PCI-1721

12-bit, 4-channel
Advanced Analog
Output Card

CE notification

The PCI-1721, developed by ADVANTECH CO., LTD., has passed the CE test for environmental specifications when shielded cables are used for external wiring. We recommend the use of shielded cables. This kind of cable is available from Advantech. Please contact your local supplier for ordering information.

On-line Technical Support

For technical support and service, please visit our support website at:

http:\\www.advantech.com/support

Note:

✎ Concerning the environment protection, we'd like to reduce the paper using for the user's manual. Starting the page of *Appendix C*, please find the PDF file of the CD-ROM.

Part No. 2003172101 2nd Edition
Printed in Taiwan October 2001

Contents

Chapter 1

1. Introduction

Thank you for buying the Advantech PCI-1721. The PCI-1721 is an advanced high-speed analog output card for PCI bus, and each of analog output channels is equipped with a 12-bit, double-buffered DAC.

It features many powerful and unique functions, like waveform output function with 10 MHz maximum update rate, auto-calibration function and Board ID.

The PCI-1721 is an ideal solution for industrial applications where high-speed continuous analog output or even real-time waveform output functions are required.

The following sections of this chapter will provide further information about features of the multifunction cards, a quick start for installation, together with some brief information on software and accessories for the PCI-1721 card.

1.1 Features

- ❑ 10 MHz maximum digital update rate
- ❑ PCI-bus mastering for data transfer
- ❑ Auto calibration function
- ❑ 4 analog output channels with 1K FIFO
- ❑ A 12-bit DAC is equipped for each of analog output channels
- ❑ Real-time waveform output function with internal/external pacer
- ❑ Synchronized output function
- ❑ Flexible output types and output range settings
- ❑ Keeping the output settings and values after system reset
- ❑ 16-ch DIO and one 10MHz 16-bit resolution counter
- ❑ Board ID

The Advantech PCI-1721 offers the following main features:

PCI-Bus Mastering Data Transfer

The PCI-1721 supports *PCI-Bus mastering DMA* for high-speed data transfer and gap-free analog output. By setting aside a block of memory in the PC, the PCI-1721 performs bus-mastering data transfers without CPU intervention, setting the CPU free to perform other more urgent tasks such as data analysis and graphic manipulation. The function allows users to run all I/O functions simultaneously at full speed without losing data.

Auto-calibration Function

The PCI-1721 provides an auto-calibration function by using a calibration utility. The built-in calibration circuitry of the PCI-1721 corrects gain and offset errors in analog output channels thereby eliminating the need for external equipment and user adjustments.

Waveform Analog Output

The PCI-1721 provides four analog output channels. Both of them can perform continuous waveform output. The analog output can be up to 10MS/s for each analog output channel. Or you can load a cyclic waveform into an on-board FIFO, which will continuously output the cyclic waveform. The on-board FIFO of the PCI-1721 can store 2 to 1024 samples of the waveform.

Keeping the Output Settings and Values after system reset

Users can independently set the four outputs to different ranges: 0~+5V, 0~+10V, ±5V, ±10V, 0~20mA or 4~20mA, and all the ranges are software selectable. When the system is hot reset (power not shut down), the PCI-1721 can either retain the last analog output settings and values, or return to its default configuration, depending on jumper setting. This practical function eliminates danger caused by mis-operation during unexpected system reset.

On-board FIFO Memory

The PCI-1721 provides an *on-board FIFO* (First In First Out) memory buffer, storing up to 1K samples for D/A conversion.

Board ID

The PCI-1721 has a built-in DIP Switch that helps define each card's ID when multiple PCI-1721 cards have been installed on the same PC chassis. The board ID setting function is very useful when users build their system with multiple PCI-1721 cards. With correct Board ID settings, you can easily identify and access each card during hardware configuration and software programming.

On Board Programmable Timer/Counter

PCI-1721 provides a programmable timer counter for generating pacer trigger for the D/A conversion. The timer/counter chip is 82C54, which includes three 16-bit counters of 10 MHz clock. 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.

Note:

✎ For detailed specifications of the PCI-1721, please refer to *Appendix A, Specifications.*

1.2 Applications

- Process control
- Programmable voltage source
- Programmable current sink
- Servo control
- Multiple loop PID control
- Simulate function generator

1.3 Installation Guide

Before you install your PCI-1721 card, please make sure you have the following necessary components:

❑ **PCI-1721 DA&C card**

❑ **PCI-1721 User's Manual**

❑ **Driver software**	Advantech DLL drivers (included in the companion CD-ROM)
❑ **Wiring cable**	PCL-10168 (optional)
❑ **Wiring board**	ADAM-3968 (optional)
❑ **Computer**	Personal computer or workstation with a PCI-bus slot (running Windows 95/98/NT/ 2000)

Some other optional components are also available for enhanced operation:

❑ **Application software**	ActiveDAQ, GeniDAQ or other third-party software packages (the waveform output function of Analog Output is not included)

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. Figure 1-1 on the next page provides a concise flow chart to give users a broad picture of the software and hardware installation procedures:

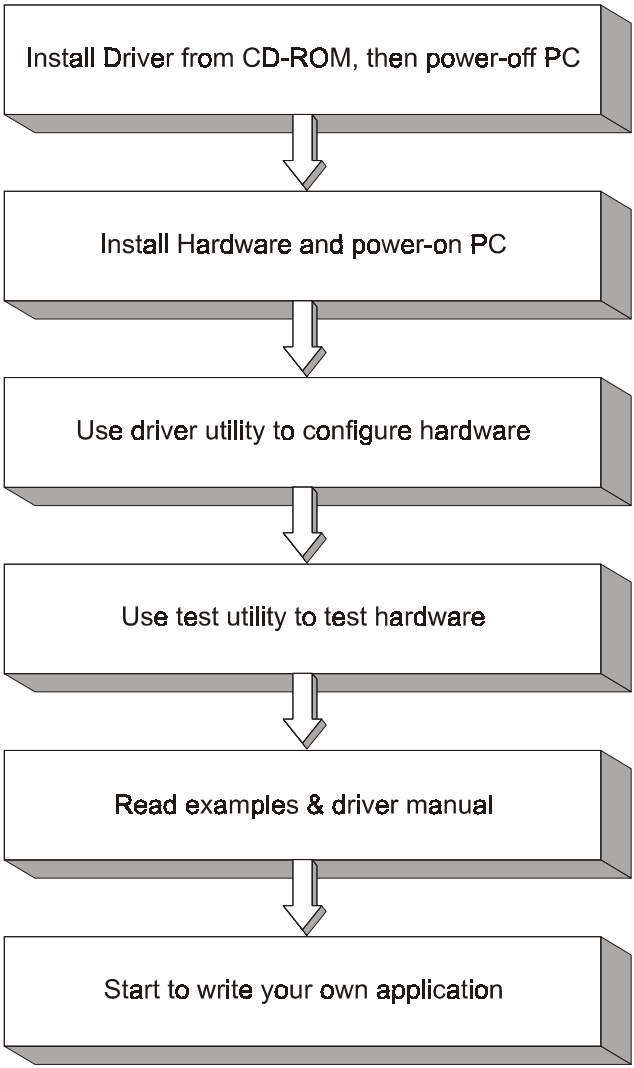


Fig. 1-1 Installation Flow Chart

1.4 Software Overview

Advantech offers a rich set of DLL drivers, third-party driver support and application software to help fully exploit the functions of your PCI-1721 card:

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

Note:

✎ The waveform output function of Analog Output is not included.

Programming choices for DA&C 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.

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 DA&C cards. Advantech's DLL driver features a complete I/O function library to help boost your application performance. The Advantech DLL driver for Windows 95/98/NT/2000 works seamlessly with development tools such as Visual C++, Visual Basic, Inprise C++ Builder and Inprise Delphi.

Register-level Programming

Register-level programming is reserved for experienced programmers who find it necessary to write codes 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.

1.5 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 Manual*. Moreover, a rich set of example source codes are also given for your reference.

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 works 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 get you very well-oriented.

