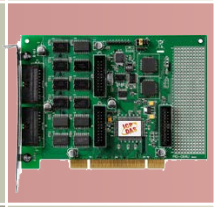
PIO-D64U/PEX-D64
User Manual

64-channel DIO board Version 1.7, Oct. 2019



SUPPORTS

Board includes PIO-D64U and PEX-D64.

WARRANTY

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

WARNING

ICP DAS assumes no liability for damages consequent to the use of this product. ICP DAS reserves the right to change this manual at any time without notice. The information furnished by ICP DAS is believed to be accurate and reliable. However, no responsibility is assumed by ICP DAS for its use, nor for any infringements of patents or other rights of third parties resulting from its use.

COPYRIGHT

Copyright © 2019 by ICP DAS. All rights are reserved.

TRADEMARK

Names are used for identification only and may be registered trademarks of their respective companies.

CONTACT US

If you have any question, please feel to contact us. We will give you quick response within 2 workdays.

Email: service@icpdas.com, service.icpdas@gmail.com

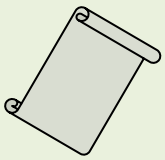
TABLE OF CONTENTS

Packing List

One PIO-D64U or PEX-D64 card as follows:

PIO-D64U

PEX-D64

One printed Quick Start Guide 

Note:

If any of these items is missing or damaged, contact the dealer from whom you purchased the product. Save the shipping materials and carton in case you need to ship or store the product in the future.

1 Introduction

The PIO-D64U supports 3.3 V/5 V PCI bus, while the PEX-D64 supports PCI Express bus. These cards provide 32-ch digital input and 32-ch digital output that consist of two 16-bit input ports and two 16-bit output ports. Those cards also offer 6-ch counter/timer with four-clock sources, 2 MHz, 1 MHz, 500 kHz and 250 kHz. The user can use the clock source from the soldering pad. 3 of the 6-ch timer/ counter are for general purposes such as frequency measurement, event counting and pulse generation; the other 3 channels are for interrupt function.

The PIO-D64U and PEX-D64 also adds a Card ID switch. Users can set Card ID and recognize the board by the ID via software when using two or more PIO-D64U/PEX-D64 cards in one computer.

These cards support various OS versions, such as DOS, Linux and 32/64-bit Windows 10/8/7/2008/2003/XP. DLL and Active X control together with various language sample programs based on Turbo C++, Borland C++, Microsoft C++, Visual C++, Borland Delphi, Borland C++ Builder, Visual Basic, C#.NET, Visual Basic.NET and LabVIEW are provided in order to help users quickly and easily develop their own applications.

1.1 Features

- Support the +3.3/+5 V PCI bus for PIO-D64U
- Support the +5 V PCI bus for PIO-D64
- Support the PCI Express x1 for PEX-D64
- 32 digital input channels (strobe control selectable) and 32 digital output channels
- Four independent programmable 16-bit timers/counters
- One 32-bit timer with a 4 MHz clock base
- Provide clock source: 2 MHz, 1 MHz, 500 KHz, 250 KHz
- Interrupt source: 3 channels
- Five 20-pin flat cable connectors
- Connect directly to DB-24PR, 24POR, DB-24C, DB-16P, DB-16R;

1.2 Specifications

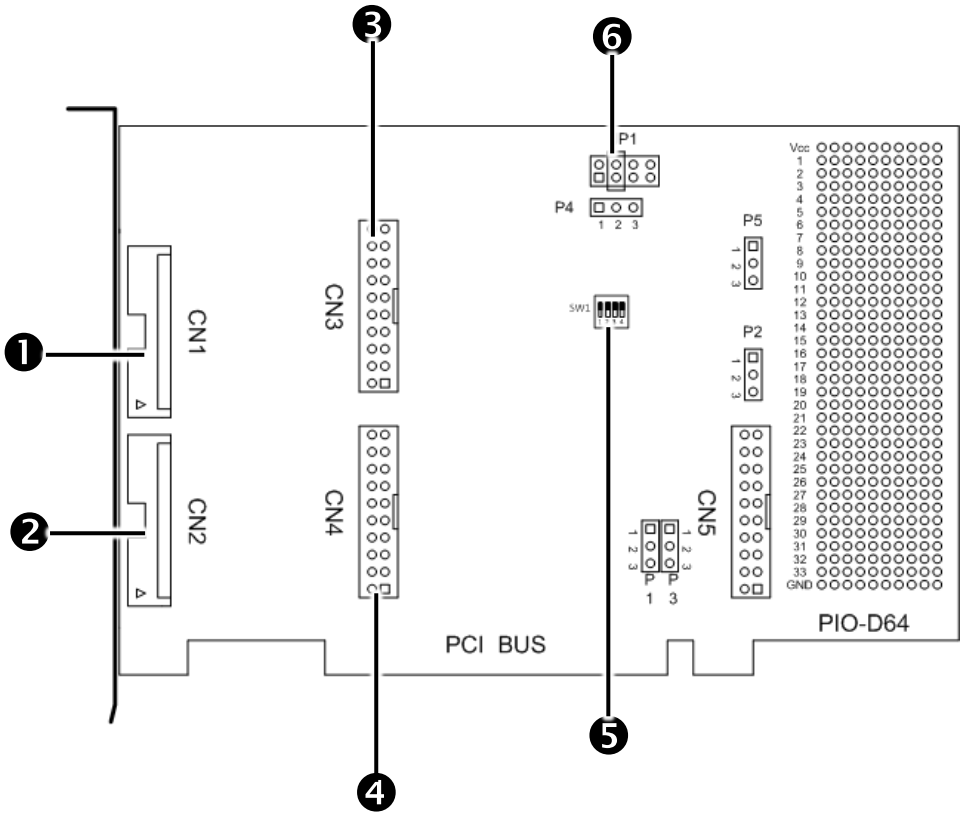
Model Name	PIO-D64	PIO-D64U	PEX-D64
Digital Input			
Channels	32		
Compatibility	5 V/TTL		
Input Voltage	Logic 0: 0.8 V max. Logic 1: 2.0 V min.		
Response Speed	1 MHz(Typical)		500 kHz
Digital Output			
Channels	32		
Compatibility	5 V/TTL		
Output Voltage	Logic 0: 0.4 V max. Logic 1: 2.4 V min.		
Output Capability	Sink: 24 mA @ 0.8 V Source: 15 mA @ 2.0 V		
Response Speed	1 MHz		500 kHz
Timer/Counter			
Channels	6 (Independent x 3/EVTIRQ x 1/TMRIRQ x 1/EXTIRQ x 1)		
Resolution	16-bit		
Compatibility	5 V/TTL		
Input Frequency	10 MHz max.		
Reference Clock	Internal: 4 MHz		
General			
Bus Type	5 V PCI, 32-bit, 33 MHz	3.3 V/5 V Universal PCI, 32-bit, 33 MHz	PCI Express x1
Data Bus	8-bit		
Card ID	No	Yes(4-bit)	
I/O Connector	20-pin box header x 5		
Dimensions (L x W)	156 mm x 110 mm		156 mm x 112 mm
Power Consumption	580 mA @ +5 V		200 mA @ +3.3 V 180 mA @ +12 V
Operating Temperature	0 ~ 60 °C		
Storage Temperature	-20 ~ 70 °C		
Humidity	5 ~ 85% RH, non-condensing		

Note:

The I/O speed is depending on I/O card, bus speed, CPU speed and system loading. Any condition changes may cause the I/O speed different.

2 Hardware Configuration

2.1 Board Layout



❶ CN1 Connector	Refer to Section 2.2 “I/O Port Location” for more details.
❷ CN2 Connector	
❸ CN3 Connector	
❹ CN4 Connector	
❺ Card ID Switch	Refer to Section 2.7 “Card ID Switch for more details.
❻Clock Source Jumper	Refer to Section 2.5.1 “Clock Source” for more details.

2.2 I/O Port Location

Connector	Description	
CN1	Digital Output Channel 0 to 15	For more detailed information related to the pin assignments for the CN1~4, refer to Section 2.3 "Pin Assignments".
CN2	Digital Input Channel 0 to 15	
CN3	Digital Output Channel 16 to 31	
CN4	Digital Input Channel 15 to 31	

Besides, there is also a connector interface (CN5) for timer and counter function, as shown in Figure 2.1.

Note:

This board is a bi-directional I/O design with default DI mode when power on.

Before switching to DO mode, the DI pull-high jumper setting may activate active-high DO devices (e.g., 24POR / 24C), or the pull-low setting may activate active-low DO devices.
Please have an appropriate jumper setting depending on the characteristics of your external device.

