PIO-821 Series Card

User Manual

45 kS/s 12-bit, 16-ch A/D, 1-ch D/A Multi-function DAQ Board

Version 2.0, Mar. 2015

SUPPORTS

Board includes PIO-821L, PIO-821H, PIO-821LU and PIO-821HU.

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

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 at:
service@icpdas.com; service.icpdas@gmail.com
We will give you quick response within 2 workdays.
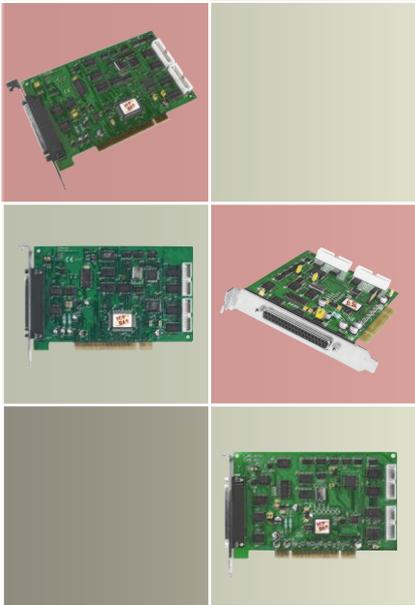
TABLE OF CONTENTS

1. Introduction

The PIO-821LU/HU card is the new generation product that ICP DAS provides to meet RoHS compliance requirement. The new PIO-821LU/HU card is designed as a drop-in replacement for the PIO-821L/H, and users can replace the PIO-821L/H by the PIO-821LU/HU directly without software/driver modification.

The PIO-821 series cards are a high performance multifunction board for PC/AT compatible computers. The PIO-821LU/HU universal PCI card supports 3.3 V/5 V PCI bus while the PIO-821L/H supports 5 V PCI bus. They feature a 12-bit ADC and the maximum sampling rate of the A/D converter reaches up to about 45K samples/sec., 16 single-ended or 8 differential analog input channels, 12-bit DAC voltage output and 16 TTL-compatible digital input and digital output channels. The PIO-821L/LU provides for low gain (1, 2, 4, 8); and the PIO-821H/HU supports high gain (1, 10, 100, 1000).

The PIO-821LU/HU also adds a Card ID switch and pull-high/pull-low resisters for DI on board. Users can set Card ID on a board and recognize the board by the ID via software when using two or more PIO-821LU/HU cards in one computer. The pull-high/ pull-low resisters allow the DI status to be specified when the DI channels are unconnected; the DI status will remain in high or low status other than floating.

These cards support various OS such as Linux, DOS, Windows 98/NT/2000 and 32/64-bit Windows 8/7/Vista/XP. DLL and Active X control together with various language sample program 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 to quickly and easily develop their own applications.
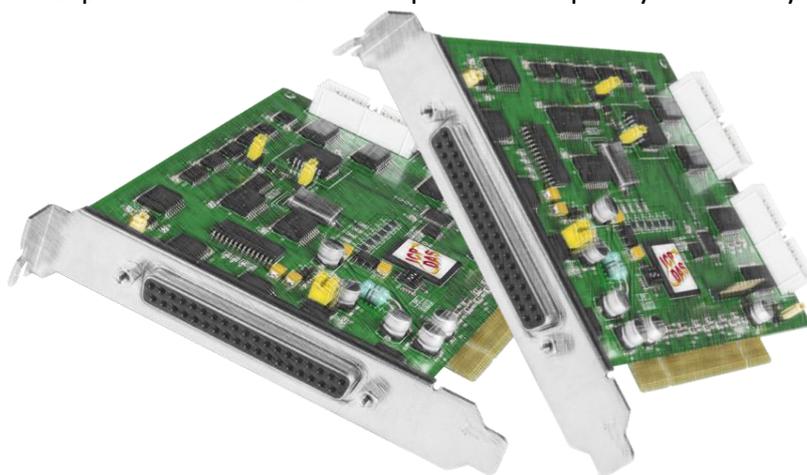
1.1   Packing List

The shipping package includes the following items:

- One PIO-821 series card hardware
- One printed Quick Start Guide
- One software utility CD

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 want to ship or store the product in the future.

1.2   Features

- Support the +5V PCI bus for PIO-821L/H
- Support the +3.3/+5 V PCI bus for PIO-821LU/HU
- The maximum sampling rate of the 12-bit A/D converter is about 45 k samples/sec.
- 16 single-ended or 8 differential analog input channels
- Software selectable input ranges
- A/D trigger mode: software trigger, pacer trigger, external trigger
- Analog input range:
  - PIO-821L/LU: ±5 V, ±2.5 V, ±1.25 V, ±0.625 V
  - PIO-821H/HU: ±5 V, ±0.5 V, ±0.05 V, ±0.005 V
- Programmable high/low gain:
  - PIO-821L/LU: 1, 2, 4, 8
  - PIO-821H/HU: 1, 10, 100, 1000
- Supports Card ID (SMD Switch) for PIO-821LU/HU
- Digital input port can be set to pull-high or pull-low for PIO-821LU/HU
- 1-channel 12-bit D/A voltage output
- 16-channel 5 V/TTL digital input
- 16-channel 5 V/TTL digital output
- Interrupt handling
- SMD, Sort card

1.3 Specifications

Model Name	PIO-821LU	PIO-821HU	PIO-821L	PIO-821H
Analog Input				
Channels	16 single-ended/ 8 differential			
A/D Converter	12-bit, 8 µs Conversion time			
Sampling Rate	45 kS/s. max.			
FIFO Size	N/A			
Over voltage Protection	Continuous +/-35 Vp-p			
Input Impedance	10 MΩ/6 pF			
Trigger Modes	Software, Internal programmable pacer			
Data Transfer	Polling, Interrupt			
Accuracy	0.01 % of FSR ±1 LSB @ 25 °C, ± 10 V			
Zero Drift	+/- 4 ppm/°C of FSR			
Analog Output				
Channels	1			
Resolution	12-bit			
Accuracy	0.01 % of FSR ±1/2 LSB @ 25 °C, ± 10 V			
Output Range	Unipolar:0 ~ 5 V, 0 ~ 10 V, 0 ~ Ext Ref			
Output Driving	+/- 5 mA			
Slew Rate	0.3 V/µs			
Output Impedance	0.1 Ω max.			
Operating Mode	Software			
Digital Input				
Channels	16			
Compatibility	5 V/TTL			
Input Voltage	Logic 0: 0.8 V max.; Logic 1: 2.0 V min.			
Response Speed	1.2 MHz (Typical)			
Digital Output				
Channels	16			
Compatibility	5 V/TTL			
Output Voltage	Logic 0: 0.4 V max.; Logic 1: 2.4 V min.			
Output Capability	Sink: 2.4 mA @ 0.8 V; Source: 0.8 mA @ 2.0 V			
Response Speed	1.2 MHz (Typical)			

Model Name	PIO-821LU	PIO-821HU	PIO-821L	PIO-821H
Timer/Counter				
Channels	3(Internal pacer x 1/Independent x 2)			
Resolution	16-bit			
Compatibility	5 V/TTL			
Input Frequency	10 MHz max.			
Reference Clock	Internal: 2 MHz			
General				
Bus Type	3.3 V/5 V Universal PCI, 32-bit, 33 MHz		5 V PCI, 32-bit, 33 MHz	
Data Bus	8-bit			
Card ID	Yes(4-bit)		No	
I/O Connector	Female DB37 x 1 20-pin box header x 2			
Dimensions (L x W x D)	159 mm x 97 mm x 22mm		183 mm x 105 mm x 22 mm	
Power Consumption	960 mA @ +5 V			
Operating Temperature	0 ~ 60 °C			
Storage Temperature	-20 ~ 70 °C			
Humidity	5 ~ 85% RH, non-condensing			

1.3.1 Analog Input Range

Model	PIO-821L/LU (Low-Gain)			
Gain	1	2	4	8
Bipolar	±5 V	±2.5 V	±1.25 V	±0.625 V
Sampling Rate Max.	45 kS/s.			
Model	**PIO-821H/HU (High-Gain)**			
Gain	1	10	100	1000
Bipolar	±5 V	±0.5 V	±0.05 V	±0.005 V
Sampling Rate Max.	45 kS/s.		10 kS/s.	1 kS/s.

1.3.2 Analog Output Range

Data Input	Analog Output
1111 1111 1111	-Vref (4095/4096)
0000 0000 0000	-Vref (1/4096)
0000 0000 0000	0 Volts

1.3.3 Programmable Timer/Counter

■ **Type:** 82C54 programmable timer/counter.

■ **Timers:** three 16-bit independent timers.

☐ **Timer 2** is used as the internal A/D pacer trigger timer.

☐ **Timer 0** and **Timer 1** are used as the machine independent timer.

☐ **Timer 0** could use external clock by setting JP5.

2. Hardware Configuration

2.1   The Block Diagrams

The block diagram of the PIO-821 series is illustrated below in order to assist users in understanding the data flow within the hardware and software system structure.
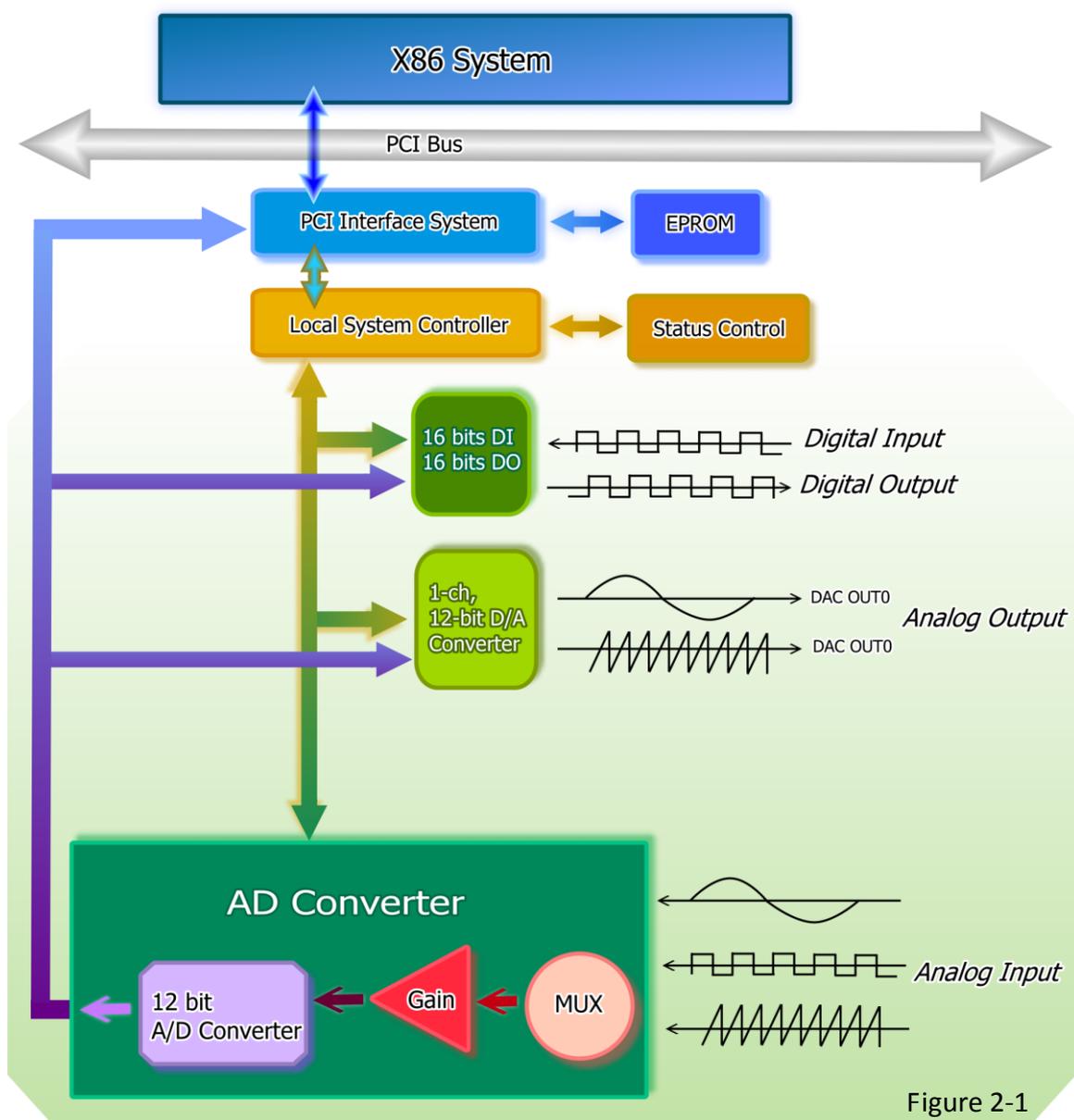
Figure 2-1

2.2   Board Layout

Figure 2.2 and Figure 2.3 shows the layout of the PIO-821L/H and PIO-821LU/HU boards and the locations of the configuration jumper and connector for signal wiring.

■   **Board Layout of the PIO-821L/H.**
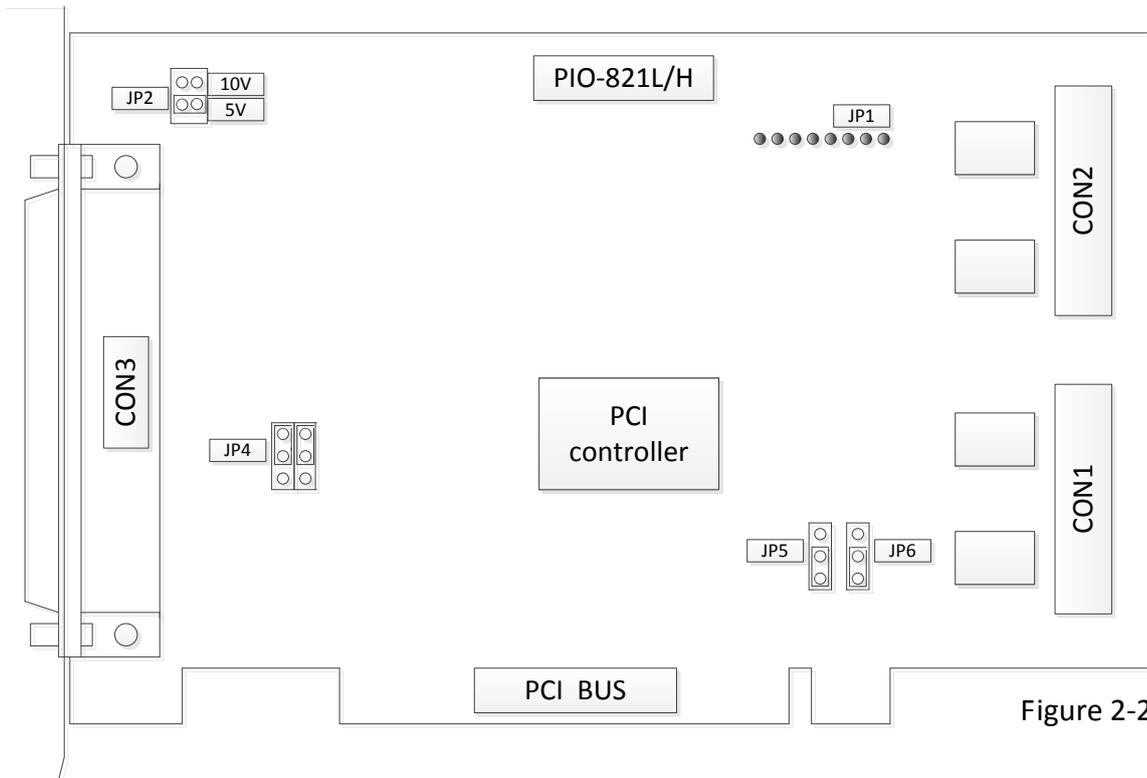
Figure 2-2

CON1	The terminal for digital input	Refer to Sec.2.6 for more detailed about pin assignments information.
CON2	The terminal for digital output	
CON3	The terminal for the A/D and D/A converters for voltage input/output	
JP1	No use	-
JP2	D/A reference voltage selection	Refer to Sec.2.4.1
JP4	A/D input type selection (Single-End or Differential)	Refer to Sec.2.4.3
JP5	External Clock/Internal Clock (2MHz)	Refer to Sec.2.5.1
JP6	External Gate/Counter0 (COUT0)	Refer to Sec.2.5.2