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 Driver for 95 and 98 (or for NT/2000)/ Driver Manual

The example source codes could 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 95/NT/2000 Application with DLL Driver* chapter and the *Function Overview* chapter on the *DLL Drivers Manual*.

Programming with DLL Driver Function Library

Advanech 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, those APIs can be categorized into several function groups:

- **Analog Output Function Group**
- **Digital Input/Output Function Group**
- **Counter Function Group**
- **Port Function Group (direct I/O)**
- **Event Function Group**

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

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 Manaul* for a detailed listing of the Error Code, Error ID and the Error Message.

1.6 Accessories

Advantech offers a complete set of accessory products to support the PCI-1721 card. These accessories include:

Wiring Cable

- **PCL-10168** The PCL-10168 shielded cable is specially designed for PCI-1721 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.

Wiring Boards

- **ADAM-3968** 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-1721 card.

Chapter 2

2. Installation

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

2.1 Unpacking

After receiving your PCI-1721 package, please inspect its contents first. The package should contain the following items:

☑ PCI-1721 card

☑ Companion CD-ROM (DLL driver included)

☑ User's Manual

The PCI-1721 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 anti-static 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, first you should:

- 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 our 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 anti-static bag for future use. You might need the original bag to store the card if you have to remove the card from PC or transport it elsewhere.

2.2 Switch and Jumper Settings

The PCI-1721 card has one function switch and five jumper settings.

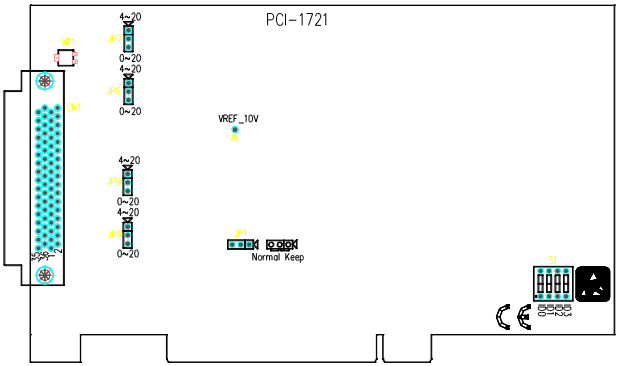


Fig. 2-1 Card connector, jumper and switch locations

Table 2-1: Summary of jumper settings

Names of Jumpers	Function description			
JP1		Keep last status after hot reset.		
		Default configuration.		
JP6~JP9		4~20 mA		0~20 mA

Setting the time to reset the analog outputs

Some users will want the capability of clearing each analog output when the system (or PC) issues a reset signal on the PCI bus. Some users will want to clear their analog output only as part of system power-on.

The PCI-1721 satisfies both these needs by providing jumper JP1. Depending on the application, this capability may allow analog outputs to be “ZERO” without requiring a complete shutdown of processes controlled by the card.

Complete loss of power to the chip clears the chip memory. Thus, no matter how JP1 is set, if the power to the PCI-1721 is disconnected, the analog output initial power-on state will be “ZERO”.

2.3 I/O Connector

Pin Assignment

Figure 2-2 shows the pin assignments for the 68-pin I/O connector on the PCI-1721.

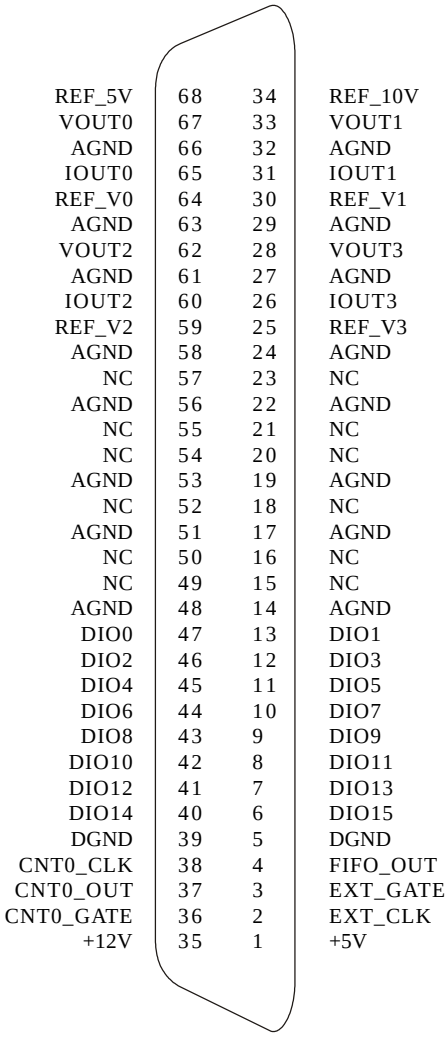


Fig. 2-2 I/O connector pin assignments for the PCI-1721

I/O Connector Signal Description

Table 2-2 I/O Connector Signal Description

Signal Name	Reference	Direction	Description
Vout<0..3>	AGND	Output	**Voltage Output, Channels 0 through 3.**
Iout<0..3>	AGND	Output	**Current Output, Channels 0 through 3.**
REF_V<0..3>	AGND	Input	**External Reference, Channel 0 through 3.**
AGND	-	-	**Analog Ground.** The two ground references (AGND and DGND) are connected together on the PCI-1721 card.
REF_10V	AGND	Output	**+10 VDC Reference.**
REF_5V	AGND	Output	**+5 VDC Reference.**
DI0<0..15>	DGND	-	**Digital Input / Output signals**. These pins are digital input / Output channel 0 to 15.
DGND	-	-	**Digital Ground.** The two ground references (AGND and DGND) are connected together on the PCI-1721 card.
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.**
FIFO_OUT	DGND	Output	**FIFO Output.** This pin pulses once for each FIFO clock when turned on. If D/A conversion is in the FIFO trigger mode, users can use this signal as a synchronous signal for other applications. A low-to-high edge triggers D/A conversion to start.
EXT_GATE	DGND	Input	**D/A External Trigger Gate**. When EXT_GATE is connected to +5 V, it will enable the external trigger signal to input. When EXT_GATE is connected to DGND, it will disable the external trigger signal to input.
EXT_CLK	DGND	Input	**D/A External Trigger Clock.** This pin is external trigger clock input for the D/A conversion. A low-to-high edge triggers D/A conversion to start.
+12V	DGND	Output	**+12 VDC Source** (from PCI bus directly with FUSE protection).
+5V	DGND	Output	**+5 VDC Source** (from PCI bus directly with FUSE protection).

Note: +12 V and +5 V can't be used as the reference voltage of Iout.

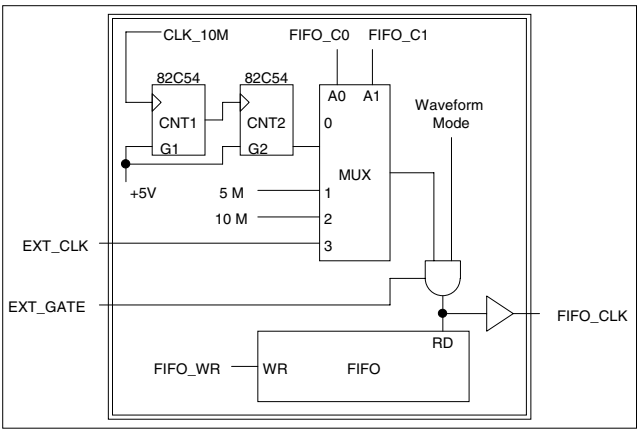


Fig. 2-3 FIFO block diagram of PCI-1721

2.4 Driver Installation

We recommend you to install the driver before you install the PCI-1721 card into your system, since this will guarantee a smooth installation process.

The 32-bit DLL driver Setup program for the PCI-1721 card is included on the companion CD-ROM that is shipped with your DA&C 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 autoplay function enabled on your system. When the Setup Program is launched, you'll see the following 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.

