PCI-1741U

16-bit, 200 kS/s High-Resolution Multifunction card

User Manual

Acknowledgements

Microsoft Windows and MS-DOS are registered trademarks of

Microsoft Corp.

All other product names or trademarks are properties of their respective owners.

Part No. 2003174100 1st Edition

Printed in Taiwan December 2004

Product Warranty (2 years)

Advantech warrants to you, the original purchaser, that each of its products will be free from defects in materials and workmanship for two years from the date of purchase.

This warranty does not apply to any products which have been repaired or altered by persons other than repair personnel authorized by Advantech, or which have been subject to misuse, abuse, accident or improper installation. Advantech assumes no liability under the terms of this warranty as a consequence of such events.

Because of Advantech's high quality-control standards and rigorous testing, most of our customers never need to use our repair service. If an Advantech product is defective, it will be repaired or replaced at no charge during the warranty period. For out-of-warranty repairs, you will be billed according to the cost of replacement materials, service time and freight. Please consult your dealer for more details.

If you think you have a defective product, follow these steps:

1. Collect all the information about the problem encountered. (For example, CPU speed, Advantech products used, other hardware and software used, etc.) Note anything abnormal and list any onscreen messages you get when the problem occurs.
2. Call your dealer and describe the problem. Please have your manual, product, and any helpful information readily available.
3. If your product is diagnosed as defective, obtain an RMA (return merchandize authorization) number from your dealer. This allows us to process your return more quickly.
4. Carefully pack the defective product, a fully-completed Repair and Replacement Order Card and a photocopy proof of purchase date (such as your sales receipt) in a shippable container. A product returned without proof of the purchase date is not eligible for warranty service.
5. Write the RMA number visibly on the outside of the package and ship it prepaid to your dealer.

CE

This product has passed the CE test for environmental specifications. Test conditions for passing included the equipment being operated within an industrial enclosure. In order to protect the product from being damaged by ESD (Electrostatic Discharge) and EMI leakage, we strongly recommend the use of CE-compliant industrial enclosure products.

FCC Class A

This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at own expense.

Technical Support and Assistance

Step 1. Visit the Advantech web site at **www.advantech.com/support** where you can find the latest information about the product.

Step 2. Contact your distributor, sales representative, or Advantech's customer service center for technical support if you need additional assistance. Please have the following information ready:
- Product name and serial number
- Description of your peripheral attachments
- Description of your software (OS, version, software, etc.)
- A complete description of the problem
- The exact wording of any error messages

Packing List

Before setting up the system, check that the items listed below are included and in good condition. If any item does not accord with the table, please contact your dealer immediately.

- 1 x PCI-1741U card
- 1 x Companion CD-ROM (DLL driver included)
- 1 x User Manual (This manual)

Contents

Chapter 1 Introduction 2

1.1 Introduction 2
1.2 Features 2
1.2.1 Auto-Calibration Function 2
1.2.2 BoardID 3
1.2.3 Plug-and-Play Function 3
1.2.4 On-board FIFO Memory 3
1.2.5 On Board Programmable Timer/Counter 3
1.3 Installation Guide 4
Figure 1.1:Installation Flow Chart 5
1.4 Software 5
1.5 Accessories 6
1.5.1 PCL-10168 Wiring Cable 6
1.5.2 ADAM-3968 Wiring Boards 6
1.5.3 PCLD-8710 6

Chapter 2 Installation 8

2.1 Unpacking 8
2.2 Driver Installation 9
Figure 2.1:Setup Screen 9
Figure 2.2:Driver Setup 10
2.3 Hardware Installation 11
Figure 2.3:Device Manager 12
2.4 Device Setup & Configuration 13
Figure 2.4:Device Manager Dialog Box 14
Figure 2.5:The Device Setting Dialog Box 15
2.5 Device Testing 16
2.5.1 Testing the Analog Input Function 17
Figure 2.6:Analog Input tab 17
2.5.2 Testing the Analog Output Function 18
Figure 2.7:Analog Output Tab 18
2.5.3 Testing Digital Input Function 18
Figure 2.8:Digital Input Tab 19
2.5.4 Testing Digital Output Function 19
Figure 2.9:Digital Output Tab 19
2.5.5 Testing Counter Function 20
Figure 2.10:Counter Tab 20

Chapter 3 Signal Connections 22

3.1 I/O Connector 22
3.1.1 I/O Connector Signal Description 22
Table 3.1:I/O Connector Signal Description 22
3.1.2 Pin Assignment 24
Figure 3.1:I/O Connector Pin Assignments 24

3.2 Analog Input Connections 25
3.2.1 Single-ended Channel Connections 25
Figure 3.2:Single-ended Channel Connections 25
3.2.2 Differential Channel Connections 26
Figure 3.3:Differential Connection - Ground Signal ... 26
Figure 3.4:Differential Connection - Floating Signal .. 27
Figure 3.5:Example 27
3.3 Analog Output Connections 28
Figure 3.6:Analog Output Connections 28
3.4 Trigger Source Connections 28
3.4.1 Internal Pacer Trigger Connection 28
3.4.2 External Trigger Source Connection 29
3.5 Field Wiring Considerations 29
3.6 BoardID Switch Settings 30
Figure 3.7:Jumpers and Switch Locations 30
Table 3.2:BoardID Setting 30
Chapter 4 Software Overview 34
4.1 Software Overview 34
4.2 Programming Choices 34
4.2.1 DLL Driver 34
4.2.2 Register-Level Programming 34
4.3 DLL Driver Programming Roadmap 34
4.3.1 Programming Tools 35
4.3.2 Programming with DLL Driver Function Library 35
4.3.3 Troubleshooting DLL Driver Error 36
Chapter 5 Calibration 38
5.1 VR Assignment 39
Figure 5.1:Locations of VR1 and TP4 39
5.1.1 PCI-1741U Calibration procedure: 39
5.1.2 Calibration Utility 40
Appendix A Specifications 44
A.1 Analog Input 44
A.2 Analog Output 45
A.3 Digital Input/Output 46
A.4 Counter/Timer 46
A.5 General 47
Appendix B Block Diagram 50
Appendix C Register Structure & Format 52
C.1 Overview 52
C.2 I/O Port Address Map 52
Table C.1:PCI-1741U Register Format (Part 1) 53
Table C.2:PCI-1741U Register Format (Part 2) 54
Table C.3:PCI-1741U Register Format (Part 1) 55
Table C.4:PCI-1741U Register Format (Part 4) 56
C.3 Ch. Number and A/D Data — BASE+0 and BASE+1 ... 57
C.3.1 BASE+0 and BASE+1 hold the result of A/D conversion

data. 57
Table C.5:PCI-1741U Register for A/D Data 57
C.4 Software A/D Trigger — BASE+0 57
C.5 A/D Channel Range Setting — BASE+2 57
Table C.6:Register for A/D Channel Range Setting 57
Table C.7:Gain Codes for PCI-1741U 58
C.6 MUX Control — BASE+4 and BASE+5..... 59
Table C.8:Register for Multiplexer Control 59
C.7 Control Register — BASE+6 60
Table C.9:Control Register 60
C.8 Status Register — BASE+6 and BASE+7 62
Table C.10:Status Register 62
C.9 Clear Interrupt and FIFO — BASE+8 and BASE+9 63
Table C.11:Register to Clear Interrupt and FIFO 63
C.10 D/A Output Channel 0 — BASE+10 and BASE+11 63
Table C.12:Register for Load D/A Channel 0 Data 63
C.11 D/A Output Channel 0 — BASE+10 and BASE+11 64
Table C.13:Register for D/A channel 0 data 64
C.12 D/A Reference Control —BASE+14 64
Table C.14:PCI-1741U Register for D/A Ref. Control 64
C.13 Digital I/O Registers — BASE+16 and BASE+17 65
Table C.15:Register for Digital Input 65
Table C.16:Register for Digital Output 65
C.14 Calibration Busy flag — BASE+18 66
Table C.17:Calibration Busy Check Flag 66
C.15 Calibration Register — BASE+18 and BASE+19 66
Table C.18:Calibration Command and Data Register . 66
Table C.19:Calibration Command Codes 67
C.16 BoardID Registers — BASE+20 68
Table C.20:Register for BoardID 68
C.17 Programmable Timer/Counter Registers BASE+24, BASE+26, BASE+28 and BASE+30 68
Appendix D 82C54 Counter Chip Function 70
D.1 The Intel 82C54..... 70
D.2 Counter Read/Write and Control Registers..... 71
Table D.1:Counter registers..... 71
D.3 Counter Operating Modes 74

CHAPTER 1

Introduction

This chapter introduces PCI-1741U and shows how to install the software. It also has information on useful accessories.

Sections include:

- Introduction
- Features
- Installation Guide
- Software
- Accessories

Chapter 1 Introduction

1.1 Introduction

Thank you for buying the Advantech PCI-1741U PCI card. PCI-1741U is a powerful high-resolution multifunction DAS card for the PCI bus. Its sampling rate of up to 200 kS/s and 16-bit resolution, fulfill the needs of most data acquisition applications. PCI-1741U provides 16 single-ended or 8 differential analog input channels, two 16-bit D/A output channels, 16 digital input/output channels, and one 10 MHz 16-bit counter channel.

The following sections will provide further information about features of the multifunction cards, a quick installation guide, and some brief information on software and accessories for the PCI-1741U card.

1.2 Features

The Advantech PCI-1741U provides users with the most requested measurement and control functions as below:

- 16-bit AI, AO high resolution
- 200 kS/s sampling rate
- Auto calibration function
- 16 S.E. or 8 Diff. AI
- Unipolar/Bipolar input range
- 1K samples FIFO for AI
- Universal PCI bus (support 3.3V or 5V PCI bus signal)
- Board ID switch

1.2.1 Auto-Calibration Function

The PCI-1741U provides an auto-calibration function with an calibration utility. The built-in calibration circuitry of the PCI-1741U corrects gain and offset errors in analog input channels, thereby eliminating the need for external equipment and user adjustments.

1.2.2 BoardID

The PCI-1741U has a built-in DIP switch that helps define each card's unique ID when multiple PCI-1741U cards have been installed on the same PC chassis. The BoardID setting function is very useful when building a system with multiple PCI-1741U cards. With the correct BoardID settings, you can easily identify and access each card during hardware configuration and software programming.

1.2.3 Plug-and-Play Function

The PCI-1741U is a Plug-and-Play device, which fully complies with PCI Specification Rev 2.2. During card installation, there is no need to set jumpers or DIP switches. Instead, all bus-related configurations such as base I/O address and interrupt are automatically done by the Plug & Play function.

1.2.4 On-board FIFO Memory

PCI-1741U provides 1K samples on-board FIFO (First In First Out) memory buffer for AD. This is an important feature for faster data transfer and more predictable performance under the Windows system.

1.2.5 On Board Programmable Timer/Counter

PCI-1741U provides a programmable timer counter for generating a pacer trigger for the A/D conversion. The timer/counter chip is 82C54, which includes three 16-bit counter 10 MHz clocks. One counter is used as an event counter for counting events coming from the input channel. The other two are cascaded together to make a 32-bit timer for pacer trigger time base.