■ **Board Layout of the PIO-821LU/HU.**
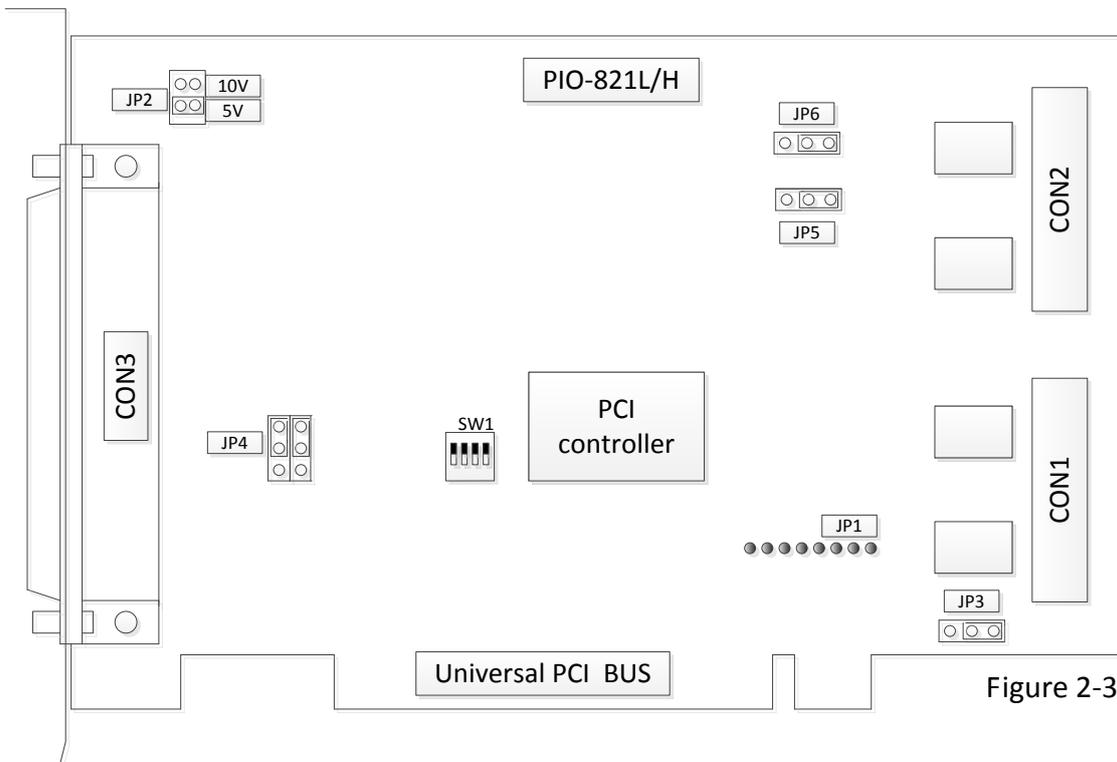
Figure 2-3

CON1	The terminal for digital input	Refer to Sec.2.6 for more detailed about pin assignments information.
CON2	The terminal for digital output	
CON3	The terminal for the A/D and D/A converters for voltage input/output	
JP1	No use	-
JP2	D/A reference voltage selection	Refer to Sec.2.4.1
JP3	Pull-high or pull-low for DI	Refer to Sec.2.4.2
JP4	A/D input type selection (Single-End or Differential)	Refer to Sec.2.4.4
JP5	External Clock/Internal Clock (2MHz)	Refer to Sec.2.5.1
JP6	External Gate/Counter0 (COUT0)	Refer to Sec.2.5.2
SW1	Card ID function	Refer to Sec.2.3

2.3   Card ID Switch

The PIO-821LU/HU has a Card ID switch (SW1) with which users can recognize the board by the ID via software when using two or more PIO-821LU/HU cards in one computer. The default Card ID is 0x0. For detail SW1 Card ID settings, please refer to Table 2-1. **Note that the Card ID function is only supported by the PIO-821LU/HU.**
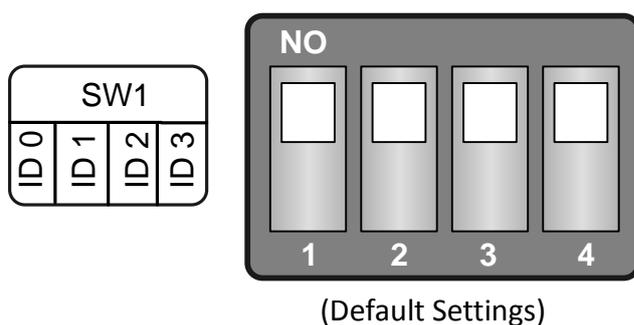
(Default Settings)

Table 2-1   (*) Default Settings; OFF → 1; ON → 0

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

2.4   Configuration of the DA/AD Output Signals

2.4.1 JP2: D/A Reference Voltage Selection

JP2 is used to configure the internal reference voltage for analog output. There are two types of internal reference voltages, which are 5 V or 10 V. The reference voltage settings are presented in Figure 2-4.   The 5 V reference voltages provide a 0 to 5 V range of analog voltage output for the A/D converter.   In the same way, the 10 V reference voltages support a functional range between 0 to 10 V of analog voltage output.

Figure 2-4

10V   10V

JP2   JP2

5V   5V

Reference Voltage = 5 V   **Reference Voltage = 10 V**
(Default Settings)
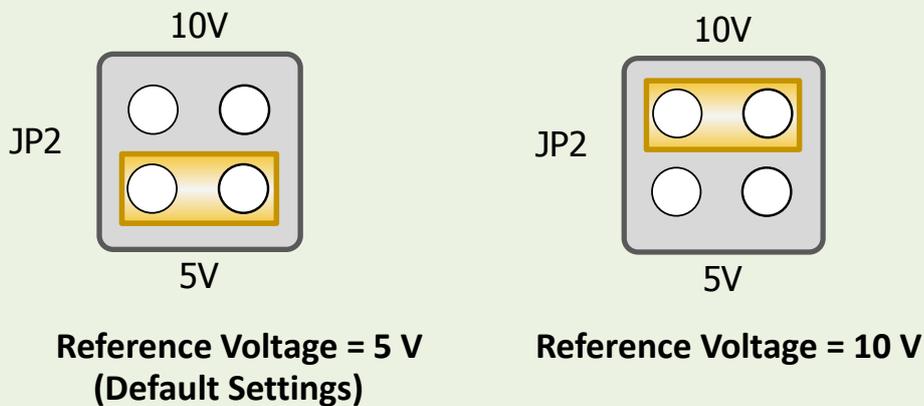
2.4.2 JP3: D/I Port Settings (Pull-High/Low)

This DI ports can be pull-high or pull-low that is selected by JP3. The pull-high/low jumpers of the card allow user to predefine the DI status instead of floating when the DI channels are unconnected or broken. The configuration is illustrated in the figure below. **Note: This function only supports PIO-821LU/HU.**

Figure 2-5

JP3   JP3

Vcc   GND   Vcc   GND

Pull-High   **Pull-Low (Default)**
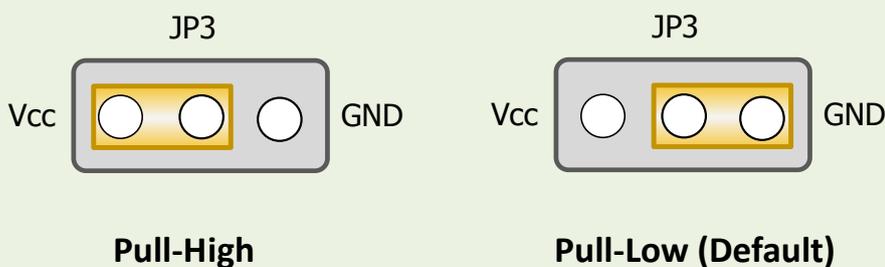
2.4.3 JP4: A/D input type selection (S.E. /Diff.)

JP4 is used to select the type of analog input between single-ended and differential inputs. As shown in **Figure 2-6 (a)**, the user needs to connect pin 1, 3 and pin 2, 4 to obtain the **single-ended** measurement for the analog input signal, which is the default setting. However, for the **differential** signal measurement, the pin 3, 5 and pin 4, 6 for the JP4 jumper should be connected as shown **Figure 2-6 (b)**.   Furthermore, based on the type of analog input configuration, the PIO-821 series cards can offer 16 single-ended or 8 differential analog input measurement channels. Notice that all the measurements of analog input channels are configured at the same time according to the chosen setting of single-ended or differential methods.

Figure 2-6

(a)

**Single-Ended
Inputs (Default)**

(b)
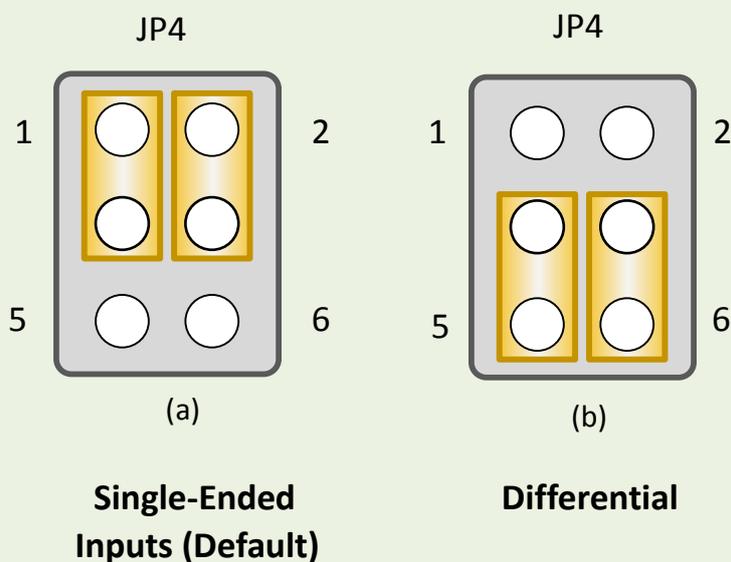
Differential

2.4.4 A/D Input Signal Connection

The PIO-821 series card can measure either single-ended or differential-type analog input signals. The user must decide which mode is most suitable for measurement purposes. Please refer to the section 2.4.3 to see how to configure the JP4 jumper based on your analog input type.

In general, there are four different analog signal connection methods, as shown in Figure 2-7 to Figure 2-10. The connection shown in Figure 2-7 is suitable for grounded analog input signals. The connection shown in Figure 2-8 is used to measure more channels than that shown in Figure 2-7, but is only suitable for large analog input signals. The connection shown in Figure 2-9 is suitable for thermocouple connections and the connection shown in Figure 2-10 is suitable for floating analog input signals.

> ⚠️ **Note: In Figure 2-9, the maximum common mode voltage between the analog input source and the AGND is 70 V_{p-p}, so the user must ensure that the input signals is within this specification. If the common mode voltage is above 70 V_{p-p}, the input multiplexer will be permanently damaged!**

The simplest way to select your input signal connection configuration is listed below.

1. Grounding source input signal	→ see Figure 2-7
2. Thermocouple input signal	→ see Figure 2-9
3. Floating source input signal	→ see Figure 2-10
4. If Vin > 0.1 V, gain<=10 and more channels are needed	→ see Figure 2-8
5. Current source input signal	→ see Figure 2-11

If you are unsure of the characteristics of your input signal, follow these test steps:

Step1: Try and record the measurement results when using Figure 2-7.
Step2: Try and record the measurement results when using Figure 2-10.
Step3: Try and record the measurement results when using Figure 2-8.
Step4: Compare the three results and select the best one.

■ **Figure 2-7:** Differential input with grounded source (Right way)
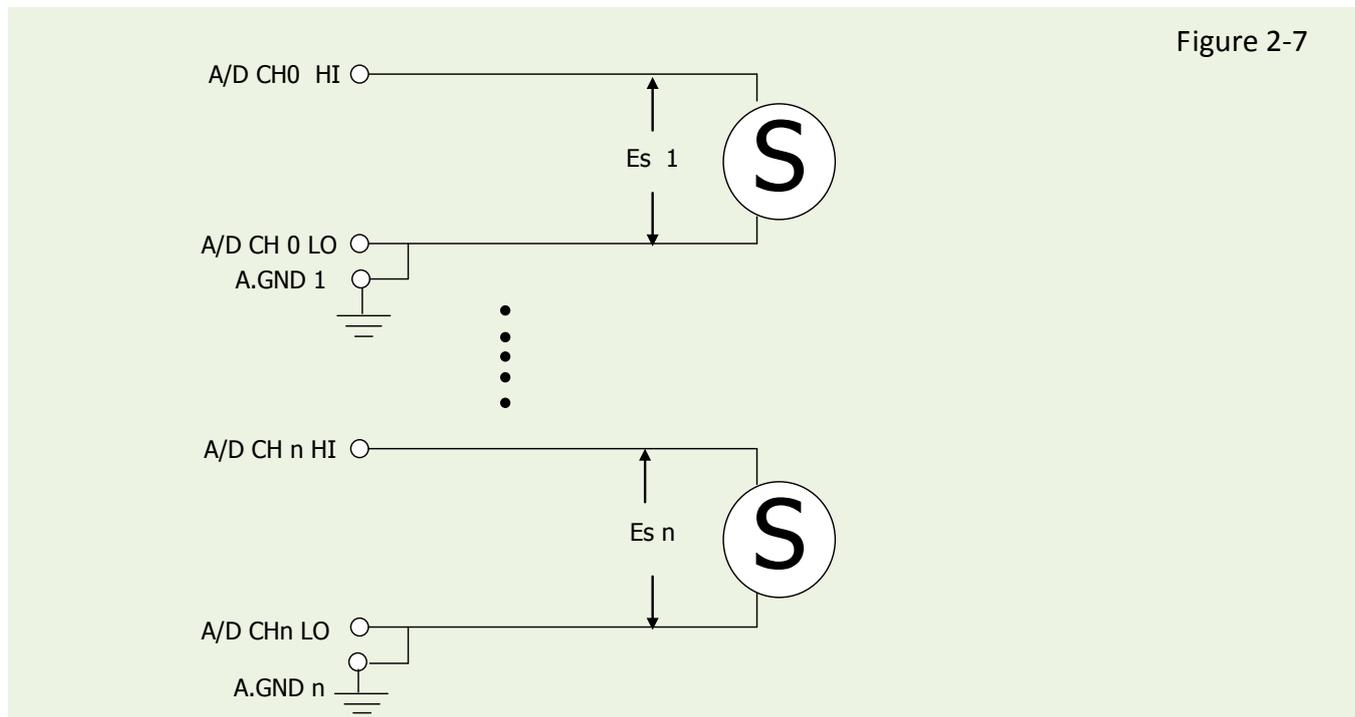
Figure 2-7

■ **Figure 2-8:** Single-ended input with floating signal source
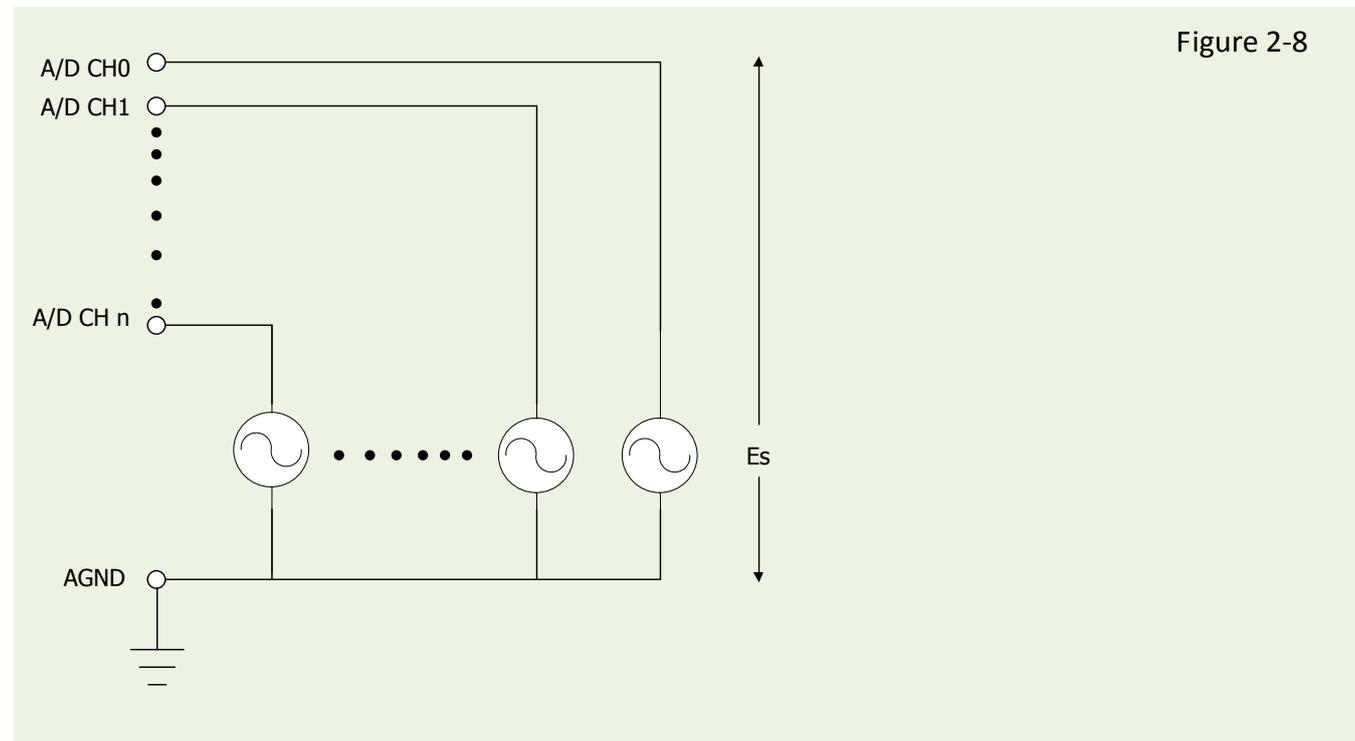
Figure 2-8

■ **Figure 2-9:** Differential input with floating thermocouple signal

Figure 2-9

Note: If the input signal is not thermocouple, the user should use an oscilloscope to measure common mode voltage of Vin before connecting to PIO-821 series card. Don't use a voltage meter or multi-meter.