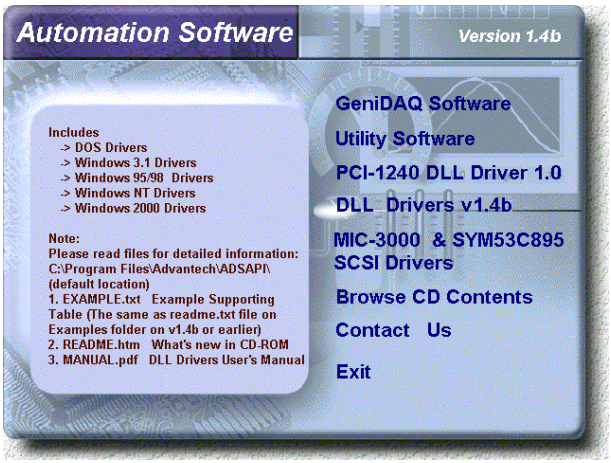


Fig. 2-4 The Setup Screen of Advantech Automation Software

Step 3: Select the *DLL Drivers* option.

Step 4: Select the *Windows 95/98 or Windows NT or Windows 2000* option according to your operating system. Just follow the installation instructions step by step to complete your DLL driver setup.

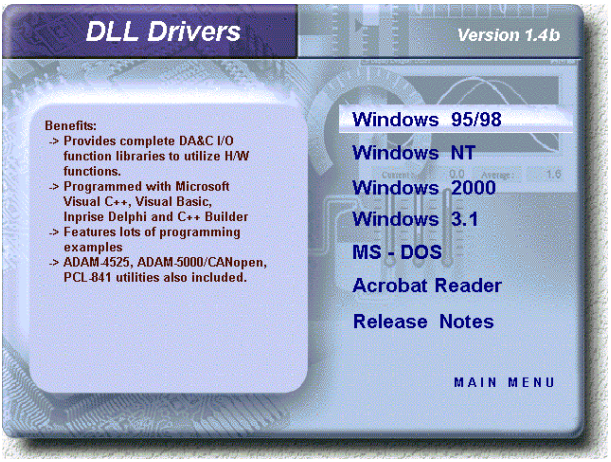


Fig. 2-5 Different options for Driver Setup

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

Start/Programs/Advantech Driver for 95 and 98 (or for NT/2000)/ Driver Manual

2.5 Hardware Installation

Note:

✎ Make sure you have installed the driver first before you install the card (please refer to *2.4 Driver Installation*)

After the DLL driver installation is completed, you can now go on to install the PCI-1721 card in any PCI slot on your computer. But it is suggested that you should refer to the computer user manual or related documentation if you have any doubt. 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 the static electricity that might be on your body.

Step 5: Insert the 1721 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, otherwise 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:

✎ In case you installed the card without installing the DLL driver first, Windows 95/98 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 *2.4 Driver Installation.*

After the PCI-1721 card is installed, you can verify whether it is properly installed on your system in the *Device Manager*:

1. Access the *Device Manager* through *Control Panel/System/Device Manager*.
2. The *device name* of the PCI-1721 should be listed on the *Device Manager* tab on the System *Property* Page.

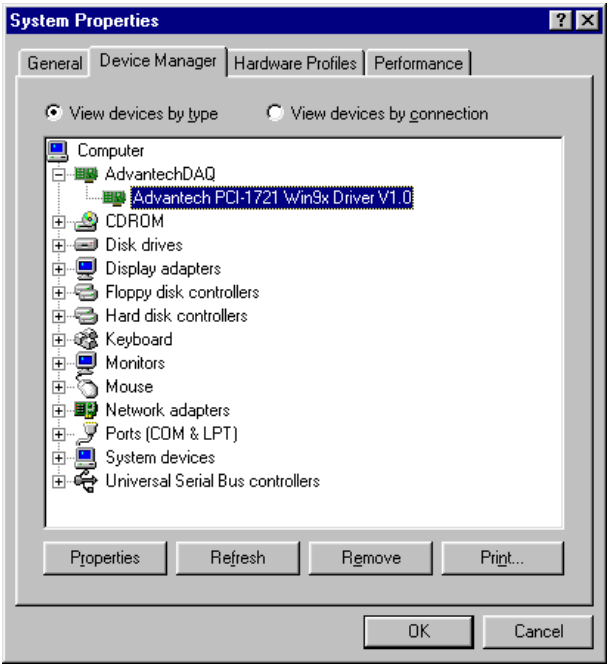


Fig. 2-6 The device name listed on the Device Manager

Note:

✎ If your card is properly installed, you should see the *device name* of your card listed on the *Device Manager* tab. **If you do see your device name listed on it 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 Installation* 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.

2.6 Device Setup & Configuration

The *Device Installation* 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 32-bit DLL drivers.

Setting 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 Driver for 95 and 98 (or for NT/2000)/Device Installation***).

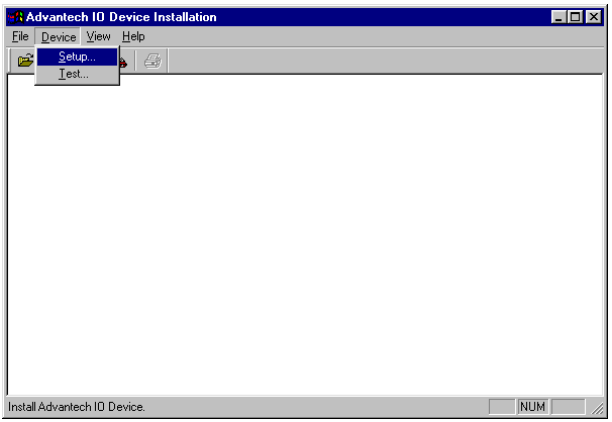


Fig. 2-7 The Advantech Device Installation utility program

Step 2: On the *Device Installation* program window, select the *Setup* menu item on the menu bar, and click the *Device* command (Fig. 2-7) to bring up the *I/O Device Installation* dialog box (Fig. 2-8). You can then view the device(s) already installed on your system (if any) on the *Installed Devices* list box. Since you haven't installed any device yet, you might see a blank list such as the one on the next page (Fig. 2-7).

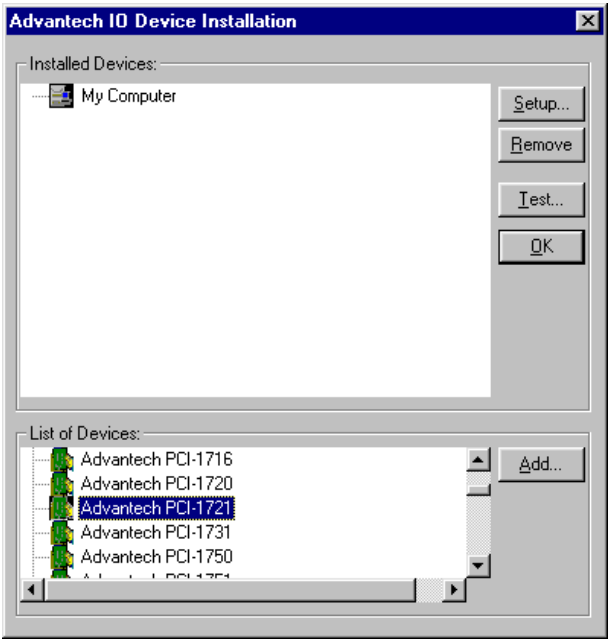


Fig. 2-8 The I/O Device Installation dialog box

Step 3: Scroll down the *List of Devices* box to find the device that you wish to install, then click the ***Add...*** button to evoke the *Device(s) found* dialog box such as one shown in Fig. 2-9. The *Device(s) found* dialog box lists all the installed devices on your system. Select the device you want to configure from the list box and press the ***OK*** button. After you have clicked ***OK***, you will see a *Device Setting* dialog box such as the one in Fig. 2-10.

