This product is obsolete.

PCI-AC5, PCIE-AC5, AC5, AND G4AC5 USER'S GUIDE

Form 1211-250204—February 2025

OPTO 22
Automation made simple.

43044 Business Park Drive • Temecula • CA 92590-3614
Phone: 800-321-OPTO (6786) or 951-695-3000
Fax: 800-832-OPTO (6786) or 951-695-2712
www.opto22.com

Product Support Services
800-TEK-OPTO (835-6786) or 951-695-3080
Fax: 951-695-3017
Email: support@opto22.com
Web: support.opto22.com

PCI-AC5, PCIe-AC5, AC5, and G4AC5 User's Guide
Form 1211-250204—February 2025

The information in this manual has been checked carefully and is believed to be accurate; however, Opto 22 assumes no responsibility for possible inaccuracies or omissions. Specifications are subject to change without notice.

Opto 22 warrants all of its products to be free from defects in material or workmanship for 30 months from the manufacturing date code. This warranty is limited to the original cost of the unit only and does not cover installation, labor, or any other contingent costs. Opto 22 I/O modules and solid-state relays with date codes of 1/96 or newer are guaranteed for life. This lifetime warranty excludes reed relay, SNAP serial communication modules, SNAP PID modules, and modules that contain mechanical contacts or switches. Opto 22 does not warrant any product, components, or parts not manufactured by Opto 22; for these items, the warranty from the original manufacturer applies. Refer to Opto 22 form 1042 for complete warranty information.

Opto 22
Automation Made Simple.

Table of Contents

1: Introduction

Opto 22 PCI-AC5, PCIe-AC5, AC5 and G4AC5 adapter cards provide an interface between a PC and Opto 22 digital input/output (I/O) mounting racks for direct connection to I/O points.

The PCI-AC5 and PCIe-AC5 offer expanded operation and support for modern computers with PCI and PCIe slots; the AC5 and G4AC5 cards support older computers with ISA slots.

The following table provides a quick look at card features; detailed specifications begin on page 19.

Feature	PCIe-AC5	PCI-AC5	AC5 and G4AC5
Interface	PCIe 1.1 bus	33 MHz PCI 2.1 bus	ISA bus
I/O Points controlled	48	48	24
Jumpers	Jumperless configuration	Jumperless configuration	Seven (used to configure base address)
LEDs	Four	Four	One

PC-Based Direct I/O SDK

For software developers building applications for the PCI-AC5 or PCIe-AC5, the PC-Based Direct I/O software development kit (SDK) is included free on the Opto 22 CD that comes with the cards.[1] (You can also download the SDK from our website.)

The SDK includes sample applications with source code, Windows drivers, and documentation. In addition, the SDK provides drivers for porting existing applications to newer platforms. For more information, see Chapter 3: Installing and Using the SDK.

Prerequisites

- This guide assumes that you are familiar with Opto 22 mounting racks and input/output modules.
- If you're planning to use the PC-Based Direct I/O SDK, this guide assumes that you are already familiar with Microsoft .NET and programming languages such as C#, Visual Basic, or Visual C++.

[1] The PC-Based Direct I/O SDK does not support AC5 and G4AC5 cards. For these cards, you must create drivers on your own. For more information, see Appendix D: AC5 and G4AC5 Technical Reference.

What's in this Guide

The guide includes the following chapters and appendices:

Chapter 1: Introduction provides Product Support information.

Chapter 2: Installing the Adapter Card describes how to install the cards.

Chapter 3: Installing and Using the SDK describes installing and using the SDK to write custom applications for the PCI-AC5 and PCIe-AC5 on Windows platforms.

Appendix A: System Specifications provides system requirements and specifications for the PCI-AC5, PCIe-AC5, AC5, and G4AC5 adapter cards, and information about the bus and LED locations.

Appendix C: PCIe/PCI-AC5 Hardware Description provides additional technical information you need to create custom applications if you aren't using the PC-Based Direct I/O SDK.

Appendix D: AC5 and G4AC5 Technical Reference includes additional technical information necessary only if you are writing your own driver for the AC5 or G4AC5.

For Help

If you have problems installing or programming the AC5, PCI-AC5, or PCIe-AC5 adapter card and cannot find the help you need in this guide, contact Opto 22 Product Support.

Phone: 800-TEK-OPTO (800-835-6786 toll-free in the U.S. and Canada)
951-695-3080
Monday through Friday,
7 a.m. to 5 p.m. Pacific Time

NOTE: Email messages and phone calls to Opto 22 Product Support are grouped together and answered in the order received.

Fax: 951-695-3017

Email: support@opto22.com

Opto 22 website: www.opto22.com

When calling for technical support, be prepared to provide the following information about your system to the Product Support engineer:

- Version of this product
- PC configuration including operating system, type of processor, speed, and memory. (A screen shot of your system information or system settings can be helpful.)
- Complete description of your hardware system, including:
  - Jumper configuration
  - Type of power supply
  - Specific error messages

2: Installing the Adapter Card

This chapter describes how to install your PCI-AC5, PCIe-AC5, AC5, or G4AC5, and how to configure the jumpers on the AC5 or G4AC5.

In this chapter:

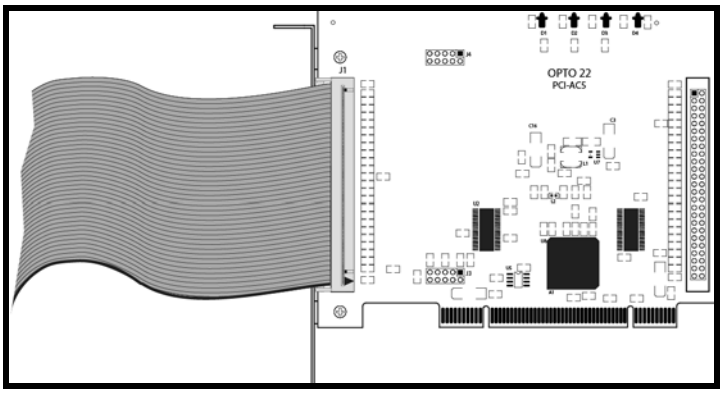
Installing the PCI-AC5 and PCIe-AC5

You can install the PCI-AC5 adapter card into any PCI expansion slot of a computer; the PCIe-AC5 can be installed into any PCIe expansion slot. **Note the power requirements** listed in the Specifications table on page 19.