1.3 Installation Guide

Before you install your PCI-1741U card, please make sure you have the following necessary components:

- PCI-1741U analog input card
- PCI-1741U User Manual
- Driver software Advantech DLL drivers (included in the companion CD-ROM)
- Wiring cable PCL-10168
- Wiring board ADAM-3968
- PCLD-8710: Industrial Wiring Terminal Board with CJC circuit for DIN-rail Mounting. (cable not included)
- Computer Personal computer or workstation with a PCI-bus slot running Windows 98/2000/XP)

Some other optional components are also available for enhanced operation:

- Application software ActiveDAQ, GeniDAQ or other third-party software packages

After you get the necessary components and maybe some of the accessories for enhanced operation of your Multifunction card, you can then begin the Installation procedures. Fig. 1-1 on the next page provides a concise flow chart to give users a broad picture of the software and hardware installation procedures:

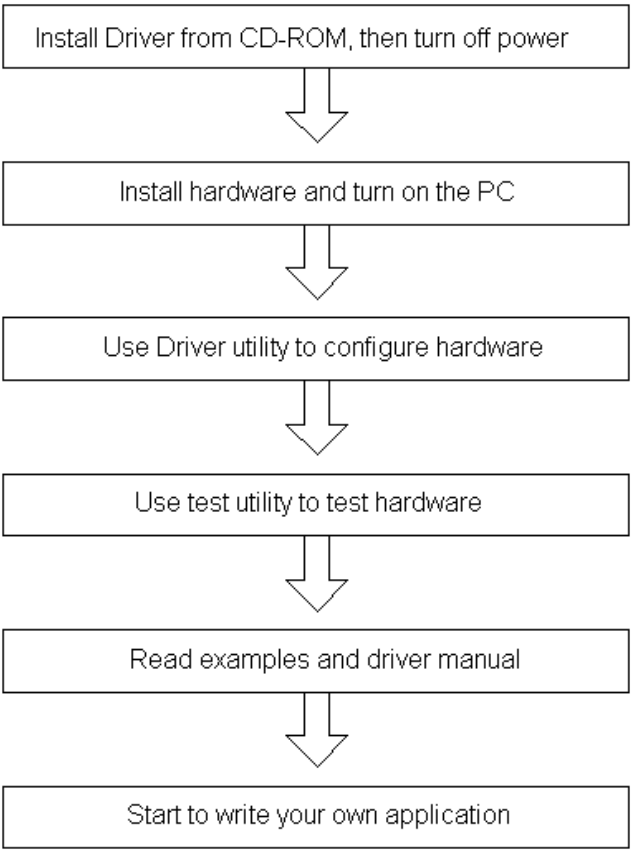


Figure 1.1: Installation Flow Chart

1.4 Software

Advantech offers a rich set of DLL drivers, third-party driver support and application software to help fully exploit the functions of your PCI-1741U card:

- DLL driver (on the companion CD-ROM)
- LabVIEW driver
- Advantech ActiveDAQ
- Advantech GeniDAQ

For more information on software, please refer to Chapter 4, Software Overview.

Users who intend to program directly at the registers of the Multifunction card have register-level programming as an option. Since register-level programming is often difficult and labor-intensive, it is usually recommended only for experienced programmers. For more information, please refer to Appendix C, Register Structure and Format.

1.5 Accessories

Advantech offers a complete set of accessory products to support the PCI-1741U cards. These accessories include:

1.5.1 PCL-10168 Wiring Cable

The PCL-10168 shielded cable is specially designed for PCI-1741U cards to provide high resistance to noise. To achieve a better signal quality, the signal wires are twisted in such a way as to form a “twisted-pair cable”, reducing cross-talk and noise from other signal sources. Furthermore, its analog and digital lines are separately sheathed and shielded to neutralize EMI/EMC problems.

1.5.2 ADAM-3968 Wiring Boards

The ADAM-3968 is a 68-pin SCSI wiring terminal module for DIN-rail mounting. This terminal module can be readily connected to the Advantech PC-Lab cards and allow easy yet reliable access to individual pin connections for the PCI-1741U card.

1.5.3 PCLD-8710

PCLD-8710 is a DIN-rail-mounted screw-terminal board to be used with any PC-LabCard products with a 68-pin SCSI connector. PCLD-8710 features the following functions:

- Two additional 20-pin flat-cable connectors for digital input and output
- Reserved space on board to meet future needs for signal-conditioning circuits (e.g. low-pass filter, voltage attenuator and current shunt)
- Industrial-grade screw-clamp terminal blocks for heavy-duty and reliable connections.

CHAPTER 2

Installation

This chapter provides a packaged item checklist, proper instructions about unpacking and step-by-step procedures for both driver and card installation.

Note that PCI-1741U is used as an example.

Sections include:

- Unpacking
- Driver Installation
- Hardware Installation
- Device Setup & Configuration
- Device Testing

Chapter 2 Installation

2.1 Unpacking

After receiving your PCI-1741U package, please inspect its contents first. The package should contain the following items:

- PCI-1741U card
- Companion CD-ROM (DLL driver included)
- User Manual

The PCI-1741U card harbors certain electronic components vulnerable to electrostatic discharge (ESD). ESD could easily damage the integrated circuits and certain components if preventive measures are not carefully paid attention to.

Before removing the card from the antistatic plastic bag, you should take following precautions to ward off possible ESD damage:

- Touch the metal part of your computer chassis with your hand to discharge static electricity accumulated on your body. Or one can also use a grounding strap.
- Touch the antistatic bag to a metal part of your computer chassis before opening the bag.
- Take hold of the card only by the metal bracket when removing it out of the bag.

After taking out the card, you should first:

- Inspect the card for any possible signs of external damage (loose or damaged components, etc.). If the card is visibly damaged, please notify our service department or the local sales representative immediately. Avoid installing a damaged card into your system.

Also pay extra caution to the following aspects to ensure proper installation:

- Avoid physical contact with materials that could hold static electricity such as plastic, vinyl and Styrofoam.

- Whenever you handle the card, grasp it only by its edges. DO NOT TOUCH the exposed metal pins of the connector or the electronic components.

Note *Keep the antistatic bag for future use. You might need the original bag to store the card if you have to remove the card from the PC or transport it elsewhere.*

2.2 Driver Installation

We recommend installing the driver before you plug the PCI-1741U Card into your system, since this will guarantee a smooth installation process.

The 32-bit DLL driver Setup program for the PCI-1741U card is included on the companion CD-ROM that is shipped with your DAS card package. Please follow the steps below to install the driver software:

Step 1: Insert the companion CD-ROM into your CD-ROM drive.

Step 2: The *Setup* program will be launched automatically if you have the *Autorun* function enabled on your system. When the *Setup* program is launched, you'll see the following setup screen.

Figure 2.1: Setup Screen

Note *If the autoplay function is not enabled on your computer, use Windows Explorer or Windows **Run** command to execute SETUP.EXE on the companion CD-ROM.*

Step 3: Select the *Installation* option, then the *Individual Drivers* option.

Step 4: Select the specific device then just follow the installation instructions step by step to complete your device driver setup.

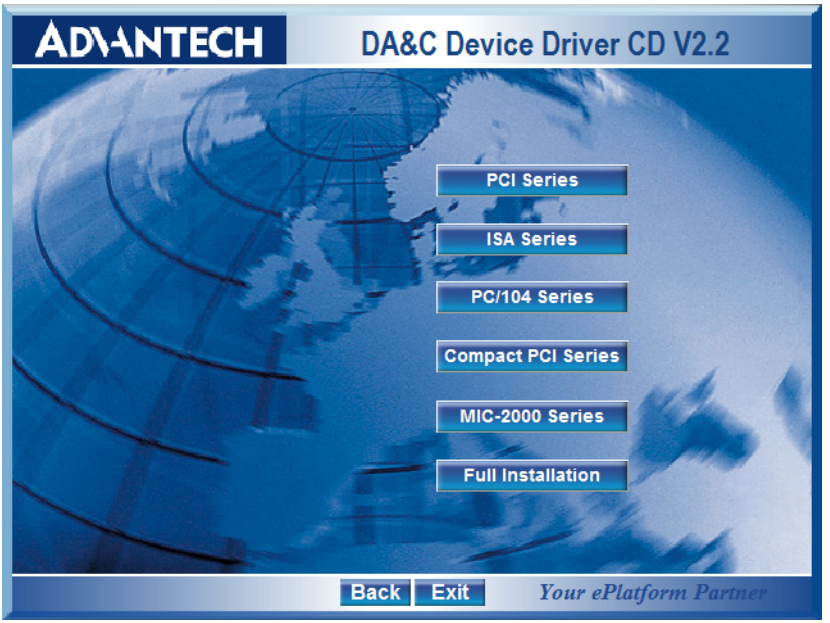


Figure 2.2: Driver Setup

For further information on driver-related issues, an online version of the *DLL Drivers Manual* is available by accessing the following path:

Start/ Programs/ Advantech Automation/ Device Manager/ Device Driver's Manual.

2.3 Hardware Installation

Note *Make sure you have installed the driver before you install the card (please refer to 2.2 Driver Installation)*

After the DLL driver installation is completed, you can go on to install the PCI-1741U card in any PCI slot on your computer. But it is suggested that you refer to the computer user manual or related documentations if you have any doubts. Please follow the steps below to install the card on your system:

Step 1: Turn off your computer and unplug the power cord and cables. TURN OFF your computer before installing or removing any components on the computer.

Step 2: Remove the cover of your computer.

Step 3: Remove the slot cover on the back panel of your computer.

Step 4: Touch the metal part on the surface of your computer to neutralize any static electricity that might be in your body.

Step 5: Insert the PCI-1741U card into a PCI slot. Hold the card only by its edges and carefully align it with the slot. Insert the card firmly into place. Use of excessive force must be avoided; or the card might be damaged.

Step 6: Fasten the bracket of the PCI card on the back panel rail of the computer with screws.

Step 7: Connect appropriate accessories (68-pin cable, wiring terminals, etc. if necessary) to the PCI card.

Step 8: Replace the cover of your computer chassis. Re-connect the cables you removed in step 2.

Step 9: Plug in the power cord and turn on the computer.

Note *If you installed the card without installing the DLL driver first, Windows 98 will recognize your card as an "unknown device" after rebooting, and will prompt you to provide the necessary driver. You should ignore the prompting messages (just click the* ***Cancel*** *button) and set up the driver according to the steps described in 2.2 Driver Installation.*

After the PCI-1741U card is installed, you can verify whether it is properly installed on your system in the *Device Manage*r:

1. Access the Device Manager through
   Start/ Control Panel/ System/ Device Manager.
2. The device name of PCI-1741U should be listed on the *Device Manager*.

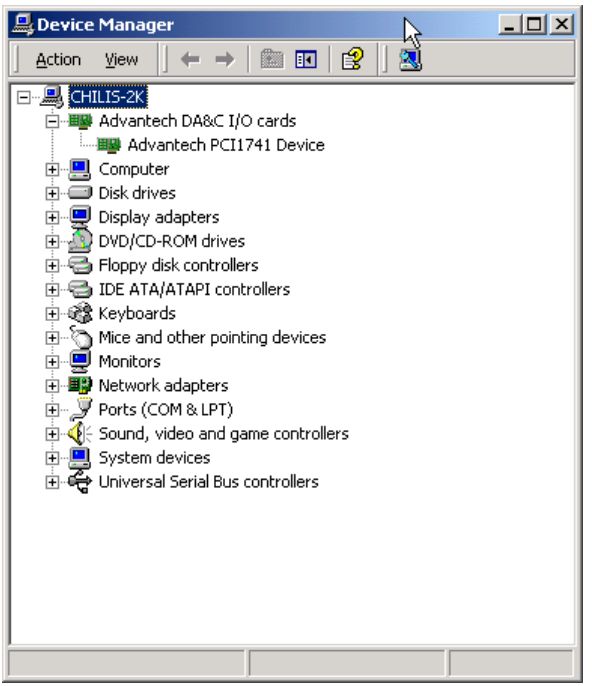


Figure 2.3: Device Manager

Note *If your card is properly installed, you should see the device name of your card listed on the Device Manager tab. If you see your device name listed, but marked with an exclamation sign "!", it means your card has not been correctly installed. In this case, remove the card device from the Device Manager by selecting its device name and press the Remove button. Then go through the driver installation process again.*

After your card is properly installed on your system, you can now configure your device using the *Device Manager* program that has itself already been installed on your system during driver setup. A complete device installation procedure should include device setup, configuration and testing. The following sections will guide you through the *setup*, *configuration* and *testing* of your device.

