JP2, respectively.
- Note 2: In Z and U axes, only one set of encoder signals is used for each axis. In Z axis, do not connect CNZ and CN7 at the same time. In U axis, do not connect CNU and CN8 at the same time.
- Note 3 : Don't connect NC (not connected) signals. Connecting these signals could cause permanent damage to your motion controller.

### RJ1 (The I/O signals of the FRnet)

The connectors RJ1 is an 8-pin RJ45 connector that enable you to connect to the signals of FRnet. Fig.3-9 shows the pin assignment for the 8-pin connector on the DN-8468Y, and the Table 3-10 shows its I/O connector signal description.



Note: Don't connect NC (not connected) signals. Connecting these signals could cause permanent damage to your motion controller.

## **B.3.3 Jumper and Switch Settings**

#### ■ JP5

Jumper 5 controls the EMG-A signal of the TB1 connector. The following diagram is shown the selection condition of the jumper 5.



#### ■ JP1, JP2

The encoder signals of axis X and axis Y can be chosen from servo driver encoder or external encoder. Fig. 3-11 shows that the encoder signals are selected from servo driver encoder. In meantime, Fig. 3-12 shows that the encoder signals are selected from external encoder.



Fig. 3-11 Primary encoder signals setting



Fig. 3-12 External encoder signals setting

The emergency stop signal for each servo ampilfier can be selected from EMG SW. The number 1, 2, 3, 4 on EMG SW are denoted as axis X, Y, Z, U, respectively. Fig. 3-13 is the default setting to connect the EMG singals to GND. The EMG signals from CN1 ~ CN4 will not take effect. If the switch is disconnected as shown in Fig. 3-14, the emergency stop signals can be controlled from EMG signals in CN1 ~ CN4.



Fig. 3-13 EMG SW setting for normally GND (Default setting)



Fig. 3-14 EMG SW setting for user controlled signals.
## **B.4 DN-8468D Daughter Board**

The DN-8468D is the daughter board for Delta ASDA-A Series Ampilifier. It has 4-axis I/O signals.

# **B4.1 Board Layout for DN-8468D**



Fig. 3-1 Board layout for the DN-8468D

## **B4.2 Signal Connections for DN-8468D**

Maintaining signal connections is one of the most important factors in ensuring that your application system is sending and receiving data correctly.

#### Pin Assignment for CON1

The I/O connector on the DN-8468D is a 68-pin SCSI II connector that enables you to connect to the I-8094 motion card. Fig. 3-2 shows the pin assignment for the 68-pin I/O connector on the DN-8468D (or on the I-8094), and refer to Table 3-2, 3-3 for description of each motion I/O signal.



Fig. 3-2 I/O connector pin assignment for the CON1

Pin name	Pin number	Description
XECA	1	Encoder A-phase signal for X axis
YECA	36	Encoder A-phase signal for Y axis
ZECA	33	Encoder A-phase signal for Z axis
UECA	68	Encoder A-phase signal for U axis
XECB	2	Encoder B-Phase signal for X axis
YECB	37	Encoder B-Phase signal for Y axis
ZECB	32	Encoder B-Phase signal for Z axis
UECB	67	Encoder B-Phase signal for U axis
XINPOS	3	In-position signal for X axis
YINPOS	38	In-position signal for Y axis
ZINPOS	31	In-position signal for Z axis
UINPOS	66	In-position signal for U axis
XALARM	4	Alarm signal for X axis
YALARM	39	Alarm signal for Y axis
ZALARM	30	Alarm signal for Z axis
UALARM	65	Alarm signal for U axis
XLMTP	5	Limit switch input signal (+) for X axis
YLMTP	40	Limit switch input signal (+) for Y axis
ZLMTP	29	Limit switch input signal (+) for Z axis
ULMTP	64	Limit switch input signal (+) for U axis
XLMTM	6	Limit switch input signal (-) for X axis
YLMTM	41	Limit switch input signal (-) for Y axis
ZLMTM	28	Limit switch input signal (-) for Z axis
ULMTM	63	Limit switch input signal (-) for U axis
XIN3	7	Input 3 signal for X axis
YIN3	42	Input 3 signal for Y axis
ZIN3	27	Input 3 signal for Z axis
UIN3	62	Input 3 signal for U axis
XIN2	8	Input 2 signal for X axis
XIN2	43	Input 2 signal for Y axis
XIN2	26	Input 2 signal for Z axis
XIN2	61	Input 2 signal for U axis
XIN1	9	Input 1 signal for X axis
YIN1	44	Input 1 signal for Y axis
ZIN1	25	Input 1 signal for Z axis
UIN1	60	Input 1 signal for U axis
XIN0	10	Input 0 signal for X axis
YIN0	45	Input 0 signal for Y axis
ZIN0	24	Input 0 signal for Z axis
UIN0	59	Input 0 signal for U axis