1. Turn off the computer. Remove the power cord and the computer's cover.

   The power cord must be removed, as a spike may cause the computer to boot.
2. Before handling the card, discharge static electricity by touching the computer's metal chassis.
3. Starting with the connector end that allows the cable to lie flat, push the ribbon cable through one of the slot openings in the computer, and then attach the cable to the adapter card.

   CAUTION: Do not scratch this card or other cards in the computer, as scratching may irreversibly damage the card or other devices.
4. Install the card in the expansion slot. Verify that the card is properly seated in the motherboard socket. Secure the card to the chassis.



5. Connect the other end of the ribbon cable to the I/O mounting rack. For cable part numbers and pinouts, see page 20.
6. Reinstall the power cord. To see the card's LED (for testing purposes), leave the computer cover off temporarily.
7. Turn on the computer. The BIOS automatically performs configuration the first time you boot up the computer with the card installed.

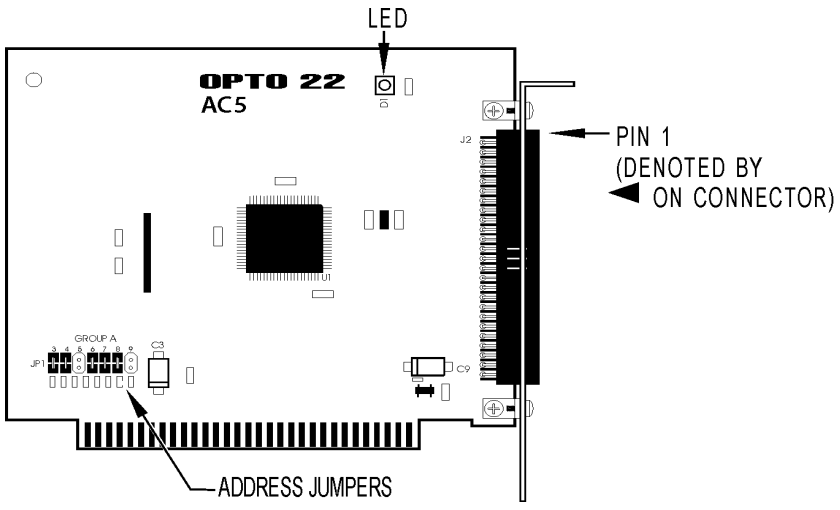
If a "Found New Hardware" message appears, simply click Cancel. The PC-Based Direct I/O SDK setup program will install the device driver automatically.

Installing the AC5 or G4AC5

The AC5 and G4AC5 adapter cards install into any ISA expansion slot of an ISA-capable computer. The driver can support up to 64 AC5 adapter cards. Before you install the AC5 or G4AC5 adapter card, you must set the address jumpers as described below.

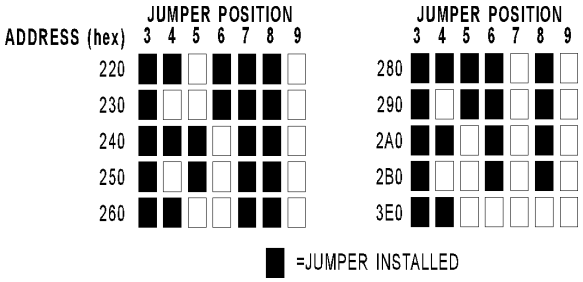
Configuring Jumpers

These instructions apply to the AC5 or G4AC5 adapter card. The card is the same for either part number; only the cable is different.



Seven jumpers (Group A, 3 through 9) define the base I/O address, which identifies a unique eight-byte address space for the adapter. The factory-set base I/O address is 220 (hex).

Select an I/O location that is not currently used by any other device. The following diagram shows examples of jumper settings you might use:



Installing the Card

A 50-conductor ribbon cable connects the adapter card to the I/O rack. The cable included with the AC5 connects the card to racks with edge connectors (such as the PB16A). The cable included with the G4AC5 connects the card to racks with header-style connectors (such as the G4PB24).

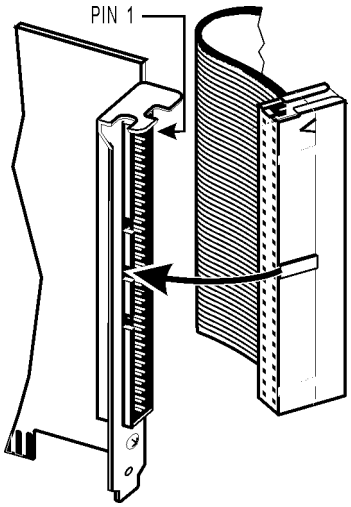
1. Turn off the computer. Remove the power cord and the computer's cover.
2. Before handling the card, discharge static electricity by touching the computer's metal chassis.
3. Install the card in one of the ISA expansion slots. Verify that the adapter card is properly seated in the motherboard ISA socket. Secure the card to the chassis.

   CAUTION: Do not scratch this card or other cards in the computer, as scratching may irreversibly damage the card or other devices.

4. Connect the cable from the adapter card to the I/O mounting rack.

   The diagram at right shows how to connect the cable to the card.

5. Reinstall the power cord.
   If you wish, leave the computer cover off temporarily to see the card's LEDs.
6. Turn on the computer.



3: Installing and Using the SDK

The PC-Based Direct I/O software development kit (SDK) helps developers build custom applications for the PCI-AC5 and PCIe-AC5. (The SDK does not support the AC5 or G4AC5.)

The SDK is available on the CD that comes with the adapter card[1], and contains sample applications with source code, Windows drivers, and user documentation. It works with .NET platform languages including **C#®** and **VB.NET®**, and supports both 64-bit and 32-bit versions of:

- Windows® 10 Professional
- Windows 8.1 Professional
- Windows 7 Professional[2]

NOTE: If you're using an unsupported operating system, you'll need to write your own driver. For details, see "Developing Custom Applications" on page 11 and Appendix D: AC5 and G4AC5 Technical Reference.