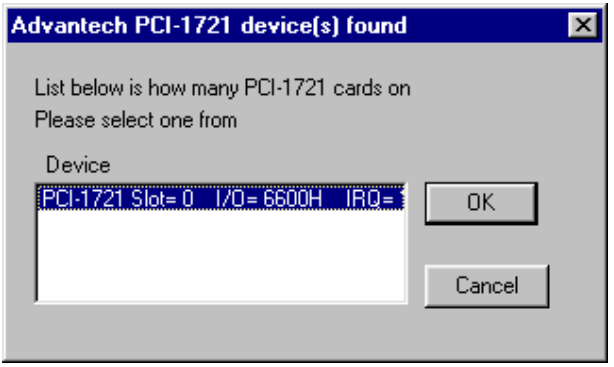


Fig. 2-9 The "Device(s) Found" dialog box

Configuring the Device

Step 4: On the *Device Setting* dialog box (Fig. 2-10), you can configure the voltage source either as *External or Internal*, and specify the voltage output range for the 4 D/A channels.

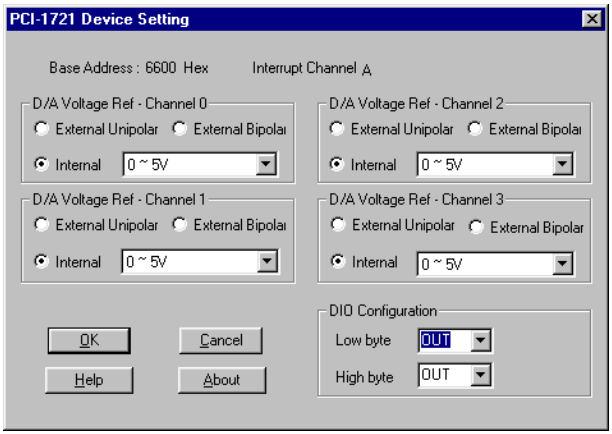


Fig. 2-10 The Device Setting dialog box

Note:

✎ Users 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*, users have six options for the output voltage ranges and current ranges: *0 ~ 5 V, 0 ~ 10 V, -5 ~ 5 V, -10 ~ 10 V, 0~20 mA* and *4~20 mA.*

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.

Step 5: After you have finished configuring the device, click ***OK*** and the *device name* will appear in the *Installed Devices* box as seen below:

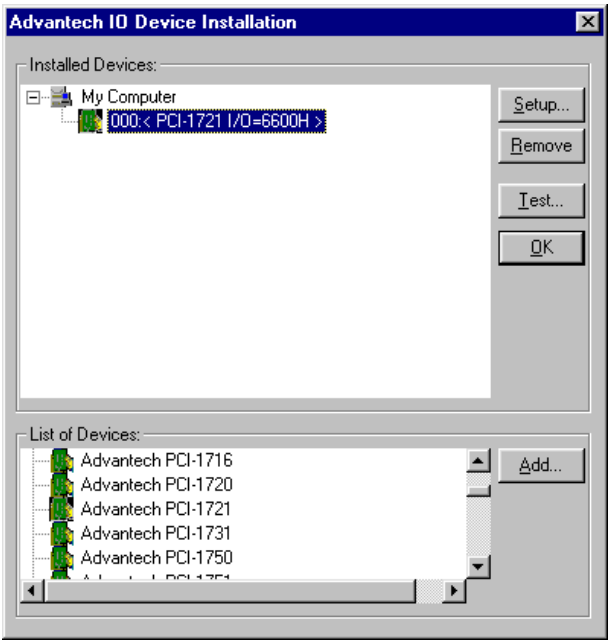


Fig. 2-11 The Device Name appearing on the list of devices box

Note:

✎ As we have noted, the *device name "000:PCI-1721 I/O=6600H"* begins with a *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.

After your card is properly installed and configured, you can click the ***Test...*** button to test your hardware by using the testing utility we supplied. For more detailed information, please refer to *Chapter 2* of the *DLL Drivers Manual*.

You can also find the rich examples on the CD-ROM to speeding up your programming.

Chapter 3

3. Signal Connections

3.1 Overview

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 the PCI-1721 via the I/O connector.

3.2 Analog Output Connections

The PCI-1721 provides four D/A output channels, *Vout_0 ~ Vout_3* and *Iout_0 ~ Iout_3*. Users may use the PCI-1721 internally-provided precision -5V (-10V) reference to generate 0 to +5 V (+10 V) D/A output range. Users also may create D/A output range through external references, *REF_V0 ~ REF_V3*. 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.

Figure 3-1 shows how to make analog output and external reference input connections on the PCI-1721.

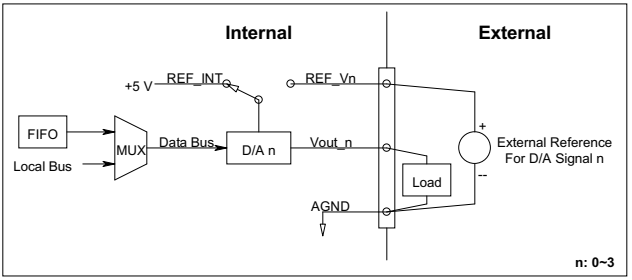


Figure 3-1: Analog output connections

3.3 Trigger Source Connections

Internal Pacer Trigger Connection

The PCI-1721 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 outputing 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_OUT)* will trigger an A/D conversion on the PCI-1721. At the same time, you can also use this signal as a synchronous signal for other applications.

External Trigger Source Connection

In addition to pacer triggering, the PCI-1721 also allows external triggering for A/D conversions. When a +5 V source is connected to *TRG_GATE*, the external trigger function is enabled. A low-to-high edge coming from *EXT_TRG* will trigger an A/D conversion on the PCI-1721. When *DGND* is connected to *TRG_GATE*, the external trigger function is thereby disabled.

3.4 Field Wiring Considerations

When you use the PCI-1721 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-1721.

- 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.

Appendix A

Appendix A. Specifications

Analog Output

Channels	4	
Resolution	12-bit	
FIFO Size	1K samples	
Operation mode	Single / Continuous / Waveform / Synchronized output	
Output Range (Internal & External Reference)	Using Internal Reference	0~+5V, 0~+10V, -5~+5V, -10~+10V, 0~20mA, 4~20mA
	Using External Reference	0 ~ +x V@ +x V (-10 ≤ x ≤ 10) -x ~ +x V@ +x V (-10 ≤ x ≤ 10)
Accuracy	Relative	±1 LSB
	Differential Non-linearity	±1 LSB (monotonic)
Offset	< 1 LSB	
Slew Rate	10 V/μs	
Driving Capability	±10 mA	
Output Impedance	0.1 Ω max.	
Max. Update Rate	10 MHz (max. for one channel)	
Settling Time	5μs (to ±1 LSB of FSR)	
External Clock Input	10 MHz Max.	
	Low	0.8 V max.
	High	2.0 V min.
External TTL Trigger Input	Low	0.8 V max.
	High	2.0 V min.

Digital Input /Output

Input Channels	16 (bi-directional)	
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	2.0 V min. @ -15 mA (source)