Do not join LO to A.GND at the computer

CAUTION: In Figure 2-9, the maximum common mode voltage between the analog input source and the AGND is 70 V_{p-p}. Make sure that the input signal is under specification first! If the common mode voltage is over 70 V_{p-p}, the input multiplexer will be permanently damaged.

■ **Figure 2-10:** Differential input with floating signal source
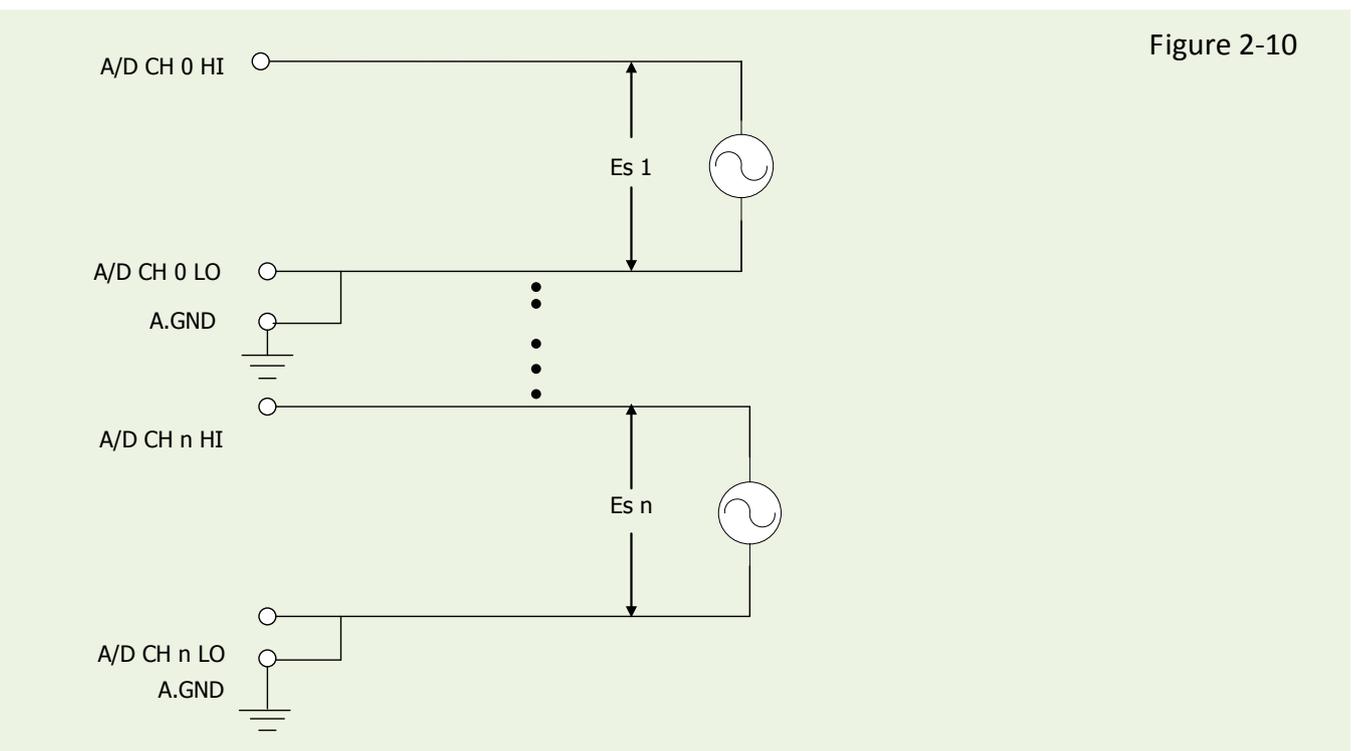
Figure 2-10

■ **Figure 2-11:** Connecting to a 4 ~ 20 mA Source
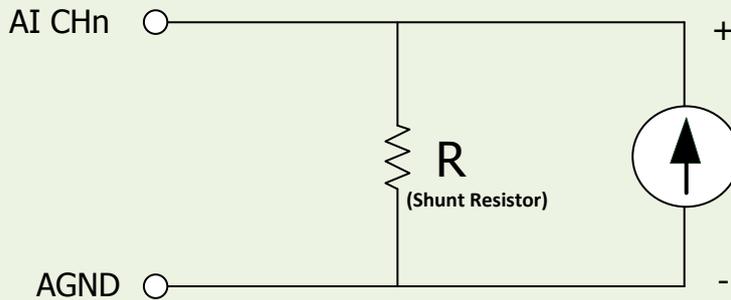
Example: A 20 mA source current through a 125 Ω resistor (e.g. 125 Ω, 0.1% DIP Resistors) between + and – terminals and the board will read a 2.5 V_{DC} voltage. You can use the **I = V/R** (Ohm's law) to calculate what value the source current should have.

Current (I) = Voltage (V) / Resistance (R)

= 2.5 V / 125 Ω

= 0.02 A

= 20 m

■ **Figure 2-12:** Signal Shielding

Signal shielding connections in Figure 2-7 to Figure 2-11 are all the same, as show in the below figure. Use a single-point connection to **frame ground (not A.GND or D.GND)**
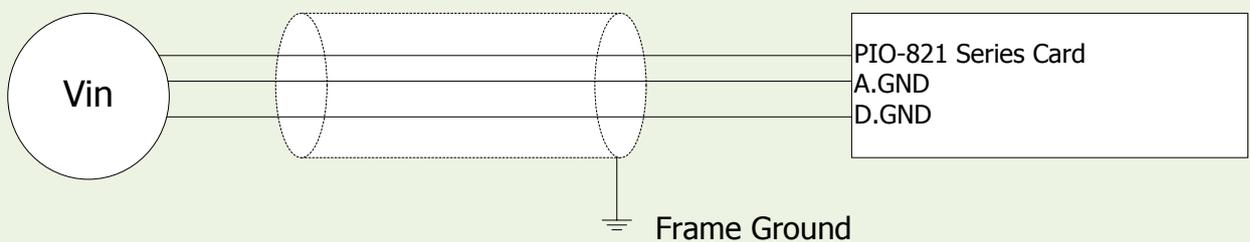
2.5   Configuration of the 8254 Chip

2.5.1 JP5: Chip 8254 Clock Settings (Internal/External Clock)

The function of the 8254 chip is used to provide the hardware sampling mechanism and counter operation.   There is two ways to provide the clock source, which are (1) On board oscillator 2 MHz called as Internal clock; (2) External clock source that comes in from the connector pin and can be provided by user. In below the detail clock source setting will be brought up.

The PIO-821 series card can be selected to use either the external or internal clock as a signal source of the timer 0 of IC 82c54.   This selection is made by the JP5 jumper, as shown in Figure 2-12. The left hand side diagram shows the setting for the internal clock at 2 MHz.   On the other hand, the right hand side diagram shows the setting for external clock.   The clock signal source is controlled by the external source.   However, the maximum frequency of the clock source is limited by 10 MHz.

Figure 2-12

	PIO-821 L/H	PIO-821 LU/HU
Internal Clock 2 MHz (Default)	EXT_CLK JP5 2 MHz	JP5 2 MHz EXT_CLK
External Clock	EXT_CLK JP5 2 MHz	JP5 2 MHz EXT_CLK
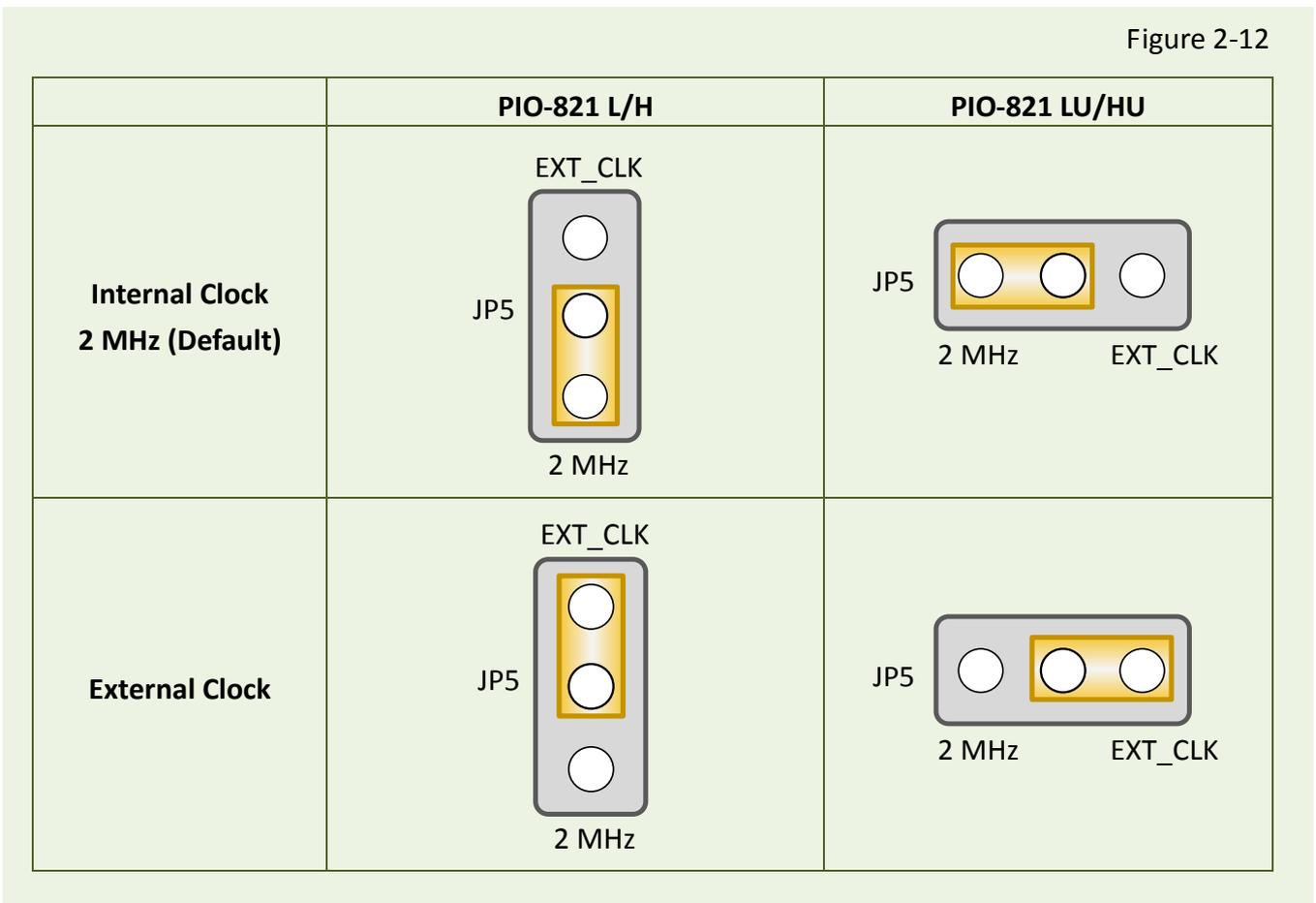
2.5.2 JP6: External Gate/Internal Counter0 Settings

The PIO-821 series card can also be set to use the external gate or internal Cout0 (Counter0) signal to control the 82c54's counter2 through the use of the JP6 jumper, as shown in Figure 2-13. The left hand diagram of Figure 2-13 is the default setting for the external gate signal control and the right hand diagram shows the internal Cout0 signal control.

Figure 2-13

	PIO-821 L/H	PIO-821 LU/HU
EXT_GATE (Default)		
COUT0		
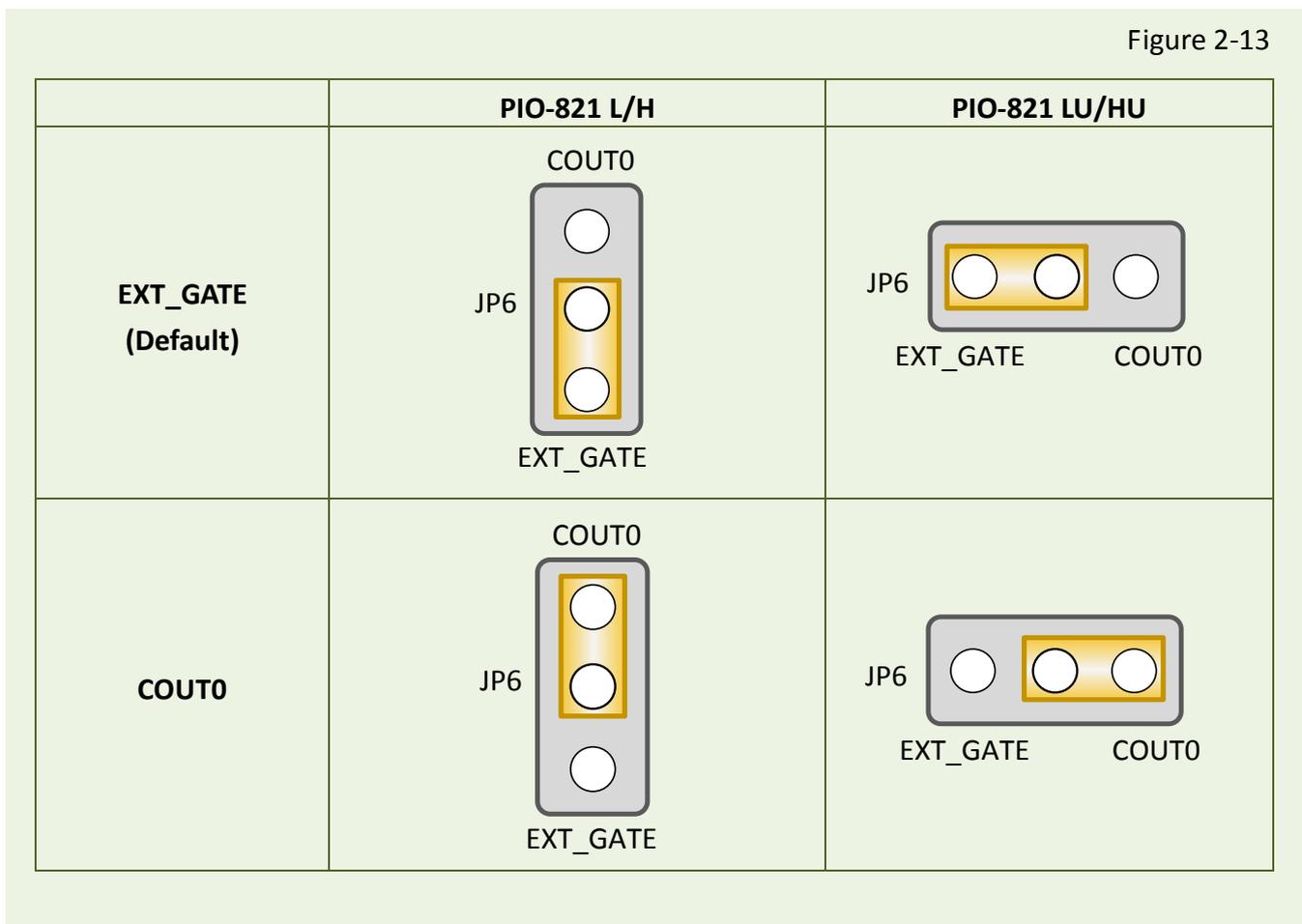
2.5.3 The 8254 Timer/Counter Connection

The 8254 Programmable timer/counter has 4 registers from **wBase+0xC0** through **wBase+0xCC**. For detailed programming information about 8254, please refer to Intel's "Microsystem Components Handbook". The block diagram is as below.