2.3 Pin Assignments

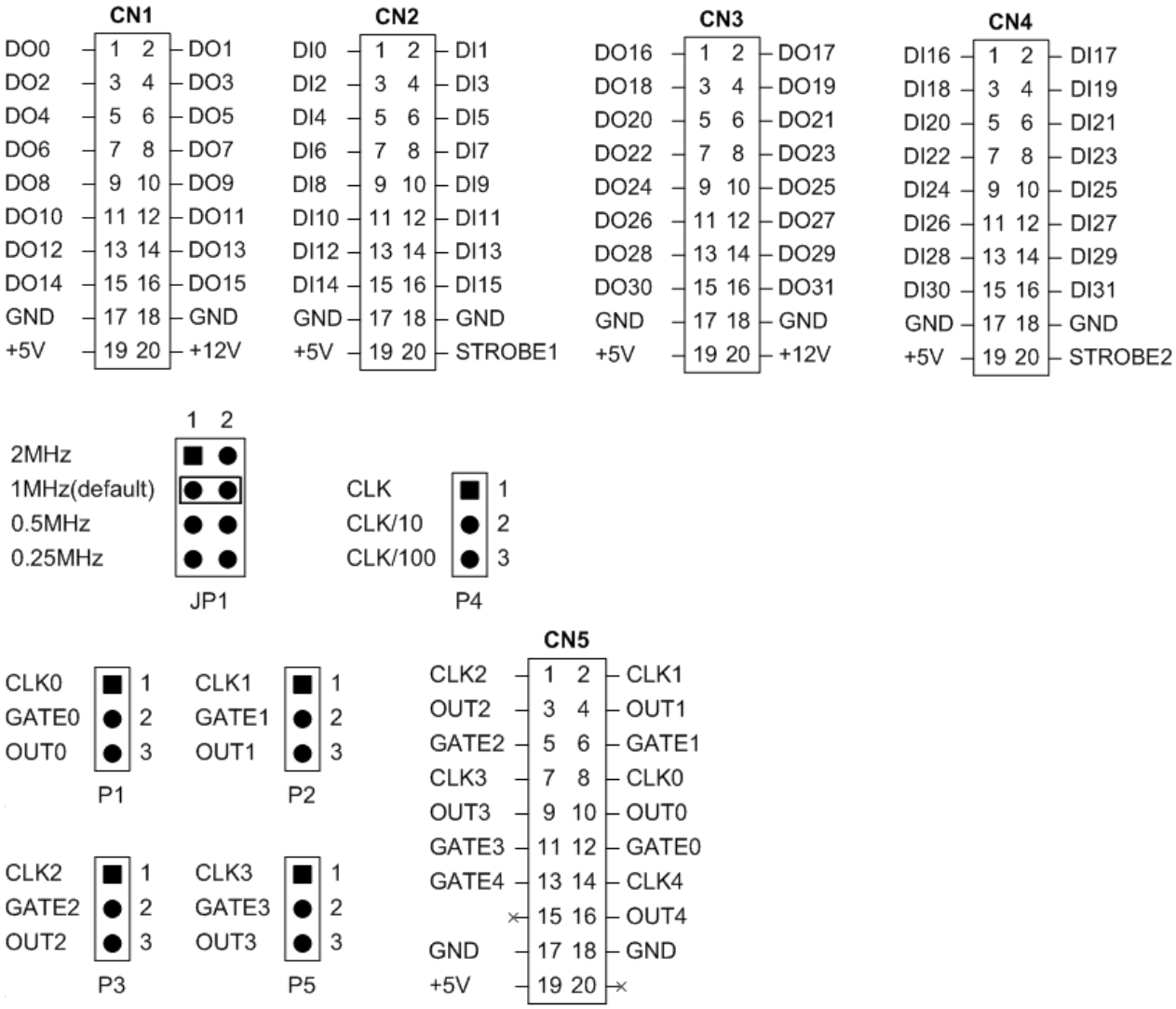


Figure 2.1

2.4 Digital I/O Operating

2.4.1 DO Port Architecture (CN1&CN3)

When the PC is power-up, all of DO states are clear to low-state by the RESET\ signal. For more information about RESET\ signal, please refer to Sec. 6.1.1. Note that the RESET\ is in Low-state in order to clear all DO states to low level signal. The detail block diagram of DO function is represented as Figure 2.2.

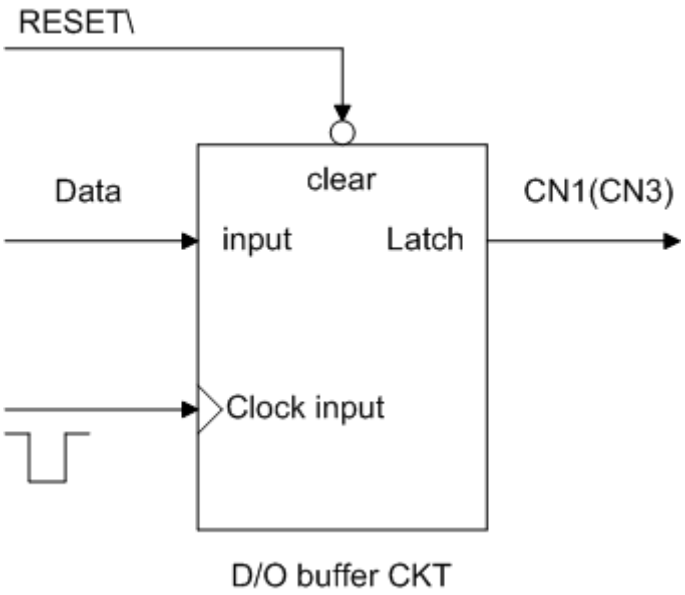


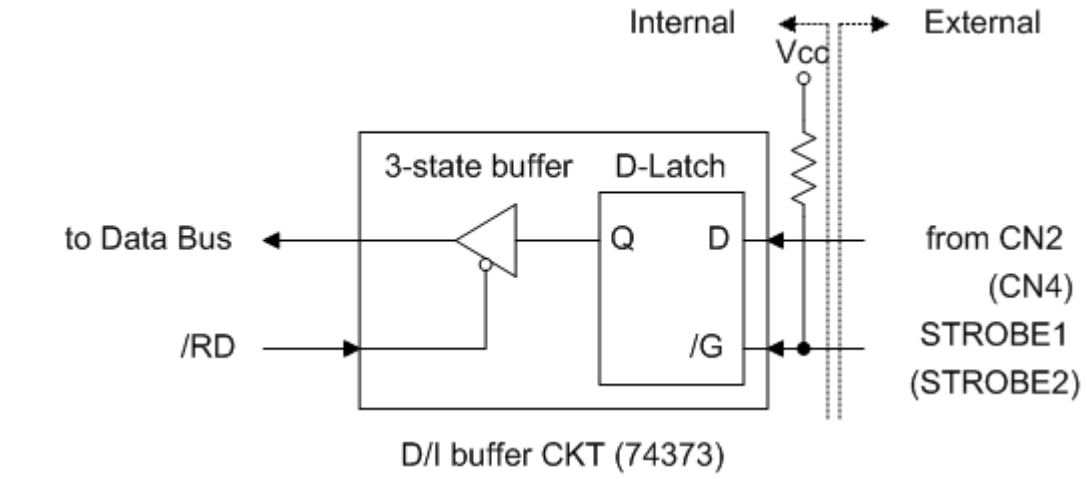
Figure 2.2: Block diagram of DO function

2.4.2 DI Port Architecture (CN2&CN4)

The enable/disable of DI port is controlled by the RESET\ signal, as depicted as below:

- **The RESET\ is in Low-state → all DI operation is disable**
- **The RESET\ is in High-state → all DI operation is enable**

Note that when the PC is power-up, all operation of DI port is disabled because RESET\ is in low level. Besides, user may need to latch input data by external strobe single in some application. We provide the following architecture, as shown in Figure 2.3, to allow user to apply the STROBE pin to latch D/I input signal. If no signal is connected to strobe pin, the input data is transparent.



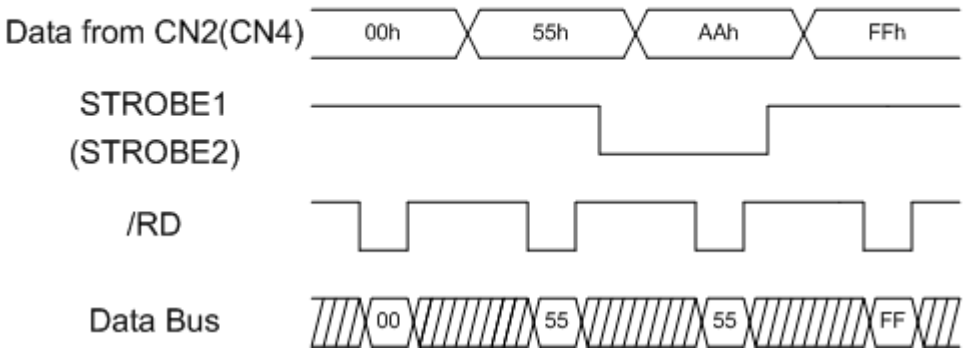
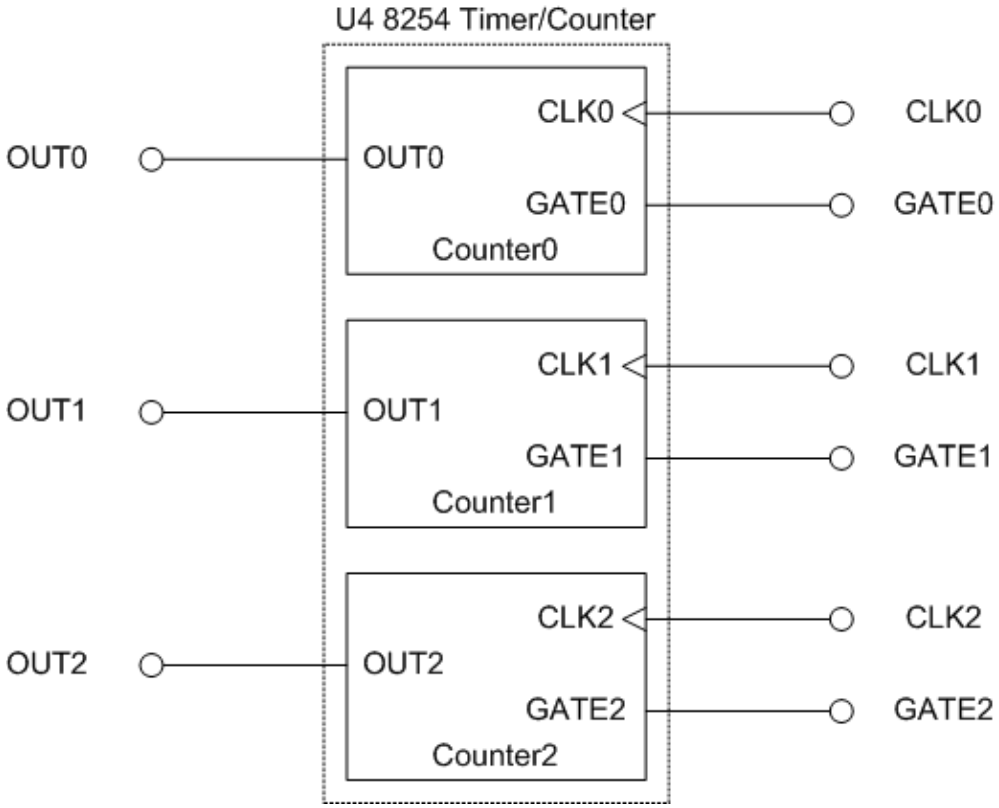