2.4 Device Setup & Configuration

The *Device Manager* program is a utility that allows you to set up, configure and test your device, and later stores your settings on the system registry. These settings will be used when you call the APIs of *Advantech Device Drivers*.

Set Up the Device

Step 1: To install the I/O device for your card, you must first run the Device Installation program by accessing:
Start/ Programs/ Advantech Automation/ Device Manager/ Advantech Device Manager.

Step 2: The Advantech Device Manager will search for the device(s) already installed on your system (if any) automatically.

Step 3: The Installed Devices box will list the device(s) found on our system, such as the one in Figure 2.4.

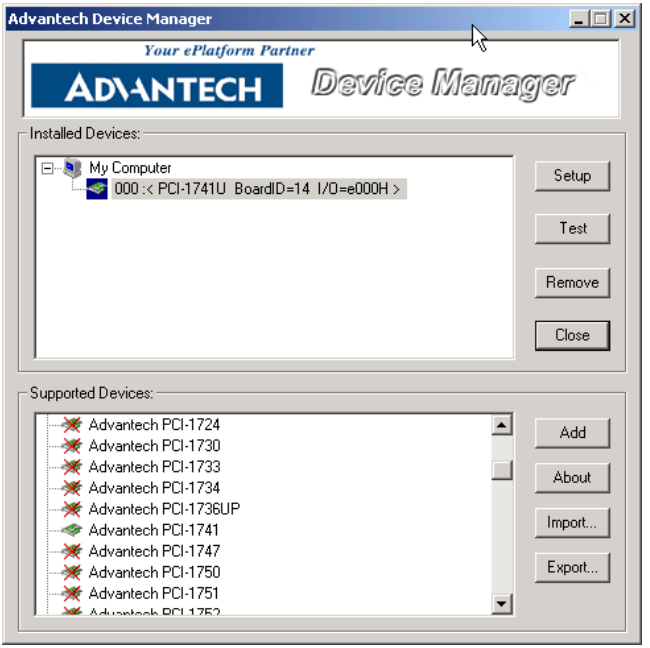


Figure 2.4: Device Manager Dialog Box

Note *As we have noted, the device name "000:<PCI-1741U BoardID=14 I/O=e000H" begins with device number "000", which is specifically assigned to each card. The device number is passed to the driver to specify which device you wish to control.*

Step 4: In the Device Setting dialog box (Figure2.5), you can configure the A/D channel as Single-ended or Differential and choose the D/A channel reference voltage source type..

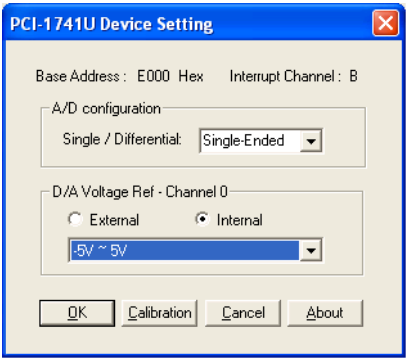


Figure 2.5: The Device Setting Dialog Box

Note

You can configure the source of D/A reference voltage either as Internal or External, and select the output voltage range. When selecting voltage source as Internal, you have two options for the output voltage range : 0 ~ 5 V and 0 ~ 10 V.

When selected as External, the output voltage range is determined by the external reference voltage in the following way:

By inputting an external reference voltage: -xV , where |x| <= 10, you will get a output voltage range: 0 to xV.

2.5 Device Testing

After following the *Setup* and *Configuration* procedure to the last step described in the previous section, you can now proceed to test the device by clicking the ***Test*** Button on the *Device Manager* dialog box (*Fig. 2-9*). A *Device Test* dialog box will appear accordingly:

On the *Device Test* dialog box, you are free to test various functions of PCI-1741U on the *Analog input, Digital input, Digital output* or *Counter* tabs. The *Analog output* function is only available for PCI-1741U.

Note: You can access the Device Test dialog box either by the previous procedure for the Device Installation Program, or simply by accessing Start/Programs/ Advantech Automation/ Device Manager/ Advantech Device Manager.

All functions are performed by the software polling method. For high speed data acquirements or output, functions have to use a corresponding VC example like ADINT or ADDMA or ADBMDMA.

2.5.1 Testing the Analog Input Function

Click the Analog Input tab to bring it up to the front of the screen. Select the input range for each channel in the Input range drop-down boxes. Configure the sampling rate on the scroll bar. Switch the channels by using the up/down arrow.

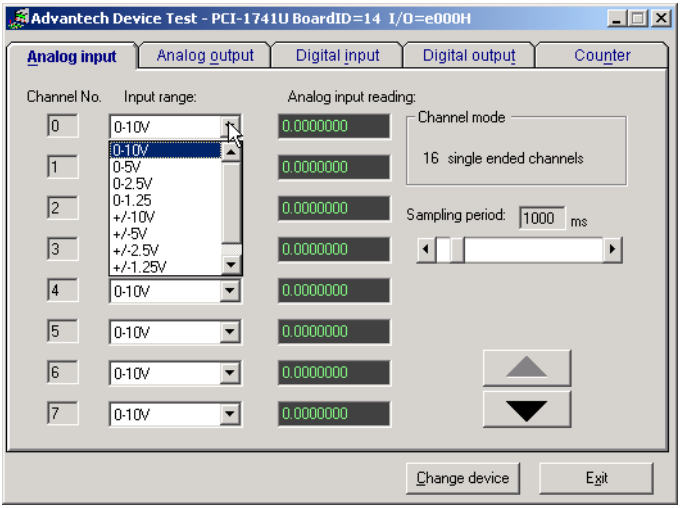


Figure 2.6: Analog Input tab

2.5.2 Testing the Analog Output Function

Click the *Analog Output* tab to bring it up to the foreground. The *Analog Output* tab allows you to output quasi-sine, triangle, or square waveforms automatically generated by the software, or output single values manually. You can also configure the waveform frequency and output voltage range.

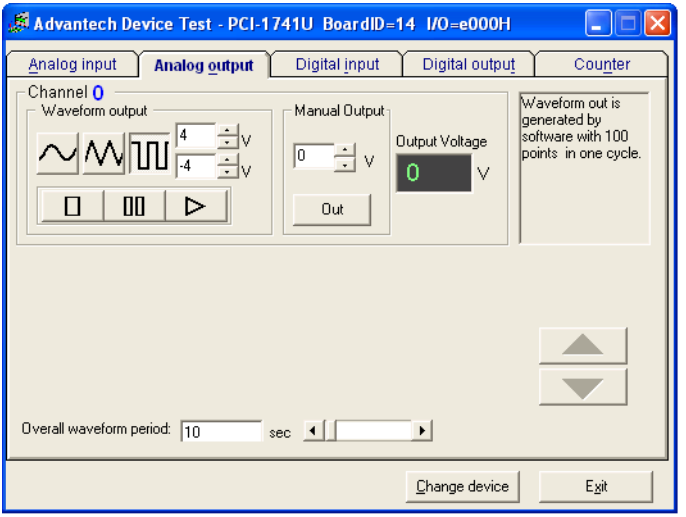


Figure 2.7: Analog Output Tab

2.5.3 Testing Digital Input Function

Click the *Digital Input* tab to show forth the *Digital Input* test panel as seen below. Through the color of the lamps, you can easily discern whether the status of each digital input channel is high or low.

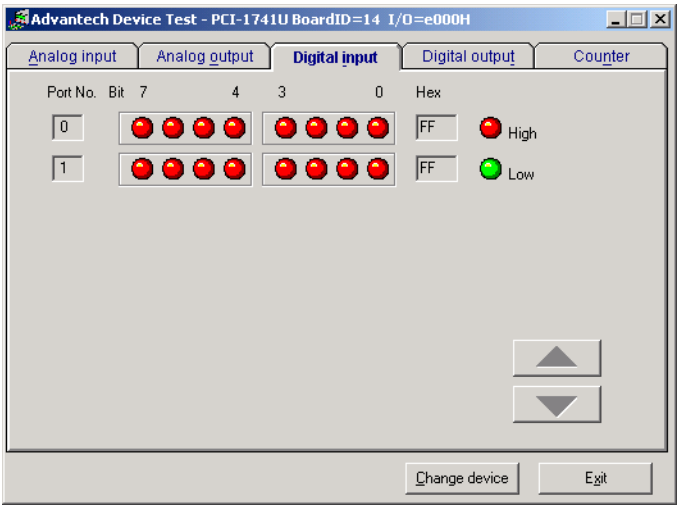


Figure 2.8: Digital Input Tab

2.5.4 Testing Digital Output Function

Click the *Digital Output* tab to bring up the *Digital Output* test panel such as the one seen on the next page. By pressing the buttons on each tab, you can easily set each digital output channel as *high* or *low* for the corresponding port.

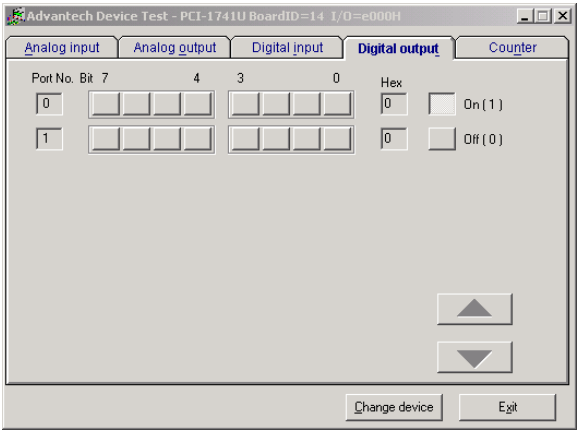


Figure 2.9: Digital Output Tab

2.5.5 Testing Counter Function

Click the *Counter* Tab to bring its test panel forth. The counter channel *(Channel 0*) offers two options: Event counting and Pulse out. If you select Event counting, you need first to connect your clock source to pin CNT0_CLK, and the counter will start counting after the pin CNT0_GATE is triggered. If you select Pulse Out, the clock source will be output to pin CNT0_OUT. You can configure the *Pulse Frequency* with the scroll bar right below it.

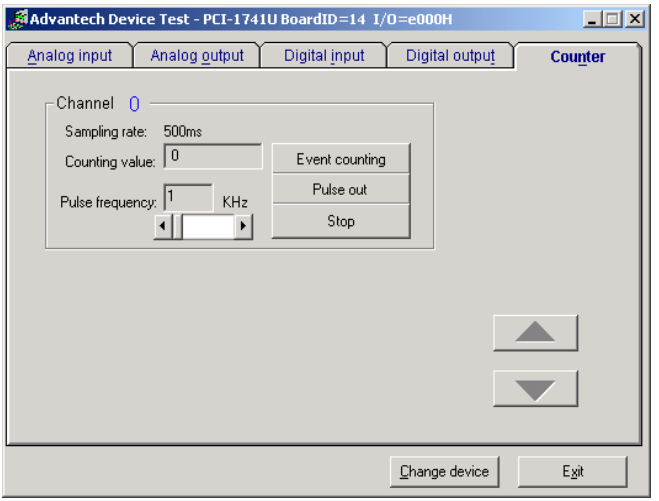


Figure 2.10: Counter Tab

Only after your card device is properly set up, configured and tested, can the device installation procedure be counted as complete. After the device installation procedure is completed, you can safely proceed to the next chapter*, Signal Connection*s.

CHAPTER 3

Signal Connections

Maintaining signal connections is one of the most important factors in ensuring that your application system is sending and receiving data correctly. A good signal connection can avoid unnecessary and costly damage to your PC and other hardware devices. This chapter provides useful information about how to connect input and output signals to PCI-1741U via the I/O connector.

Sections include:

- I/O Connector
- Analog Input Connections
- Analog Output Connections
- Trigger Source Connections
- Field Wiring Considerations

Chapter 3 Signal Connections

3.1 I/O Connector

The I/O connector on the PCI-1741U is a 68-pin connector that enables you to connect to accessories with the PCL-10168 shielded cable.