Figure 2-14
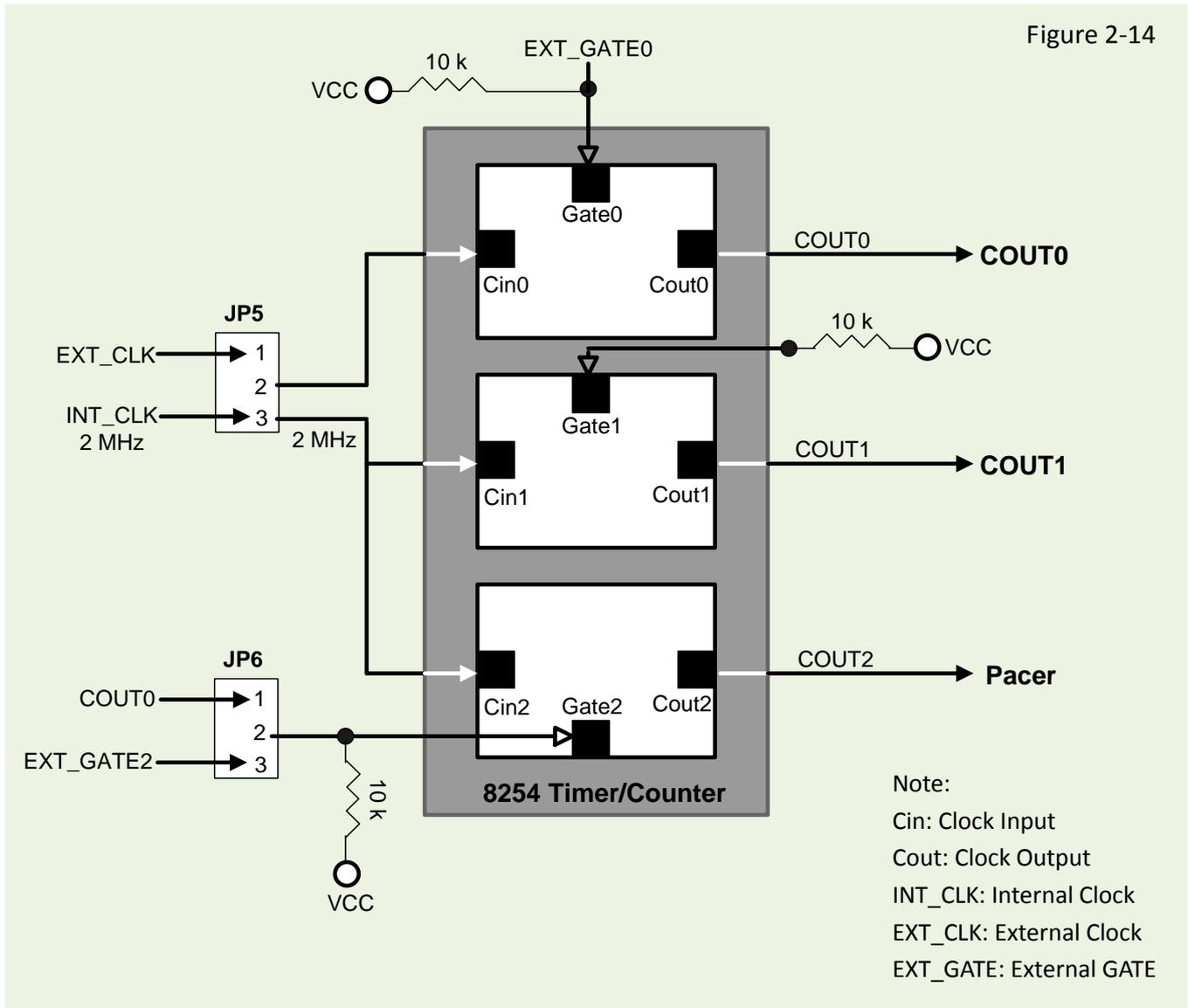
The counter0, counter1 and counter2 are all 16-bit counter. The counter0 and counter2 can be cascaded as a 32-bit timer. The **counter2** is used in **pacer trigger**, **PIO821_SetChannelConfig()** and **PIO821_Delay()** functions, so they cannot be used in the same time.

2.6 Pin Assignments

The Pin assignments of CON1, CON2 and CON3 on the PIO-821 series cards are represented in the figure below.

■ CON1/CON2: 20-pin flat-cable headers for digital input/output.

CON1/2 are TTL compatible	
High (1)	2.0 ~ 5.0 V(Voltage over 5.0V will damage the device)
None Define	2.0 V ~ 0.8 V
Low (0)	Under 0.8 V

■ CON3: 37-pin D-type female connector for analog input/output and Timer/Counter.

Pin Assignment	Terminal No.		Pin Assignment
AI_0	01		
AI_1	02	20	AI_8
AI_2	03	21	AI_9
AI_3	04	22	AI_10
AI_4	05	23	AI_11
AI_5	06	24	AI_12
AI_6	07	25	AI_13
AI_7	08	26	AI_14
A.GND	09	27	AI_15
A.GND	10	28	A.GND
N.C.	11	29	A.GND
N.C.	12	30	DAOUT
+12V	13	31	N.C.
A.GND	14	32	GATE0
D.GND	15	33	N.C.
COUT0	16	34	GATE2
N.C.	17	35	COUT2
COUT1	18	36	EXT_INT
VCC	19	37	EXT_CLK

CON3

Pin Assignment	Terminal No.		Pin Assignment
DI 0	01	02	DI 1
DI 2	03	04	DI 3
DI 4	05	06	DI 5
DI 6	07	08	DI 7
DI 8	09	10	DI 9
DI 10	11	12	DI 11
DI 12	13	14	DI 13
DI 14	15	16	DI 15
GND	17	18	GND
+5V	19	20	+12V

CON1

Pin Assignment	Terminal No.		Pin Assignment
DO 0	01	02	DO 1
DO 2	03	04	DO 3
DO 4	05	06	DO 5
DO 6	07	08	DO 7
DO 8	09	10	DO 9
DO 10	10	12	DO 11
DO 12	12	14	DO 13
DO 14	14	16	DO 15
GND	16	18	GND
+5V	18	20	+12V

CON2

3. Hardware Installation

> *Note:*
> *It's recommended to install driver first, since some operating system (such as Windows 2000) may ask you to restart the computer again after driver installation. This reduces the times to restart the computer.*

To install the PIO-821 series card, follow the procedure described below:

> Step 1: Installing PIO-821 series card driver on your computer first.

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

> Step 2: Configuring Card ID by the SW1 DIP-Switch for PIO-821LU/HU only.

For detailed information about the card ID (SW1), please refer to Sec. 2.3 Car ID Switch .

Step 3: Shut down and power off your computer.

Step 4: Remove all covers from the computer.

Remove

Step 5: Select an empty PCI slot.
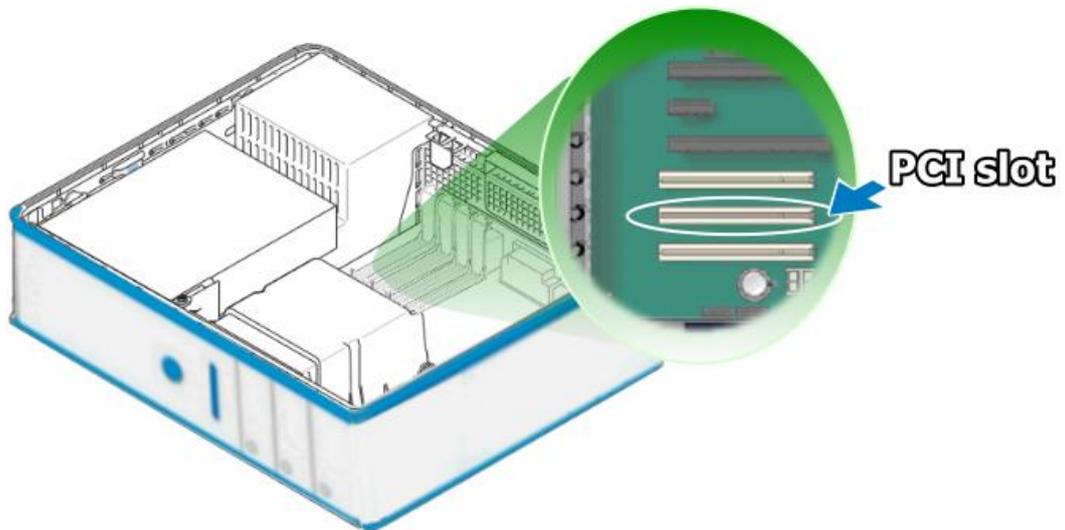
PCI slot

Step 6: Remove the PCI slot cover form the PC.
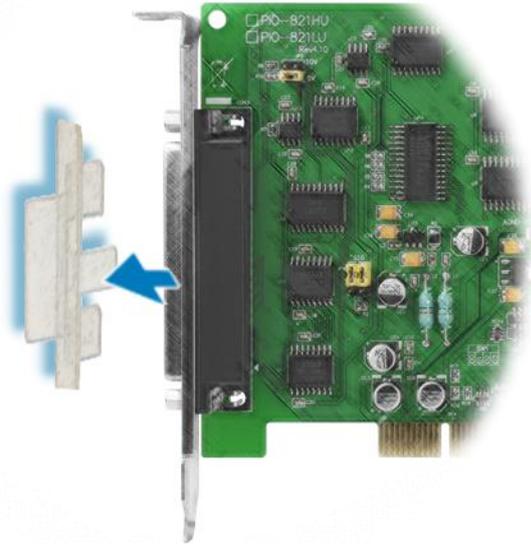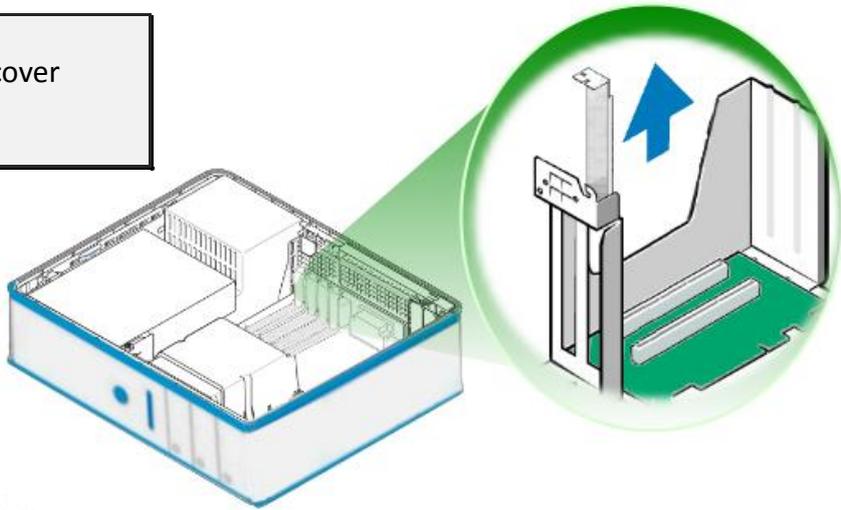
Step 7: Remove the connector cover form the PIO-821 series card.

Step 8: Carefully insert your PIO-821 series card into the PCI slot.

PCI slot
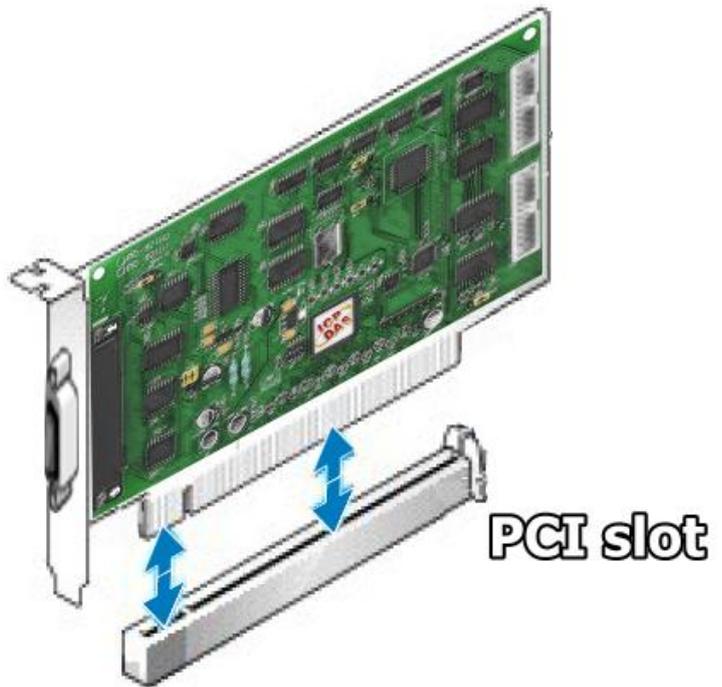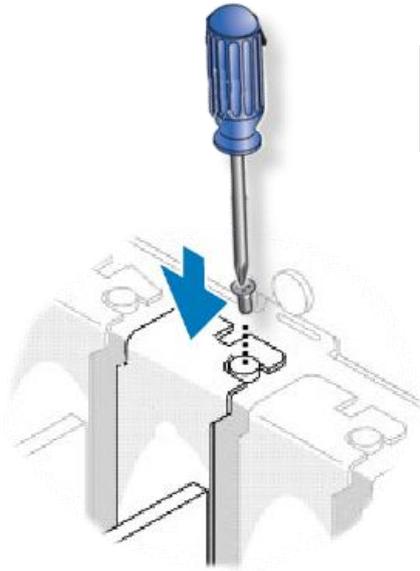
Step 9: Tighten the screw.

Confirm the PIO-821 series card is mounted on the motherboard.

Step 10: Replace the computer cover.
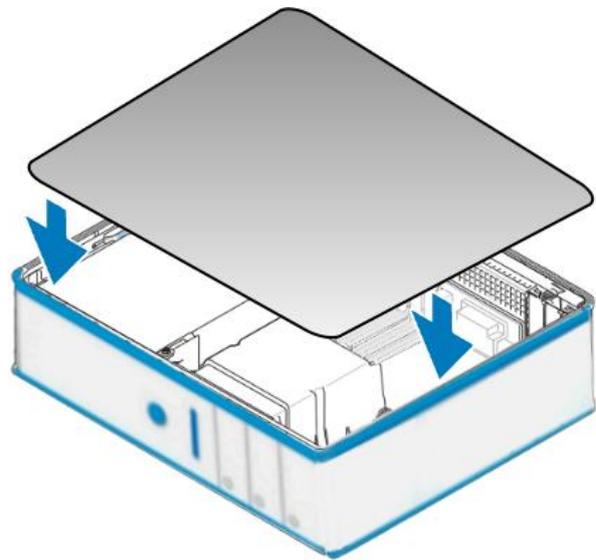
Step 11: Power on the computer.

Follow the prompt message to finish the Plug&Play steps, please refer to Chapter 4 Software Installation.

4. Software Installation

This chapter provides a detailed description of the process for installing the PIO-821 series driver and how to verify whether the PIO-821 was properly installed. PIO-821 series card can be used on DOS, Linux and Windows 98/ME/2000 and 32-/64-bit XP/2003/Vista/7/8 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 installer package for the PIO-821 series card can be found on the supplied CD-ROM, or can be obtained from the ICP DAS FTP web site. Install the appropriate driver for your operating system. The location and addresses are indicated in the Table4-1 and Table 4-2 below.

Table 4-1: UniDAQ Driver/SDK

OS	Windows 2000、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
Driver Name	UniDAQ Driver/SDK (unidaq_win_setup_xxxx.exe)
CD-ROM	CD:\\ NAPDOS\PCI\UniDAQ\DLL\Driver\
Web Site	http://ftp.icpdas.com/pub/cd/iocard/pci/napdos/pci/unidaq/dll/driver/
Installing Procedure	For detailed information about the UniDAQ driver installation, please refer to UniDAQ DLL Software Manual. The user manual is contained in: CD:\NAPDOS\PCI\UniDAQ\Manual\ http://ftp.icpdas.com/pub/cd/iocard/pci/napdos/pci/unidaq/manual/

Table 4-2: PIO-821 Series Classic Driver

OS	Windows 95/98/ME、Windows NT、Windows 2000、32-bit Windows XP、32-bit Windows 2003、32-bit Windows Vista、32-bit Windows 7
Driver Name	**PIO-821 Series Classic (PIO-821_Win_Setup_xxx_x86.exe)**
CD-ROM	CD:\\ NAPDOS\PCI\PIO-821\DLL\Driver\
Web Site	http://ftp.icpdas.com/pub/cd/iocard/pci/napdos/pci/pio-821/dll/driver/
Installing Procedure	For detailed information about the PIO-821 series classic driver installation, please refer to PIO-821 series classic driver DLL Software Manual. The user manual is contained in: CD:\NAPDOS\PCI\PIO-821 \Manual\ http://ftp.icpdas.com/pub/cd/iocard/pci/napdos/pci/pio-821/manual/

4.2 PnP Driver Installation

Power off the computer and install the PIO-821 series cards. Turn on the computer and Windows 95/98/ME/NT/2000 and 32-/64-bit Windows XP/2003/Vista/7/8 should automatically defect the new PCI device(s) and then ask for the location of the driver files for the hardware. If a problem is encountered during installation, refer to the PnPinstall.pdf file for more information.

4.3   Verifying the Installation

Please open the **Device Manager** to verify the installation. Below are the steps for entering the Device Manager in each of the major versions of windows. Refer to appropriate for your OS, continue to complete the following steps:

4.3.1 How do I get into Windows Device Manager?

■   **Microsoft Windows 95/98/ME users**

Step 1:   On the desktop right-click on **"My Computer"** and click **"Properties"** or open the **"Control Panel"** and double-click the **"System"** icon.

Step 2:   Click the "Device Manager" tab.

■   **Microsoft Windows 2000/XP users**

Step 1:   Select **"Start→ Settings→ Control Panel"** and double-click the **"System"** icon.

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

■   **Microsoft Windows 2003 users**

Step 1:   Open the **"Administrative Tools"** in Control Panel.

Step 2:   Within the Administrative Tools click **"Computer Management"**.