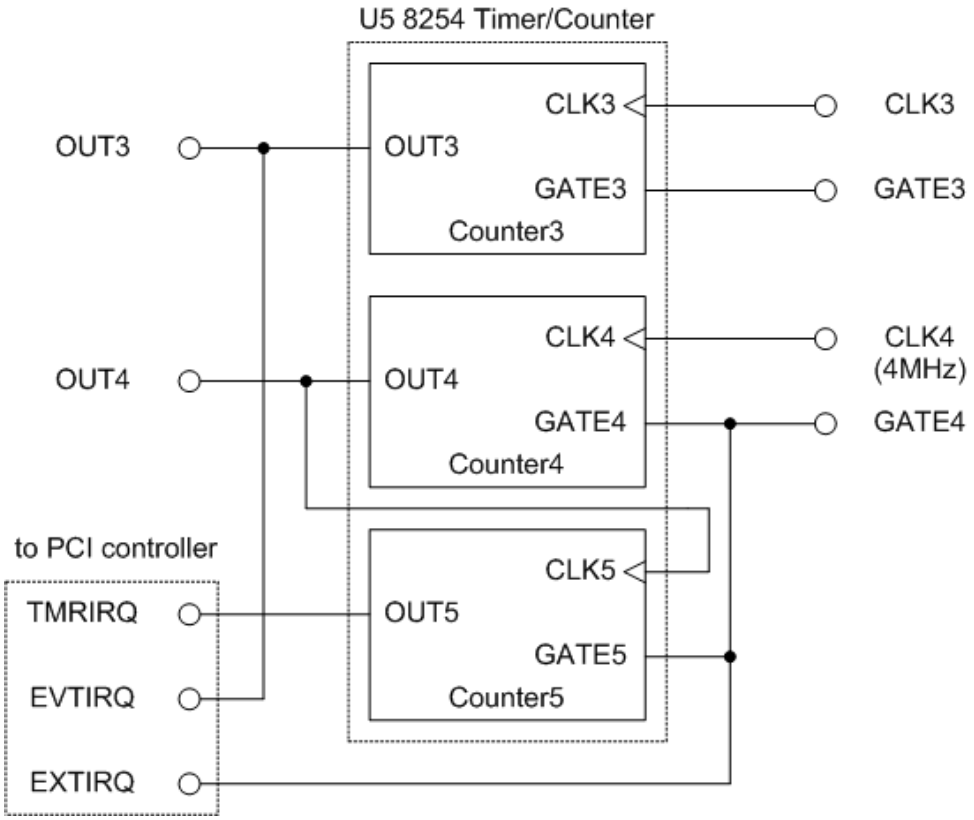
Figure 2.3

2.5 Timer/Counter Operation

PIO-D64/PIO-D64U has two timer/counter chips, 8254. The first 8254 chip is used as general purpose timer/counter, as shown in Figure 2.4.The pin assignment is presented in Section 2.3 “Pin Assignments”.



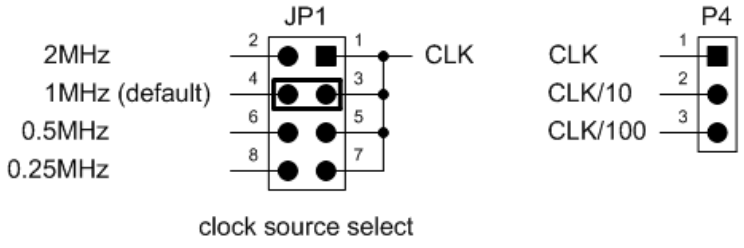
The second 8254 chip is used to generate interrupt trigger signals, as shown in Figure 2.5. The Counter3 accept event signal and will generate trigger signal of the interrupt. And the Counter4 and Counter5 are cascaded together, which has clock source 4 MHz. It is used to generate pacer timer trigger of the interrupt.



Note: Refer to Section 2.3 "Pin Assignments" for more information about pin assignment.
Refer to Section 2.6 "Interrupt Operation" for more information about operation of interrupt.

2.5.1 Clock Source

The PIO-D64 series provides wide range clock source as below table. By jumper setting of JP1, user can select suitable clock output from the corresponding P4 soldering pad.



JP1 setting	P4 soldering pad clock output		
	P4.1	P4.2	P4.3
1-2	2MHz	200KHz	20KHz
3-4 (default)	1MHz	100KHz	10KHz
5-6	500KHz	50KHz	5KHz
7-8	250KHz	25KHz	2.5KHz

2.6 Interrupt Operation

There are three interrupt sources in Card. These three signals are named as **INT_CHAN_0, INT_CHAN_1 and INT_CHAN_2**. Their signal sources are given as follows: (Refer to Sec. 2.5 for the source of interrupt signal)

INT_CHAN_0: EXTIRQ
INT_CHAN_1: EVTIRQ
INT_CHAN_2: TMRIRQ

If only one interrupt signal source is used, the interrupt service routine does not have to identify the interrupt source. Refer to DEMO3.C, DEMO4.C and DEMO5.C of DOS operating system for more information.

If there are more than one interrupt source, the interrupt service routine has to identify the active signals as follows: (refer to DEMO6.C of DOS operation system)

Read the new status of all interrupt signal sources (refer to Sec 6.3.3)
Compare the new status with the old status to identify the active signals
If INT_CHAN_0 is active, service it
If INT_CHAN_1 is active, service it
If INT_CHAN_2 is active, service it
Update interrupt status

Note that if the interrupt signal is too short, the new status may be as same as old status. In that condition the interrupt service routine cannot identify which interrupt source is active. So the interrupt signal must be hold_active long enough until the interrupt service routine is executed. This hold_time is different for different operating system. The hold_time can be as short as micro-second or as long as second. In general, 20 ms is enough for all operating system.

2.6.1 Interrupt Block Diagram

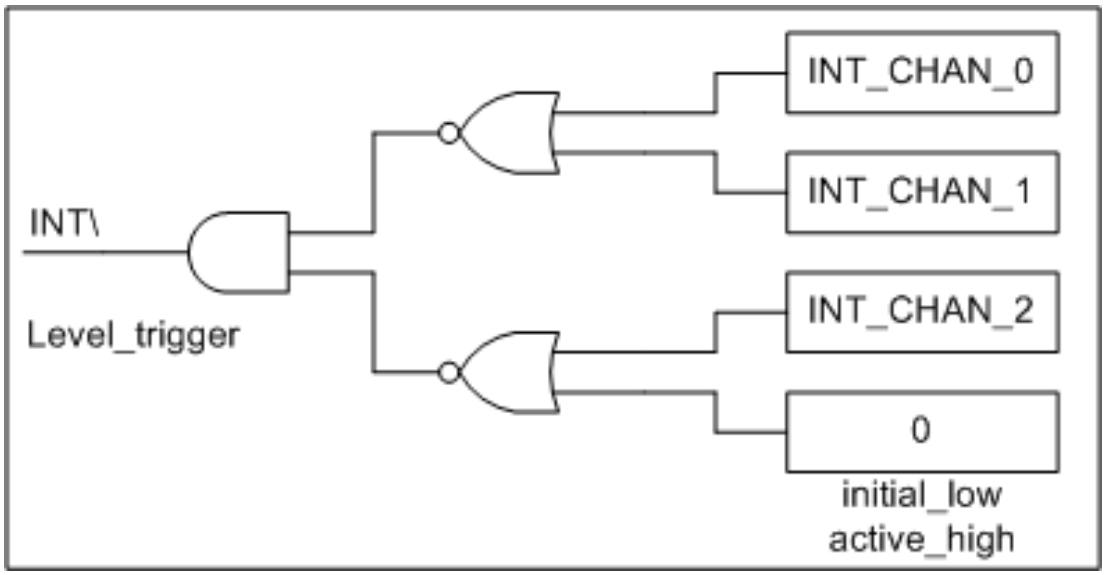


Figure 2.6

The interrupt output signal, INT\, is level-trigger & Active_Low. If the INT\ generates a low-pulse, it will interrupt the PC once a time. If the INT\ is fixed in low level, it will interrupt the PC continuously. Therefore, for the normal application, the INT_CHAN_0/1/2 must be controlled in a pulse_type signals. That is, they must be fixed in low level state normally and generate a high_pulse to interrupt the PC.

The priority of INT_CHAN_0/1/2 is the same. If all these three signals are active at the same time, then INT\ will be active only once a time. So the interrupt service routine has to read the status of all interrupt channels for multi-channel interrupt. Refer to DEMO6.C in DOS operating system for demonstrate the application under the condition of both INT_CHAN_1 and INT_CHAN_2.

If only one interrupt source is used, the interrupt service routine doesn't have to read the status of interrupt source. The demo programs, DEMO3.C, DEMO4.C and DEMO5.C in DOS operating system, are designed for single-channel interrupt application as follows:

DEMO3.C → for INT_CHAN_0 only
DEMO4.C → for INT_CHAN_1 only
DEMO5.C → for INT_CHAN_2 only

2.6.2 INT_CHAN_0

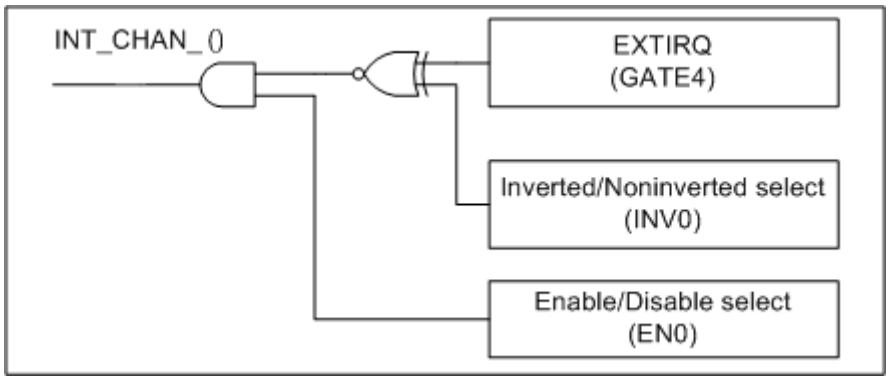


Figure 2.7

Figure 2.7 illustrates the control method of external interrupt. Note that the signal source comes from GATE4. The INV0 is used to invert/non-invert the trigger signal source and EN0 is used to disable/enable the timer interrupt (Pin13 of CN5) (Refer to Sec. 2.6 for the source of interrupt signal). The INT_CHAN_0 must be fixed in low level state normally and generated a high_pulse to interrupt the PC.

The EN0 can be used to enable/disable the INT_CHAN_0 as follows:
(refer to Figure 2.7 and Sec. 6.1.2)
EN0=0 → INT_CHAN_0=disable
EN0=1 → INT_CHAN_0=enable

The INV0 can be used to invert/non-invert the EXTIRQ as follows:
(Refer to Figure 2.7 and Sec. 6.1.4)
INV0=0 → INT_CHAN_0=inverted state of EXTIRQ
INV0=1 → INT_CHAN_0=non-inverted state of EXTIRQ

NOTE: Refer to DEMO3.C in DOS operating system for more information.