Fig. 3-1 shows the pin assignments for the 68-pin I/O connector on the PCI-1741U, and table 3-1 shows the I/O connector signal description.

3.1.1 I/O Connector Signal Description

Table 3.1: I/O Connector Signal Description			
AI<0…15>	AIGND	Input	Analog Input Channels 0 through 15. Each channel pair, AI<i, i+1> (i = 0, 2, 4...14), can be configured as either two single-ended inputs or one differential input.
AIGND	-	-	Analog Input Ground. The three ground references (AIGND, AOGND, and DGND) are connected together.
AO0_REF	AOGND	Input	Analog Output Channel 0 External Reference.
AO0_OUT	AOGND	Output	Analog Output Channels 0.
AOGND	-	-	Analog Output Ground. The analog output voltages are referenced to these nodes. The three ground references (AIGND, AOGND, and DGND) are connected together.
DI<0..15>	DGND	Input	Digital Input channels.
DO<0..15>	DGND	Output	Digital Output channels.
DGND	-	-	Digital Ground. This pin supplies the reference for the digital channels at the I/O connector as well as the +5VDC supply. The three ground references (AIGND, AOGND, and DGND) are connected together.

Table 3.1: I/O Connector Signal Description			
CNT0_CLK	DGND	Input	Counter 0 Clock Input. The clock input of counter 0 can be either external (up to 10 MHz) or internal (1 MHz), as set by software.
CNT0_OUT	DGND	Output	Counter 0 Output.
CNT0_GATE	DGND	Input	Counter 0 Gate Control.
PACER_OUT	DGND	Output	Pacer Clock Output. This pin pulses once for each pacer clock when turned on. If A/D conversion is in the pacer trigger mode, users can use this signal as a synchronous signal for other applications. A low- to- high edge triggers A/D conversion to start.
TRG_GATE	DGND	Input	A/D External Trigger Gate. When TRG _GATE is connected to +5 V, it will enable the external trigger signal to input. When TRG _GATE is connected to DGND, it will disable the external trigger signal to input.
EXT_TRG	DGND	Input	A/D External Trigger. This pin is external trigger signal input for the A/D conversion. A low-to-high edge triggers A/D conversion to start.
+12V	DGND	Output	+12 VDC Source.
+5V	DGND	Output	+5 VDC Source.

3.1.2 Pin Assignment

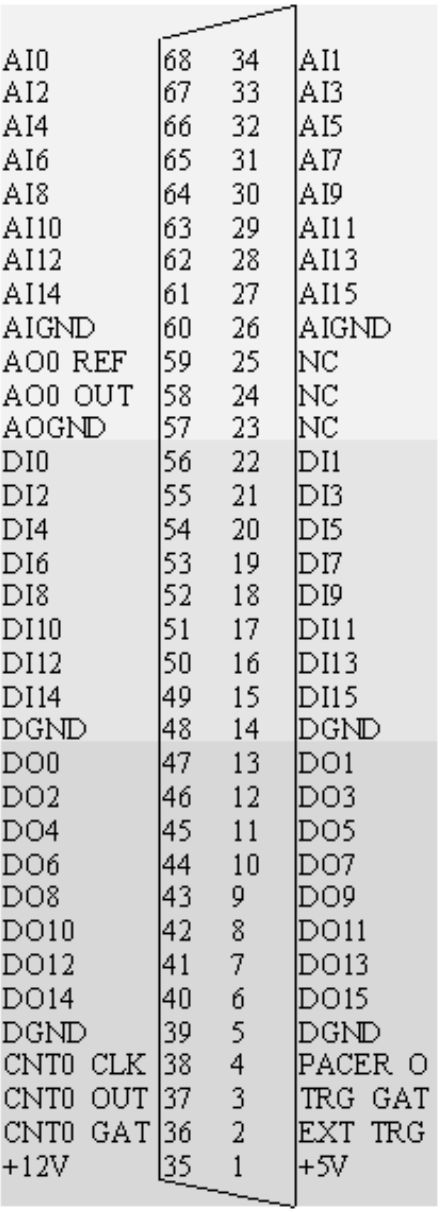


Figure 3.1: I/O Connector Pin Assignments

3.2 Analog Input Connections

3.2.1 Single-ended Channel Connections

The single-ended input configuration has only one signal wire for each channel, and the measured voltage (Vm) is the voltage of the wire as referenced against the common ground.

A signal source without a local ground is also called a “floating source”. It is fairly simple to connect a single-ended channel to a floating signal source. In this mode, the PCI-1741U provides a reference ground for external floating signal sources. *Fig. 3-2* shows a single-ended channel connection between a floating signal source and an input channel on PCI-1741U.

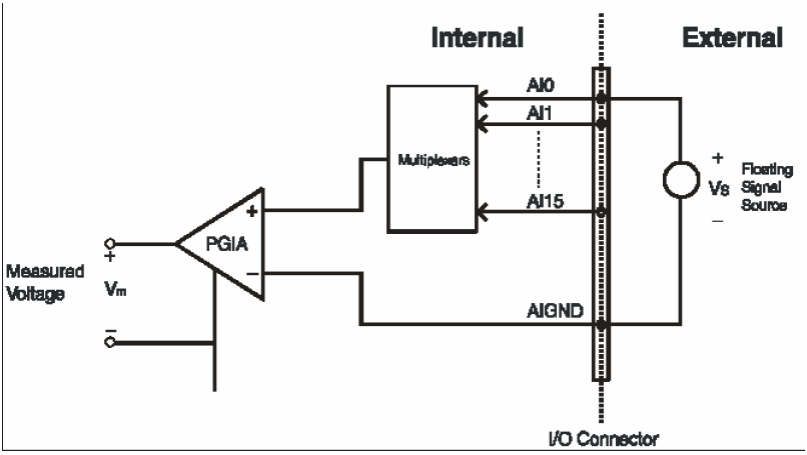


Figure 3.2: Single-ended Channel Connections

3.2.2 Differential Channel Connections

The differential input channels operate with two signal wires for each channel, and the voltage difference between both signal wires is measured. On PCI-1741U, when all channels are configured to differential input, up to 8 analog channels are available.

If one side of the signal source is connected to a local ground, the signal source is ground-referenced. Therefore, the ground of the signal source and the ground of the card will not be exactly of the same voltage. The difference between the ground voltages forms a common-mode voltage (V cm).

To avoid the ground loop noise effect caused by common-mode voltages, you can connect the signal ground to the Low input. *Figure 3-3* shows a differential channel connection between a ground-reference signal source and an input channel on the PCI-1741U. With this connection, the PGIA rejects a common-mode voltage V_{cm} between the signal source and the PCI-1741U ground, shown as V_{cm} in *Figure 3-3*.

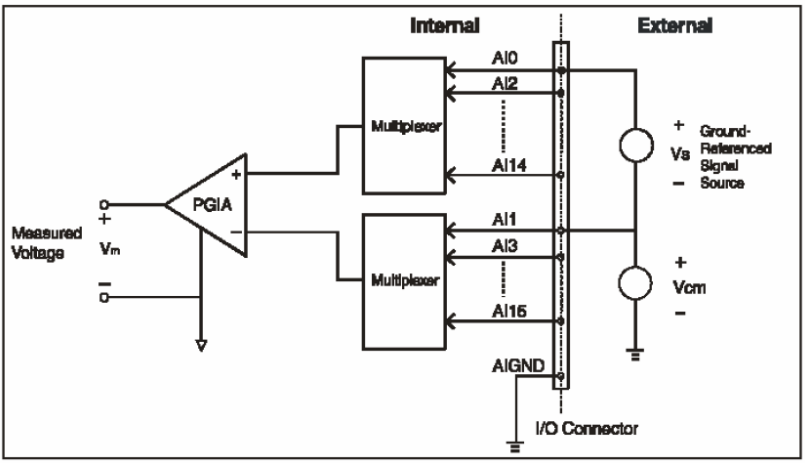


Figure 3.3: Differential Connection - Ground Signal

If a floating signal source is connected to the differential input channel, the signal source might exceed the common-mode signal range of the PGIA, and the PGIA will be saturated with erroneous voltage-readings. You must therefore reference the signal source against the AIGND.

Figure 3.4 shows a differential channel connection between a floating signal source and an input channel on PCI-1741U. In this figure, each side of the floating signal source is connected through a resistor to the AIGND. This connection can reject the common-mode voltage between the signal source and the PCI-1741U ground.

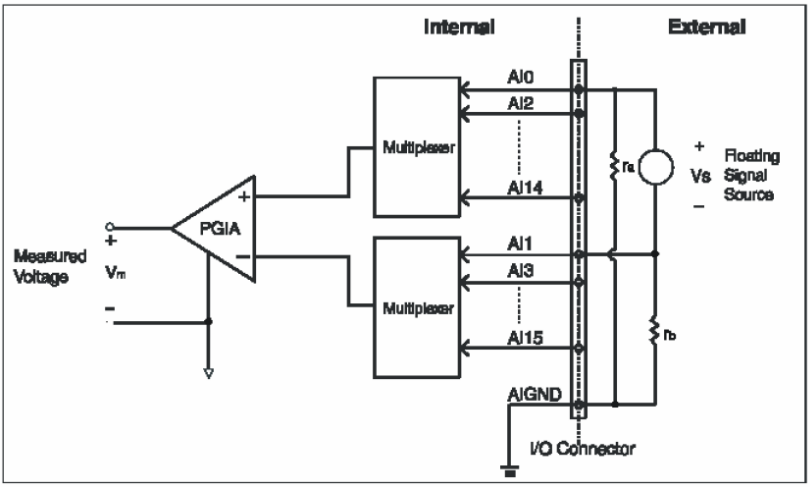


Figure 3.4: Differential Connection - Floating Signal

However, this connection has the disadvantage of loading the source down with the series combination (sum) of the two resistors. For ra and rb, for example, if the input impedance rs is 1 kW, and each of the two resistors are 100 kW, then the resistors load down the signal source with 200 kW (100 kW + 100 kW), resulting in a –0.5% gain error. The following gives a simplified representation of the circuit and calculating process.

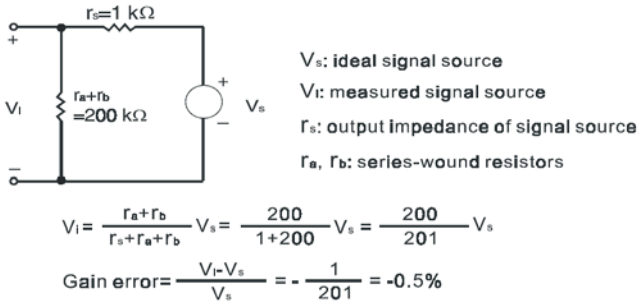


Figure 3.5: Example

3.3 Analog Output Connections

The PCI-1741U provides one D/A output channel, **AO0**. You may use the PCI-1741U internally-provided precision -5V (-10V) reference to generate 0 to +5 V (+10 V) D/A output range. You may also create a D/A output range through the external reference, **AO0_REF**. The external reference input range is +/-10 V. For example, connecting with an external reference of -7 V will generate 0 ~ +7 V D/A output.

Fig. 3-5 shows how to make analog output and external reference input connections on the PCI-1741U.