To help you get started, the SDK includes a fully functioning Visual C# solution (.sln file) that interfaces with the PCI-AC5 and PCIe-AC5. You can use this solution as an example for building your own custom applications. When you install the SDK, the solution is copied to the C:\Program Files\Opto22\PC-DIRECT-SDK\Examples\DirectScan folder.

For help troubleshooting your custom application, see "Error Messages and Troubleshooting" on page 15.

To update existing applications to run on Windows 10, Windows 8.1, or Windows 7, see "Porting Legacy Applications" on page 16.

In this chapter:

[1] If you don't have the CD, you can download the SDK from our website, or you can request the CD by calling Opto 22 at 800-321-6786 (toll-free in the U.S. and Canada) or 951-695-3000.

[2] For Windows 7 to properly identify the SDK's digital signatures and files, Microsoft Security Advisory 3123479 update (or higher) must be installed. To download the update, see https://support.microsoft.com/en-us/kb/3123479.

Installing the SDK

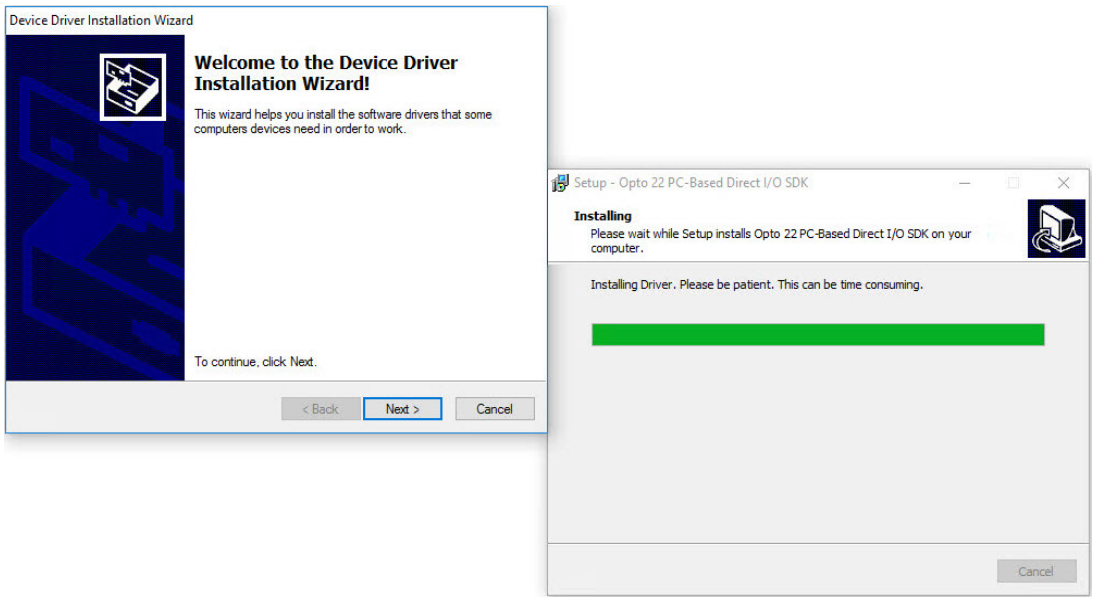
NOTE: The SDK does not support the AC5 or G4AC5 adapter cards. To create drivers for these cards, you must build them yourself. For technical specifications, see Appendix D: AC5 and G4AC5 Technical Reference.

TIP: To ensure reliable control operations and prevent applications from closing unexpectedly, disable Sleep and Hibernation modes (by setting Windows Power Options to "Never") on computers that test or run applications built with the SDK. For more information, see https://support.microsoft.com/en-us/help/13770/windows-shut-down-sleep-hibernate-your-pc.

1. To start the PC-Based Direct I/O SDK, insert the CD in your CD-ROM drive (or, if you've downloaded the SDK from our website, double-click the installer).

   NOTE: If Windows Security dialog boxes appear, click the appropriate buttons (for example, Yes, Open, Run, or Install) to continue.

2. Follow the Setup Wizard's prompts to accept the License Agreement and select the location where the SDK will be installed. Note that if the SDK is already installed, the Setup Wizard prompts you to uninstall it before continuing.



3. On the Device Driver Installation Wizard screen, click Next to install the driver.

After the driver is installed, this screen appears:

4. Click Finish to complete the driver installation, and then click Finish again to complete the SDK installation and launch DirectScan.exe.

Using the Direct I/O Scan Utility

Direct I/O Scan is a diagnostic utility that allows you to configure points as either input or output, and test your system by turning digital points on and off.

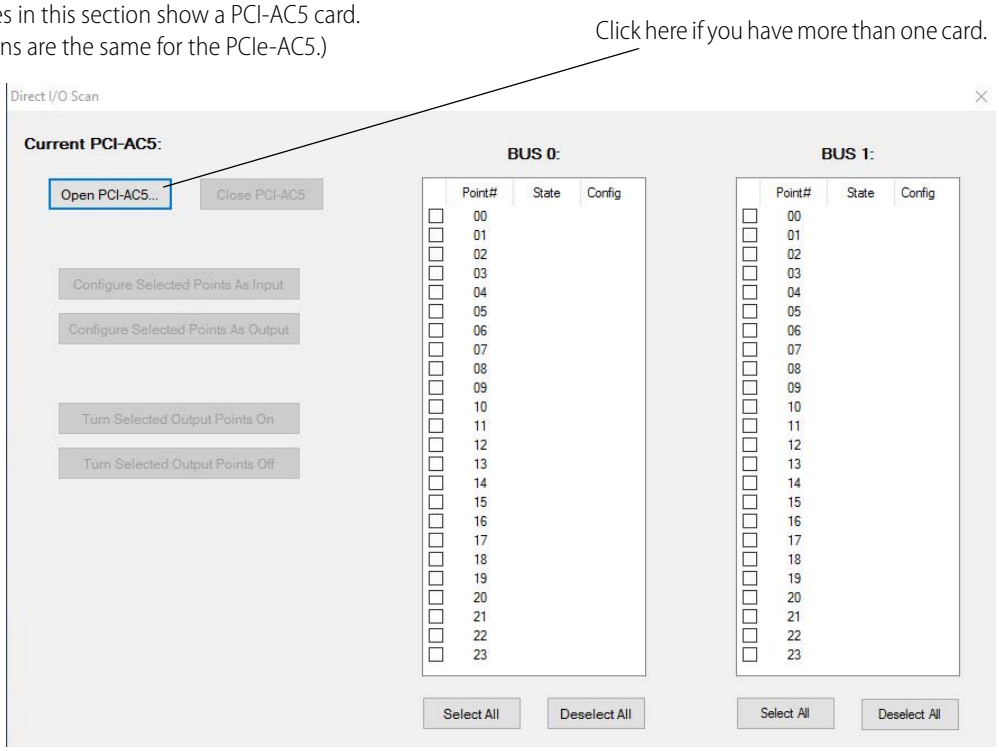
For software developers, the Direct I/O Scan source code can be an example for developing your own custom code. Direct I/O Scan is written in Visual C#, and the source code (DirectScan.sln) is located in the C:\Program Files\Opto22\PC-DIRECT-SDK\Examples\DirectScan folder.