■ **Microsoft Windows Vista/7 users**

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

Step 2: In the **Start Search box type device manager** and then press enters.
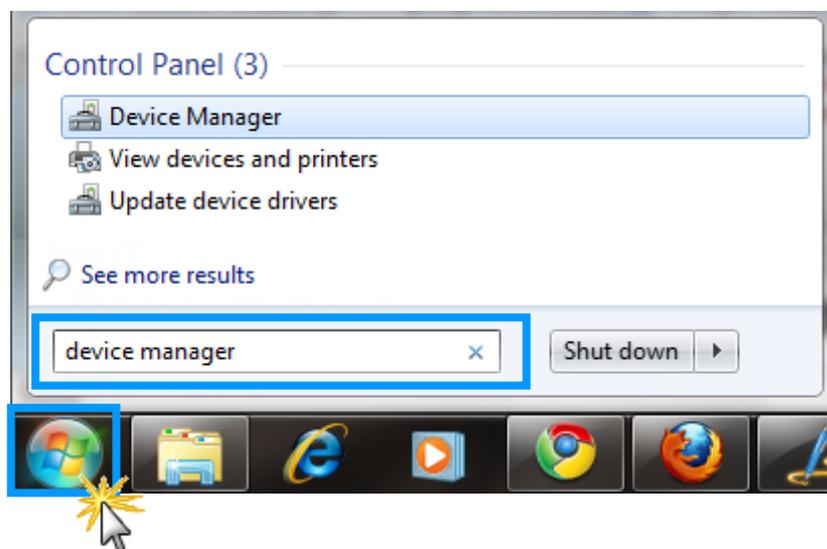
■ **Microsoft Windows 8 users**

Step 1: To show the **Start screen icon** from the desktop view, simply hover your cursor over the **bottom-left corner** of your screen. (Or using keyboard shortcuts, click **[Windows Key] +[X]** to open the Start Menu.)

Step 2: **Right-click** on the Start screen icon then click on **"Device Manager"**.
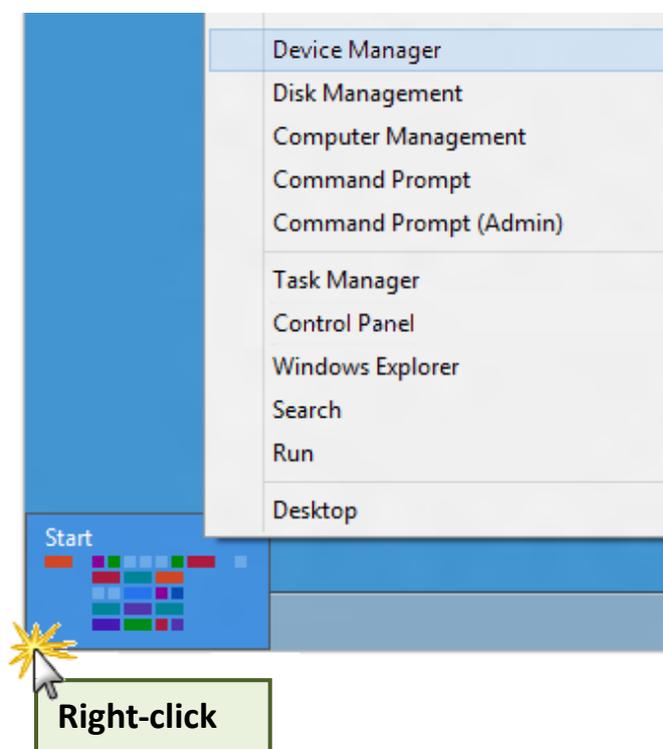
4.3.2 Check that the Installation

Check the PIO-821 series card which listed correctly or not, as illustrated below.
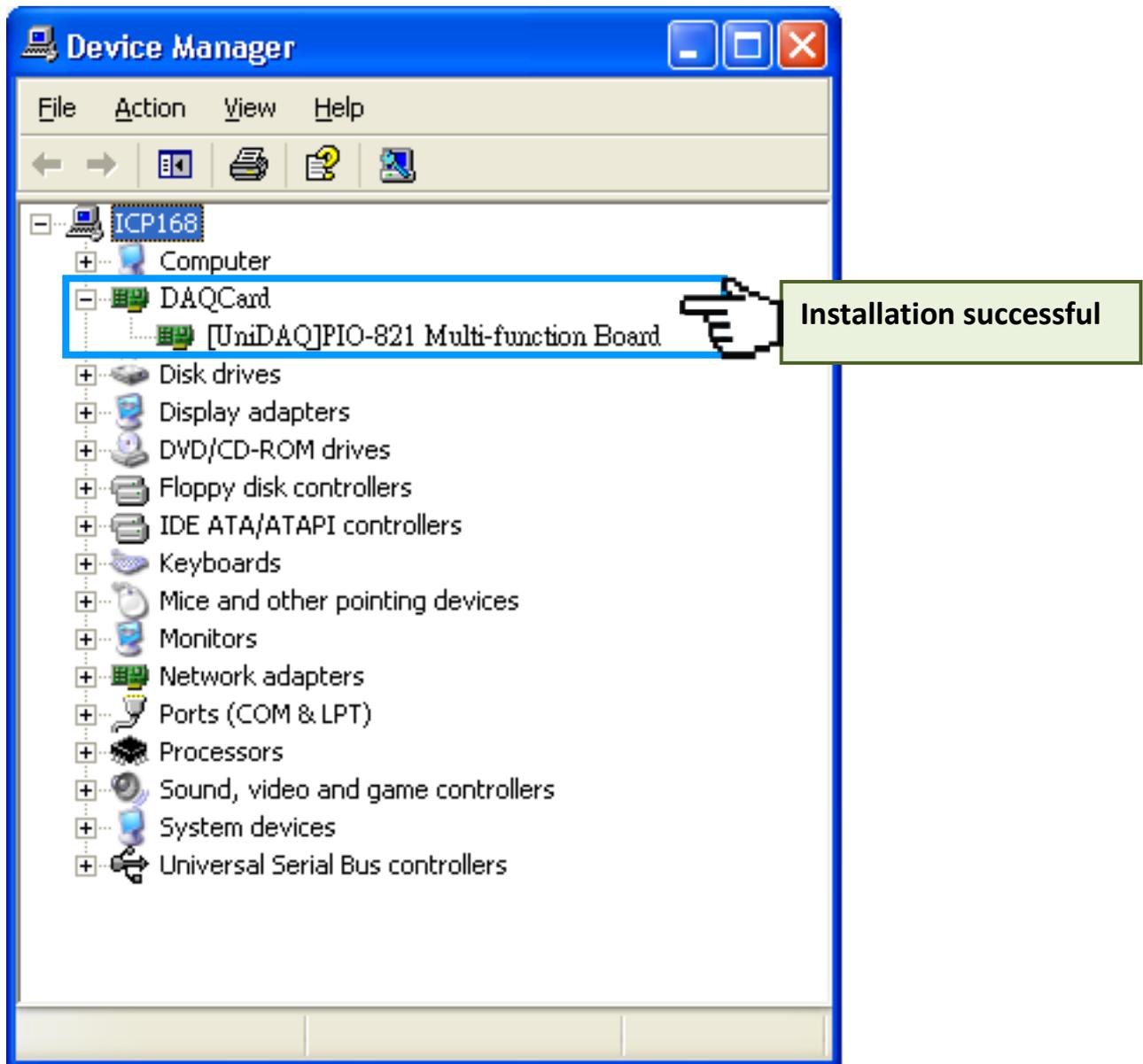
5. Testing PISO-821 Series Card

This chapter can give you the detail steps about self-test. In this way, user can confirm that PIO-821 series card well or not. Before the self-test, you must complete the hardware and driver installation. For detailed information about the hardware and driver installation, please refer to Chapter 3 Hardware Installation and Chapter 4 Software Installation.

5.1 Self-Test Wiring

5.1.1 DIO Test Wiring

Use the CA-2002 cable (optional) to connect the CON1 with CON2 on the PIO-821 series card.
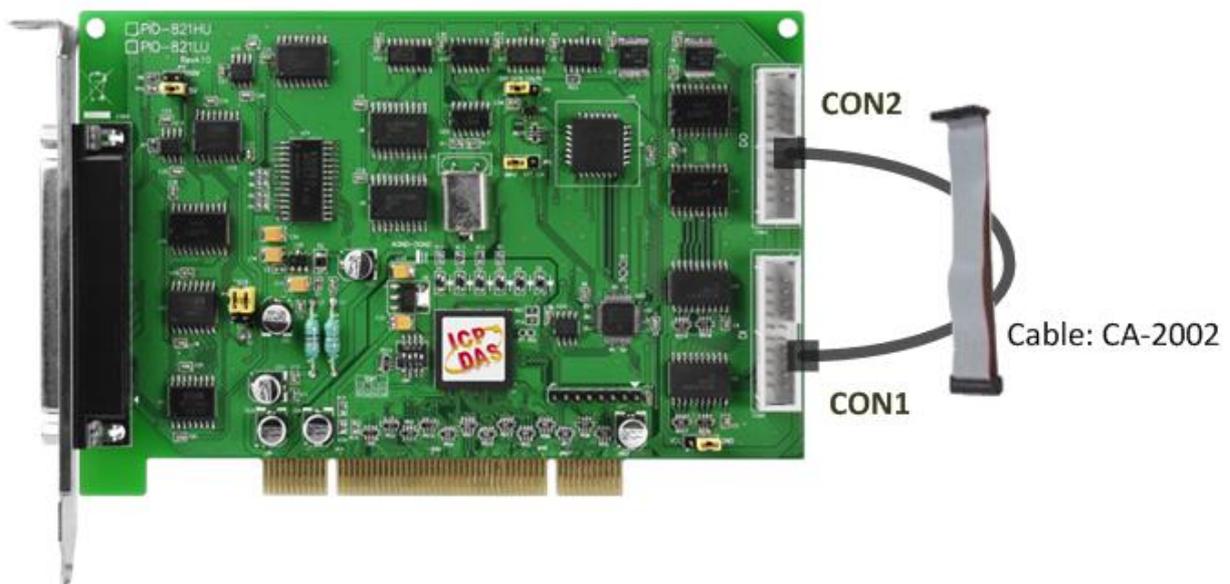
5.1.2 Analog Input Test Wiring

■ Prepare for device:

☑ DN-37 (optional) wiring terminal board.

☑ Provide a stable signal source. (For example, dry battery)

Step 1: Use the **DN-37 to connect the CON3** on the PIO-821 series card.

Step 2: Wire the **signal source to A/D channel0**, and then keep set the **JP4 jumper to Single-Ended** (default settings, refer to Sec. 2.4.3 for more detailed), and wire the signals as follows:

Connect the **AI 0 (Pin01) to signal positive (+)** and then **A.GND (Pin09)** to **signal negative (-)**.
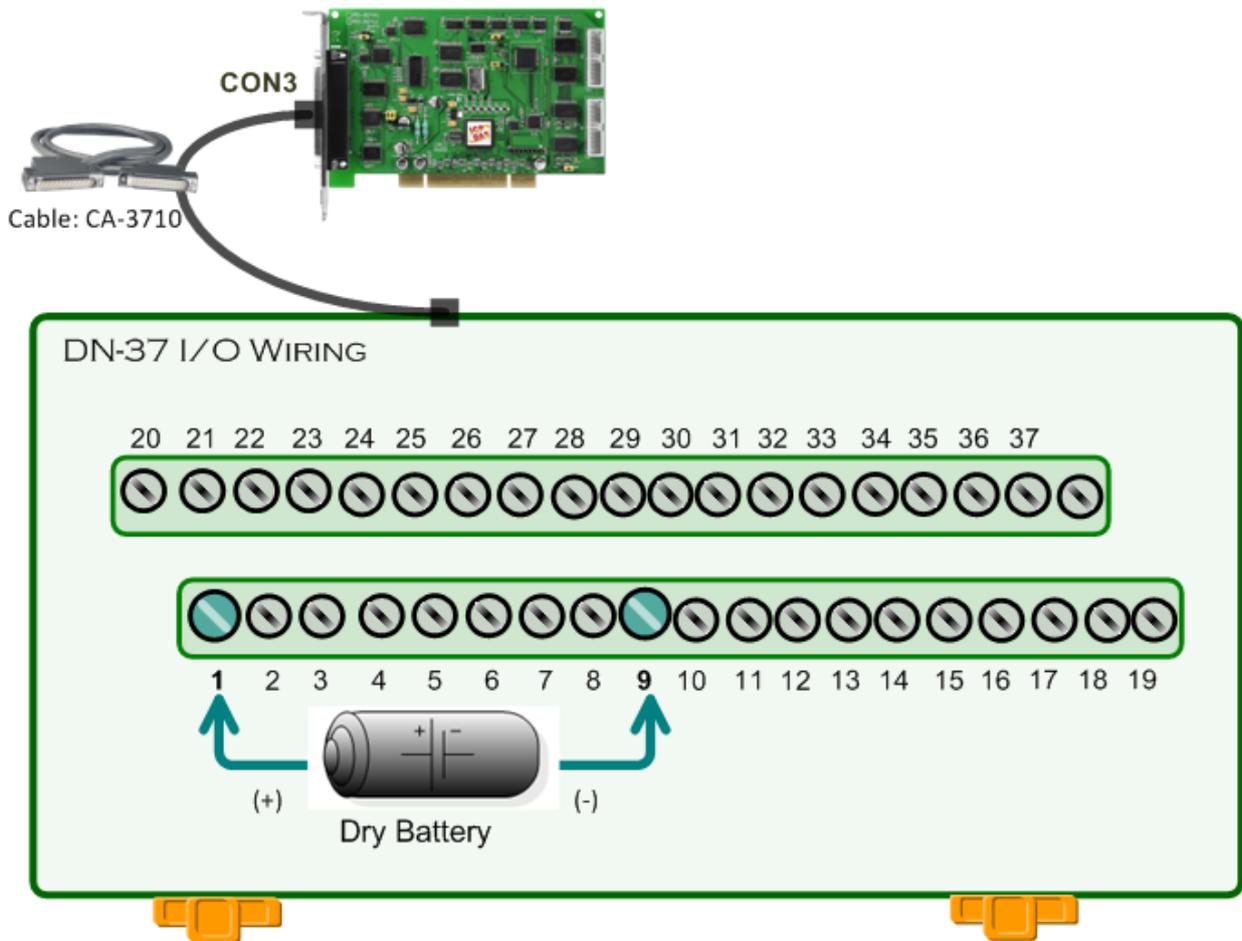
5.1.3 Analog Output Test Wiring

■ Prepare for device:

☑ DN-37 (optional) wiring terminal board.

☑ Digital Multi-Meter.

Step 1: Use the **DN-37 to connect the CON3** on the PIO-821 series card.

Step 2: Wire the **signal source to D/A channel**, and then keep set the **JP2 jumper to 5 V** (default settings, refer to Sec. 2.4.1 for more detailed), and wire the signals as follows:

Connect the **positive probe (+) of Multi-meter to DAOUT (Pin 30)**, and then the **negative probe (-) of Multi-meter to A.GND (Pin 29)**.
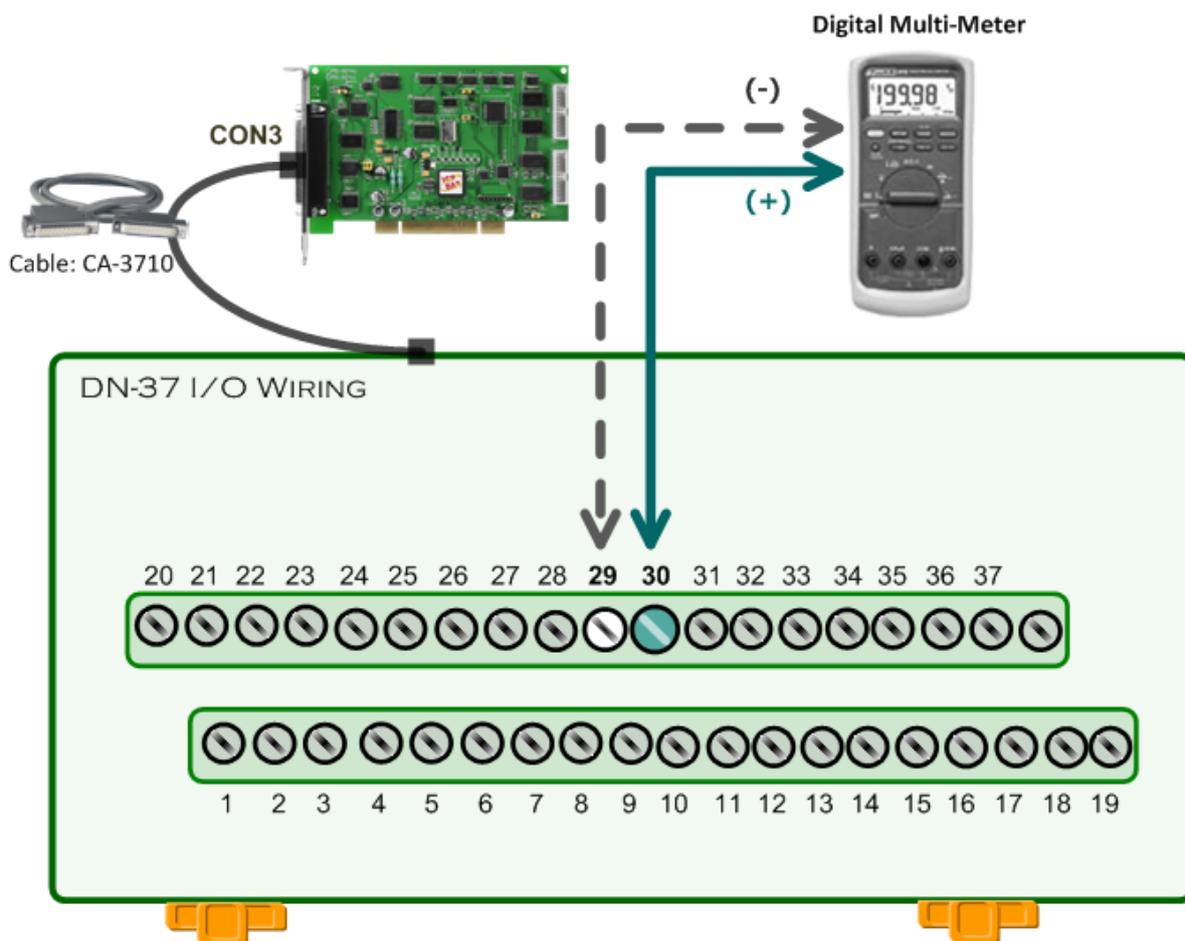
5.2   Execute the Test Program

The following example use UniDAQ driver to perform self-test. If you install the PIO-821 series classic driver, please refer to Quick Start Guide of the PIO-821 (http://ftp.icpdas.com/pub/cd/iocard/pci/napdos/pci/pio-821/manual/quickstart/classic/pio-821_quickstart_eng_v11.pdf) to execute the self-test.