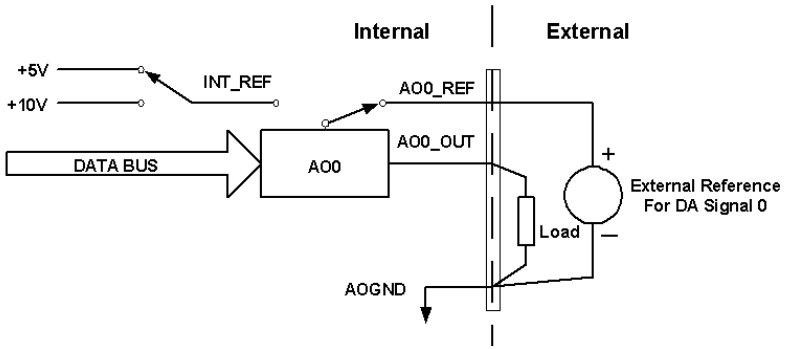


Figure 3.6: Analog Output Connections

3.4 Trigger Source Connections

3.4.1 Internal Pacer Trigger Connection

PCI-1741U includes one 82C54 compatible programmable Timer/ Counter chip which provides three 16-bit counters connected to a 10 MHz clock, each designated specifically as Counter 0, Counter 1 and Counter 2. Counter 0 is a counter which counts events from an input channel or outputting pulse. Counter 1 and Counter 2 are cascaded to create a 32-bit timer for pacer triggering. A low-to-high edge from the Counter 2 output **(PACER_OU**T) will trigger an A/D conversion on the PCI-1741U. At the same time, you can also use this signal as a synchronous signal for other applications.

3.4.2 External Trigger Source Connection

In addition to pacer triggering, the PCI-1741U also allows external triggering for A/D conversions. When a +5 V source is connected to **TRG_GAT**E, the external trigger function is enabled. A low-to-high edge coming from **EXT_TRG** will trigger an A/D conversion on the PCI-1741U. When **DGND** is connected to **TRG_GAT**E, the external trigger function is thereby disabled.

3.5 Field Wiring Considerations

When you use the PCI-1741U to acquire data from outside, noises in the environment might significantly affect the accuracy of your measurements if due cautions are not taken. The following measures will be helpful to reduce possible interference running signal wires between signal sources and the PCI-1741U.

- The signal cables must be kept away from strong electromagnetic sources such as power lines, large electric motors, circuit breakers or welding machines, since they may cause strong electromagnetic interference. Keep the analog signal cables away from any video monitor, since it can significantly affect a data acquisition system.
- If the cable travels through an area with significant electromagnetic interference, you should adopt individually shielded, twisted-pair wires as the analog input cable. This type of cable has its signal wires twisted together and shielded with a metal mesh. The metal mesh should only be connected to one point at the signal source ground.
- Avoid running the signal cables through any conduit that might have power lines in it.
- If you have to place your signal cable parallel to a power line that has a high voltage or high current running through it, try to keep a safe distance between them. Or, you should place the signal cable at a right angle to the power line to minimize the undesirable effect.

The signals transmitted on the cable will be directly affected by the quality of the cable. In order to ensure better signal quality, we recommend that you use the PCL-10168 shielded cable.

3.6 BoardID Switch Settings

The PCI-1741U card has one Board ID switch setting. The following image shows the layout of PCI-1741U:

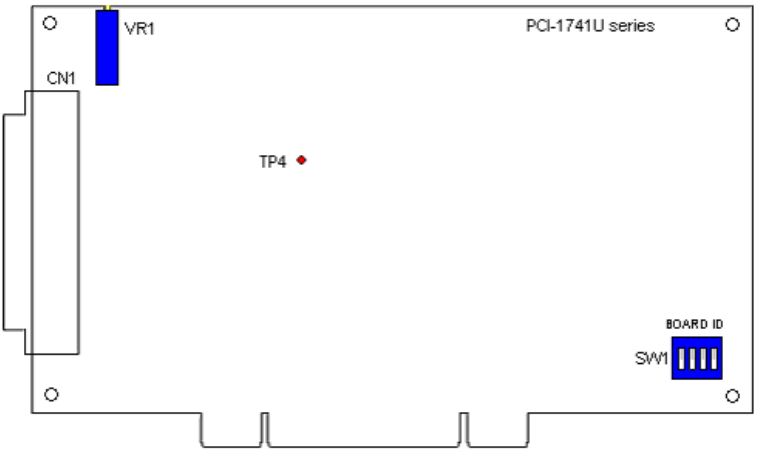


Figure 3.7: Jumpers and Switch Locations

The PCI-1741U has a built-in DIP-switch (SW1), which is used to define each the BoardID. You can determine the BoardID switch on the register. When there are multiple cards on the same chassis, this BoardID switch setting function is useful for identifying each card device number through BoardID switch. The factory setting for the PCI-1741U BoardID switch is 0. If you need to adjust it to other BoardID switchs, set SW1 while referring to Table 3.2.

Table 3.2: BoardID Setting

ID3*	ID2	ID1	ID0	Board ID
1	1	1	1	0
1	1	1	0	1
1	1	0	1	2
1	1	0	0	3
1	0	1	1	4
1	0	1	0	5

Table 3.2: BoardID Setting				
1	0	0	1	6
1	0	0	0	7
0	1	1	1	8
0	1	1	0	9
0	1	0	1	10
0	1	0	0	11
0	0	1	1	12
0	0	1	0	13
0	0	0	1	14
0	0	0	0	15

Note: On: 1, Off: 0

*ID3: the most significant bit (MSB) of BoardID

CHAPTER 4

Software Overview

This chapter gives you an overview of the software programming choices available and a quick reference to source codes examples that can help you be better informed when programming. After following the instructions given in *Chapter 2*, it is hoped that you feel comfortable enough to proceed further.

Sections include:

- Software Overview
- Programming Choices
- DLL Driver Programming

Chapter 4 Software Overview

4.1 Software Overview

Programming choices for DAS cards: You may use Advantech application software such as Advantech DLL driver. On the other hand, advanced users are allowed another option for register-level programming, although not recommended due to its laborious and time-consuming nature.

4.2 Programming Choices

4.2.1 DLL Driver

The Advantech DLL Drivers software is included on the companion CD-ROM at no extra charge. It also comes with all the Advantech DAS cards. Advantech's DLL driver features a complete I/O function library to help boost your application performance. The Advantech DLL driver for *Windows 98/2000/XP* works seamlessly with development tools such as Visual C++, Visual Basic, Borland C++ Builder and Borland Delphi.

4.2.2 Register-Level Programming

Register-level programming is reserved for experienced programmers who find it necessary to write code directly at the level of device registers. Since register-level programming requires much effort and time, we recommend that you use the Advantech DLL drivers instead. However, if register-level programming is indispensable, you should refer to the relevant information in *Appendix C, Register Structure and Format*, or to the example codes included on the companion CD-ROM.

4.3 DLL Driver Programming Roadmap

This section will provide you a roadmap to demonstrate how to build an application from scratch using Advantech DLL driver with your favorite development tools such as Visual C++, Visual Basic, Delphi and C++ Builder. The step-by-step instructions on how to build your own applications using each development tool will be given in the *DLL Drivers Manua*l. Moreover, a rich set of example source codes are also given for your reference.

4.3.1 Programming Tools

Programmers can develop application programs with their favorite development tools:

- Visual C++
- Visual Basic
- Delphi
- C++ Builder

For instructions on how to begin programming in each development tool, Advantech offers a *Tutorial* Chapter in the *DLL Drivers Manual* for your reference. Please refer to the corresponding sections in this chapter on the *DLL Drivers Manual* to begin your programming efforts. You can also take a look at the example source codes provided for each programming tool, since they can be very helpful.

The *DLL Drivers Manual* can be found on the companion CD-ROM. Or if you have already installed the DLL Drivers on your system, The *DLL Drivers Manual* can be readily accessed through the ***Start*** button:

Start/Programs/Advantech Automation/Device Manager/Device Driver's Manual

The example source codes can be found under the corresponding installation folder such as the default installation path:

\Program Files\ADVANTECH\ADSAPI\Examples

For information about using other function groups or other development tools, please refer to the *Creating Windows 98/2000/XP Application with DLL Driver* chapter and the *Function Overview* chapter on the *DLL Drivers Manua*l.

4.3.2 Programming with DLL Driver Function Library

Advantech DLL driver offers a rich function library to be utilized in various application programs. This function library consists of numerous APIs that support many development tools, such as Visual C++, Visual Basic, Delphi and C++ Builder.

According to their specific functions or services, these APIs can be categorized into several function groups:

- Analog Input Function Group
- Analog Output Function Group
- Digital Input/Output Function Group
- Counter Function Group
- Temperature Measurement Function Group
- Alarm Function Group
- Port Function Group
- Communication Function Group
- Event Function Group

For the usage and parameters of each function, please refer to the *Function Overview* chapter in the *DLL Drivers Manual*.

4.3.3 Troubleshooting DLL Driver Error

Driver functions will return a status code when they are called to perform a certain task for the application. When a function returns a code that is not zero, it means the function has failed to perform its designated function. To troubleshoot the DLL driver error, you can pass the error code to **DRV_GetErrorMessage** function to return the error message. Or you can refer to the *DLL Driver Error Codes Appendix* in the *DLL Drivers Manual* for a detailed listing of the Error Code, Error ID and the Error Message.

CHAPTER 5

Calibration

This chapter provides brief information on PCI-1741U calibration. Regular calibration checks are important to maintain accuracy in data acquisition and control applications.

Sections include:

- VR Assignment

Chapter 5 Calibration

This chapter provides brief information on PCI-1741U calibration. Regular calibration checks are important to maintain accuracy in data acquisition and control applications. We provide the calibration programs or utility on the companion CD-ROM to assist you in calibration.

Note *If you installed the program to another directory, you can find these programs in the corresponding subfolders in your destination directory.*

PCI-1741U has been calibrated at the factory for initial use. However, a calibration of the analog input and the analog output function every six months is recommended.

These calibration programs make calibration an easy job. With a variety of prompts and graphic displays, these programs will lead you through the calibration and setup procedures, showing you all the correct settings and adjustments.

To perform a satisfactory calibration, you will need a 7½-digit digital multi-meter and a voltage calibrator or a stable, noise-free D.C. voltage source.

Note *Before you calibrate the PCI-1741U, you must turn on the power at least **15 minutes** to make sure the DAS card is stable.*

The calibration demo programs are included on the companion CD-ROM
- 1741UCAL.Calibration program source file

- 1741UCAL.EXE Calibration program execution file

The calibration program is designed only for the DOS environment.

Access the program from the default location:

C:\Program Files\Advantech\ADSAPI\Utility\PCI1741U

5.1 VR Assignment

There is one variable resistor (VR1) on the PCI-1741U to adjust the accurate reference voltage on the PCI-1741U. We provide a test point (See TP4 in Figure 5.1) for you to check the reference voltage on board. Before you start to calibrate A/D and D/A channels, please adjust VR1 until the reference voltage on TP4 has reached +5.0000 V. Figure 5.1 shows the locations of VR1 and TP4.

Figure 5.1: Locations of VR1 and TP4

5.1.1 PCI-1741U Calibration procedure:

Users can calibrate the PCI-1741U manually. 1741UCAL.C illustrates the standard calibration procedures for your reference. If you want to calibrate the hardware in your own way, the following steps will guide you.