To use the Direct I/O Scan application:

1. If Direct I/O Scan is not already open, start it by pressing the Windows Start key, typing `directscan.exe` and then pressing the Enter key.

   In the Direct I/O Scan window, Bus 0 and Bus 1 refer to the two ribbon cable interfaces connected to the card. For more information, see "Bus and LED Locations" on page 22.

The images in this section show a PCI-AC5 card.
(Instructions are the same for the PCIe-AC5.)

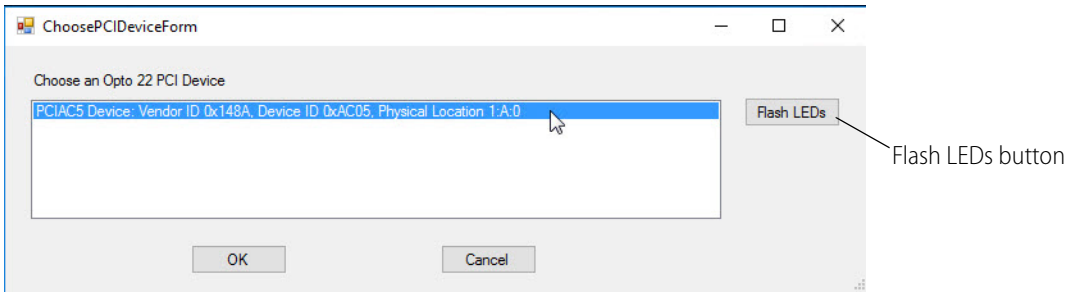


2. If you have only one PCI-AC5 or PCIe-AC5 card installed, continue to step 3.

   If you have more than one card installed:

   a. Click the Open button.

   b. In the Choose PCI Device Form dialog box, select a card.



   c. (Optional) To identify which card you selected, you can flash its LEDs:

   – Open your computer's case to see the installed adapter cards.

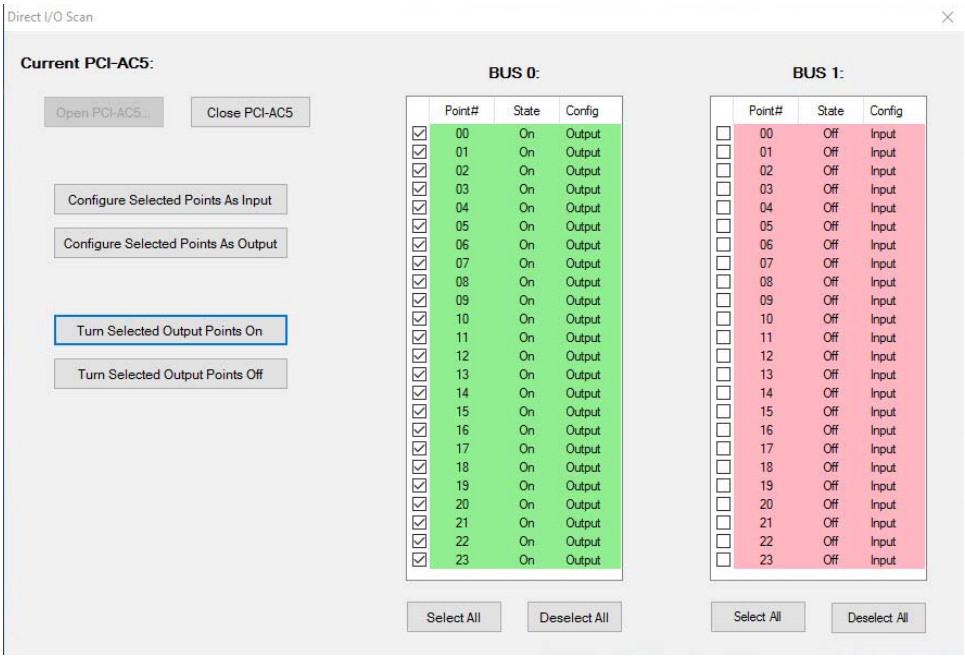
   – Click Flash LEDs.

   The LEDs of the selected card will flash five times. If the LEDs don't flash on the card you are looking at, you may have selected the wrong card. Try selecting a different card from the list, and then click Flash LEDs again to make sure you've selected the right card.

   d. When you're done, click OK to close the dialog box.

3. Use the main screen to experiment with selecting points, configuring points as inputs or outputs, and turning points on and off until you have finished testing the application.