2.6.3 INT_CHAN_1

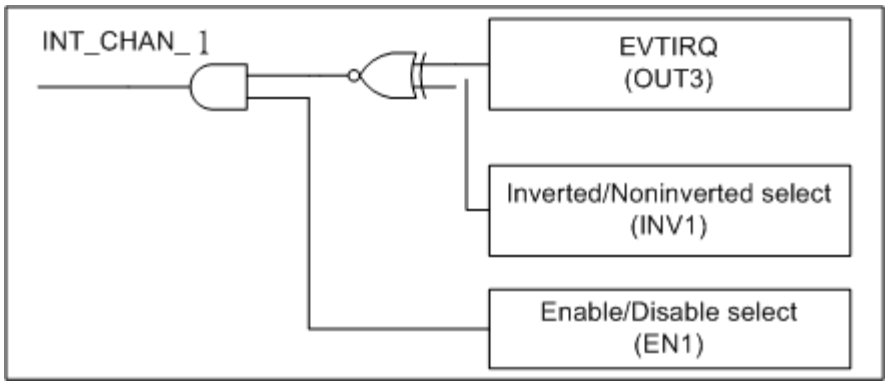


Figure 2.8

Figure 2.8 illustrates the control method of event interrupt. Note that the signal source comes from OUT3. The INV1 is used to invert/non-invert the trigger signal source and EN1 is used to disable/enable the timer interrupt (Refer to Sec. 2.6 for the source of interrupt signal). User can use Counter3 as event counter to count the event signal that comes from Pin7 of CN5. When the amount of event is the same as counter3 setting, the interrupt of INT_CHAN_1 will be trigger. The INT_CHAN_1 must be fixed in low level state normally and generated a high_pulse to interrupt the PC.

The EN1 can be used to enable/disable the INT_CHAN_1 as follows:
(refer to Figure 2.8 and Sec. 6.1.2)
EN1=0 → INT_CHAN_1=disable
EN1=1 → INT_CHAN_1=enable

The INV1 can be used to invert/non-invert the EVTIRQ as follows:
(Refer to Figure 2.8 and Sec. 6.1.4)
INV1=0 → INT_CHAN_1=inverted state of EVTIRQ
INV1=1 → INT_CHAN_1=non-inverted state of EVTIRQ

NOTE: Refer to DEMO4.C in DOS operating system for more information.

2.6.4 INT_CHAN_2

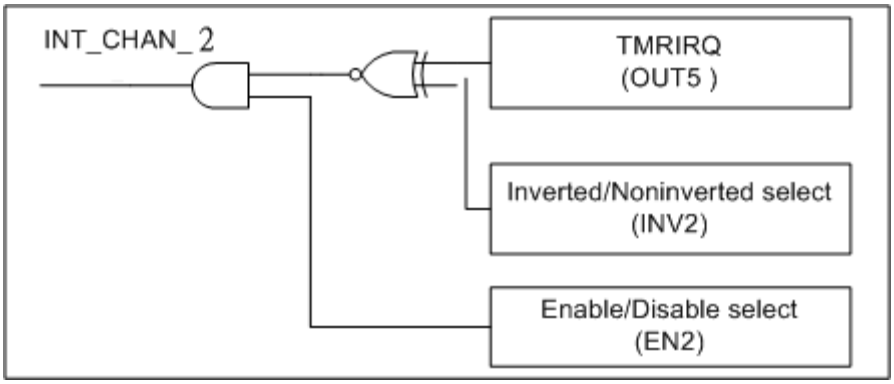


Figure 2.9

Figure 2.9 illustrates the control method of timer interrupt. Note that the signal source comes from OUT5. The INV2 can be used to invert/non-invert the Trigger signal source and EN2 is used to disable/enable the timer interrupt (Refer to Sec. 2.6 for the source of interrupt signal). Note that the INT_CHAN_2 must be fixed in low level state normally and generated a high_pulse to interrupt the PC. Because Counter4 and Counter5 are cascaded together, it can be used as 32-bit timer base on 4 MHz clock source.

The EN2 can be used to enable/disable the INT_CHAN_2 as follows:
(refer to Figure 2.9 and Sec. 6.1.2)
EN2=0 → INT_CHAN_2=disable
EN2=1 → INT_CHAN_2=enable

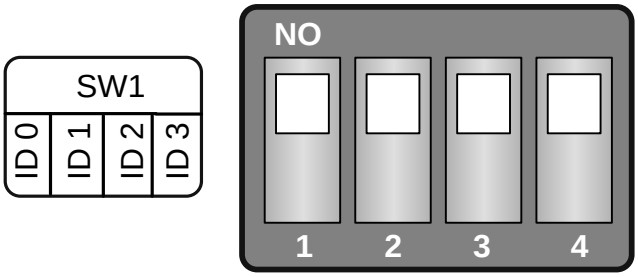
The INV2 can be used to invert/non-invert the TMRIRQ as follow2:
(Refer to Figure 2.9 and Sec. 6.1.4)
INV2=0 → INT_CHAN_2=inverted state of TMRIRQ
INV2=1 → INT_CHAN_2=non-inverted state of TMRIRQ

NOTE: Refer to DEMO5.C in DOS operating system for more information.

2.7 Card ID Switch

Before installing the card into your computer, configure the Card ID according to your requirements.

The PIO-D64U/PEX-D64 card has a Card ID switch (SW1) with which users can recognize the board by the ID via software when using two or more PIO-D64/PEX-D64 cards in one computer. The default Card ID is 0x0. For detailed information about the SW1 Card ID settings, refer to Table 2-5.



(Default Settings)

Table 2-5: Card ID Settings (SW1)

Card ID (Hex)	1 ID0	2 ID1	3 ID2	4 ID3
(*) 0x0	ON	ON	ON	ON
0x1	OFF	ON	ON	ON
0x2	ON	OFF	ON	ON
0x3	OFF	OFF	ON	ON
0x4	ON	ON	OFF	ON
0x5	OFF	ON	OFF	ON
0x6	ON	OFF	OFF	ON
0x7	OFF	OFF	OFF	ON
0x8	ON	ON	ON	OFF
0x9	OFF	ON	ON	OFF
0xA	ON	OFF	ON	OFF
0xB	OFF	OFF	ON	OFF
0xC	ON	ON	OFF	OFF
0xD	OFF	ON	OFF	OFF
0xE	ON	OFF	OFF	OFF
0xF	OFF	OFF	OFF	OFF

Note: (*) Default Settings; OFF → 1; ON → 0

3 Hardware Installation

Note:

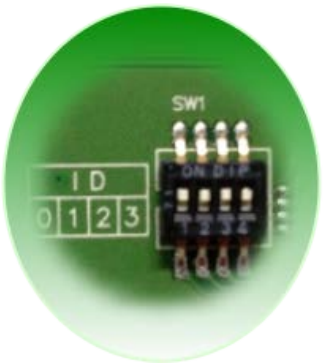
It is recommended that the driver is installed before installing the hardware as the computer may need to be restarted once the driver is installed in certain operating systems, such as Windows 2000 or Windows XP, etc. Installing the driver first helps reduce the time required for installation and restarting the computer.

To install your PIO-D64U/PEX-D64 card, follow the procedure described below:

Step 1: Install the driver for your board on Host computer.

For detailed information about the driver installation, please refer to Chapter 4 "Software Installation".

Step 2: Configure the Card ID using the DIP Switch (SW1) on PIO-D64U/PEX-D64.

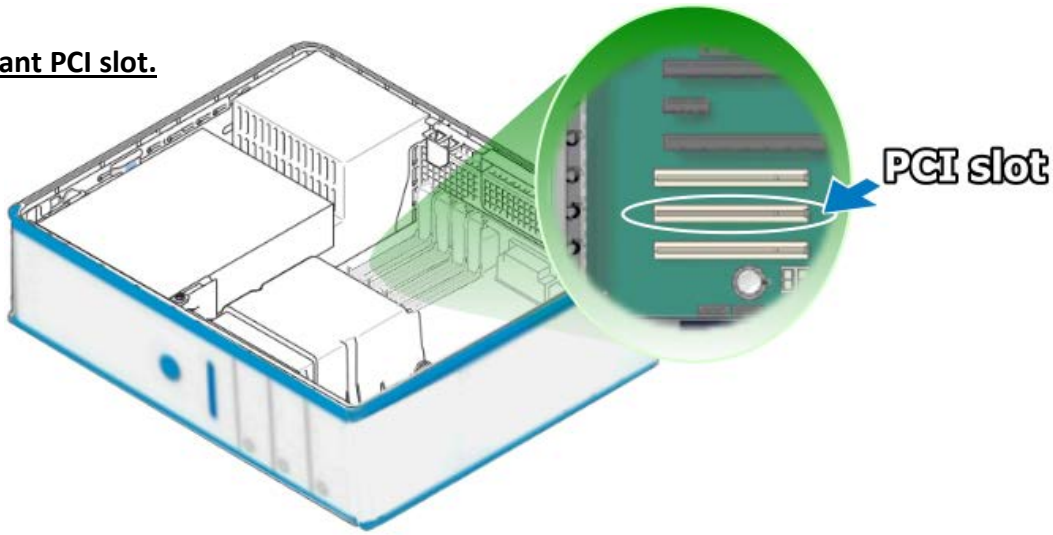


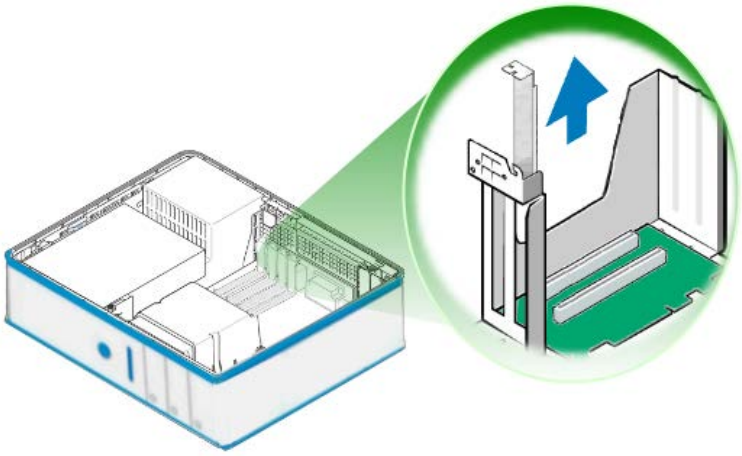
For detailed information about the card ID (SW1), please refer to Section 2.7 "Card ID Switch".