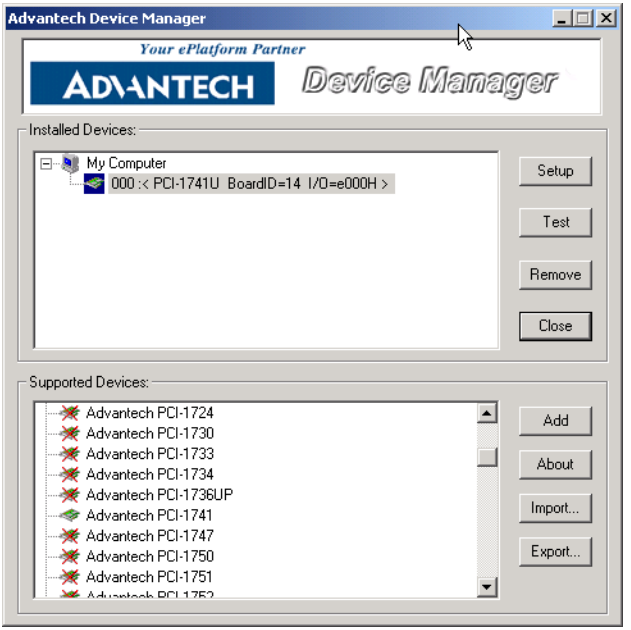
1. On board reference 5V calibration (adjust VR1 until TP4 = 5.0V)
2. AI PGA offset calibration
3. Use On board reference to calibrate AI ±5V input range
4. Use On board reference to calibrate AI 0 to 5V input range
5. Use AI ±5V input range to calibrate AO CH0
6. Use AO CH0 to calibrate AI ±10V input range
7. Use AO CH0 to calibrate AI ±2.5V input range
8. Use AO CH0 to calibrate AI ±1.25V input range

9. Use AO CH0 to calibrate AI ±0.625V input range
10. Use AO CH0 to calibrate AI 0 to 10V input range
11. Use AO CH0 to calibrate AI 0 to 2.5V input range
12. Use AO CH0 to calibrate AI 0 to 1.25V input range

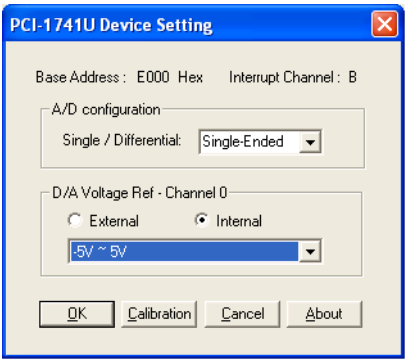
5.1.2 Calibration Utility

Using Advantech Device Manager – PCI-1741U Device Setting Dialog, the user can easily finish the calibration procedures automatically. The calibration steps are as follows:

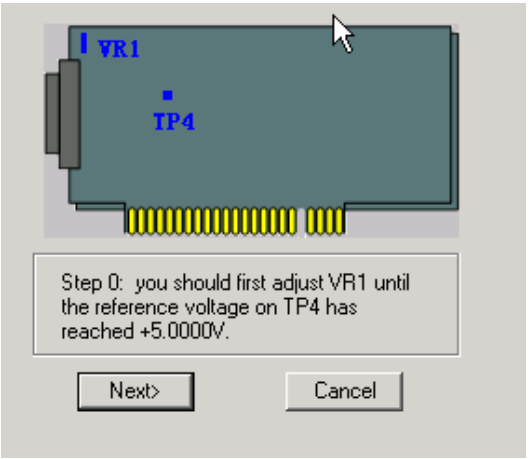
1. Step 0: Open Advantech Device Manager, select PCI-1741U card.



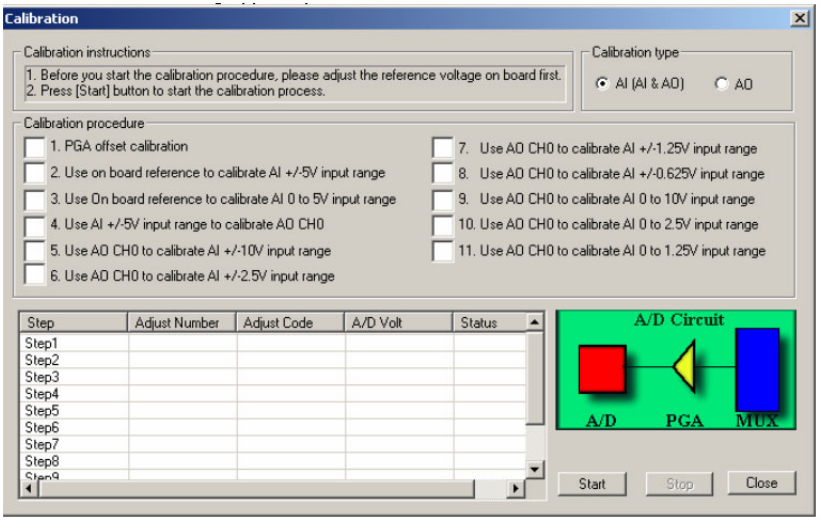
2. Click the “Setup” button, open PCI-1741U device setting dialog:



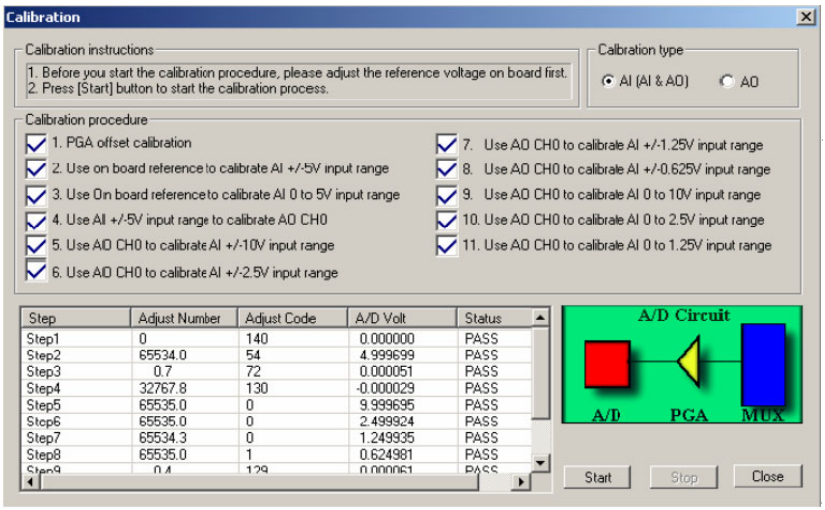
3. Click the "Calibration"button, start calibration procedure:



4. After VR1 is adjusted and the reference voltage on TP4 has reached +5.0000V, click "Next>" to view the calibration dialog box.



5. Select Calibration type AI or AO. AI calibration: this card calibration needs 11 procedures (1-11). AO calibration: this card calibration needs 4 procedures (1-4).
6. Click “Start” to start calibration. The first calibration procedure is enabled.
7. Auto-calibration is finished.



APPENDIX A

Specifications

Appendix A Specifications

A.1 Analog Input

Channels	16 single-ended or 8 differential					
Resolution	16-bit					
FIFO Size	1K samples					
Max. Sampling Rate	200 kS/s					
Input range and Gain List	Gain	0.5	1	2	4	8
	Unipolar	N/A	0~10	0~5	0~2.5	0~1.25
	Bipolar	±10	±5	±2.5	±1.25	±0.625
Bandwidth for PGA (MHz)	Gain	0.5	1	2	4	8
	Bandwidth	4.0	4.0	2.0	1.5	0.65
Common mode voltage	±11 V max. (operational)					
Max. Input voltage	±20 V (protection)					
Input Impedance	100 MΩ/10pF(Off); 100 MΩ/100pF(On)					
Accuracy	DC	DNLE: ±1LSB				
		INLE: ±1LSB				
		Zero (Offset) error: Adjustable to ±1 LSB				
		Gain	0.5	1	2	4	8
		Gain error(% FSR)	0.03	0.02	0.02	0.03	0.04
	AC	THD: -90 dB					
		ENOB: 13.5 bits					
Clocking and Trigger Inputs	Trigger Mode	Software, on-board programmable pacer or external					
	A/D pacer clock	200 kHz (max.); 2.328mHz (min.)					

Note *The sampling rate depends on the computer hardware architecture and software environment. The rates may vary due to programming language, code efficiency, CPU utilization and so on.*

A.2 Analog Output

Channels	1	
Resolution	16-bit	
Operation mode	Single output	
Throughput	200 kS/s max. per channel (FSR)	
Output Range (Internal & External Reference)	Using Internal Reference	0~+5V,0~+10 V, -5~+5V,-10~+10V
	Using External Reference	0 ~ +x V@ +x V (-10=< x =<10) -x ~ +x V@ +x V (-10=< x =<10)
Accuracy	DC	DNLE: ±1LSB (monotonic)
		INLE: ±1LSB
		Zero (Offset) error: Adjustable to ±1 LSB
		Gain (Full-scale) error: Adjustable to ±1 LSB
Dynamic Performance	Settling Time	5μs (to 4 LSB of FSR)
	Slew Rate	20 V/μs
Drift	10 ppm/°C	
Driving Capability	±20mA	
Output Impedance	0.1 Ω max.	

A.3 Digital Input/Output

Input channels	16	
Output channels	16	
Number of ports	2	
Input voltage	Low	0.8 V max.
	High	2.0 V min.
Output voltage	Low	0.5 V max. @ +24 mA (sink)
	High	2.4 V min. @ -15 mA (source)

A.4 Counter/Timer

Counter chip	82C54 or equivalent
Channels	3 channels, 2 channels are permanently configured as programmable pacers; 1 channel is free for user application
Counter 0	16-bit counter
Counter 1, 2	Cascade as a 32-bit clock divider for pacer clock for A/D conversion
Resolution	16-bit
Base Clock	Channel 1: 10MHz Channel 2: Takes input from output of channel 1 Channel 0: Internal 100 kHz or external (Max.10 MHz)
Compatibility	TTL level
Clock Input	Low0.8 V max.
	High2.0 V min.
Gate Input	Low0.8 V max.
	High2.0 V min.
Counter Output	Low0.5 V max.@+24 mA (sink)
	High2.4 V min.@-15 mA (source)

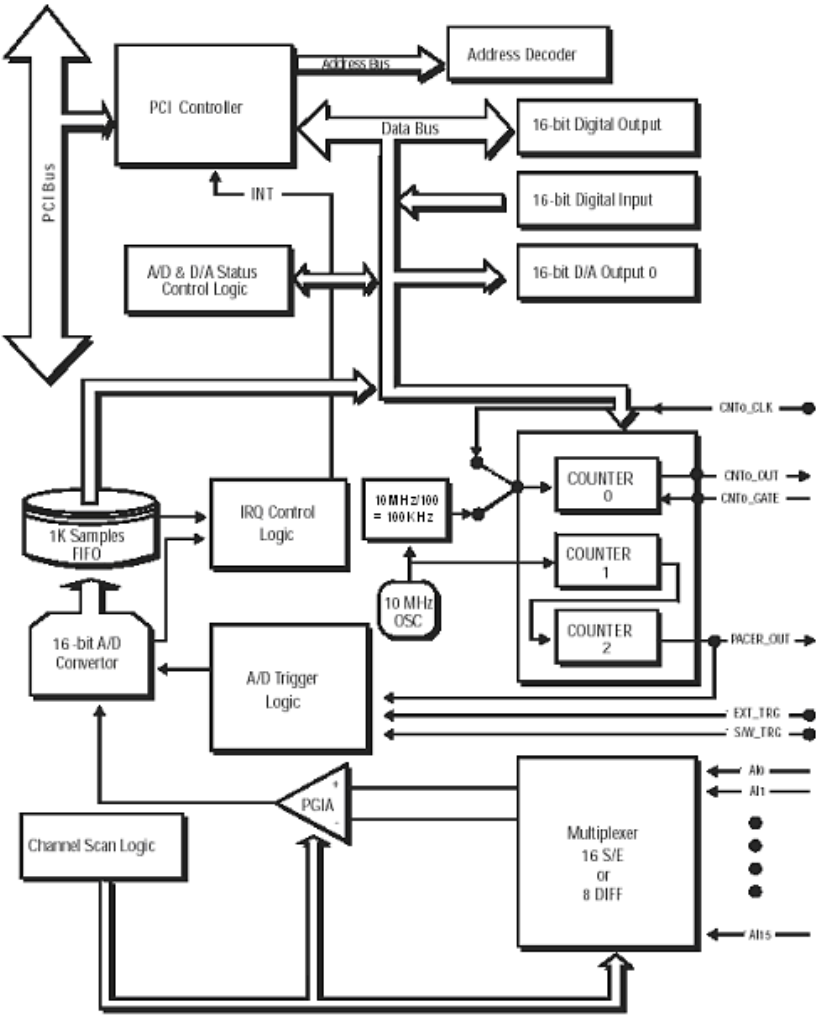
A.5 General

I/O Connector Type	68-pin SCSI-II female	
Dimensions	175 x 100 mm (6.9" x 3.9")	
Power Consumption	Typical	+5 V @ 850 mA +12 V @ 600 mA
	Max.	+5 V @ 1 A +12 V @ 700 mA
Temperature	Operation	0~+60°C(32~158°F) (refer to *IEC 68-2-1,2)*
	Storage	-20~+85°C(-4~185°F)
Relative Humidity	5~95%RH non-condensing (refer to *IEC 68-2-3*)	
Certification	CE certified	

APPENDIX B

Block Diagram

Appendix B Block Diagram



APPENDIX C

Register Structure and Format

Appendix C Register Structure & Format

C.1 Overview

PCI-1741U is delivered with an easy-to-use 32-bit DLL driver for user programming under the *Windows 98/2000 or XP* operating systems. We advise users to program the PCI-1741U using the 32-bit DLL driver provided by Advantech to avoid the complexity of low-level programming by register.