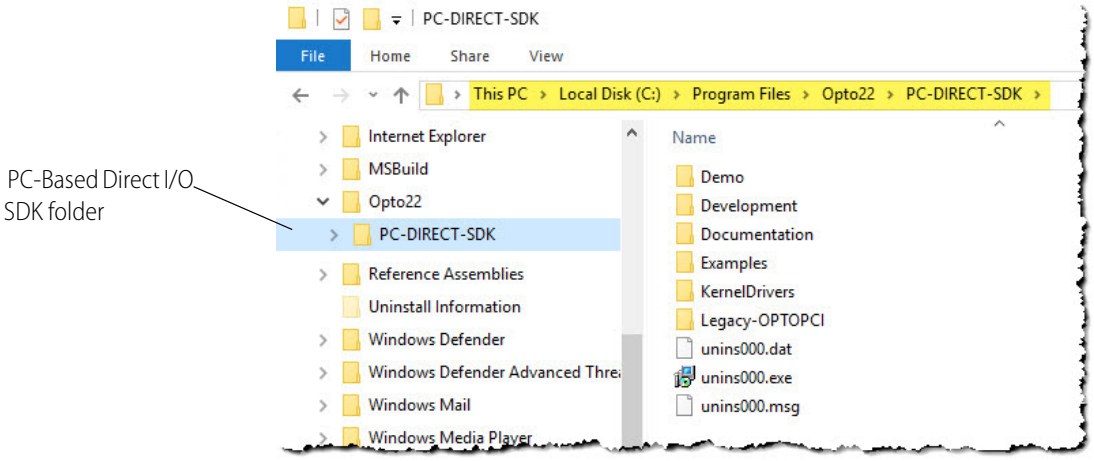
To display the settings shown in this image:
First, click the Select All button for BUS 0.
Then, click the Configure Selected Points As Output button.
Finally, click the Turn Selected Output Points On button.



Developing Custom Applications

NOTE: To upgrade existing applications to work with Windows 10, Windows 8.1, and Windows 7 operating systems, see page 16.

When you install the SDK, new folders are created in the C:\Program Files\Opto22\PC-DIRECT-SDK directory. These folders contain the sample code, documentation, demo utility, and, most importantly, software stacks and kernel driver installers that you need for building custom applications.



The SDK contains several .NET managed software stacks. Applications built with any of the software stacks will work with both the PCI-AC5 and PCIe-AC5 adapter cards.

Although the files in each stack have the same filenames, each stack is different. When you create a custom application, you must ***use the stack that matches your development environment and the target platform***.

If you're developing on a computer with this kind of operating system:	...and your .NET solution is configured to run on this platform:		
	Any CPU	**x86**	**x64**
32-bit	Use a 32-bit SDK software stack.	Use a 32-bit SDK software stack.	Not applicable.*
64-bit	Use a 64-bit SDK software stack.	Use a 32-bit SDK software stack.	Use a 64-bit SDK software stack.
*This option isn't applicable, because if you're coding on a 32-bit machine, you won't be able to run or debug code targeted to a 64-bit platform.			

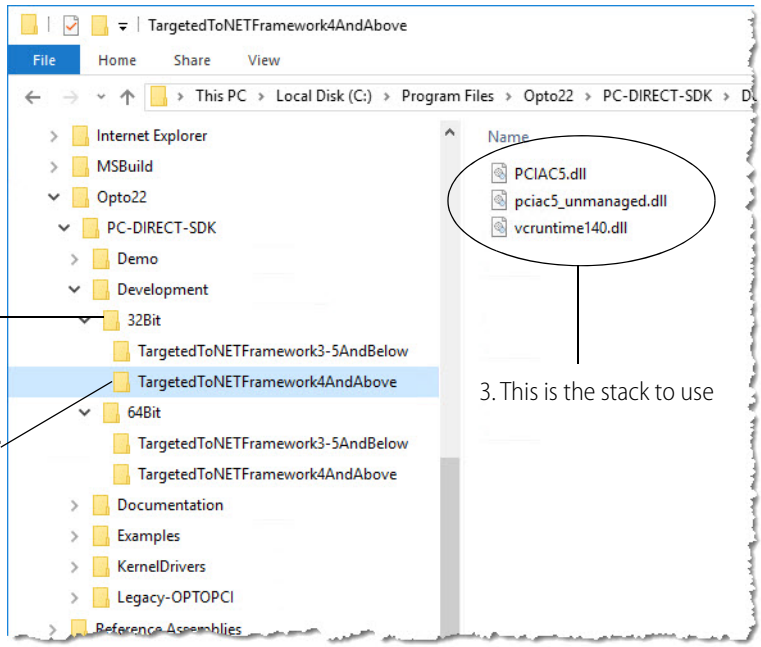
Software stacks are located in the C:\Program Files\Opto22\PC-DIRECT-SDK\Development directory, and folder names are designed to help you find the stack you need.

For example, if:

- Your development computer is a 64-bit machine
- You're are using .NET 4.0
- Your .NET solution is configured for a 32-bit platform

...then use the software stack located in C:\Program Files\Opto22\PC-DIRECT-SDK\Development**32Bit\TargetedToNETFramework4AndAbove**.

1. *Building an app for a 32-bit platform?* Open the **32Bit folder**
2. *Developing with .NET 4.0 or higher?* Choose a stack from the **TargetedToNETFramework4AndAbove folder**



Developing the Application

TIP: To ensure reliable control operations and prevent applications from closing unexpectedly, disable Sleep and Hibernation modes (by setting Windows Power Options to "Never") on computers that test or run applications built with the SDK. For more information, see https://support.microsoft.com/en-us/help/13770/windows-shut-down-sleep-hibernate-your-pc.

For a functional reference of the classes and methods in the SDK, see page 16.

To use the SDK in a .NET application:

1. Copy the appropriate software stack to the folder where the application's executable (.exe) file resides.[3]
2. Add the PCIAC5.dll from the chosen software stack as a reference to your solution.

   To identify which software stack to use, see page 12.
3. Add this .NET using statement to your solution:

   ```
   using Opto22.Adapters.DirectIO
   ```
4. Instantiate the PCIAC5_CardList object by accessing the public property TheCardList.

   Example:
   ```
   public PCIAC5_CardList pciac5cards;
   pciac5cards = PCIAC5_CardList.TheCardList;
   ```
5. Call GetCard( ) on the PCIAC5_CardList object.

   Example:
   ```
   pciac5cards.GetCard(0,
   PCIAC5_Card.eResetLevel.BrainSetForResetActiveLow, True);
   ```
6. Call WriteDigitalConfig( ) to configure input and output points.
7. Create a loop that reads or writes points.
8. When the loop is complete, call ReleaseCard( ) to close the card properly.

Special Considerations

When developing custom applications, here are some special considerations to keep in mind:

Error Codes. If one of the function calls returns false, it's an indication that an error or warning has occurred. To inspect the message, use the **lastError** property on either the PCIAC5_CardList object or the PCIAC5_Card object, depending on what function was called.