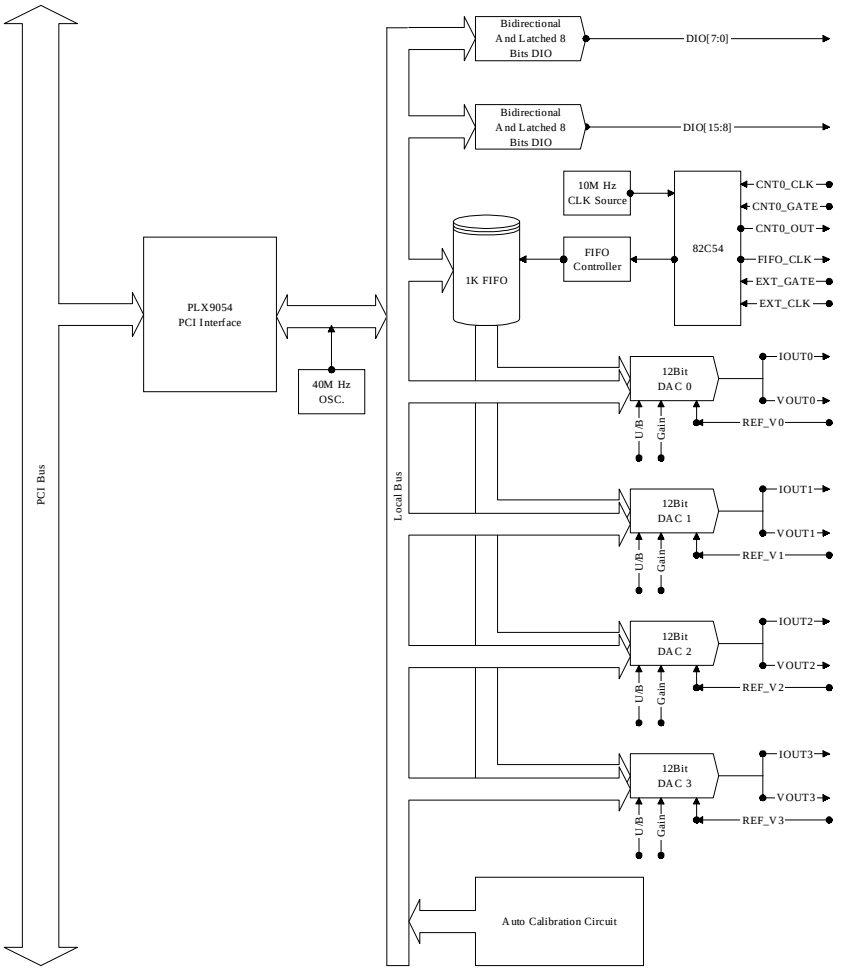
Counter/Timer

Channels	1	
Resolution	16-bit	
Compatibility	TTL level	
Base Clock	10 MHz	
Max. Input Frequency	10 MHz	
Clock Input	Low	0.8 V max.
	High	2.0 V min.
Gate Input	Low	0.8 V max.
	High	2.0 V min.
Counter Output	Low	0.5 V max. @ +24 mA
	High	2.4 V min. @ -15 mA

General

I/O Connector Type	68-pin SCSI-II female	
Dimensions	175 mm 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 (32~140° 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 C. Register Structure and Format

C.1 Overview

The PCI-1721 is delivered with an easy-to-use 32-bit DLL driver for user programming under the Windows 95/98/NT/2000 operating system. We advise users to program the PCI-1721sing 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-1721 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

The PCI-1721 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.

The table C-1 shows the function of each register of the PCI-1721 or driver and its address relative to the card's base address.

Table C-1 PCI-1721 register format (Part 1)

Base Address + HEX		PCI-1721 Register Format															
		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0H	W	D/A Channel 0 Data															
		X	X	X	X	DA11	DA10	DA9	DA8	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0
	R	N/A															
2H	W	D/A Channel 1 Data															
		X	X	X	X	DA11	DA10	DA9	DA8	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0
	R	N/A															
4H	W	D/A Channel 2 Data															
		X	X	X	X	DA11	DA10	DA9	DA8	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0
	R	N/A															
6H	W	D/A Channel 3 Data															
		X	X	X	X	DA11	DA10	DA9	DA8	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0
	R	N/A															
10H	W	N/A															
	R	Board ID															
		0	0	0	0	0	0	0	0	0	0	0	0	BD3	BD2	BD1	BD0
12H	W	D/A Control Command															
		VREF_3	VREF_2	VREF_1	VREF_0	MOD_3	MOD_2	MOD_1	MOD_0	UB_3	R_3	UB_2	R_2	UB_1	R_1	UB_0	R_0
	R	D/A Control Status															
		VREF_3	VREF_2	VREF_1	VREF_0	MOD_3	MOD_2	MOD_1	MOD_0	UB_3	R_3	UB_2	R_2	UB_1	R_1	UB_0	R_0
16H	W	Write Calibration Result to EEPROM															
		X	X	UB	R	X	SEL1	SEL0	P	D7	D6	D5	D4	D3	D2	D1	D0
	R	N/A															

Table C-1 PCI-1721 register format (Part 2)

Base Address + HEX		PCI-1721 Register Format															
		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
22H	W	All D/A channels Synchronized Setting Command															
		X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	SYNC
	R	All D/A channels Synchronized Setting Status															
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SYNC
24H	W	Synchronization Strobe															
		X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
	R	N/A															
26H	W	Calibration Setting Command															
		X	X	UB	R	X	SEL1	SEL0	P	D7	D6	D5	D4	D3	D2	D1	D0
	R	Calibration Setting Status															
		0	0	UB	R	0	SEL1	SEL0	P	D7	D6	D5	D4	D3	D2	D1	D0
28H	W	N/A															
	R	Calibration Status															
		0	0	0	0	0	0	0	0	0	0	0	0	0	CALI_ REQ	CALI_ INIT	CALI_ RDY
2AH	W	I/O Setting Command															
		X	X	X	X	X	X	X	X	X	FIFO_ C1	FIFO_ C0	HDIO	LDIO	X	GATE0	8254_ CNT0
	R	I/O Setting Status															
		0	0	0	0	0	0	0	0	0	FIFO_ C1	FIFO_ C0	HDIO	LDIO	0	GATE0	8254_ CNT0
2CH	W	Clear FIFO strobe															
		X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
	R	N/A															
2EH	W	N/A															
	R	FIFO Status															
		0	0	0	0	0	0	0	0	0	0	0	0	0	F/F	F/H	F/E

Table C-1 PCI-1721 register format (Part 3)

Base Address + HEX		PCI-1721 Register Format															
		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
30H	W	82C54 Counter 0 Command															
		X	X	X	X	X	X	X	X	D7	D6	D5	D4	D3	D2	D1	D0
	R	82C54 Counter 0 Status															
		0	0	0	0	0	0	0	0	D7	D6	D5	D4	D3	D2	D1	D0
32H	W	82C54 Counter 1 Command															
		X	X	X	X	X	X	X	X	D7	D6	D5	D4	D3	D2	D1	D0
	R	82C54 Counter 1 Status															
		0	0	0	0	0	0	0	0	D7	D6	D5	D4	D3	D2	D1	D0
34H	W	82C54 Counter 2 Command															
		X	X	X	X	X	X	X	X	D7	D6	D5	D4	D3	D2	D1	D0
	R	82C54 Counter 2 Status															
		0	0	0	0	0	0	0	0	D7	D6	D5	D4	D3	D2	D1	D0
36H	W	82C54 Control Command															
		X	X	X	X	X	X	X	X	D7	D6	D5	D4	D3	D2	D1	D0
	R	82C54 Control Status															
		0	0	0	0	0	0	0	0	D7	D6	D5	D4	D3	D2	D1	D0
3EH	W	DIO Write Strobe															
		D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
	R	DIO Read Strobe															
		D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
40H	W	DMA FIFO Data Buffer (lower word)															
		D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
	R	N/A															
42H	W	DMA FIFO Data Buffer (upper word)															
		D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
	R	N/A															

C.3 D/A Channel 0/1/2/3 Data - BASE+0/2/4/6H

Table C-2 PCI-1721 Register for D/A channel 0 data

Base Addr.		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0H	W	D/A Channel 0 Data															
		X	X	X	X	DA11	DA10	DA9	DA8	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0
2H	W	D/A Channel 1 Data															
		X	X	X	X	DA11	DA10	DA9	DA8	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0
4H	W	D/A Channel 2 Data															
		X	X	X	X	DA11	DA10	DA9	DA8	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0
6H	W	D/A Channel 3 Data															
		X	X	X	X	DA11	DA10	DA9	DA8	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0

DA11 ~ DA0 **D/A data**

DA0 the least significant bit (LSB) of the D/A data

DA11 the most significant bit (MSB)

C.4 Board ID - BASE+10H

The PCI-1721 offers Board ID register **BASE+10H**. With correct Board ID settings, user can easily identify and access each card during hardware configuration and software programming.