The most important consideration in programming the PCI-1741U at the register level is to understand the function of the card's registers. The information in the following sections is provided only for users who would like to do their own low-level programming.

C.2 I/O Port Address Map

PCI-1741U requires 32 consecutive addresses in the PC's I/O space. The address of each register is specified as an offset from the card's base address. For example, BASE+0 is the card's base address and BASE+7 is the base address plus seven bytes.

Table C-1 shows the function of each register of PCI-1741U or driver and its address relative to the card's base address.

Table C.1: PCI-1741U Register Format (Part 1)								
Base Address +decimal	Read							
	7	6	5	4	3	2	1	0
	A/D Data							
1	AD15	AD14	AD13	AD12	AD11	AD10	AD9	AD8
0	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0
	N/A							
3								
2								
	N/A							
5								
4								
	A/D Status Register							
7	CAL				IRQ	F/F	F/H	F/E
6	AD16/ 12	CNT0	ONE/ FH	IRQE N	GATE	EXT	PACE R	SW
	N/A							
9								
8								
	D/A channel 0 data							
11	DA15	DA14	DA13	DA12	DA11	DA10	DA9	DA8
10	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0
13								
12								
	N/A							
15								
14								

Table C.2: PCI-1741U Register Format (Part 2)								
Base Address +decimal	Read							
	7	6	5	4	3	2	1	0
	Digital Input							
17	DI15	DI14	DI13	DI12	DI11	DI10	DI9	DI8
16	DI7	DI6	DI5	DI4	DI3	DI2	DI1	DI0
	Calibration Command busy flag							
19								
18	BUSY							
	Board ID							
21								
20					BD3	BD2	BD1	BD0
	N/A							
23								
22								
	Counter 0							
25								
24	D7	D6	D5	D4	D3	D2	D1	D0
	Counter 1							
27								
26	D7	D6	D5	D4	D3	D2	D1	D0
	Counter 2							
29								
28	D7	D6	D5	D4	D3	D2	D1	D0
	N/A							
31								
30								

Table C.3: PCI-1741U Register Format (Part 1)								
Base Address +decimal	Write							
	7	6	5	4	3	2	1	0
	Software A/D Trigger							
1								
0								
	A/D Channel Range Setting							
3								
2			S/D	B/U		G2	G1	G0
	Multiplexer Control							
5					Stop channel			
4					Start channel			
	A/D Control Register							
7	CAL							
6	AD16 /12	CNT0	ONE/ FH	IRQE N	GATE	EXT0	PACE R	SW
	Clear Interrupt and FIFO							
9	Clear FIFO							
8	Clear interrupt							
	D/A Output Channel 0							
11	DA15	DA14	DA13	DA12	DA11	DA10	DA9	DA8
10	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0
13								
12								
	D/A Control Register							
15								
14					DA0_ LDEN	DA0_ I/E	DA0_ B/U	DA0_ 5/10

Table C.4: PCI-1741U Register Format (Part 4)								
Base Address +decimal	Write							
	7	6	5	4	3	2	1	0
	Digital Output							
17	DO15	DO14	DO13	DO12	DO11	DO10	DOI9	DO8
16	DO7	DO6	DO5	DO4	DO3	DO2	DO1	DO0
	Calibration Register							
19					CM3	CM2	CM1	CM0
18	D7	D6	D5	D4	D3	D2	D1	D0
	N/A							
21								
20								
	N/A							
23								
22								
	Counter 0							
25								
24	D7	D6	D5	D4	D3	D2	D1	D0
	Counter 1							
27								
26	D7	D6	D5	D4	D3	D2	D1	D0
	Counter 2							
29								
28	D7	D6	D5	D4	D3	D2	D1	D0
	Counter Control							
31								
30	D7	D6	D5	D4	D3	D2	D1	D0

C.3 Ch. Number and A/D Data — BASE+0 and BASE+1

C.3.1 BASE+0 and BASE+1 hold the result of A/D conversion data.

The 16 bits of data from the A/D conversion are stored in BASE+1 bit 7 to bit 0 and BASE+0 bit 7 to bit 0.

Table C.5: PCI-1741U Register for A/D Data

Read	**A/D Data**							
Bit #	7	6	5	4	3	2	1	0
BASE + 1	AD15	AD14	AD13	AD12	AD11	AD10	AD9	AD8
BASE + 0	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0

AD15 ~ AD0 Result of A/D Conversion

AD0 The least significant bit (LSB) of A/D data

AD15 The most significant bit (MSB) of A/D data

C.4 Software A/D Trigger — BASE+0

You can trigger an A/D conversion by software, the card's on-board pacer or an external pulse. BASE+6, Bit 2 to bit 0, select the trigger source.

(see *Section C.7, Control Register -- BASE+6*)

If you select software triggering, a write to the register BASE+0 with any value will trigger an A/D conversion.

C.5 A/D Channel Range Setting — BASE+2

All A/D channels have the same input range, controlled by the gain code G2 to G0 and B/U. The B/U control the input range is bipolar or unipolar. Please refer to the input range Table C-x. The PCI-1741U could configure as 16 channels for single-ended analog input or 8 channels for differential by write the S/D bit.

Table C.6: Register for A/D Channel Range Setting

Write	**A/D Channel Range Setting**							
Bit #	7	6	5	4	3	2	1	0
BASE + 2			S/D	B/U		G2	G1	G0

S/D Single-ended or Differential control bit

0 single-ended

1 differential.

B/U Bipolar or Unipolar control bit

0 bipolar

1 unipolar.

G2 to G0 Gain Code

Table C.7: Gain Codes for PCI-1741U

Gain	Input Range(V)	B/U	Gain Code		
			G2	G1	G0
1	-5 to +5	0	0	0	0
2	-2.5 to +2.5	0	0	0	1
4	-1.25 to +1.25	0	0	1	0
8	-0.625 to +0.625	0	0	1	1
0.5	-10 to +10	0	1	0	0
	N/A	0	1	0	1
	N/A	0	1	1	0
	N/A	0	1	1	1
1	0 to 10	1	0	0	0
2	0 to 5	1	0	0	1
4	0 to 2.5	1	0	1	0
8	0 to 1.25	1	0	1	1
	N/A	1	1	0	0
	N/A	1	1	0	1
	N/A	1	1	1	0
	N/A	1	1	1	1

C.6 MUX Control — BASE+4 and BASE+5

Table C.8: Register for Multiplexer Control								
Write	**Multiplexer Control**							
Bit #	7	6	5	4	3	2	1	0
BASE + 5					STO3	STO2	STO1	STO0
BASE + 4					STA3	STA2	STA1	STA0

STA3 ~ STA0 Start Scan Channel Number

STO3 ~ STO0 Stop Scan Channel Number

The write-only registers of BASE +4 and BASE+5 control how the multiplexers (Multiplexer) scan.

- BASE+4 bit 3 to bit 0, STA3 ~ STA0, hold the start scan channel number.
- BASE+5 bit 3 to bit 0, STO3 ~ STO0, hold the stop scan channel number.

Writing to these two registers automatically initializes the scan range of the Multiplexer. Each A/D conversion trigger also sets the Multiplexer to the next channel. With continuous triggering, the Multiplexer will scan from the start channel to the stop channel and then repeat. For differential mode, STO0 and STA0 are not available, i.e. only even channels are considered. The following examples show the scan sequences of the Multiplexer.

Example 1

If the PCI-1741U is configured as single-ended and the start scan input channel is AI3 and the stop scan input channel is AI7, then the scan sequence is AI3, AI4, AI5, AI6, AI7, AI3, AI4, AI5, AI6, AI7, AI3, AI4…

Example 2

If PCI-1741U is configured as single-ended and the start scan channel is AI13 and the stop scan channel is AI2, then the scan sequence is AI13, AI14, AI15, AI0, AI1, AI2, AI13, AI14, AI15, AI0, AI1, AI2, AI13, AI14...

Example 3

If PCI-1741U is configured as differential and the start scan channel is AI2 and the stop scan channel is AI6, then the scan sequence is AI2, AI4, AI6, AI2, AI4, AI6...

Example 4

If the PCI-1741U is configured as differential and the start scan channel is AI4 and the stop scan channel is AI2, then the scan sequence is AI4, AI6, AI8, AI10, AI12, AI14, AI0, AI2, AI4, AI6...

C.7 Control Register — BASE+6

The write-only register BASE+6 and BASE+7 allows users to set an A/D trigger source and an interrupt source.

Table C.9: Control Register

Write	**A/D Status Register**							
Bit #	7	6	5	4	3	2	1	0
BASE + 7	CAL							
BASE + 6	AD16/12	CNT0	ONE/FH	IRQEN	GATE	EXT	PACER	SW

SW Software trigger enable bit

1 enable

0 disable.

PACER Pacer trigger enable bit

1 enable

0 disable.

EXT External trigger enable bit

1 enable;

0 disable.

*Note: Users should **not** enable SW, PACER and EXT concurrently.*

GATE External trigger gate function enable bit.

0 Disable 1 Enable

IRQEN Interrupt enable bit.

0 Disable 1 Enable

ONE/FH Interrupt source bit

0 Interrupt when an A/D conversion occurs

1 Interrupt when the FIFO is half full.

CNT0 Counter 0 clock source select bit

0 The clock source of Counter 0 comes from the internal clock 1 MHz

1 The clock source of Counter 0 comes from the external clock maximum up to 10 MHz

AD16/12 Analog Input resolution selection bit

0 16 bit

1 12 bit

CAL Analog I/O calibration bit

0 Normal mode

All analog input and outputs channels are connected to 68 pin SCSI-II connector respectively.

1 A/D and D/A calibration mode

The wiring becomes that AI0 is connected to 0 V (AGND), AI2 is connected to +5 V reference, AI4 is connected to AO0 automatically.

C.8 Status Register — BASE+6 and BASE+7

The registers of BASE+6 and BASE+7 provide information for A/D configuration and operation.

Table C.10: Status Register

Write	**A/D Control Register**							
Bit #	7	6	5	4	3	2	1	0
BASE + 7	CAL				IRQ	F/F	F/H	F/E
BASE + 6	AD16/ 12	CN T0	ONE/ FH	IRQE N	GATE	EXT	PACE R	SW

The content of the status register of BASE+6 is the same as that of the control register except the F/E, F/H, F/F, and IRQ as below.

F/E FIFO Empty flag

This bit indicates whether the FIFO is empty.

1 means that the FIFO is empty.

F/H FIFO Half-full flag

This bit indicates whether the FIFO is half-full.

1 means that the FIFO is half-full.

F/F FIFO Full flag

This bit indicates whether the FIFO is full.

1 means that the FIFO is full.

IRQ Interrupt flag

This bit indicates the interrupt status.

1 means that an interrupt has occurred.

C.9 Clear Interrupt and FIFO — BASE+8 and BASE+9

Writing data to either of these two bytes clears the interrupt or the FIFO.