Step 3: **Shut down and switch off the power to the computer, and then disconnect the power supply.**

Step 4: **Remove the cover from the computer.**

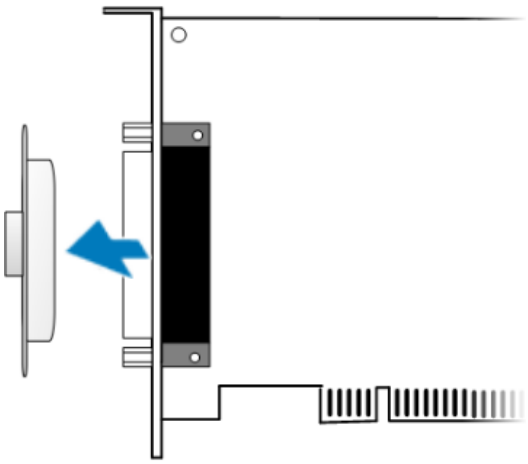
Step 5: **Select a vacant PCI slot.**



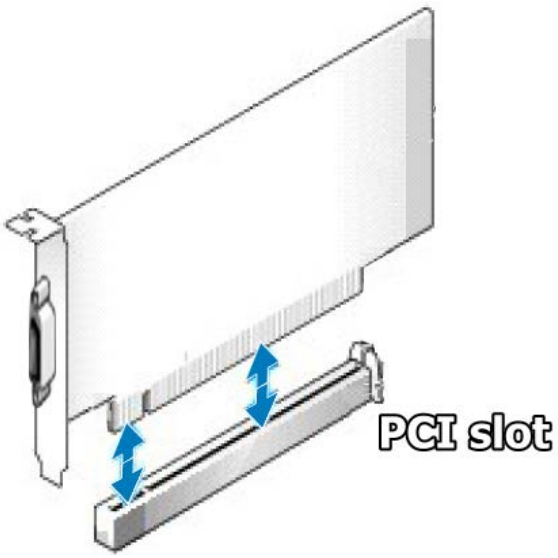


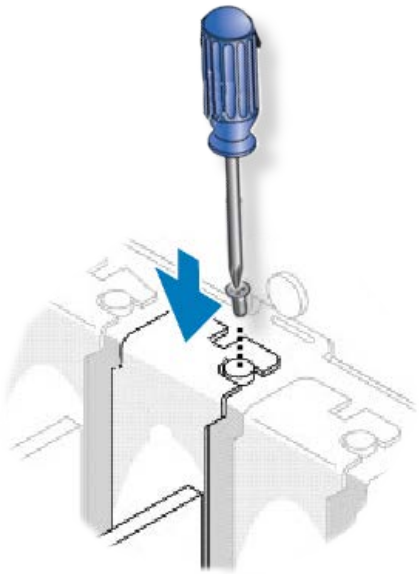
Step 6: **Unscrew and remove the PCI slot cover from the computer case.**

Step 7: **Remove the connector cover from your board.**

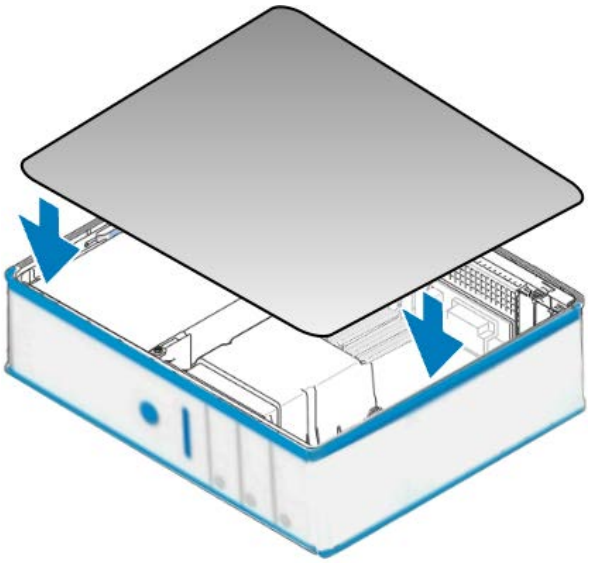


Step 8: **Carefully insert your board into the PCI or PCIe slot by gently pushing down on both sides of the board until it slides into the PCI connector.**





Step 9: **Confirm that the board is correctly inserted in the motherboard, and then secure your board in place using the retaining screw that was removed in Step 6.**



Step 10: **Replace the covers on the computer.**

Step 11: **Re-attach any cables, insert the power cord and then switch on the power to the computer.**

Once the computer reboots, follow any message prompts that may be displayed to complete the Plug and Play installation procedure. Refer to Chapter 4 "Software Installation" for more information.

4 Software Installation

This chapter provides a detailed description of the process for installing the PIO-D64 series driver and how to verify whether the PIO-D64 series cards was properly installed. PCI-D96SU/D128SU card can be used on DOS and 32/64-bit XP/2003/2008/7/8/10 based systems, and the drivers are fully Plug and Play (PnP) compliant for easy installation.

4.1 Obtaining/Installing the Driver Installer Package

The driver installation package for PIO-D64 series card can be found the ICP DAS FTP web site. Install the appropriate driver for your operating system. The location and website addresses for the installation package are indicated below.

- **UniDAQ Driver/SDK**

OS	32/64-bit Windows XP, 32/64-bit Windows 2003, 32/64-bit Windows Vista, 32/64-bit Windows 7, 32/64-bit Windows 2008, 32/64-bit Windows 8, 32/64-bit Windows 10
Driver Name	UniDAQ Driver/SDK (unidaq_win_setup_xxxx.exe)
Web Site	http://ftp.icpdas.com/pub/cd/iocard/pci/napdos/pci/unidaq/dll/driver/
Installing Procedure	To install the UniDAQ driver, follow the procedure described below. Step 1: Double-click the UniDAQ_Win_Setupxxx.exe icon to begin the installation process.

Installation Procedure	Step 2: When the “Welcome to the ICP DAS UniDAQ Driver Setup Wizard” screen is displayed, click the “Next>” button to start the installation. Step 3: On the “Information” screen, verify that the DAQ board is included in the list of supported devices, then click the “Next>” button. Step 4: On the “Select Destination Location” screen, click the “Next>” button to install the software in the default folder, C:\ICPDAS\UniDAQ. Step 5: On the “Select Components” screen, verify that the DAQ board is in the list of device, and then click the “Next>” button to continue. Step 6: On the “Select Additional Tasks” screen, click the “Next>” button to continue. Step 7: On the “Download Information” screen, click the “Next>” button to continue. Step 8: Once the installation has completed, click “No, I will restart my computer later”, and then click the “Finish” button. For more detailed information about how to install the UniDAQ driver, refer to Section 2.2 “Install UniDAQ Driver DLL” of the UniDAQ Software Manual, which can be downloaded from: http://ftp.icpdas.com/pub/cd/iocard/pci/napdos/pci/unidaq/manual/

4.2 PnP Driver Installation

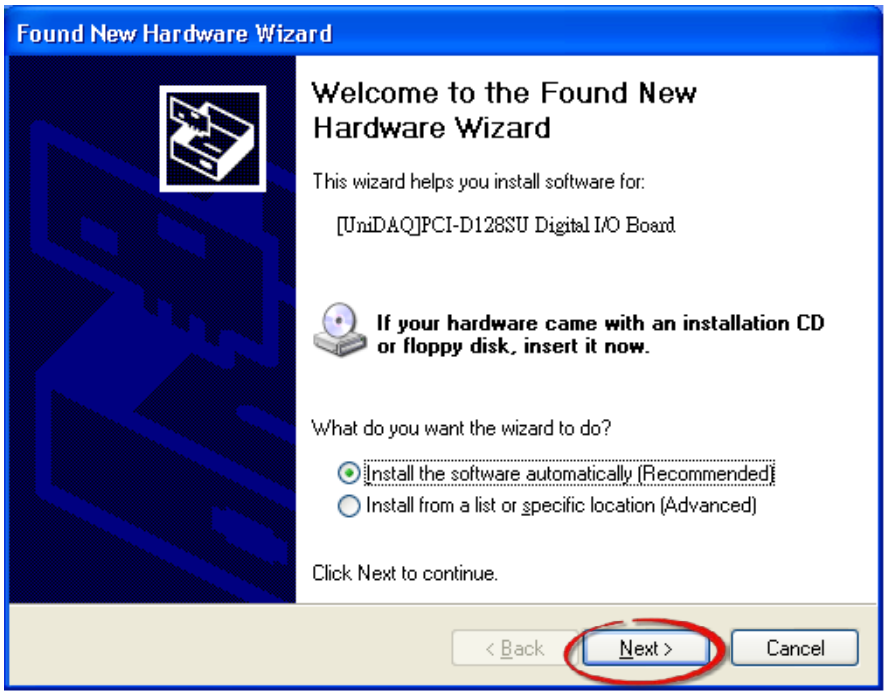
Step 1: Correctly shut down and power off your computer and disconnect the power supply, and then install your board into the computer. For detailed information about the hardware installation of PCI-D96SU/D128SU card, refer to Chapter 3 “Hardware Installation”.

Step 2: Power on the computer and complete the Plug and Play installation**.**

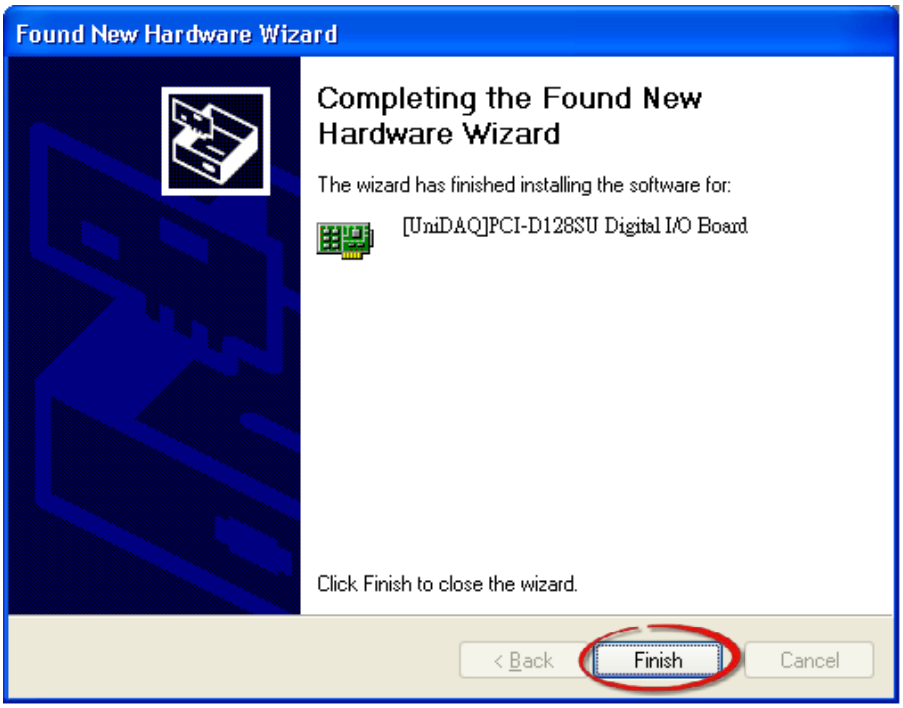
Note:

Recent operating systems, such as Windows 7/8/10 will automatically detect the new hardware and install the necessary drivers etc., so Steps 3 to 5 can be skipped.