Step 1: Execute the UniDAQ Utility Program. The UniDAQ Utility.exe will be placed in the default path (C:\ICPDAS\UniDAQ\Driver\) after completing installation.

1. Double click the **"UniDAQUtility.exe"**

2. Confirm the PIO-821 series card had successfully installed to PC. It starts form 0.

3. Click the **"TEST"** button to start test.
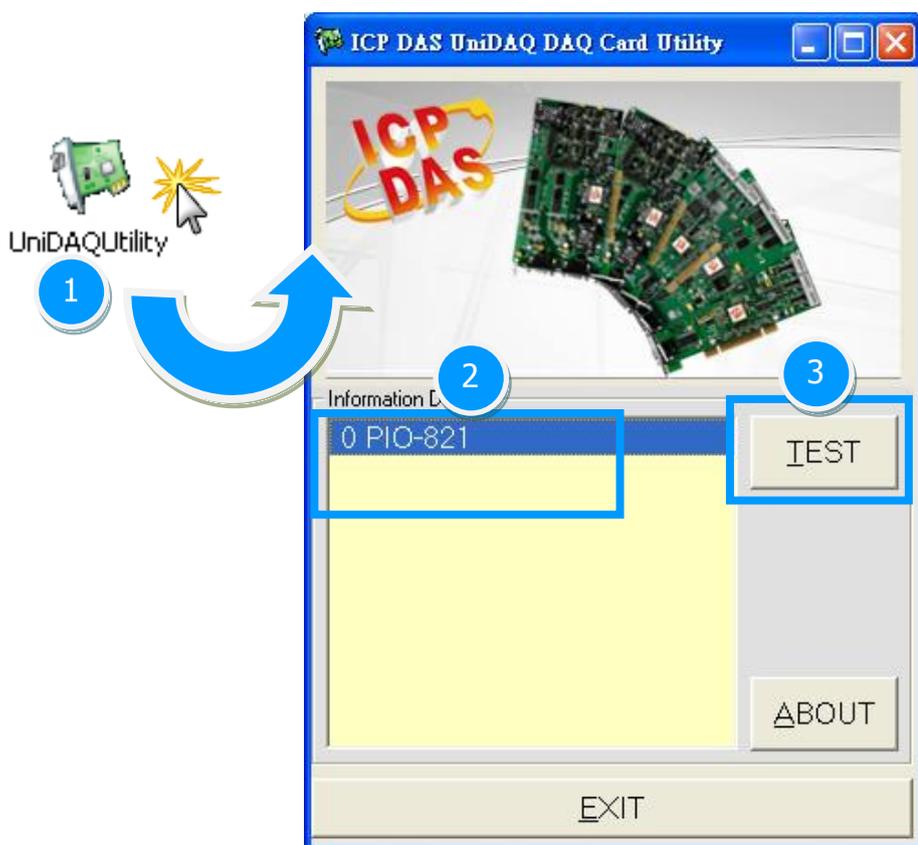
Step 2: Get **DIO function** test result.

Click "Digital Output" item.

④

0 PIO-821 (CARD ID:F)

| Analog Input | Analog Output | Digital Input | **Digital Output** | Timer/Counter | Debug |

7 6 5 4   3 2 1 0

ON(1)

OFF(0)

⑥ Check channel 0, 2, 4, 6

Select the "Port 0"

⑤

Port Number [0 ▾] HEX [55]

EXIT

Click "Digital Input" item.

⑦

0 PIO-821 (CARD ID

| Analog Input | Analog Output | **Digital Input** | Digital Output | Timer/Counter | Debug |

7 6 5 4 3 2 1 0

● ON(1)

● OFF(0)

⑨ The corresponding D/I becomes red for channel 0, 2, 4, 6 of D/O is ON.

Select the "Port 0"

⑧

Port Number [0 ▾] HEX [55]

EXIT

Step 3: Get <u>**Analog Input function**</u> test result.

10 Click "Analog Input" item.

11 Confirm the configuration setting.

12 Click this button to start test.

13 Check analog input on Channel 0 textbox.
The other channels value for floating number.
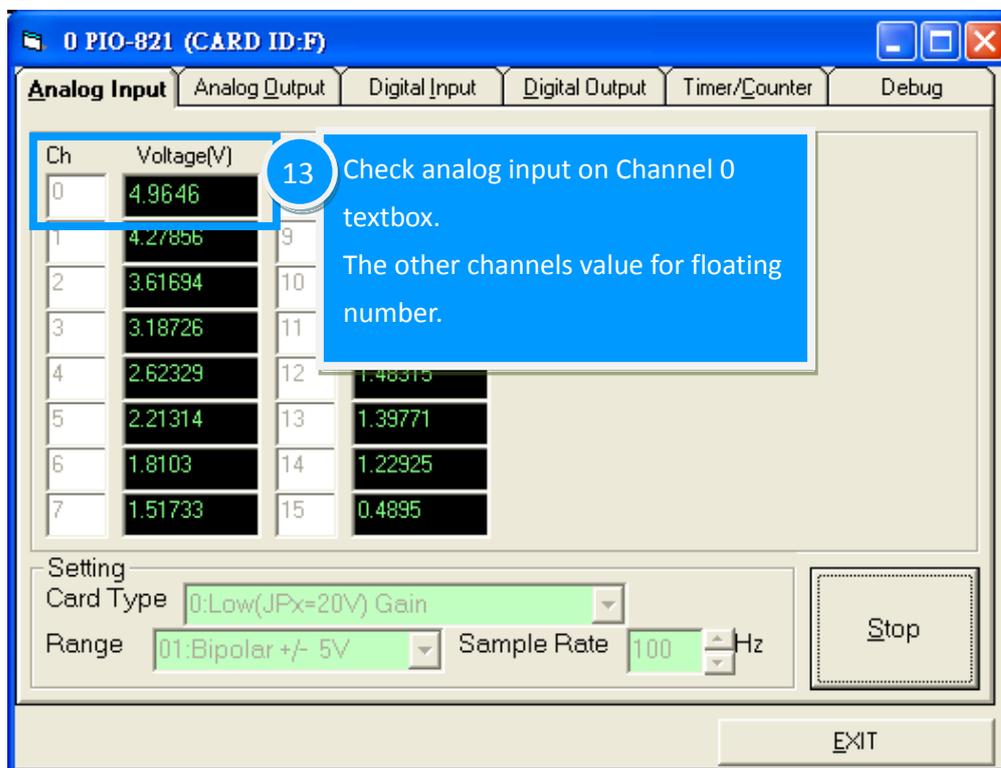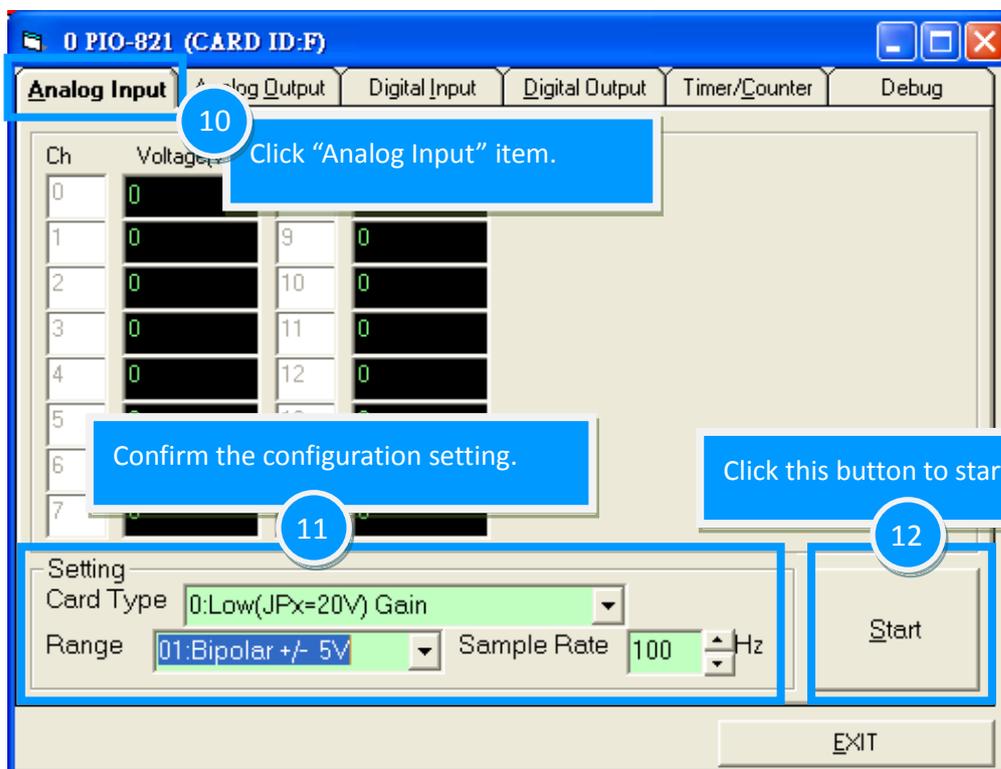
Step 4: Get <u>**Analog Output function**</u> test result.

Click "Analog Output" item.

14

Click this button.

18

15   **16**   **17**

Select the "Channel 0"

Select the "0 ~ 5V"

Type the voltage value

The value read on meter may be a little difference from the DA value because of the resolution limit of meter or the measurement error.

Complete

2.499

19

Check the value on **multi-meter**, they should be identical to the values set in program.

6. Calibration

The PIO-821 is already fully calibrated when shipped from the factory including the calibration coefficients which are stored in the EEPROM on board. For more precise application of voltages at the "system end", the following procedure provides a method that allows you to calibrate the board within your system, so that you can achieve the correct voltages at your field connection. This calibration allows the user to remove the effects of voltage drops caused by IR loss in the cable and/or connector.

At first the user has to prepare the equipment for calibration: the precise multi-meter. Note that the calibrated values for analog output and analog input channels are stored within 3 words in the address of the EEPROM, as show in Table6-1. The calibration procedure will be demonstrated below:

Table 6-1: Calibration values stored in the EEPROM address

	The address of the EEPROM for Analog output	The address of the EEPROM for Analog input
5 V	0	
10 V	1	
0 V		2

The **calibration.exe program** for the ICP DAS PIO-821 series can be found on the supplied CD-ROM, or can be obtained from the ICP DAS FTP web site. The location and addresses are indicated in the table 6-2 below: (Note: Before beginning the **"A/D and D/A Calibration"** process, the **PIO-821 series classic driver** installation must be completed. For detailed information regarding PIO-821 series classic driver installation, refer to section 4.1.2.)

Table 6.2

	CD:\\ NAPDOS\PCI\PIO-821\DLL\Calibrate\
	http://ftp.icpdas.com/pub/cd/iocard/pci/napdos/pci/pio-821/dll/calibrate/

6.1   D/A Calibration

Figure 6-1
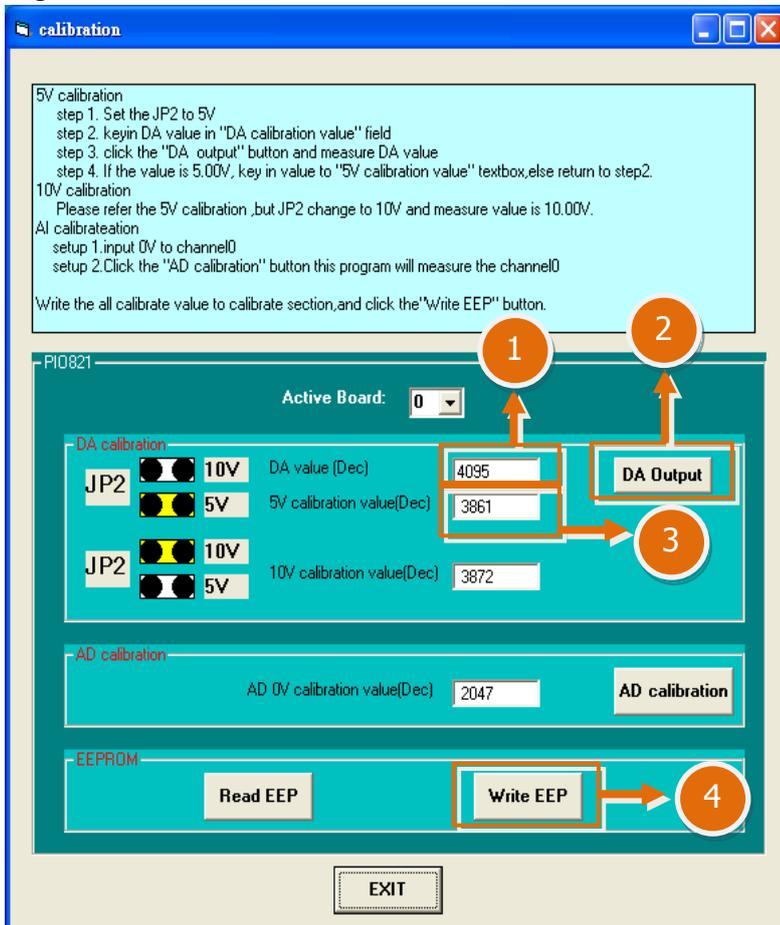
■   **Calibration for 5 V mode**

Step 1: Set the **JP2** jumper on PIO-821 to the **5V** position (refer to Sec. 2.4.1).

Step 2: Run the calibration tool, which is located in **"/program files/ DAQpro/PIO-821/calibation.exe"** in order to open the configuration interface, as shown in Figure 6-1.

Step 3: If you want to calibrate the DA, for example, then let the **"DA value"** be set at 4095, as shown in Figure 6-1 **(1)**.

Step 4: Click the **"DA Output"** button and then use particular multi-meter to measure the analog output, as shown in Figure 6-1 **(2)**.

Step 5: If the analog output is smaller or bigger than the allowed maximum (5.00 V) value of analog output, then go to step 3 to step 4 to change the output value (for example: 4093, 4092,…) and change the output voltage until its value is equal to the allowed maximum voltage output value, as shown in Figure 6-1 **(3)**.

Step 6: If the analog output is equal to the maximum allowed analog output level, please key in the calibration value to **"5 V calibration value"** in the textbox and Click the **"Write EEP"** button. The DA calibration process is finished for the setting range of analog output.   It is shown in Figure 6-1 **(4)**.

■   **Calibration for 10 V mode**

Please refer to the **Calibration for 5 V mode**, but the **JP2 jumper** has to be configured at the **10 V** output range and the allowance set to the **maximum value at 10.00 V**.