Table 3-2 DN-8468D I/O connector signal description (part 1)

### Table 3-3 DN-8468D I/O connector signal description (part 2)

Pin name	Pin number	Description
XEXPP	11	EXT pulsar input signal (+) for X axis
YEXPP	46	EXT pulsar input signal (+) for Y axis
ZEXPP	23	EXT pulsar input signal (+) for Z axis
UEXPP	58	EXT pulsar input signal (+) for U axis
XEXPM	12	EXT pulsar input signal (-) for X axis
YEXPM	47	EXT pulsar input signal (-) for Y axis
ZEXPM	22	EXT pulsar input signal (-) for Z axis
UEXPM	57	EXT pulsar input signal (-) for U axis
XDRIVE	13	Driver enable signal for X axis
YDRIVE	48	Driver enable signal for Y axis
ZDRIVE	21	Driver enable signal for Z axis
UDRIVE	56	Driver enable signal for U axis
XPP	14	Driving pulsar signal (+) for X axis
YPP	49	Driving pulsar signal (+) for Y axis
ZPP	20	Driving pulsar signal (+) for Z axis
UPP	55	Driving pulsar signal (+) for U axis
XPM	15	Driving pulsar signal (+) for X axis
YPM	50	Driving pulsar signal (+) for Y axis
ZPM	19	Driving pulsar signal (+) for Z axis
UPM	54	Driving pulsar signal (+) for U axis
XOUT1	16	Output 1 signal for X axis
YOUT1	48	Output 1 signal for Y axis
ZOUT1	21	Output 1 signal for Z axis
UOUT1	56	Output 1 signal for U axis
EXPLSN1	17	EXT pulse input signal for interpolation
EMGN1	52	Emergency stop input signal
FRnetA	16	FRnet port A
FRnetB	18	FRnet port B
XDCC	51	Deviation Counter Clear for X axis
YDCC	53	Deviation Counter Clear for Y axis
GND	34	Ground
VCC	35	External power (12~24V)

The connector TB1 is 7-pin connector that enables you to connect to the signals of your motor drivers. Fig.3-3 shows the pin assignment for the 7-pin connector on the DN-8468D, and the Table 3-4 shows its I/O connector signal description.



Fig. 3-3 Pin definition for TB1

<b>0</b>		
Name	Description	
FR-A	FRnet port A	
FR-B	FRnet port B	
X-DCC	Deviation Counter Clear for X axis	
Y-DCC	Deviation Counter Clear for Y axis	
E-PLS	EXT pulse signal	
EMG-A	EMG input signal for all axes	
E-GND	EXT power ground	

### Table 3-4 TB1 Signal Connection

### ■ TB2

The connector TB2 is 5-pin connector that enables you to connect to the signals of your motor drivers. Fig.3-4 shows the pin assignment for the 5-pin connector on the DN-8468D, and the Table 3-5 shows its I/O connector signal description.



Table 3-5 TB2 Signal Connection		
Pin name	Description	
E-PWR	EXT power supply +24V	
E-GND	EXT power ground	
FGND	Frame ground	

A A:

► Note: Don't reverse connect signals with E-PWR and E-GND. Serious damage to your motion card and motion controller might be happened.

The connectors CNX, CNY, CNZ, and CNU are 50-pin connectors that enable you to connect to the CN1 connector of Delta ASDA-A series motor drivers. Fig.3-5 shows the pin assignment for the 50-pin connector on the DN-8468D, and the Table 3-6 shows its I/O connector signal description.



Fig. 3-5 Pin definition for CNX, CNY, CNZ, CNU

N	Number	Description
Name	Number	Description
A+	21	Encoder A-Phase (+)
A-	22	Encoder A-Phase (-)
B+	25	Encoder B-Phase (+)
B-	23	Encoder B-Phase (-)
Z+	50	Encoder Z-Phase (+)
Z-	24	Encoder Z-Phase (-)
P+	41	Positive Direction Pulse Output(+)
P-	43	Positive Direction Pulse Output(-)
N+	37	Negative Direction Pulse Output(+)
N-	36	Negative Direction Pulse Output(-)
INP	1	Servo In Position
RDY	7	Servo Ready
SVON	9	Servo On
ALM-RST	33	Alarm Reset
CCLR	10	Error Counter Clear
ALARM	28	Servo Alarm
EMG	30	Emergent Stop
E-PWR	11	EXT power +24V
E-GND	2,4,6,12, 13,19,26, 27,31,32, 44,45,47, 49	EXT power ground
NC	3,5,8,14, 15,16,17, 18,20,29, 34,35,38 39,40,42, 46 48	No connection

Table 3-6 CN 1 Signal Connection

- Note 1: There are two sets encoder signals for X and Y axes. In X axis, one is from CNX and the other is from CN5. In Y axis, one is from CNY and the other is from CN6. Users can select encoder signals from JP1 and JP2, respectively.
- Note 2: In Z and U axes, only one set of encoder signals is used for each axis. In Z axis, do not connect CNZ and CN7 at the same time. In U axis, do not connect CNU and CN8 at the same time.
- Note 3 : Don't connect NC (not connected) signals. Connecting these signals could cause permanent damage to your motion controller.