Step 3: Select **“Install the software automatically [Recommended]”** and click the **“Next>”** button.



Step 4: Click the **"Finish"** button.



Step 5: Windows pops up **"Found New Hardware"** dialog box again.

4.3 Verifying the Installation

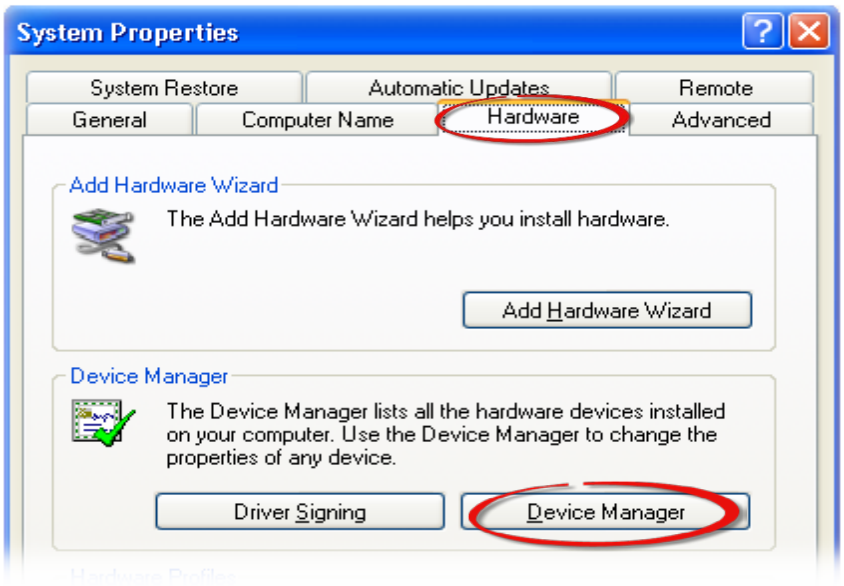
To verify that the driver was correctly installed, use the Windows **Device Manager** to view and update the device drivers installed on the computer, and to ensure that the hardware is operating correctly. The following is a description of how access the Device Manager in each of the major versions of Windows. Refer to the appropriate description for the specific operating system to verify the installation.

4.3.1 Accessing Windows Device Manager

- **Windows 2000/XP**

Step 1: Click the "**Start"** button and then point to "**Settings" and click** "**Control Panel**". Double-click the "**System**" icon to open the **"System Properties"** dialog box.

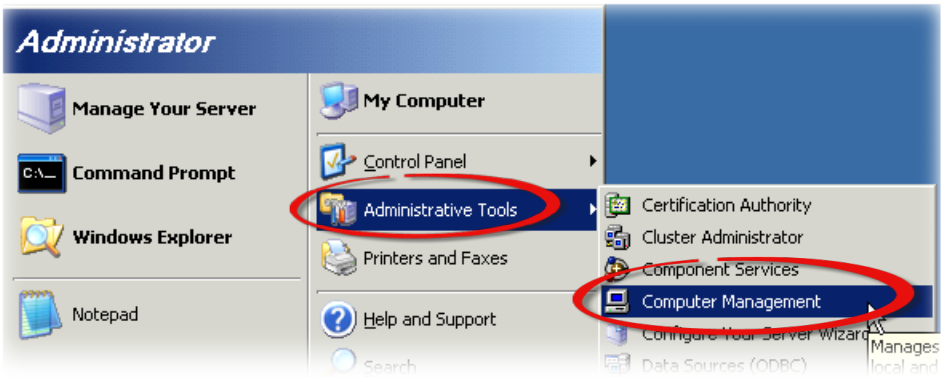
Step 2: Click the "**Hardware**" tab and then click the "**Device Manager**" button.



➢ Windows Server 2003

Step 1: Click the **"Start"** button and point to **"Administrative Tools"**, and then click the **"Computer Management"** option.

Step 2: Expand the **"System Tools"** item in the console tree, and then click **"Device Manager"**.



➢ Windows 7/10

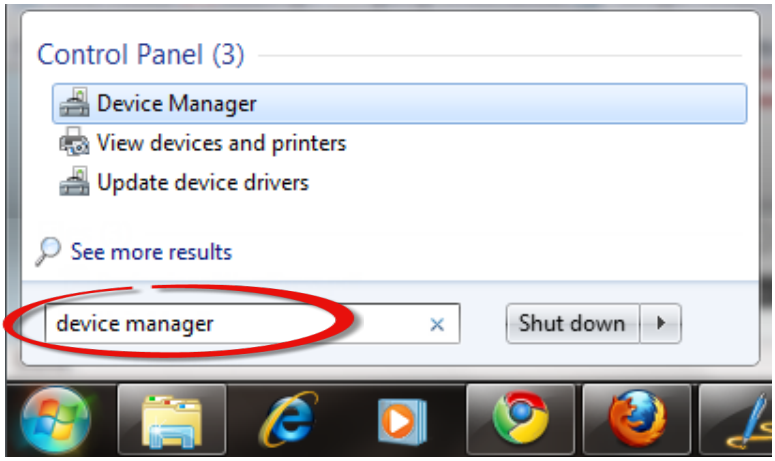
Step 1: Click the **"Start"** button, and then click **"Control Panel"**.

Step 2: Click **"System and Maintenance"**, and then click **"Device Manager"**.

Alternatively,

Step 1: Click the **"Start"** button.

Step 2: In the **Search field,** type **Device Manager** and then press Enter.



Note:

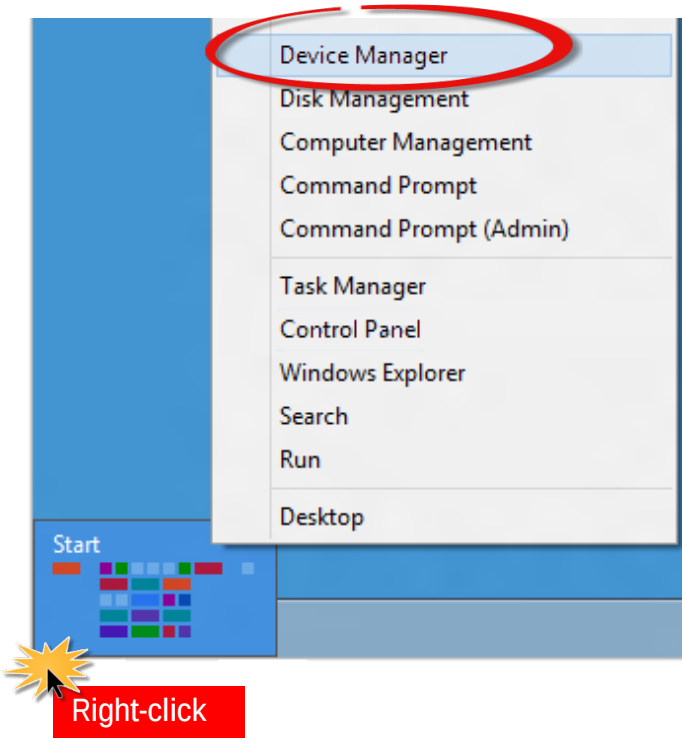
Administrator privileges are required for this operation. If you are prompted for an administrator password or confirmation, enter the password or provide confirmation by clicking the "Yes" button in the User Account Control message.

➢ Windows 8

Step 1: To display the **Start screen icon** from the desktop view, hover the mouse cursor over the **bottom-left corner** of screen.

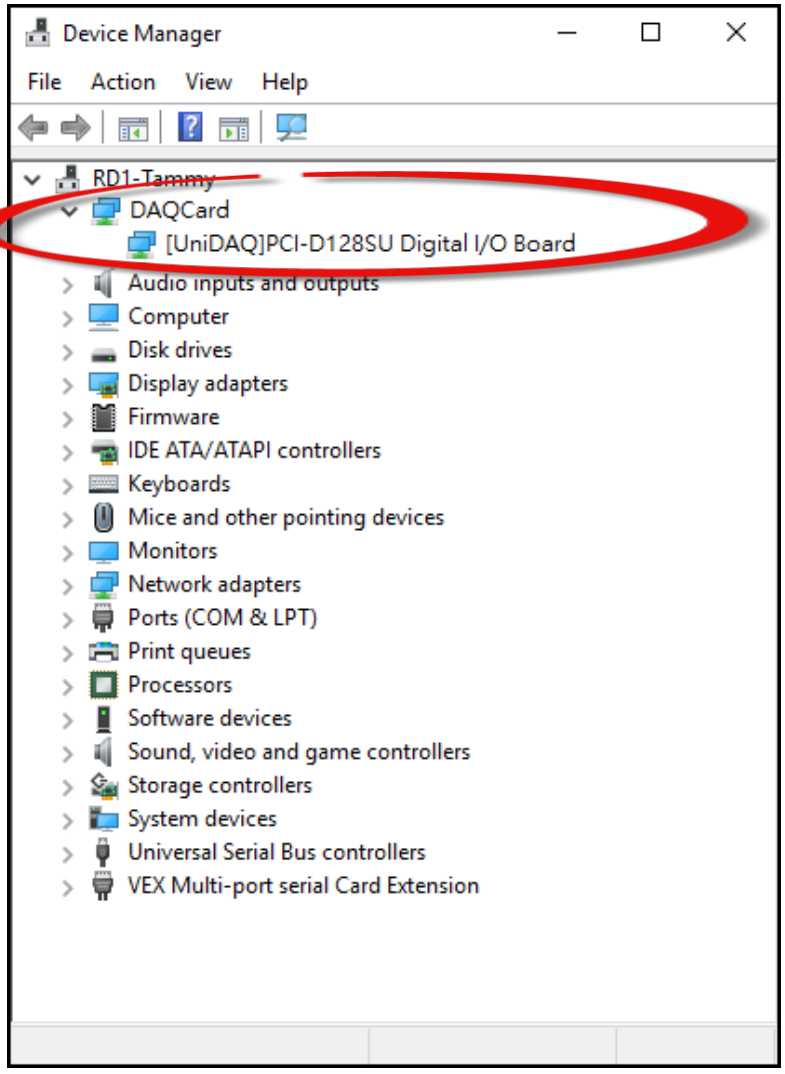
Step 2: **Right-click** the Start screen icon and then click "**Device Manager"**.