6.2   A/D Calibration

Step 1: Set the **JP4** jumper on PIO-821 to the **differential input** position (refer to Sec. 2.4.3) and connect the AI0+ (CON3, pin1) and AI0- (CON3, pin20) of analog input channel 0 together, as shown in the following Figure6-2.
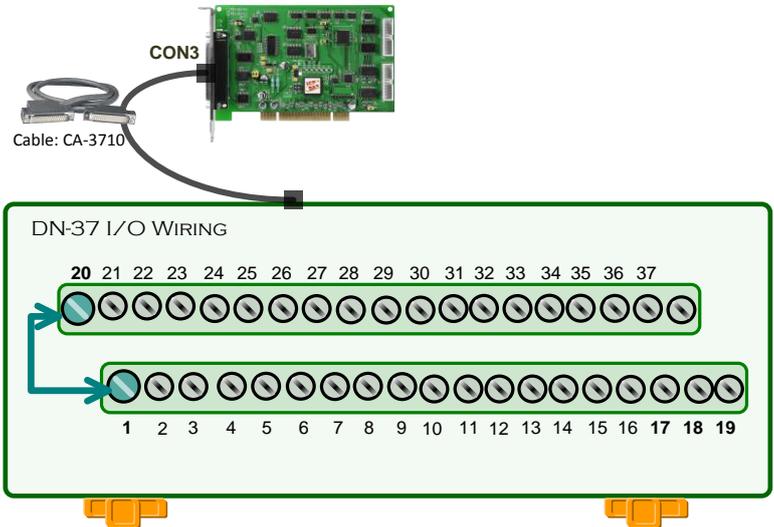
Figure 6-2
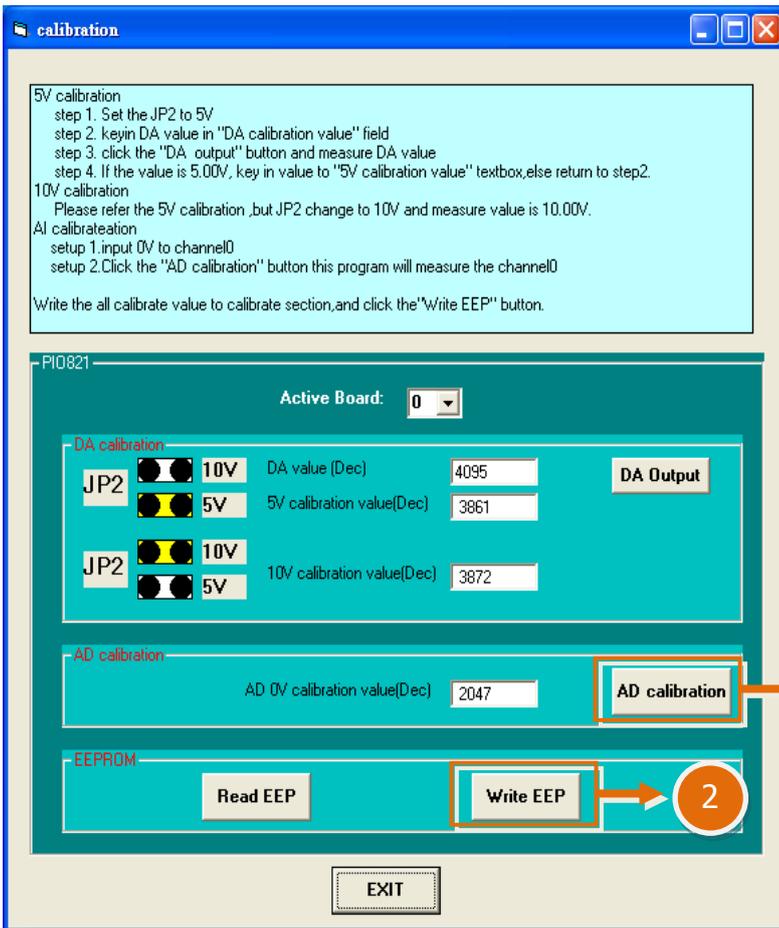
Step 2: Click the **"AD calibration"** button to write the value of the analog input channel 0 into the text field of the **"AD calibration value"** , as shown in Figure 6-3 **(1)**.

Step 3: Click the **"Write EEP"** button to save the calibration data into the EEPROM to finish the calibration process, as shown in Figure 6-3 **(2)**.

Figure 6-3

7.   I/O Control Register

7.1   How to Find the I/O Address

The plug&play BIOS will assign a proper I/O address to every PIO/PISO series card in the power-on stage. The fixed IDs for the PIO-821 series cards are given as follows:

Table 7-1:

PIO-821L/H (Rev 1.0)		PIO-821L/H (Rev 2.0 or above) PIO-821LU/HU	
Vendor ID	0xE159	Vendor ID	0xE159
Device ID	0x02	Device ID	0x01
Sub-Vendor ID	0x80	Sub-Vendor ID	0x00FF 0x40FF
Sub-Device ID	0x03	Sub-Device ID	0x03
Sub-Aux ID	0x00	Sub-Aux ID	0x00

We provide all necessary functions as follows:
1. **PIO_DriverInit(&wBoard, wSubVendor, wSubDevice, wSubAux)**
2. **PIO_GetConfigAddressSpace(wBoardNo,*wBase,*wIrq, *wSubVendor,
   *wSubDevice, *wSubAux, *wSlotBus, *wSlotDevice)**
3. **Show_PIO_PISO(wSubVendor, wSubDevice, wSubAux)**

All functions are defined in PIO.H. Refer to **PIO-821 DLL software manual** for more information. The important driver information is given as follows:

1. Resource-allocated information:
- **wBase:** BASE address mapping in this PC
- **wIrq:** IRQ channel number allocated in this PC

2. PIO/PISO identification information:

- **wSubVendor:** subVendor ID of this board

- **wSubDevice:** subDevice ID of this board

- **wSubAux:** subAux ID of this board

3. PC's physical slot information:

- **wSlotBus:** hardware slot ID1 in this PC's slot position

- **wSlotDevice:** hardware slot ID2 in this PC's slot position

The PIO_PISO.EXE utility will detect and show all PIO/PISO cards installed in this PC. Refer to Sec. 7.1.1 for more information.

7.1.1 PIO_PISO Utility

The PIO_PISO.EXE is valid for all PIO/PISO cards. This program shows all PCI hardware ID regarding the PIO and PISO series DAQ cards. It is useful to test if the card Plug & Play successfully when the computer bootup. If the PIO or PISO series card does not shown in the screen correctly, please try to use another PCI slot and try again.

The user can execute the PIO_PISO.EXE to get the following information:

- List all PIO/PISO cards installed in this PC

- List all resources allocated to every PIO/PISO cards

- List the wSlotBus and wSlotDevice for specified PIO/PISO card identification. (refer to Sec. 7.2 for more information about the assignment of I/O Address)

■ **For Windows OS**

The **PIO_PISO.EXE for Windows** is contained in:

CD:\NAPDOS\PCI\Utility\Win32\PIO_PISO

http://ftp.icpdas.com/pub/cd/iocard/pci/napdos/pci/utility/win32/pio_piso/

After executing the utility, the detail information for all PIO/PISO cards that installed in the PC will be shown as follows:
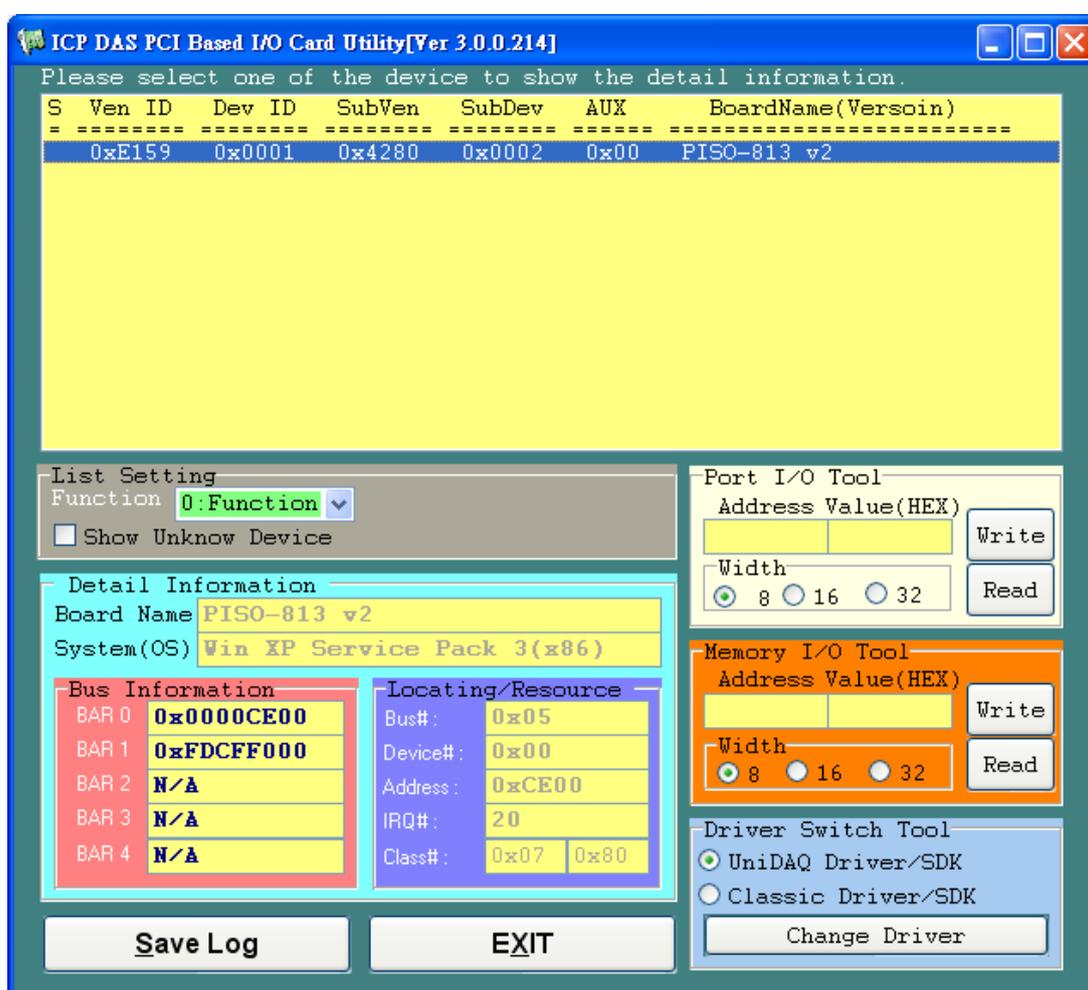
■ **For DOS**

The **PIO_PISO.EXE for DOS** is contained in:

CD:\NAPDOS\PCI\Utility\DOS

http://ftp.icpdas.com/pub/cd/iocard/pci/napdos/pci/utility/dos/

The PIO_PISO program source is given as follows:

```
/* -------------------------------------------------------- */
/* Find all PIO_PISO series cards in this PC system */
/* step 1 : plug all PIO_PISO cards into PC            */
/* step 2 : run PIO_PISO.EXE                           */
/* -------------------------------------------------------- */

#include "PIO.H"

WORD wBase,wIrq;
WORD wBase2,wIrq2;

int main()
{
int    i,j,j1,j2,j3,j4,k,jj,dd,j11,j22,j33,j44;
WORD wBoards,wRetVal;
WORD wSubVendor,wSubDevice,wSubAux,wSlotBus,wSlotDevice;
char c;
float ok,err;

clrscr();
wRetVal=PIO_DriverInit(&wBoards,0xff,0xff,0xff); /*for PIO-PISO */
printf("\nThrer are %d PIO_PISO Cards in this PC",wBoards);
if (wBoards==0 ) exit(0);

printf("\n-------------------------------------------------");
for(i=0; i<wBoards; i++)
    {
    PIO_GetConfigAddressSpace(i,&wBase,&wIrq,&wSubVendor,
                &wSubDevice,&wSubAux,&wSlotBus,&wSlotDevice);

    printf("\nCard_%d:wBase=%x,wIrq=%x,subID=[%x,%x,%x],
                SlotID=[%x,%x]",i,wBase,wIrq,wSubVendor,wSubDevice,
                wSubAux,wSlotBus,wSlotDevice);
    printf(" --> ");
    ShowPioPiso(wSubVendor,wSubDevice,wSubAux);
    }

PIO_DriverClose();
}
```

7.2   The Assignment of I/O Address

The Plug&Play BIOS will assign the proper I/O address to a PIO/PISO series card. If there is only one PIO/PISO board, the user can identify the board as card_0. If there are two PIO/PISO boards in the system, it is very difficult to identify which board is card_0. The software driver can support a maximum of 16 boards. Therefore, the user can install 16 PIO/PSIO series cards onto one PC system. The methods used to find and identify card_0 and card_1 is demonstrated below.

The simplest way to identify which card is card_0 is to use wSlotBus and wSlotDevice in the following manner:

> **Step 1:** Remove all PIO-821 series boards from the PC.
>
> **Step 2:** Install one PIO-821 series into the PC's PCI_slot1, run PIO_PISO.EXE.
>   Then record the wSlotBus1 and wSlotDevice1 information.
>
> **Step 3:** Remove all PIO-821 series boards from the PC.
>
> **Step 4:** Install one PIO-821 series into the PC's PCI_slot2 and run PIO_PISO.EXE.
>   Then record the wSlotBus2 and wSlotDevice2 information.
>
> **Step 5:** Repeat Steps(3) and (4) for every PCI_slot and record all information from wSlotBus and wSlotDevice.

The records may look similar to the table follows:
Table 7-2:

PC's PCI Slot	WslotBus	WslotDevice
Slot_1	0	0x07
Slot_2	0	0x08
Slot_3	0	0x09
Slot_4	0	0x0A
PCI-BRIDGE		
Slot_5	1	0x0A
Slot_6	1	0x08
Slot_7	1	0x09
Slot_8	1	0x07

The above procedure will record all the wSlotBus and wSlotDevice information on a PC. These values will be mapped to this PC's physical slot and this mapping will not be changed for any PIO/PISO cards. Therefore, this information can be used to identify the specified PIO/PISO card by following steps:

Step1: Using the wSlotBus and wSlotDevice information from Table 7-2.

Step2: Enter the board number into PIO_GetConfigAddressSpace(…) function to get the information for a specific card, especially the wSlotBus and wSlotDevice details.

Step3: Identify the specific PIO/PISO card by comparing the data of the wSlotBus and wSlotDevice from Step1 and Step2.

Note: that normally the card installed in slot 0 is card0 and the card installed in slot1 is card1 for PIO/PISO series cards.

7.3 The I/O Address Map

The I/O address for PIO-821 series cards are automatically assigned by the main board ROM BIOS. The I/O address can 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-821 series card. The I/O address for the PIO-821 series card are given in the table below, all of which are based on the base address of each card.

Table 6-3: Refer to Sec. 7.1 for more information about wBase.

Address	Read	Write
wBase+0x0	RESET\ Control Register	RESET\ Control Register
wBase+0x2	AUX Control Register	AUX Control Register
wBase+0x3	AUX Data Register	AUX Data Register
wBase+0x5	INT Mask Control Register	INT Mask Control Register
wBase+0x7	AUX Pin Status Register	AUX Pin Status Register
wBase+0x2a	INT Polarity Control Register	INT Polarity Control Register
wBase+0xc0	8254-Counter0	8254-Counter0
wBase+0xc4	8254-Counter1	8254-Counter1
wBase+0xc8	8254-Counter2	8254-Counter2
wBase+0xcc	-	8254 Control Word
wBase+0xd0	A/D Low Byte	D/A Low Byte
wBase+0xd4	A/D High Byte	D/A High Byte
wBase+0xd8	DI Low Byte	DO Low Byte
wBase+0xdc	DI High Byte	DO High Byte
wBase+0xe0	-	A/D Gain Control and Multiplexer Control
wBase+0xe4	-	A/D Mode Control
wBase+0xe8	-	A/D Software Trigger Control
wBase+0xec	A/D Status	-
wBase+0xf0	Read Card ID	-

7.3.1 RESET\ Control Register

(Read/Write): wBase+0x0

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/A command applications are initiated.

7.3.2 AUX Control Register

(Read/Write): wBase+0x2

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

This register is designed for feature extension and for enable or disable of the reading/writing data from or to the EEPROM. And it is reversed for internal utilization and do not apply this control register under any consideration.

7.3.3 AUX Data Register

(Read/Write): wBase+0x3

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

This register controls the read/write function of the EEPROM on board. There are all reversed by ICPDAS internal use. If the user wants to access this EEPROM, please refer to the function read/write of the EEPROM provided by the driver toolkit.

7.3.4   INT Mask Control Register

(Read/Write): wBase+0x5

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

The INT mask control register is presented as following table. The detail function for these control register is described as below.

INT0 : ADC ready

INT1 : Timer 0

EN0=0 →   disable INT0 to be an interrupt signal (default)

EN0=1 →   enable INT0 to be an interrupt signal

EN1=0 →   disable INT1 to be an interrupt signal (default)

EN1=1 →   enable INT1 to be an interrupt signal

The following is the partial programs for DOS C development environment enable or disable interrupt function.   For more information, please refer to the DOS demo program demo1.c.

```
For example:
 outportb(wBase+5,0);                // disable all interrupts
 outportb(wBase+5,1);                // enable interrupt of INT0
 outportb(wBase+5,2);                // enable interrupt of INT1
 outportb(wBase+5,3);                // enable all two channels of interrupt
```

7.3.5   AUX Status Register

(Read/Write): wBase+0x7

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

Based on the auxiliary status register, Aux0 (bit 0) and Aux 1(bit 1) stand as INT0 and INT1 signal respectively. Aux2 to Aux3 (bit2 to 3) represents the control register of the EEPROM and Aux4 to Aux7 (bit4 to 7) depicts the Aux-ID. Generally, the Aux 0 to 1 are used as interrupt sources. Interrupt service has to check this register to start service routing.

7.3.6   Interrupt Polarity Control 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	-	-	INV1	INV0

The interrupt polarity control register is presented as following table. It is used to invert the interrupt signal or not.   The detail function for these control register is described as below.