The connectors CN1~CN4 are 11-pin connectors that enable you to connect to the signals of your motor drivers. Fig.3-7 shows the pin assignment for the 20-pin connector on the DN-8468D, and the Table 3-8 shows its I/O connector signal description.



Pin name	Description
E-PWR	EXT power supply +24V
EMG	EMG input signal
LMT+	Limit Switch Input Signal (+)
LMT-	Limit Switch Input Signal (-)
INPUT3	Input Signal (IN3)
NRHOME	Near Home Sensor Input Signal
HOME	Home Sensor Input Signal
RESET	Reset input signal
EXP+	EXT Positive Direction Pulse (+)
EXP-	EXT Negative Direction Pulse (-)
E-GND	EXT power ground

Table 3-8 CN1~4 Signal Connection

The connectors CN5~CN8 are 15-pin connectors that enable users to connect the signals to external motor drivers. Fig.3-8 shows the pin assignment for the 15-pin connector on the DN-8468D, and the Table 3-9 shows its I/O connector signal description.



Fig. 3-8 Pin definition for CN5 ~ CN8

Name	No.	Description
A+	9	Encoder A-Phase (+)
A-	4	Encoder A-Phase (-)
B+	10	Encoder B-Phase (+)
B-	5	Encoder B-Phase (-)
Z+	11	Encoder Z-Phase (+)
Z-	12	Encoder Z-Phase (-)
P+	1	Positive Direction Pulse Output(+)
P-	7	Positive Direction Pulse Output(-)
N+	2	Negative Direction Pulse Output(+)
N-	8	Negative Direction Pulse Output(-)
E-PWR	15	EXT power +24V
E-GND	3	EXT power ground
EXT-5V	6	EXT power +5V
NC	13, 14	No connection

Table 3-9 CN5~8

- Note 1: There are two sets encoder signals for X and Y axes. In X axis, one is from CNX and the other is from CN5. In Y axis, one is from CNY and the other is from CN6. Users can select encoder signals from JP1 and JP2, respectively.
- Note 2: In Z and U axes, only one set of encoder signals is used for each axis. In Z axis, do not connect CNZ and CN7 at the same time. In U axis, do not connect CNU and CN8 at the same time.
- Note 3 : Don't connect NC (not connected) signals. Connecting these signals could cause permanent damage to your motion controller.

The connectors RJ1 is an 8-pin RJ45 connector that enable you to connect to the signals of FRnet. Fig.3-9 shows the pin assignment for the 8-pin connector on the DN-8468D, and the Table 3-10 shows its I/O connector signal description.



Table 3-10 RJ1		
Pin name	Description	
FRnetA	FRnet port A	
FRnetB	FRnet port B	
NC	No connection	

Fig. 3-9 Pin definition for RJ1

Note: Don't connect NC (not connected) signals. Connecting these signals could cause permanent damage to your motion controller.

## **B4.3 Jumper and Switch Settings**

#### ■ JP5

Jumper 5 controls the EMG-A signal of the TB1 connector. The following diagram is shown the selection condition of the jumper 5.



#### ■ JP1, JP2

The encoder signals of axis X and axis Y can be chosen from servo driver encoder or external encoder. Fig. 3-11 shows that the encoder signals are selected from servo driver encoder. In meantime, Fig. 3-12 shows that the encoder signals are selected from external encoder.







Fig. 3-12 External encoder signals setting

#### SW1

The emergency stop signal for each servo ampilfier can be selected from SW1. The number 1, 2, 3, 4 on SW1 are denoted as axis X, Y, Z, U, respectively. Fig. 3-13 is the default setting to connect the EMG singals to GND. The EMG signals from CN1 ~ CN4 will not take effect. If the switch is disconnected as shown in Fig. 3-14, the emergency stop signals can be controlled from EMG signals in CN1 ~ CN4.



Fig. 3-13 SW1 setting for normally GND (Default setting)



Fig. 3-14 SW1 setting for user controlled signals.

#### ■ JP10 ~ JP13

Jumper 10 ~ Jumper 13 can select the reset function in CN1 ~ CN4 for each axis. The following diagram is shown the selection condition of the JP10.



Fig. 3-15 JP 10 ~ 13 setting