Table C.11: Register to Clear Interrupt and FIFO

Write	**Clear Interrupt and FIFO**							
Bit #	7	6	5	4	3	2	1	0
BASE + 9	Clear FIFO							
BASE + 8	Clear Interrupt							

C.10 D/A Output Channel 0 — BASE+10 and BASE+11

The PCI-1741U provides the innovative design as *gate control* for Analog Output function. It works as general Analog Output function when you disable the flag (bit 3 (DA0_LDEN) of BASE+14). That means the data will be output immediately. However, when you enable the flag, you need to read these two registers BASE+10 and BASE+11 to output the data to the Analog Output channel.

Table C.12: Register for Load D/A Channel 0 Data

Read	**Load D/A Channel 0 data**							
Bit #	7	6	5	4	3	2	1	0
BASE + 11								
BASE + 10								

C.11 D/A Output Channel 0 — BASE+10 and BASE+11

The write-only registers of BASE+10 and BASE+11 accept data for D/A Channel 0 output.

Table C.13: Register for D/A channel 0 data

Write	D/A Output Channel 0							
Bit #	7	6	5	4	3	2	1	0
BASE + 11	DA15	DA14	DA13	DA12	DA11	DA10	DA9	DA8
BASE + 10	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0

DA15 ~ DA0 Digital to analog data

DA0 LSB of the D/A data

DA15 MSB of the D/A data

C.12 D/A Reference Control —BASE+14

The write-only register of BASE+14 allows users to set the D/A reference source.

Table C.14: PCI-1741U Register for D/A Reference Control

Write	D/A Output Channel 1							
Bit #	7	6	5	4	3	2	1	0
BASE + 15								
BASE + 14					DA0_LDEN	DA0/I/E	DA0_B/U	DA0_5/10

DA0_5/10 The internal reference voltage for the D/A output channel 0

0 - 5 V

1 - 10 V

DA0_B/U for D/A output channel 0

0 Bipolar

1 Unipolar

DA0_I/E Internal or external reference voltage for D/A output channel 0

0 Internal source

1 External source

DA0_LDEN for Gate Control of D/A output channel 0 (Please refer to C.10 and C.11)

0 Disable

1 Enable

C.13 Digital I/O Registers — BASE+16 and BASE+17

The PCI-1741U offers 16 digital input channels and 16 digital output channels. These I/O channels use the input and output ports at addresses BASE+16 and BASE+17.

Table C.15: Register for Digital Input

Read	**Digital Input**							
Bit #	7	6	5	4	3	2	1	0
BASE + 17	DI15	DI14	DI13	DI12	DI11	DI10	DI9	DI8
BASE + 16	DI7	DI6	DI5	DI4	DI3	DI2	DI1	DI0

Table C.16: Register for Digital Output

Write	**Digital Output**							
Bit #	7	6	5	4	3	2	1	0
BASE + 17	DO15	DO14	DO13	DO12	DO11	DO10	DO9	DO8
BASE + 16	DO7	DO6	DO5	DO4	DO3	DO2	DO1	DO0

Note *The default configuration of the digital output channels is logic 0. This avoids damaging external devices during system start-up or reset since the power on status is set to the default value.*

C.14 Calibration Busy flag — BASE+18

Table C.17: Calibration Busy Check Flag								
Read	**Calibration Command and Data**							
Bit #	7	6	5	4	3	2	1	0
BASE + 18	BUSY							

BUSY Calibration busy check flag, 1 means busy, 0 means not busy.

When busy is 1, program should wait until busy is 0 to write next command.

C.15 Calibration Register — BASE+18 and BASE+19

The PCI-1741U offers Calibration registers BASE+18 and BASE+19 for user to calibrate the A/D and D/A.

Table C.18: Calibration Command and Data Register								
Write	**Calibration Command and Data**							
Bit #	7	6	5	4	3	2	1	0
BASE + 19					CM3	CM2	CM1	CM0
BASE + 18	D7	D6	D5	D4	D3	D2	D1	D0

D7 to D0 Calibration data register

D0 LSB of the calibration data

D7 MSB of the calibration data

The calibration data register set the value from 0 to 255 providing 256 steps variation for calibration command.

CM3 to CM0 Calibration command register

CM0 LSB of the calibration command

CM3 MSB of the calibration command

Table C.19 lists the command codes for PCI-1741U

Table C.19: Calibration Command Codes

Meaning	Command Code			
	CM3	CM2	CM1	CM0
A/D bipolar offset adjust	0	0	0	0
A/D unipolar offset adjust	0	0	0	1
PGA offset adjust	0	0	1	0
A/D gain adjust	0	0	1	1
D/A 0 gain 1 adjust (10 V)	0	1	0	0
D/A 0 gain 2 adjust (5 V)	0	1	0	1
D/A 0 bipolar offset adjust	0	1	1	0
D/A 0 unipolar offset adjust	0	1	1	1

Note *Users have to follow the calibration procedure to calibrate the PCI-1741U.*

C.16 BoardID Registers — BASE+20

PCI-1741U offers a BoardID register at BASE+20. With correct Board ID settings, user can easily identify and access each card during hardware configuration and software programming.

Table C.20: Register for BoardID

Read	**Board ID**							
Bit #	7	6	5	4	3	2	1	0
BASE + 20					BD3	BD2	BD1	BD0

C.17 Programmable Timer/Counter Registers BASE+24, BASE+26, BASE+28 and BASE+30

The four registers of BASE+24, BASE+26, BASE+28 and BASE+30 are used for the 82C54 programmable timer/counter. Please refer to *Appendix D 82C54 Counter Chip Functions* for detailed application information.

Note *Users have to use a 16-bit (word) command to read/ write each register.*

APPENDIX D

82C54 Counter Chip Function

Appendix D 82C54 Counter Chip Function

D.1 The Intel 82C54

The PCI-1741U uses one Intel 82C54-compatible programmable interval timer/counter chip. The popular 82C54 chip offers three independent 16-bit counters, counter 0, counter 1 and counter 2. You can program each counter for maximum count values from 2 to 65535.

The 82C54 has a maximum input clock frequency of 10 MHz. The PCI-1741U provides 10 MHz input frequencies to the counter chip from an on-board crystal oscillator.

Counter 0

On the PCI-1741U, counter 0 can be a 16-bit timer or an event counter as chosen by the user. When the clock source is set as an internal source, counter 0 is a 16-bit timer.

Counter 1 & 2

Counter 1 and counter 2 of the counter chip are cascaded to create a 32-bit timer for the pacer trigger. A low-to-high edge of counter 2 output will trigger an A/D conversion.

D.2 Counter Read/Write and Control Registers

The 82C54 programmable interval timer uses four registers at addresses BASE + 24(Dec), BASE + 26(Dec), BASE + 28(Dec) and BASE + 30(Dec) for read, write and control of counter functions. Register functions appear below:

Table D.1: Counter registers	
Register	Function
BASE + 24(Dec)	Counter 0 read/write
BASE + 26(Dec	Counter 1 read/write
BASE + 28(Dec)	Counter 2 read/write
BASE + 30(Dec)	Counter control word

Since the 82C54 counter uses a 16-bit structure, each section of read/write data is split into a least significant byte (LSB) and most significant byte (MSB). To avoid errors, it is important that you make read/write operations in pairs and keep track of the byte order.

The data format for the control register is as below:

BASE + 30(Dec) 82C54 control								
Bit	D7	D6	D5	D4	D3	D2	D1	D0
Value	SC1	SC0	RW1	RW0	M2	M1	M0	BCD

Description

SC1 & SC0 Select counter		
Counter	SC1	SC0
0	0	0
1	0	1
2	1	0
Read-back command	1	1

RW1 & RW0 Select read / write operation		
Operation	RW1	RW0
counter Latch	0	0
Read/write LSB	0	1
Read/write MSB	1	0
Read/write LSB first, then MSB	1	1

M2, M1 & M0 Select operating mode				
M2	M1	M0	Mode	Description
0	0	0	0	Stop on terminal count
0	0	1	1	Programmable one shot
X	1	0	2	Rate generator
X	1	1	3	Square wave rate generator
1	0	0	4	Software triggered strobe
1	0	1	5	Hardware triggered strobe

BCD Select binary or BCD counting	
BCD	Type
0	Binary counting 16-bits
1	Binary coded decimal (BCD) counting

If you set the module for binary counting, the count can be any number from 0 up to 65535. If you set it for BCD (Binary Coded Decimal) counting, the count can be any number from 0 to 9999.

If you set both SC1 and SC0 bits to 1, the counter control register is in read-back command mode. The control register data format then becomes:

BASE + 30(Dec) 82C54 control, read-back mode								
Bit	D7	D6	D5	D4	D3	D2	D1	D0
Value	1	1	CNT	STA	C2	C1	C0	X

CNT = 0 Latch count of selected counter(s)

STA = 0 Latch status of selected counter(s)

C2, C1 & C0 Select counter for a read-back operation

C2 = 1 select Counter 2

C1 = 1 select Counter 1

C0 = 1 select Counter 0

If you set both SC1 and SC0 to 1 and STA to 0, the register selected by C2 to C0 contains a byte which shows the status of the counter. The data format of the counter read/write register then becomes:

BASE +24/26/28(Dec) Status read-back mode								
Bit	D7	D6	D5	D4	D3	D2	D1	D0
Value	OUT	NC	RW1	RW2	M2	M1	M0	BCD

OUT Current state of counter output

NC Null count is 1 when the last count written to the counter register has been loaded into the counting element

D.3 Counter Operating Modes

The 82C54 programmable interval timer uses four registers at addresses BASE + 24(Dec), BASE + 26(Dec), BASE + 28(Dec) and BASE + 30(Dec) for read, write and control of counter functions. Register functions appear below:

MODE 0 - Stop on Terminal Count

The output will initially below after you set operation to this mode. After you load the count into the selected count register, the output will remain low and the counter will count. When the counter reaches the terminal count, its output will go high and remain high until you reload it with the mode or a new count value. The counter continues to decrement after it reaches the terminal count. Rewriting a counter register during counting has the following results:

1. Writing to the first byte stops the current counting.
2. Writing to the second byte starts the new count.

MODE 1 - Programmable One-shot Pulse

The output is initially high. The output will go low on the count following the rising edge of the gate input. It will then go high on the terminal count. If you load a new count value while the output is low, the new value will not affect the duration of the one-shot pulse until the succeeding trigger. You can read the current count at any time without affecting the one-shot pulse. The one-shot is re-triggerable, thus the output will remain low for the full count after any rising edge at the gate input.

MODE 2 - Rate Generator

The output will be low for one period of the input clock. The period from one output pulse to the next equals the number of input counts in the counter register. If you reload the counter register between output pulses, the present period will not be affected, but the subsequent period will reflect the value.

The gate input, when low, will force the output high. When the gate input goes high, the counter will start from the initial count. You can thus use the gate input to synchronize the counter.

With this mode the output will remain high until you load the count register. You can also synchronize the output by software.

MODE 3 - Square Wave Generator

This mode is similar to Mode 2, except that the output will remain high until one half of the count has been completed (for even numbers), and will go low for the other half of the count. This is accomplished by decreasing the counter by two on the falling edge of each clock pulse. When the counter reaches the terminal count, the state of the output is changed, the counter is reloaded with the full count and the whole process is repeated.

If the count is odd and the output is high, the first clock pulse (after the count is loaded) decrements the count by 1. Subsequent clock pulses decrement the count by 2. After time-out, the output goes low and the full count is reloaded. The first clock pulse (following the reload) decrements the counter by 3. Subsequent clock pulses decrement the count by two until time-out, then the whole process is repeated. In this way, if the count is odd, the output will be high for (N+1)/2 counts and low for (N-1)/2 counts.

MODE 4 -Software-Triggered Strobe

After the mode is set, the output will be high. When the count is loaded, the counter will begin counting. On terminal count, the output will go low for one input clock period then go high again. If you reload the count register during counting, the new count will be loaded on the next CLK pulse. The count will be inhibited while the GATE input is low.

MODE 5 - Hardware-Triggered Strobe

The counter will start counting after the rising edge of the trigger input and will go low for one clock period when the terminal count is reached. The counter is re-triggerable.