Table C-3 PCI-1721 Board ID data

Base Addr.		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
10H	R	Board ID															
		0	0	0	0	0	0	0	0	0	0	0	0	BD3	BD2	BD1	BD0

BD3 ~ DB0 **Board ID**

BD0 LSB of the Board ID

BD3 MSB of the Board ID

C.5 D/A Control - BASE+12H

Table C-4 PCI-1721 Register for D/A control

Base Addr.		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
12H	W	D/A Control Command															
		VREF_3	VREF_2	VREF_1	VREF_0	MOD_3	MOD_2	MOD_1	MOD_0	UB_3	R_3	UB_2	R_2	UB_1	R_1	UB_0	R_0
	R	D/A Control Status															
		VREF_3	VREF_2	VREF_1	VREF_0	MOD_3	MOD_2	MOD_1	MOD_0	UB_3	R_3	UB_2	R_2	UB_1	R_1	UB_0	R_0

R_*n* **D/A channel *n* output voltage range**

0 5V

1 10V

UB_*n* **D/A channel *n* unipolar or bipolar output**

0 Unipolar

1 Bipolar

MOD_*n* **D/A channel *n* output mode**

0 Direct output (normal)

1 Waveform output

VREF_*n* **D/A channel *n* voltage reference source**

0 Internal

1 External

C.6 Write Calibration Result to EEPROM - BASE+16H

Table C-5 PCI-1721 Register for write calibration result to EEPROM

Base Addr.		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
16H	W	Write Calibration Result to EEPROM															
		X	X	UB	R	X	SEL1	SEL0	P	D7	D6	D5	D4	D3	D2	D1	D0

D7 ~ D0 **Calibration result data**

D0 the least significant bit (LSB) of the data

D7 the most significant bit (MSB)

P **Gain result or Offset result**

0 Gain

1 Offset

SEL1 ~ SEL0 **Select D/A channel**

00 Channel 0

01 Channel 1

10 Channel 2

11 Channel 3

R **D/A channel output voltage range**

0 5V

1 10V

UB **D/A channel unipolar or bipolar output**

0 Unipolar

1 Bipolar

C.7 All D/A channels Synchronized Setting - BASE+22H

The PCI-1721 provides the innovation function which all D/A channels can output the data synchronization.

Table C-6 PCI-1721 Register for all D/A channels synchronized setting

Base Addr.		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
22H	W	All D/A channels Synchronized Setting Command															
		X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	SYNC
	R	All D/A channels Synchronized Setting Status															
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SYNC

SYNC	All D/A channels synchronized setting
0	Disable
1	Enable

C.8 Synchronization strobe - BASE+24H

Write any values to **BASE+24H**, all D/A channels will output data synchronization.

Table C-7 PCI-1721 Register for synchronization strobe

Base Addr.		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
24H	W	Synchronized Strobe															
		X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

C.9 Calibration Setting - BASE+26H

Table C-8 PCI-1721 Register for calibration command and status

Base Addr.		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
26H	W	Calibration Setting Command															
		X	X	UB	R	X	SEL1	SEL0	P	D7	D6	D5	D4	D3	D2	D1	D0
	R	Calibration Setting Status															
		0	0	UB	R	0	SEL1	SEL0	P	D7	D6	D5	D4	D3	D2	D1	D0

D7 ~ D0 **Calibration result data**

D0 the least significant bit (LSB) of the data

D7 the most significant bit (MSB)

P **Gain result or Offset result**

0 Gain

1 Offset

SEL1 ~ SEL0 **Select D/A channel**

00 Channel 0

01 Channel 1

10 Channel 2

11 Channel 3

R **D/A channel output voltage range**

0 5V

1 10V

UB **D/A channel unipolar or bipolar output**

0 Unipolar

1 Bipolar

C.10 Calibration Status - BASE+28H

Table C-9 PCI-1721 Register for calibration status

Base Addr.		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
28H	R	Calibration Status															
		0	0	0	0	0	0	0	0	0	0	0	0	0	CALI_ REQ	CALI_ INIT	CALI_ RDY

CALI_RDY **Calibration activate bit**
This bit indicates whether the calibration command is finish.
0 means that the calibration command is finish.

CALI_INIT **Calibration initial bit**
This bit indicates whether the calibration controller is initialed.
0 means that the calibration controller is initial.

CALI_REQ **Calibration request bit**
This bit indicates whether the calibrate command send to the calibration controller is completed.
0 means that the command is sent completed.

C.11 I/O Setting Command and Status - BASE+2AH

Table C-10 PCI-1721 Register for I/O setting

Base Addr.		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2AH	W	I/O Setting Command															
		X	X	X	X	X	X	X	X	X	FIFO_C1	FIFO_C0	HDIO	LDIO	X	GATE0	8254_CNT0
	R	I/O Setting Status															
		0	0	0	0	0	0	0	0	0	FIFO_C1	FIFO_C0	HDIO	LDIO	0	GATE0	8254_CNT0

8254_CNT0 **Select clock source of 82C54 counter 0**

0 10 MHz clock

1 82C54 counter 0 clock (from pin 38 of CN1)

GATE0 **Gate control of 82C54 counter 0**

0 Disable 82C54 counter 0

1 Enable 82C54 counter 0

LDIO **Setting the LOW byte DIO as input or output**

0 Output

1 Input

HDIO **Setting the HIGH byte DIO as input or output**

0 Output

1 Input

FIFO_C1 ~ FIFO_C0

Select FIFO clock source

00 Clock from 82C54. The limitation is 2.5 MHz

01 5 MHz

10 10 MHz

11 Clock from external. The limitation is 10 MHz

Note:

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

You can refer to Fig.2-3 *(FIFO block diagram)* to get more clear description.

C.12 Clear FIFO strobe - BASE+2CH

Write any values to **BASE+2CH** to clear the FIFO.

Table C-11 PCI-1721 Register for clear FIFO strobe

Base Addr.		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2CH	W	Clear FIFO strobe															
		X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

C.13 Clear FIFO strobe - BASE+2EH

Table C-12 PCI-1721 Register for clear FIFO strobe

Base Addr.		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2EH	R	FIFO Status															
		0	0	0	0	0	0	0	0	0	0	0	0	0	F/F	F/H	F/E

F/E **FIFO Empty flag**
This bit indicates whether the FIFO is empty.
0 means that the FIFO is empty.

F/H **FIFO Half-full flag**
This bit indicates whether the FIFO is half-full.
0 means that the FIFO is half-full.

F/F **FIFO Full flag**
This bit indicates whether the FIFO is full.
0 means that the FIFO is full.

C.14 82C54 Counter Chip - BASE+30/32/34/36H

Table C-13 PCI-1721 Register for82C54 counter chip

Base Addr.		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
30H	**W**	**82C54 Counter 0 Command**															
		X	X	X	X	X	X	X	X	D7	D6	D5	D4	D3	D2	D1	D0
	R	**82C54 Counter 0 Status**															
		0	0	0	0	0	0	0	0	D7	D6	D5	D4	D3	D2	D1	D0
32H	**W**	**82C54 Counter 1 Command**															
		X	X	X	X	X	X	X	X	D7	D6	D5	D4	D3	D2	D1	D0
	R	**82C54 Counter 1 Status**															
		0	0	0	0	0	0	0	0	D7	D6	D5	D4	D3	D2	D1	D0
34H	**W**	**82C54 Counter 2 Command**															
		X	X	X	X	X	X	X	X	D7	D6	D5	D4	D3	D2	D1	D0
	R	**82C54 Counter 2 Status**															
		0	0	0	0	0	0	0	0	D7	D6	D5	D4	D3	D2	D1	D0
36H	**W**	**82C54 Control Command**															
		X	X	X	X	X	X	X	X	D7	D6	D5	D4	D3	D2	D1	D0
	R	**82C54 Control Status**															
		0	0	0	0	0	0	0	0	D7	D6	D5	D4	D3	D2	D1	D0

The four registers of **BASE+30/32/34/36H** are used for the 82C54 programmable timer/counter. For detailed information, Intel® 82C54 User's Manual is available by accessing the following path on CD-ROM:

\Document\Intel 82C54 manual.pdf

C.15 DIO Write/Read- BASE+3EH

Table C-14 PCI-1721 Register for DIO write/read strobe

Base Addr.		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
3EH	W	Write DO data															
		D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
	R	Read DI data															
		D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0

C.16 DMA FIFO Data Buffer - BASE+40/42H

The PCI-1721 provides the 32-bit FIFO. Write the data (pattern) which you want to output to **BASE+40/42H**, then the data will transfer to the FIFO.