Alternatively, press **[Windows Key] +[X]** to open the Start Menu, and then select Device Manager from the options list.



4.3.2 Check that the Installation

Check that the PIO-D64U card is correctly listed in the **Device Manager** window, as illustrated below.



5. Board Testing

This chapter provides detailed information about the **"Self-Test"** process, which is used to confirm that the PIO-D64 series card is operating correctly. Before beginning the **"Self-Test"** process, ensure that both the hardware and driver installation procedures are fully completed. For detailed information about the hardware and driver installation, refer to Chapter 3 "Hardware Installation" and Chapter 4 "Software Installation".

5.1 Self-Test Wiring

Before beginning the **"Self-Test"** procedure, ensure that the following items are available:

☑ CA-2002 (optional) cable

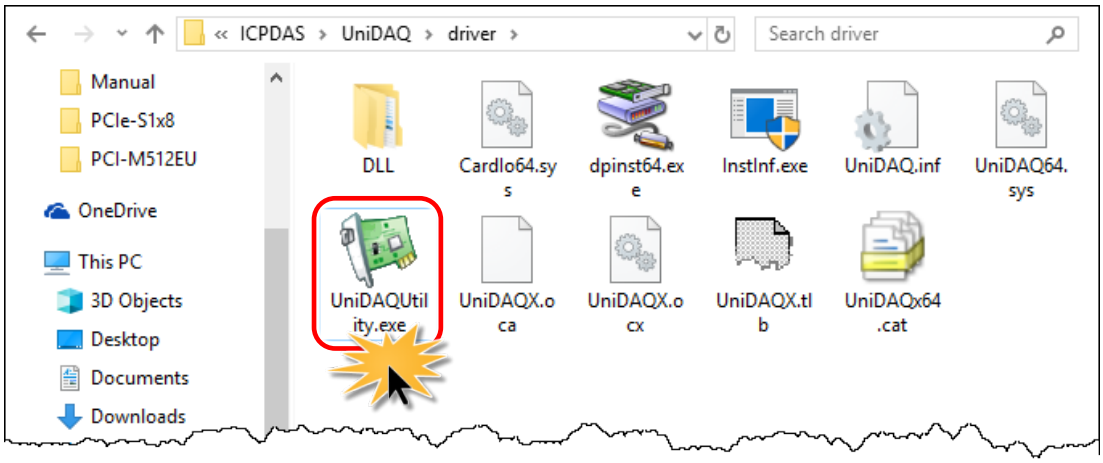
Connect the **CON1** to **CON2** on board using the **CA-2002** cable.

CA-2002 Cable

5.2 Launch the Test Program

Step 1: Double-click the **UniDAQ Utility** software.

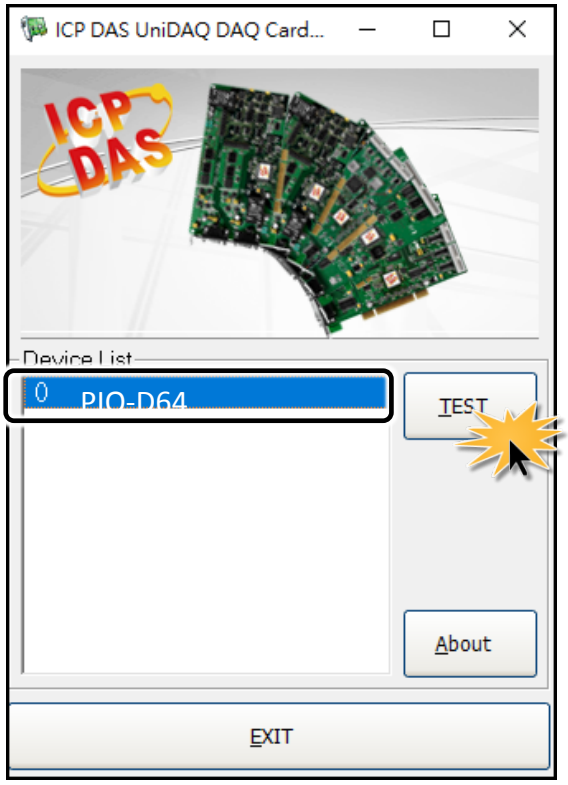
The UniDAQ Utility will be placed in the **default path "C:\ICPDAS\UniDAQ\Driver"** after completing installation.



Step 2: Confirm that your board has been successfully i nstalled in the Host system.

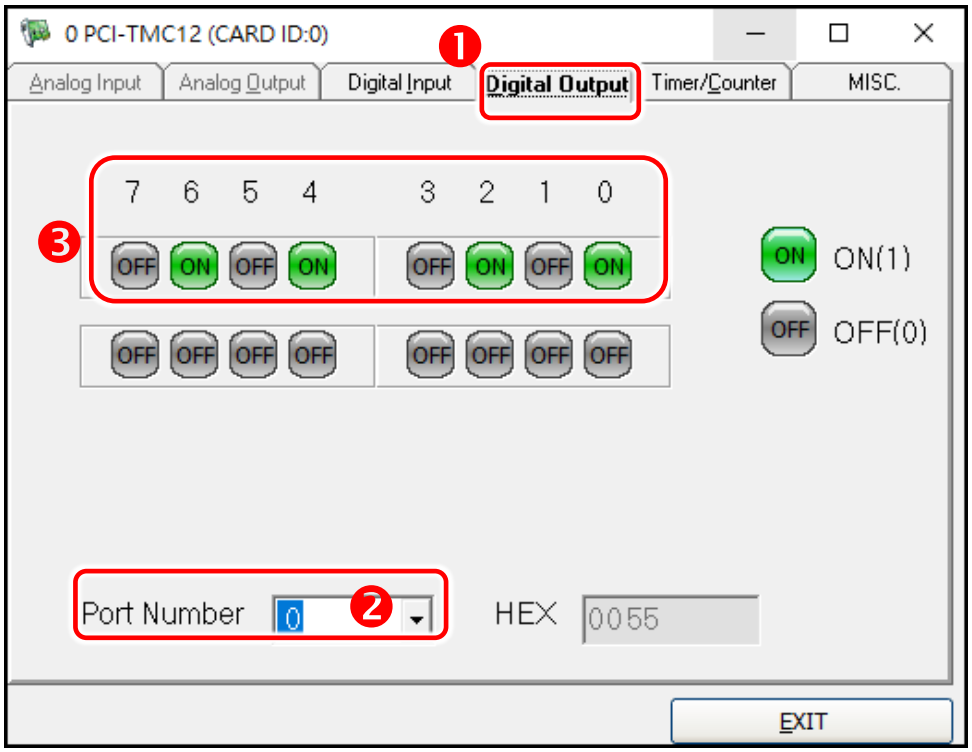
Note that the device number starts from 0.

Step 3: Click the **"TEST"** button to start the test.

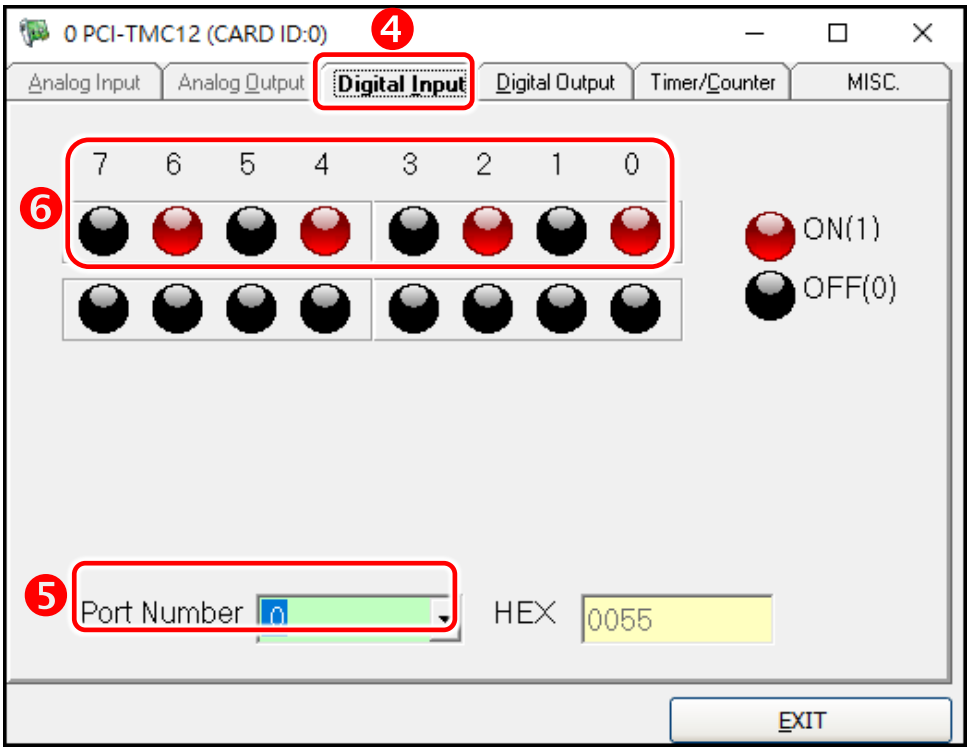


Step 4: Check the results of the **Digital Input and Digital Output** functions test.

1. Click the **"Digital Output"** tab.
2. Select **"Port 0"** from the **"Port Number"** drop-down menu.
3. Check the checkboxes for **DO channels 0, 2, 4 and 6**.



4. Click the **“Digital Input”** tab.
5. Select **“Port 1”** from the **“Port Number”** drop-down menu.
6. The DI indicators will turn **red** when the corresponding **DO channels 0, 2, 4 and 6 are ON**.