Error and warning messages are also logged in the Application area of the Windows Event Log. The messages are grouped under the application name, Opto 22 PCI-AC5.

Multi-threaded Applications. The PC-Based Direct I/O SDK permits only a single handle to a card. If your application uses multiple threads, you should implement a mutex on the handle to avoid thread collision.

Also, multiple applications cannot access the adapter card simultaneously. If you want multiple applications to access the card, you'll need to use or build a program to control and synchronize access.

[3] In Visual Studio®, the bin folder is the default location for a solution's .exe files. For details on how .NET applications locate DLLs, see the Microsoft Dev Center, https://msdn.microsoft.com/en-us/library/windows/desktop/ms682586(v=vs.85).aspx

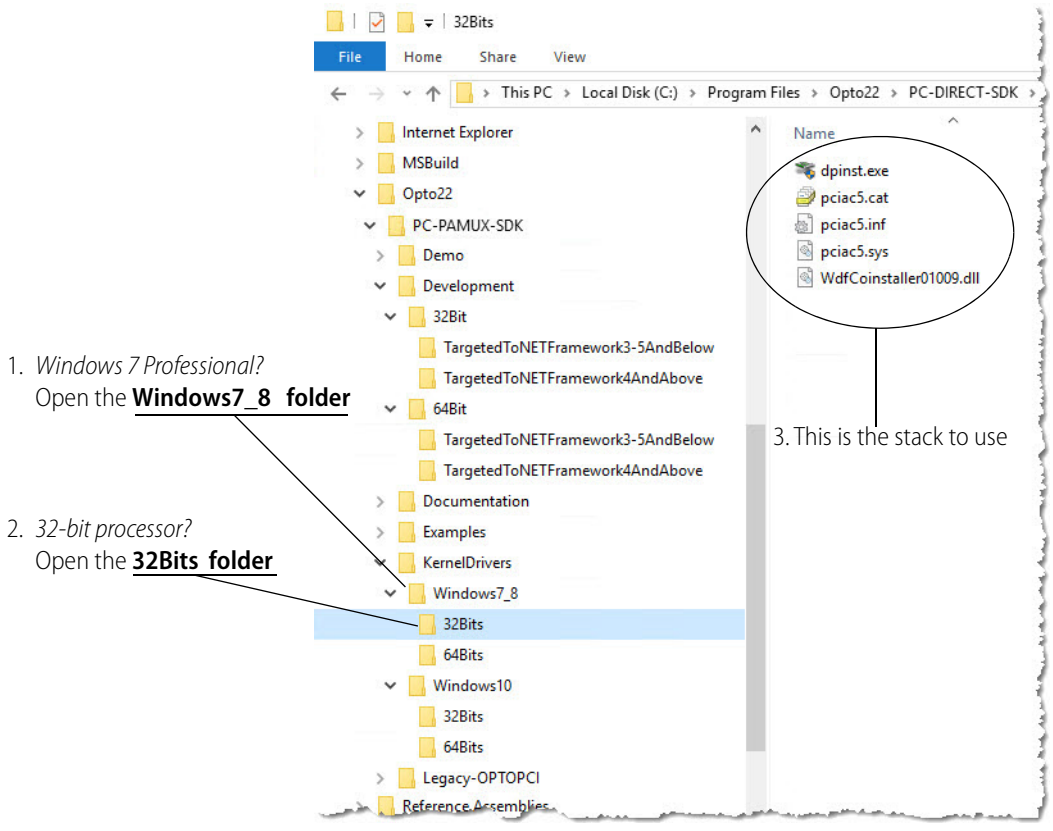
PCI Express Slot Width. While Opto 22's PCI-Express adapter cards are a 1x form factor, they may be installed in wider PCI-Express sockets, like 4x, 8x, and 16x. For expansion slot availability, see the user's manuals for your computer or motherboard.

Building an Application and Driver Installer

The SDK's adapter card installer places both 32-bit and 64-bit driver and library components in the application installation directory. This allows you to build and distribute your application to specific platforms.

To install the adapter card kernel mode driver, the SDK includes 32-bit and 64-bit versions of Microsoft's Driver Package Installer (dpinst.exe).

If your operating system is:	...and your processor is:	...Use the Driver Package Installer in this folder:
Windows 7 Professional or Windows 8.1 Professional	32-bit	C:\Program Files\Opto22\PC-DIRECT-SDK\KernelDrivers\Windows7_8\32Bits
Windows 10 Professional	32-bit	C:\Program Files\Opto22\PC-DIRECT-SDK\KernelDrivers\Windows10\32Bits
Windows 7 Professional or Windows 8.1 Professional	64-bit	C:\Program Files\Opto22\PC-DIRECT-SDK\KernelDrivers\Windows7_8\64Bits
Windows 10 Professional	64-bit	C:\Program Files\Opto22\PC-DIRECT-SDK\KernelDrivers\Windows10\64Bits



To install the driver, run dpinst.exe in the folder that contains the driver's installation components.

For more information about the Driver Package Installer, see Microsoft's Hardware Dev Center website at https://developer.microsoft.com/en-us/windows/hardware.

Opto 22 encourages OEMs that distribute software to wrap their applications and the necessary DLLs in an installer.

Error Messages and Troubleshooting

If one of the function calls returns false, it is an indication that an error or warning has occurred. To inspect the message, use the lastError property on either the PCIAC5_CardList object or the PCIAC5_Card object (depending on which function was called).

All error or warning messages are also logged in the Application section of the Windows Event Viewer (Control Panel > Administrative Tools > Event Viewer). The messages are grouped under the application name, "Opto 22 PCI-AC5.