INT0 : ADC ready

INT1 : Timer 0

INV0=0 →   invert signal from INT0;
INV0=1 →   do not invert signal from INT0;
INV1=0 →   invert signal from INT1;
INV1=1 →   do not invert signal from INT0;

The following is the partial programs for DOS C development environment enable or disable inverting function for interrupt signal.

```
For example:
outportb(wBase+0x2a,0);  /* select the inverted input from all 2 channels     */
outportb(wBase+0x2a,3);  /* select the non-inverted input from all 2 channels */
outportb(wBase+0x2a,2);  /* select the inverted input of INT0               */
                         /* select the non-inverted input from the others    */
```

7.3.7   8254 Counter

Address	Read	Write
wBase+0xc0	8254-Counter0	8254-Counter0
wBase+0xc4	8254-Counter1	8254-Counter1
wBase+0xc8	8254-Counter2	8254-Counter2
wBase+0xcc	-	8254 Control Word

The 8254 Programmable timer/counter has 4 registers from **wBase+0xc0 through wBase+0xcc**. For detailed programming information about 8254, please refer to **Intel's "Microsystem Components Handbook"**.

7.3.8   A/D Buffer Register

(Read): wBase+0xd0   →A/D Low Byte Data Format

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

(Read): wBase+0xd4   →A/D High Byte Data Format

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	D11	D10	D9	D8

(Read): wBase+0xec   →Read A/D Status

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
-	-	-	-	-	-	-	READY

A/D 12 bits data: D11 to D0, D11=MSB, D0=LSB

READY   = 0: A/D 12 bits data not ready

= 1: A/D 12 bits data is ready

The low 8 bits A/D data are stored in address wBase+0xd0 and the high 4 bits data are stored in address wBase+0xd4. The READY bit is used as an indicator for A/D conversion. When an A/D conversion is completed, the READY bit will be set to 1.

7.3.9   D/A Buffer Register

(Write): wBase+0xd0   →D/A Low Byte Data Format

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

(Write): wBase+0xd4   →D/A High Byte Data Format

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	D11	D10	D9	D8

The D/A converter will convert the 12 bits digital data to analog output. The low 8 bits of D/A channel are stored in address wBase +4 and high 4 bits are stored in address wBase +5. The D/A output latch registers are designed as a "double buffered" structure, so the analog output latch registers will be updated until the high 4 bits digital data are written. The user must send low 8 bits first and then send high 4 bits to update the 12 bits AD output latch register.

7.3.10   D/I Input Buffer Register

Read): wBase+0xd8   →D/I Low Byte Data Format

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

(Read): wBase+0xdc   →D/I High Byte Data Format

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
D15	D14	D13	D12	D11	D10	D9	D8

DI 16 bits input data: D15 to D0, D15=MSB, D0=LSB

The PIO-821 series card provides 16 TTL compatible digital inputs. The low 8 bits are stored in address wBase +0xd8. The high 8 bits are stored in address wBase +0xdc.

7.3.11 D/O Output Buffer Register

(Write): wBase+0xd8 →D/O Low Byte Data Format

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

(Write): wBase+0xdc →D/O High Byte Data Format

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
D15	D14	D13	D12	D11	D10	D9	D8

The PIO-821L/H provides 16 TTL compatible digital outputs. The low 8 bits are stored in address wBase+0xd8. The high 8 bits are stored in address wBase+0xdc.

7.3.12 A/D Gain Control and Multiplex Control Register

(Write): wBase+0xe0 →D/O Low Byte Data Format

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
-	-	MUX3	MUX2	MUX1	MUX0	GAIN1	GAIN0

PIO-821L/LU GAINS Control Code Table:

GAIN	Input Range	GAIN1	GAIN0	Setting Time
1	±5 V	0	0	23 µs
2	±2.5 V	0	1	23 µs
4	±1.25 V	1	0	25 µs
8	±0.6255 V	1	1	28 µs

PIO-821H/HU GAINS Control Code Table:

GAIN	Input Range	GAIN1	GAIN0	Setting Time
1	±5 V	0	0	23 µs
10	±2.5 V	0	1	28 µs
100	±1.25 V	1	0	140 µs
1000	±0.6255 V	1	1	1300 µs

The only difference between PIO-821L/LU and PIO-821H/HU is the GAIN control function. The PIO-821L/LU provides gain factor of 1, 2, 4, 8 and PIO-821H/HU provides 1, 10, 100, 1000. The gain control registers control the gain of A/D input signal.

Note:

If gain control code changed, the hardware needs to delay extra gain settling time. The gain settling time is different for different gain control code. The software driver does not take care of the gain settling time, so the user needs to delay the gain settling time if gain changed.

A/D input channel selection data = 4 bits: MUX3 to MUX0, MUX3=MSB, D0= MUX0,
" - "=don't care
Single-ended mode: MUX3... MUX0
Differential mode: MUX2... MUX0, MUX3 =don't care

7.3.13   A/D Mode Control Register

(Write): wBase+0xe4

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	1	1	1	0	ExtTrig	PacerTrig	SoftTrig

The PIO-821 series card provides software trigger, pacer trigger and external trigger modes to get A/D input signal.

Bit 2 to Bit 0 =0   : reset the AD trigger mode
SoftTrig = 1   : Enable Software Trigger Mode
PacerTrig = 1   : Enable Pacer Trigger Mode
ExtTrig = 1   : Enable External Trigger Mode

The following is the partial programs for DOS C development environment to set A/D mode:

```
For example:
outportb(wBase+0xe4, 0xF1);        // enable software trigger mode
outportb(wBase+0xe4, 0xF2);        // enable pacer trigger mode
outportb(wBase+0xe4, 0xF4);        // enable external trigger mode
outportb(wBase+0xe4, 0xF0);        // reset the AD trigger mode
```

7.3.14   A/D Software Trigger Control Register

(Write): wBase+0xe8

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	0	0	0	A/D

The A/D converter will convert the analog input data by software control. After setting section 7.3.13 A/D Mode Control Register to software trigger mode, writing 1 to A/D software trigger control register will trigger A/D converter to complete an A/D conversion.

> For example:
> outportb(wBase+0xe8, 0x01);   // convert A/D data once

7.3.15   Card ID Register

(Read): wBase+0xf0

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	ID3	ID2	ID1	ID0

> For example:
> wCardID = inportb(wBase+0xf0);   /* read Card ID */

Note: The Card ID function is only supported by the PIO-821LU/HU (Ver. 1.0 or above)

8. Demo Programs

8.1 Demo Program for Windows

All demo programs will not work properly if the DLL driver has not been installed correctly. During the DLL driver installation process , the install-shields will register the correct kernel driver to the operation system and copy the DLL driver and demo programs to the correct position based on the driver software package you have selected (Win98/Me/NT/2K and 32-/64-bit winXP/2003/Vista/7/8). Once driver installation is complete, the related demo programs and development library and declaration header files for different development environments will be presented as follows.

■ **Demo Program for PIO-821 Series Classic Driver**

The demo program is contained in:

CD:\NAPDOS\PCI\PIO-821\DLL\Demo

http://ftp.icpdas.com/pub/cd/iocard/pci/napdos/pci/pio-821/dll/demo/

⊕ BCB4 → for Borland C++ Builder 4
PIO821.H → Header files
PIO821.LIB → Linkage library for BCB only

⊕ Delphi4 → for Delphi 4
PIO821.PAS → Declaration files

⊕ VC6 → for Visual C++ 6
PIO821.H → Header files
PIO821.LIB → Linkage library for VC only

⊕ VB6 → for Visual Basic 6
PIO821.BAS → Declaration files

⊕ VB.NET2005 → for VB.NET2005
PIO821.vb → Visual Basic Source files

⊕ CSharp2005 → for C#.NET2005
PIO821.cs → Visual C# Source files

For detailed information about the DLL function of the PIO-821 series card, please refer to PIO-821 DLL Software Manual (CD:\NAPDOS\PCI\PISO-821\Manual\)

■ **Demo Program for UniDAQ SDK Driver**

The demo program is contained in:

CD:\NAPDOS\PCI\UniDAQ\DLL\Demo

http://ftp.icpdas.com/pub/cd/iocard/pci/napdos/pci/unidaq/dll/demo/

⊕ BCB6 → for Borland C++ Builder 6
UniDAQ.H → Header files
UniDAQ.LIB → Linkage library for BCB only

⊕ Delphi6 → for Delphi 6
UniDAQ.PAS → Declaration files

⊕ VB6 → for Visual Basic 6
UniDAQ.BAS → Declaration files

⊕ CSharp2005 → for C#.NET2005
UniDAQ.cs → Visual C# Source files

⊕ VC6 → for Visual C++ 6
UniDAQ.H → Header files
UniDAQ.LIB → Linkage library for VC only

⊕ VB.NET2005 → for VB.NET2005
UniDAQ.vb → Visual Basic Source files

⊕ VC.NET2005 → for VC.NET2005 (32-bit)
UniDAQ.H → Header files
UniDAQ.LIB → Linkage library for VC only

⊕ VC.NET2005 → for VC.NET2005 (64-bit)
UniDAQ.H → Header files
UniDAQ.LIB → Linkage library for VC only

For detailed information about the DLL function and demo program of the UniDAQ, please refer to UniDAQ DLL Software Manual (CD:\NAPDOS\PCI\UniDAQ\Manual\)

8.2   Demo Program for DOS

The demo program is contained in:

CD:\NAPDOS\PCI\PIO-821\DOS

http://ftp.icpdas.com/pub/cd/iocard/pci/napdos/pci/pio-821/dos/

⊕   \TC*.*   → for Turbo C 2.xx or above

⊕   \TC\LIB*.*   → for TC Library
⊕   \TC\DEMO*.*   → for TC demo program
⊕   \TC\DIAG*.*   → for TC diagnostic program

⊕   \TC\LIB\PIO.H   → TC Declaration File
⊕   \TC\LIB\TCPIO_L.LIB   → TC Large Model Library File
⊕   \TC\LIB\TCPIO_H.LIB   → TC Huge Model Library File

For detailed information about the DLL function of the DOS, please refer to PIO-821 DLL Software Manual (CD:\NAPDOS\PCI\PIO-821\Manual\)

Appendix: Daughter Board

A1. DB-37 and DN-37

■ **DB-37:** The DB-37 is a general purpose daughter board for D-sub 37 pins. It is designed for easy wire connection via pin-to-pin. Use a 37-pin cable (e.g. CA-3710, etc.) to connect DB-37 to CON1 of the PIO-821 series card.
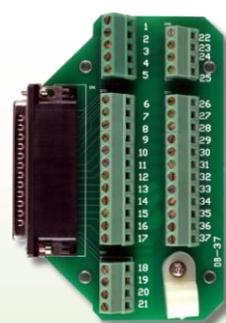
DB-37

■ **DN-37:** The DN-37 is a general purpose daughter board for DB-37 pins with DIN-Rail Mountings. They are also designed for easy wire connection via pin-to-pin.
Use a 37-pin cable (e.g. CA-3710, etc.) to connect to CON1 of the PISO-813 series card by DN-37.

DN-37

A2. DB-8225

The DB-8225 provides an onboard CJC (Cold Junction Compensation) circuit for thermocouple measurement, and terminal block for easy signal connection and measurement. The CJC is connected to A/D channel_0 and CON3 on the PIO-821 can be directly connected to the DB-8225 via a 37-pin D-Sub connector. Refer to the "DB-8225 User Manual" for details.
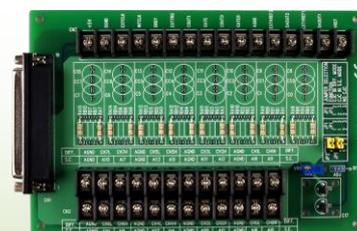
DB-8225

A3. DB-16P Isolated Input Board

The DB-16P is a 16-channel isolated digital input daughter board. The optically isolated inputs of the DB-16P are consisted of are bi-directional optocoupler with resistor for current sensing. You can use the DB-16P to sense DC signal from TTL levels up to 24 V or use the DB-16P to sense a wide range of AC signals. You can use this board to isolate the computer from large common-mode voltage, ground loops and transient voltage spike that often occur in industrial environments.
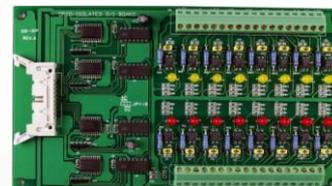
DB-16P
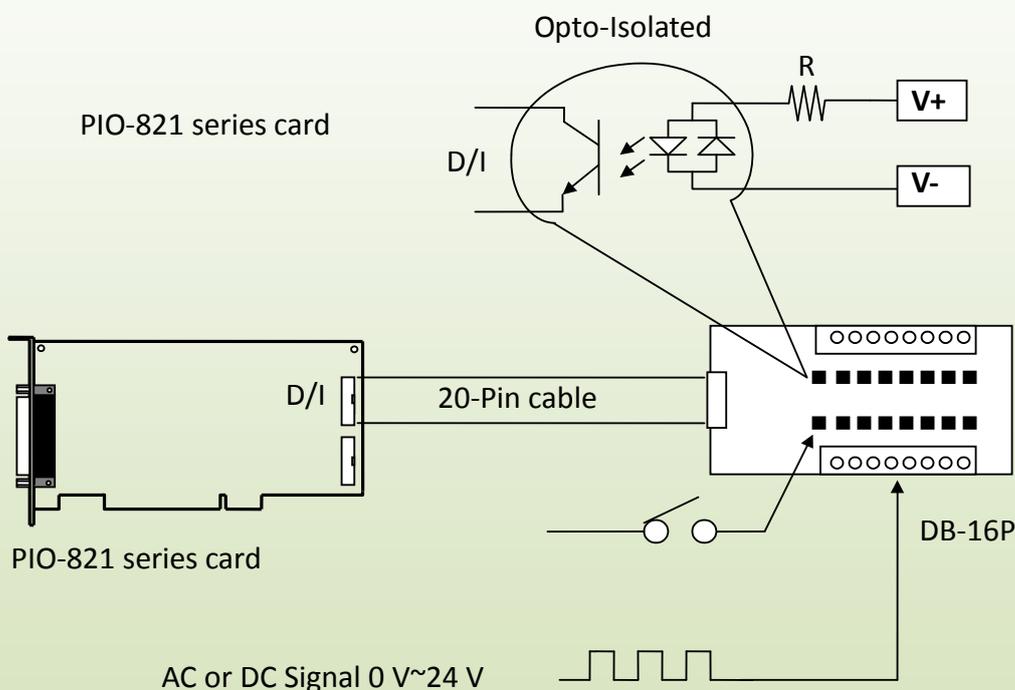
A4. DB-16R Relay Board

The DB-16R is a 16-channel relay output board consisting of 16 Form C relays that enable efficient switching of a load using programmable control. It is both a connector and functionally is compatible with 785 series boards, but with an industrial type terminal block. The relay is powered by applying a 5 V signal to the appropriate relay channel on the 20-pin flat connector. There are 16 LEDs for each relay, which illuminated when their associated relay is activated. This board includes a screw terminal that can be used to connect an external power supply in order to prevent overloading your PC's power supply.
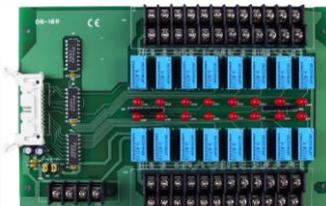
DB-16R

The application example for the DB-16R in the PIO-821 is illustrated in below figure.

Form C Relay

Normally Open

Normally Closed

Com

D/O   20-Pin cable

PIO-821 series card

DB-16R
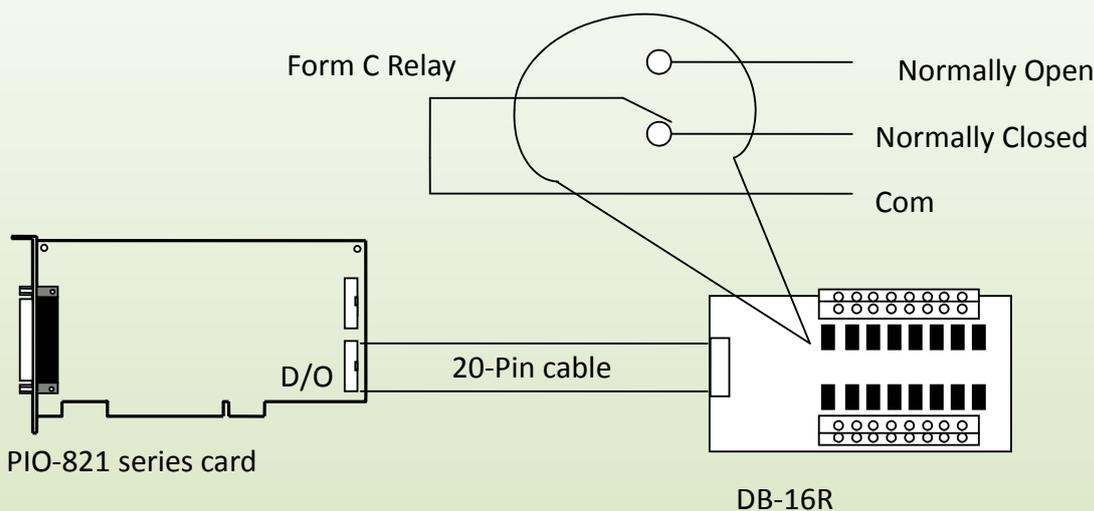
Note:

Channel: 16 Form C Relay

Relay: Switching up to 0.5 A at 110 VAC/ 1 A at 24 V_{DC}