Table C-15 PCI-1721 Register for DMA FIFO data buffer

Base Addr.		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
40H	W	DMA FIFO Data Buffer (lower word)															
		D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
42H	W	DMA FIFO Data Buffer (upper word)															
		D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0

Appendix D. Calibration

This chapter provides brief information on PCI-1721 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 D/A calibration.

Note:

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

The PCI-1721 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 4½-digit digital multi-meter and a voltage calibrator or a stable, noise-free D. C. voltage source.

Note:

✎ Before you calibrate the D/A function, you must turn on the power at least **15 minutes** to make sure the DA&C card getting stable.

A calibration utility, AutoCali, is included on the companion CD-ROM :

AutoCali.EXE PCI-1721 calibration utility

This calibration utility is designed for the Microsoft® Windows environment. Access this program from the default location:

C:\Program Files\Advantech\ADSAPI\Utility\PCI1721

D.1 VR Assignment

There is one variable resistor (VR1) on the PCI-1721 to adjust the accurate reference voltage on the PCI-1721. We have provided a test point (See J6 in Figure D-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 J6 has reached +10.0000 V. Figure D-1 shows the locations of VR1 and J6.

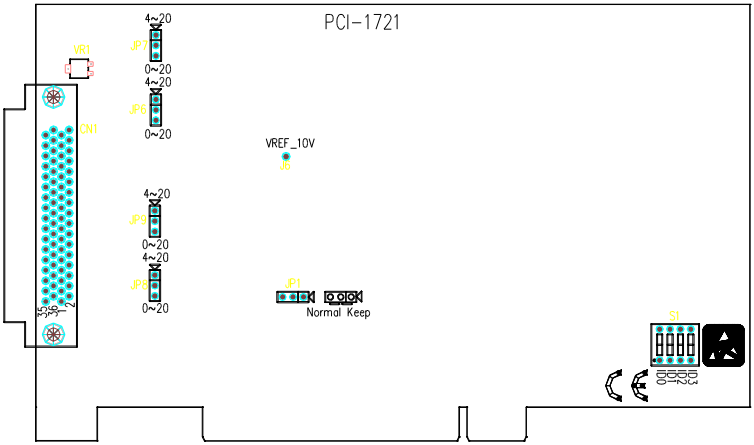


Figure D-1: PCI-1721 VR assignment

D.2 D/A Calibration

You can select an on-board +5V or +10V *internal reference voltage* or an *external voltage* as your analog output reference voltage. If you use an external reference, connect the reference voltage within the ±10V range to the reference input of the D/A output channel you want to calibrate. Then adjust the gain value, unipolar offset voltage, bipolar offset voltage, respectively, of D/A channels 0 and 1 with the *Calibration Command and Data register* (BASE+26H).

Note:

✎ Using a precision voltmeter to calibrate the D/A outputs is recommended.

The auto-calibration program AutoCali.EXE helps you finish the D/A calibration procedure automatically. Although the procedure is not necessary, the following calibration steps are provided below for your reference in case you want to implement the calibration yourself:

1. To adjust those four jumpers JP6~JP9 to 0~20 mA.
2. Setting the output range as 0~5 V for channel 0.
3. Adjust **UNIPOLAR** offset calibration. First, output 0x0000 to channel 0. Then writing the ***value*** from 0x00 to 0xFF sequentially to *Calibration Command and Data register* (BASE+26H), and to see whether the output voltage is less then 1 LSB. If so, to go to next step. Otherwise, you must change the ***value*** and repeat all the procedure in this step again until the output voltage is less then 1 LSB.
4. Adjust **GAIN 5V** calibration. First, output 0x0FFF to channel 0. Then writing the ***value*** from 0x00 to 0xFF sequentially to *Calibration Command and Data register* (BASE+26H), and to see whether the output voltage is less then 1 LSB. If so, to go to next step. Otherwise, you must change the ***value*** and repeat all the procedure in this step again until the output voltage is less then 1 LSB.
5. Setting the output range as ±10 V for channel 0.
6. Adjust **BIPOLAR** offset calibration. First, output 0x0800 to channel 0. Then writing the ***value*** from 0x00 to 0xFF sequentially to *Calibration Command and Data register* (BASE+26H), and to see whether the output voltage is less then 1 LSB. If so, to go to next step. Otherwise, you must change the ***value*** and repeat all the procedure in this step again until the output voltage is less then 1 LSB.
7. Adjust **GAIN 10V** calibration. First, output 0x0FFF to channel 0. Then writing the *value* from 0x00 to 0xFF sequentially to *Calibration Command and Data register* (BASE+26H), and to see whether the output voltage is less then 1 LSB. If so, to go to next step. Otherwise, you must change the ***value*** and repeat all the procedure in this step again until the output voltage is less then 1 LSB.
8. To change to another channel and repeat steps 2 to 7 to calibrate all four channels.

Table D-1 D/A binary code table

A/D code		Mapping Voltage	
Hex.	**Dec.**	**Bipolar**	**Unipolar**
0000h	0	-FS	0
07FFh	2047	-1 LSB	0.5 FS - 1 LSB
0800h	2048	0	0.5 FS
0FFFh	4095	+FS - 1 LSB	FS - 1 LSB

Note:

✎ 1 LSB = FS / 4095 for Unipolar
(For example: 1LSB = 10 / 4095, while the range is 0 V to10 V)
1 LSB = +FS / 4095 for Bipolar
(For example: 1LSB = 10 / 4095, while the range is -5 V to +5 V)

D.3 Calibration Utility

The calibration utility, AutoCali, provides four functions - auto D/A calibration and manual D/A calibration. The program helps the user to easily finish the calibration procedures automatically, however, the user can calibrate the PCI-1721 manually. The following steps will guide you through the PCI-1721 software calibration.

Step 1: Access the calibration utility program ***AutoCali.exe*** from the default location:

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

Note:

✎ If you installed the program to another directory, you can find this program in the corresponding subfolders in your destination directory.

Step 2: Select PCI-1721 in the ADSDAQ dialog box.

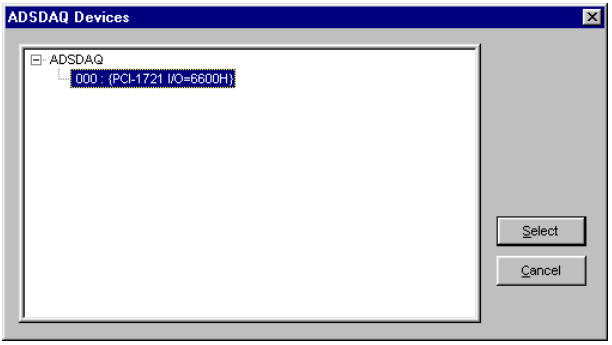


Figure D-2: Selecting the device you want to calibrate

Step 3: After you start to calibrate the PCI-1721, please don't forget to adjust VR1.

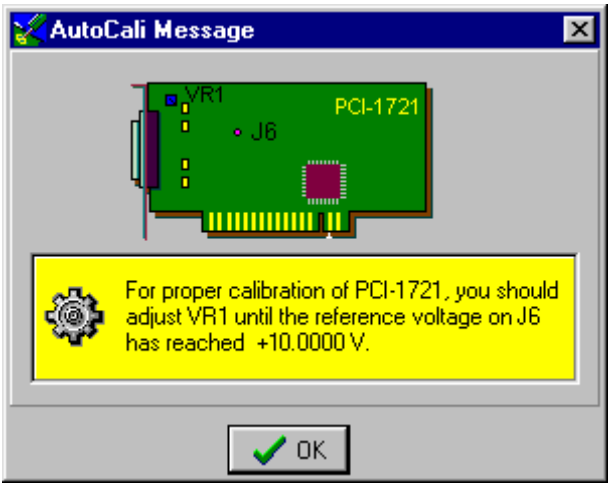


Figure D-3: Warning message before start calibration

D/A channel Auto-Calibration

Step 4: There are four D/A channels in PCI-1721, select the output range for each channel and then press the ***Start*** button to calibrate D/A channels (Fig. D-4).