Troubleshooting

Symptom	Remedy
Outputs don't turn on	• May indicate that the points were not configured as outputs. Try configuring the point using the function WriteConfig. Also, remember that point numbering begins with zero, not one. • Measure the voltage at the terminals on the rack and verify that it is 5.00 to 5.20 VDC (using a high quality meter). • Check the continuity of the 50-wire ribbon cable with a meter. • Make sure the cable is connected to the proper bus. • Make sure that output modules are installed in correct locations. • If output module or rack LEDs are on, this indicates the issue is on the field side of the module. Inspect wiring diagrams and see if there is excitation across the module's field output. Inspect the output module fuse with a meter. • For DC outputs that are always on, the outputs may be reverse biased (wired backwards with respect to polarity). • If AC outputs stick on, it may be that the steady state load current is too low compared to the module's leakage current. A resistive shunt may be necessary across the load device to shunt excess current. For details, see form 1104, *Guide to Troubleshooting Legacy Opto 22 Products*.
Inputs don't read correctly	• Verify that input modules are plugged into the correct locations. • Measure the voltage at the terminals on the rack and verify that it is 5.00 to 5.20 VDC (using a high quality meter). • Verify that the input modules are on by looking at the LED indicators. • Ensure that the 50-wire ribbon cable is properly seated. • Ensure that the 50-wire ribbon cable is plugged into the proper bus number. • Validate continuity of the 50-wire ribbon cable.
The computer's CPU usage is always at 100% or the computer runs slowly when the control application is running	• In time-independent applications, consider adding a Thread.Sleep(1) into your control or monitoring loop. This will allow the application to release time back to the operating system. Try larger numbers in Sleep's argument if the improvement is not too noticeable.

Porting Legacy Applications

NOTE: To convert applications for AC5 and G4AC5 cards to run on PCI-AC5 and PCIe-AC5 cards, see Appendix B: Converting Applications to a Newer Card.

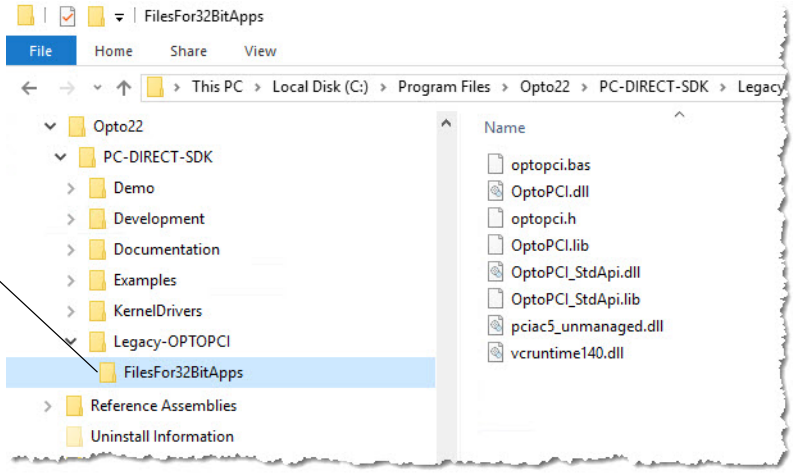
Programs developed for the PCI-AC5 or PCIe-AC5 with older versions of the PC-Based Direct I/O SDK will not work with Windows 10, Windows 8.1, or Windows 7 due to an incompatibility at the driver level of these operating systems.

To update existing PCI-AC5 and PCIe-AC5 applications to run on these newer operating systems, you must use the Legacy-OPTOPCI unmanaged software stack. When you install the SDK, these files are copied to the C:\Program Files\Opto22\PC-DIRECT-SDK\Legacy-OptoPCI\FilesFor32BitApps folder.

The files in this stack are 32-bit only and consist of these components:

- OptoPCI_StdApi.dll
  NOTE: Include this file only for applications that require a STDAPI calling convention with OptoPCI.dll (for example, Delphi).
- OptoPCI.dll
- pciac5_unmanaged.dll
- vc140runtime.dll

Copy the files in this folder to the folder that contains the legacy app's .exe file



To upgrade a legacy program to work with Windows 10, Windows 8.1, and Windows 7:

1. Copy the legacy program files to the target computer.
2. Install the PC-Based Direct I/O SDK on the target computer. (For details, see page 8.)
3. Copy the files from C:\Program Files\Opto22\PC-DIRECT-SDK\Legacy-OptoPM32\FilesFor32BitApps to the folder where the legacy application's executable (.exe) file resides.

To recompile the legacy program on the target operating system:

- Include this file with the legacy program files: optopci.h
- Link this file to legacy program: OptoPCI.lib

To run the legacy program without recompiling:

- Include these files with the legacy program files: OptoPCI.dll
  pciac5_unmanaged.dll

Converting VB Legacy Programs

When converting Visual Basic (VB) legacy programs, be sure to register OCX files.

For example, to register COMCT232.OCX:

1. Open a command prompt with "run as administrator" permissions.
2. Navigate to the directory where COMCT232.OCX resides. Type `regsvr32 COMCT232.OCX` and then press Enter.

A dialog box will notify you of the results of the registration (success/failure).

Changing Hardware Access

Modern operating systems use interface drivers to prevent direct access to hardware.[4] Interface drivers access hardware via a handle.The handle also provides a simple reference method for the application. When using the PC-Based Direct I/O SDK to migrate a legacy application, the first operation your application must perform is to acquire a handle to the hardware from the operating system. At the end of the operation, the application must close the handle to release hardware access back to the operating system. In other words, when updating legacy applications, you must replace the direct hardware access logic, and instead, use a handle:

1. Open an adapter handle.
2. Configure the adapter.
3. Read and write to the adapter.
4. Close the adapter handle.

Updating Data Sizes

When updating a legacy application to run on a newer version of Windows, you must make sure that data sizes in the legacy code are compatible with the driver on the current version of the PC-Based Direct I/O SDK.

The PC-Based Direct I/O SDK supports both 64-bit and 32-bit versions of:

- Windows 10 Professional
- Windows 8.1 Professional
- Windows 7 Professional[5]

[4] In legacy applications that use the inp, outp, or PortIO functions, hardware is accessed directly and in two steps: first, you configure the adapter, then you read and write to the adapter. The direct access method relies on the assumption that no other process will attempt to access the hardware. This can cause resource conflicts, which is part of the reason Microsoft changed the driver architecture in Windows 7 (and higher).