6. I/O Register Addresses

The PIO-D64 series cards are I/O mapped devices that are easily configured from any language. The following is a summary of the address registry that can be used with the PIO-D64 series.

6.1 Hardware ID

During the power-on stage, the Plug and Play BIOS will assign an appropriate I/O address to each PIO-D64 series card installed in the system. Each board includes four fixed ID numbers that are used to identify the board, and are indicated below table 6-1:

Table 6-1: Hardware ID

Model	PIO-D64	PIO-D64U	PEX-D64
Vendor ID (HEX)	E159		
Device ID (HEX)	0002	0001	
Sub-Vendor ID (HEX)	0080	4080	
Sub-Device ID (HEX)	0001		
Aux ID(HEX)	20		

6.2 I/O Address Mapping

The I/O address for PIO-D64 series cards are automatically assigned by the main board ROM BIOS. The I/O address may also be re-assigned by the user. It is strongly recommended that users do not change the I/O address. The Plug&Play BIOS will effectively perform the assignment of proper I/O addresses to each PIO-D64 series card. The I/O address for the PIO-D64/D64U and PEX-D64 are given in the table below, all of which are based on the base address of each card.

Table 6-4: Refer to Sec. 6.1 for more information about wBase.

Address	Read	Write
wBase+0	-	RESET\ Control Register
wBase+5	INT mask control register	Same
wBase+7	Aux pin status register	Same
wBase+0x2a	INT polarity control register	Same
wBase+0xc0	DI0~DI7	DO0~DO7
wBase+0xc4	DI8~DI15	DO8~DO15
wBase+0xc8	DI16~DI23	DO16~DO23
wBase+0xcc	DI24~DI31	DO24~DO31
wBase+0xd0	Read 8254-Counter0	Write 8254-Counter0
wBase+0xd4	Read 8254-Counter1	Write 8254-Counter1
wBase+0xd8	Read 8254-Counter2	Write 8254-Counter2
wBase+0xdc	Read 8254 Control Word	Writer 8254 Control Word
wBase+0xe0	Read 8254-Counter3	Write 8254-Counter3
wBase+0xe4	Read 8254-Counter4	Write 8254-Counter4
wbase+0xe8	Read 8254-Counter5	Write 8254-Counter5
wBase+0xec	Read 8254 Control Word	Writer 8254 Control Word
wBase+0xf4	Read Card ID	-

6.3 BAR 0

6.3.1 RESET\Control Register

(Read/Write): wBase+0

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	RESET\

When the PC's power is first turned on, RESET\ signal is in a Low-state. **This will disable all D/I/O operations.** The user has to set the RESET\ signal to a High-state before any D/I/O command applications are initiated.

For example:

```
outportb (wBase,1);          /* RESET\=High → all D/I/O are enable now */
outportb (wBase,0);          /* RESET\=Low → all D/I/O are disable now */
```

6.3.2 INT Mask Control Register

(Read/Write): wBase+5

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	EN3	EN2	EN1	EN0

EN0=0→ Disable INT_CHAN_0 as an interrupt signal (Default).
EN0=1→ Enable INT_CHAN_0 as an interrupt signal

EN1=0→ disable INT_CHAN_1 as a interrupt signal (Default)
EN1=1→ enable INT_CHAN_1 as a interrupt signal

EN2=0→ disable INT_CHAN_2 as a interrupt signal (Default)
EN2=1→ enable INT_CHAN_2 as a interrupt signal

EN3=0→ disable INT_CHAN_3 as a interrupt signal (Default)
EN3=1→ enable INT_CHAN_3 as a interrupt signal

For example:

```
outportb(wBase+5,0);        /*Disable all interrupt */
outportb(wBase+5,1);        /* Enable interrupt of INT_CHAN_0 */
outportb(wBase+5,2);        /* Enable interrupt of INT_CHAN_1 */
outportb(wBase+5,4);        /* Enable interrupt of INT_CHAN_2 */
outportb(wBase+5,0x07);     /* Enable all four channels of interrupt */
```

Refer to the following demo program for more information:

DEMO3.C of DOS → for INT_CHAN_0 only

DEMO4.C of DOS → for INT_CHAN_1 only

DEMO5.C of DOS → for INT_CHAN_2 only

DEMO6.C of DOS → for INT_CHAN_1 and INT_CHAN_2

6.3.3 Aux Status Register

(Read/Write): wBase+7

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Aux7	Aux6	Aux5	Aux4	Aux3	Aux2	Aux1	Aux0

Aux0=INT_CHAN_0, Aux1=INT_CHAN_1, Aux2=INT_CHAN_2, Aux3=INT_CHAN_3, Aux7~4=Aux-ID. The Aux 0~3 are used as interrupt sources. The interrupt service routine has to read this register for interrupt source identification. Refer to Sec. 2.6 for more information.

6.3.4 Interrupt Polarity Register

(Read/Write): wBase+0x2A

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	0	INV2	INV1	INV0

This register provides a function to control invert or non-invert for the interrupt signal source. A detailed application example is given below.

INV0/1/2=0→ select the invert signal from INT_CHAN_0/1/2
INV0/1/2=1→ select the non-invert signal from INT_CHAN_0/1/2

INV0= Control interrupt channel_0
INV1= Control interrupt channel_1
INV2= Control interrupt channel_2

For example:

```
outportb(wBase+0x2a,0);       /* select the invert input from all 3 channels       */
outportb(wBase+0x2a,0x0f);    /* select the non-invert input from all 3 channels */

outportb(wBase+0x2a,0x0e);    /* select the inverted input of INT_CHAN_0         */
                              /* select the non-inverted input from the others    */

outportb(wBase+0x2a,0x0c);    /* select the inverted input of INT_CHAN_0 &       */
                              /*                              INT_CHAN_1         */
                              /* select the non-inverted input from the others    */
```

Refer to Sec. 2.6 and DEMO6.C (DOS) for more information.

6.3.5 Read/Write I/O Port

(Read/Write): wBase+0xC0/ 0xC4/ 0xCC/ 0xCC

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
D7	D6	D5	D4	D3	D2	D1	D0

6.3.6 Read/Write 8254

8254 control word

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SC1	SC0	RL1	RL0	M2	M1	M0	BCD

SC1,SC0: 00: counter0
01: counter1
10: counter2
11: read -back command

RL1,RL0: 00: counter latch instruction
01: read/write low counter byte only
10: read/write high counter byte only
11: read/write low counter byte first, then high counter byte

M2,M1,M0: 000: mode0 interrupt on terminal count
001: mode1 programmable one-shot
010: mode2 rate generator
011: mode3 square-wave generator
100: mode4 software triggered pulse
101: mode5 hardware triggered pulse

BCD: 0: binary count
1: BCD count

6.3.7 Card ID Register

(Read): 0x0F4 **Card ID (Switch)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	SW3	SW2	SW1	SW0

```
wCardID = inportb(wBase+0xF4);                    /* read Card ID*/
```

Note: The Card ID function supports the model:
PIO-D64U (Ver1.0 or above)

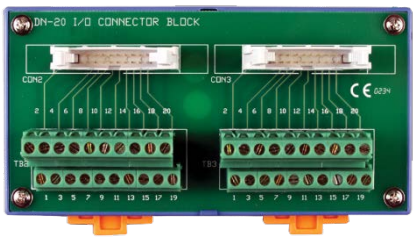
7. Windows API Function

For more details regarding the Windows API Functions for the card, refer to UniDAQ SDK User manual, which can be downloaded from:
http://ftp.icpdas.com/pub/cd/iocard/pci/napdos/pci/unidaq/manual/

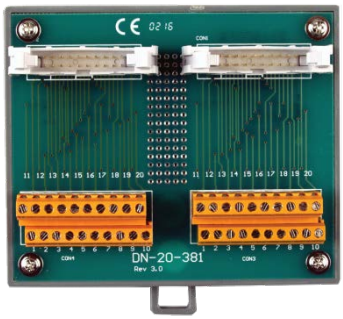
8. Daughter Boards

8.1 DN-20/DN-20-381

The DN-20/20-381 is a general-purpose DIN-Rail mountable daughter board containing 20-pin header I/O Connectors and is designed to allow easy field wiring connections. Pins 01 to 20 on the DN-20 daughter board are connected to the CN1/CN2 connector on the card using a 20-pin flat cable.

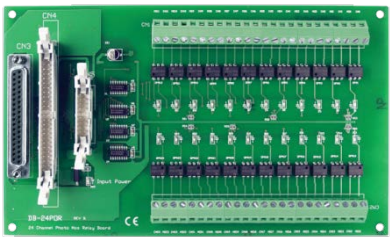


DN-20



DN-20-381

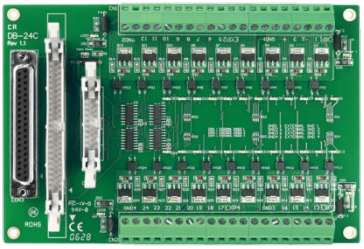
8.2 DB-24POR



The DB-24POR includes 24 normally open, form A, Photo-MOS relays. The board interface to field logic signals, eliminating ground-loop problems and isolating the host computer from damaging voltages. The user can use the DB-24POR to switch load, up to 350 V_{AC} and up to 130 mA. The DB-24POR has one 37-pin D-sub connector, one 50-pin OPTO-22 compatible male header and one 20-pin male header.

The relay is energized by applying a 5 voltage signal to the appropriate relay channel on the n the 50-pin header or 20-pin header or 37-pin D-sub connector. Twenty-four indicators LEDs, one for each relay, light when their associated relay is activated. The DB-24POR daughter board is connected to the CN1/CN2 connector on the card using a 20-pin flat cable.

8.3 DB-24C



The DB-24C has 24 channels of optically isolated digital outputs, arranged into four isolated banks. Each digital output offers a Darlington transistor and integral suppression diode for inductive load. The board interface to field logic signals, eliminating ground-loop problems and isolating the host computer from damaging voltages. The DB-24C has one 37-pin D-sub connector, one 50-pin OPTO-22 compatible male header and one 20-pin male header.

The transistor is energized by applying a 5-voltage signal to the appropriate input channels on the 50-pin header or 20-pin header or 37-pin D-sub connector. Twenty-four enunciator LEDs, one for each transistor, light when their associated transistor is activated. The DB-24C daughter board is connected to the CN1/CN2 connector on the card using a 20-pin flat cable.