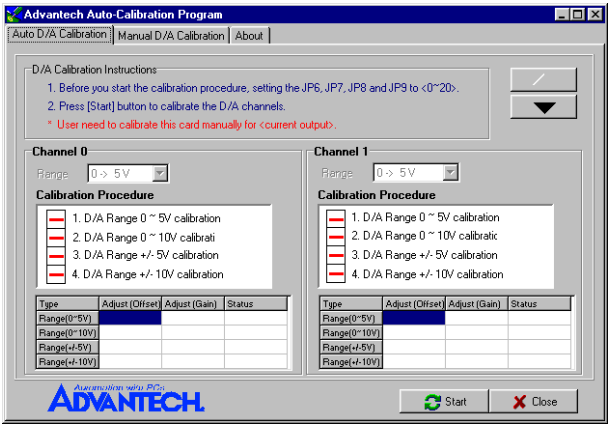


Figure D-4: Range Selection in D/A Calibration

Step 5: Clicking the ***Start*** button, the utility will test some registers to make sure the PCI-1721 has stable enough to process the calibrate procedure (Fig. D-5)

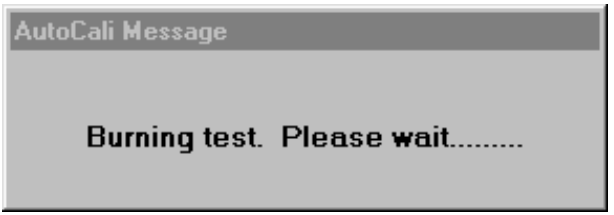


Figure D-5: Burning test

Step 6: D/A channel 0 calibration is enabled (Fig. D-6)

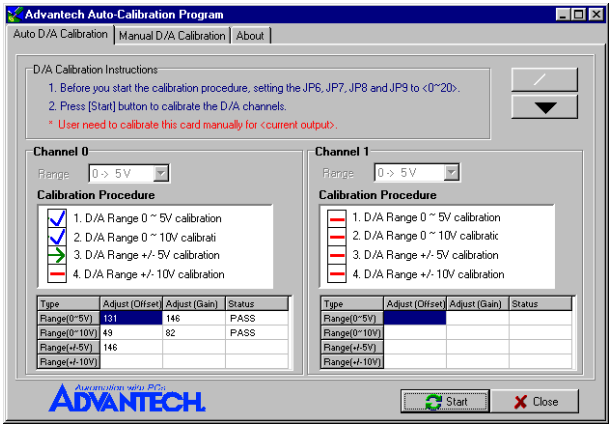


Figure D-6: Calibrating D/A Channel 0

Step 7: D/A channel 1 calibration is enabled (Fig. D-7)

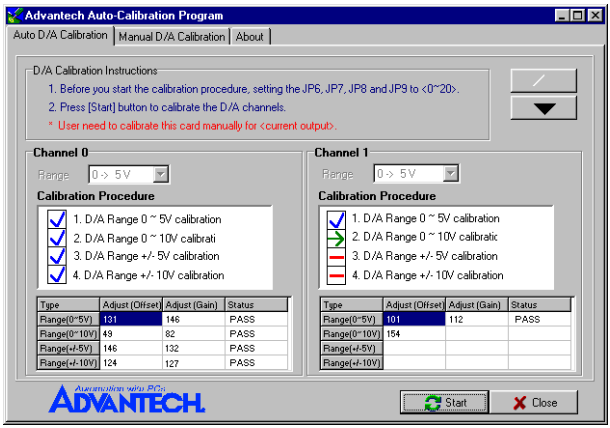


Figure D-7: Calibrating D/A Channel 1

Step 8: Auto-calibration is finished (Fig. D-8)

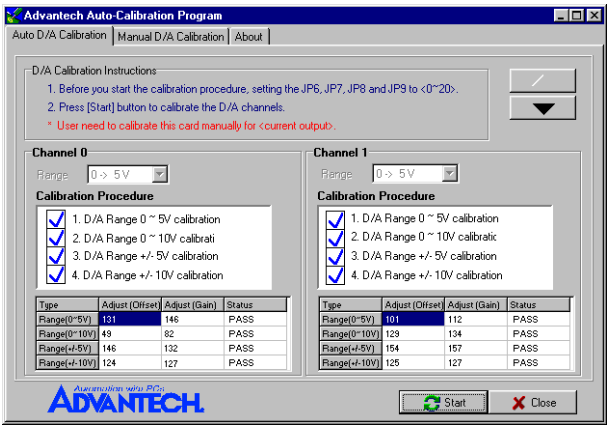


Figure D-8: D/A Calibration is finished

D/A channel Manual-Calibration

Step 1: Click the *Manual D/A Calibration* tab to show the D/A channel manual calibration panel. Four D/A channels are individually calibrated . Before calibrating, output desired voltage from the D/A channels and measure it through an external precision multi-meter.

Step 2: For example, choose channel 0; select the Range and select the wished output voltage code or value from the radio buttons (Fig. D-9 and Fig. D-10).

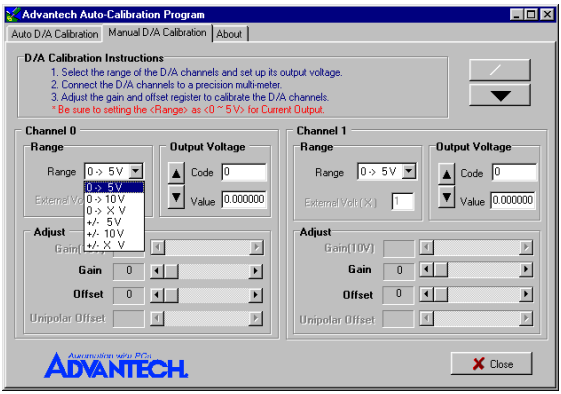


Figure D-9: Selecting D/A Range

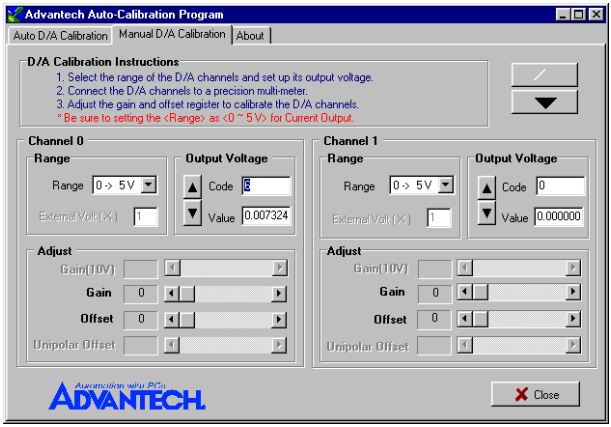


Figure D-10: Choosing Output Voltage

Step 3: According to the difference between the output voltage from D/A channel and the value in the multi-meter, adjust the *gain* and *offset* registers (Fig. D-11)

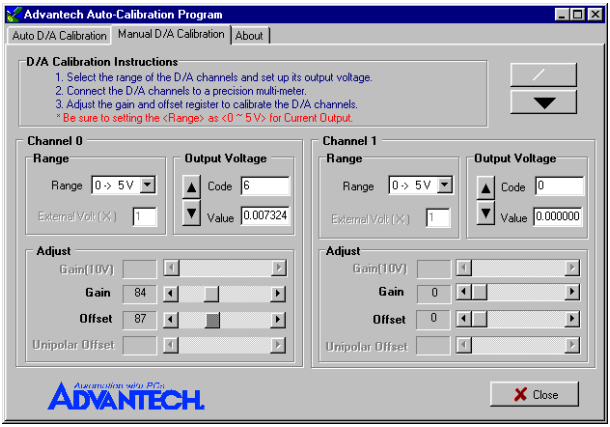


Figure D-11: Adjusting registers

Step 4: Adjust registers until they fall between the output voltage from the D/A channel and the value in the multi-meter.

Note:

✎ The "Waveform Editor Utility" also comes with PCI-1721. Access this program from the default location:
C:\Program Files\Advantech\ADSAPI\Utility\PCI-1721\WaveformEditor
Kindly refer to the "readme" file for detailed information.