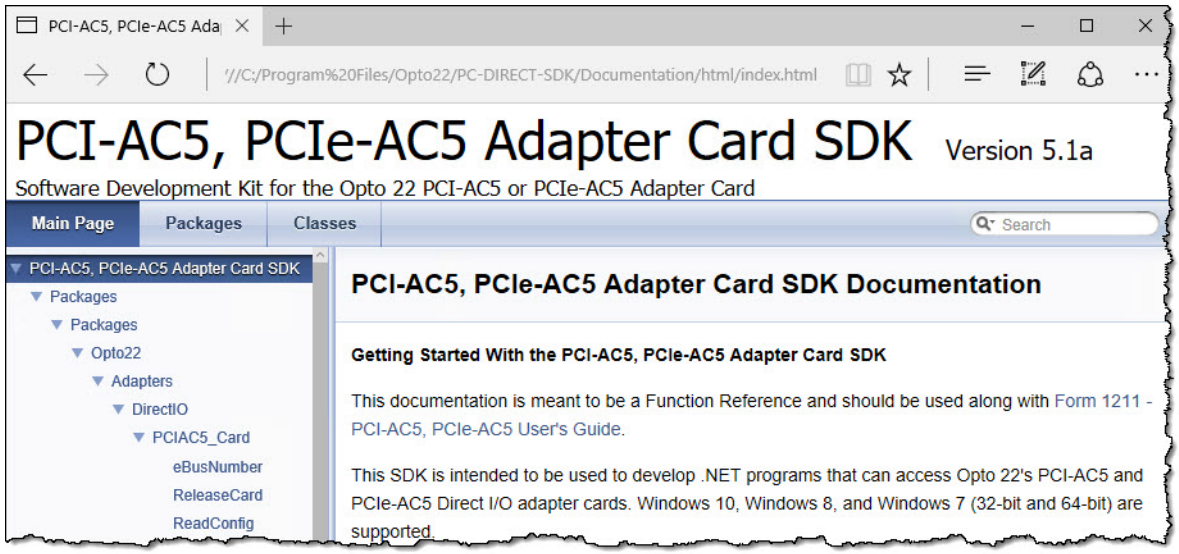
[5] For Windows 7 to properly identify the SDK's digital signatures and files, Microsoft Security Advisory 3123479 update (or higher) must be installed. To download the update, see https://support.microsoft.com/en-us/kb/3123479.

Function Reference Help File

The Function Reference help file is installed on your PC when you install the SDK.

To open the Function Reference, press the Windows Start key, and in the list of Programs, click Opto 22 > PCI-AC5 and PCIe-AC5 Function Reference.

If your browser blocks the ActiveX controls, make sure to unblock them so that you can see the navigation pane.



A: System Specifications

In this appendix:

Specifications

The following table shows specifications for the PCI-AC5, PCIe-AC5, AC5, and G4AC5 adapter cards, plus system requirements for the SDK:

	PCIe-AC5	PCI-AC5	AC5 and G4AC5
Interface	PCIe (1.x)	PCI	ISA
I/O points controlled	48	48	24
Computer compatibility	PCIe 1.1 bus	32-bit, 33 MHz PCI 2.1 bus	ISA bus
Power requirements for card (from the PCI or ISA bus on the PC)	12 VDC @ 50 mA and 3.3 VDC @ 500 mA	Rev C card[1]: 5 VDC @ 250 mA *and* 3.3 VDC @ 250 mA Rev B card[1]: 5.0 VDC @ 600 mA	5 VDC @ 600 mA
Compatible modules	All cards: 5 VDC logic modules, such as the IDC5, ODC5, G4IDC5, G4ODC5, and SNAP-IAC5.[2]		
SDK compatibility	• 64- and 32-bit versions of Microsoft Windows 10 Professional, 8.1 Professional, and 7 Professional[3] • Supports C# and VB.NET	• 64- and 32-bit versions of Microsoft Windows 10 Professional, 8.1 Professional, and 7 Professional[3] • Supports C# and VB.NET	See Note.[4]
Jumpers	Jumperless configuration	Jumperless configuration	Used to configure base address

	PCIe-AC5	PCI-AC5	AC5 and G4AC5
LEDs	Four	Four	One
Operating temperature Storage temperature	0 to 60 °C -30 to 85 °C	0 to 70 °C -30 to 85 °C	0 to 70 °C -30 to 85 °C
Agency certifications	Compliant with DFARS	Compliant with DFARS	Compliant with DFARS
Warranty	30 months	30 months	30 months

[1] Rev C cards show **9278** on a white label; older Rev B cards show a number beginning with **8939**. Rev C cards require *both* 5.0 and 3.3 volts. These cards are not compatible with computers that supply 5 VDC only.

[2] Choose 5 VDC logic modules that are compatible with the mounting rack used.
WARNING: Do not use 15 VDC or 24 VDC modules (such as the IDC15 and IDC24). Using these modules with 15 or 24 VDC logic power can cause serious damage to the adapter card and to the computer.

[3] For Windows 7 to properly identify the SDK's digital signatures and files, Microsoft Security Advisory 3123479 update (or higher) must be installed. To download the update, see https://support.microsoft.com/en-us/kb/3123479.

[4] The PC-Based Direct I/O SDK no longer supports the AC5 or G4AC5 adapter cards. For these cards, you can continue to use version R5.0b of the PC-Based Direct I/O SDK (PC_Based_Direct _IO_SDK_R5.0b.exe). It can be downloaded from Opto 22's FTP site.

Connecting Cables

The AC5 adapter card comes with a 6-foot cable with an edge connector (part number OD6). The G4AC5 is the same adapter card, but with a 6-foot cable with a header connector (PN HH6).

PCI-AC5 and PCIe-AC5 adapter cards come with two 6-foot cables with header connectors (PN HH6). The following cables are also available separately:

Edge Connector	
PN	**Length**
OD2	2 feet
OD4	4 feet
OD6	6 feet
OD8	8 feet
OD10	10 feet

Header Connector	
PN	**Length**
HH1.5	1.5 feet
HH4	4 feet
HH6	6 feet
HH8	8 feet
HH10	10 feet

Pinouts for Connecting Cables

The following table shows pinouts for connecting 50-wire ribbon cable:

Pin	Function
49	No connection
47	Point 0
45	Point 1
43	Point 2

Pin	Function
23	Point 12
21	Point 13
19	Point 14
17	Point 15

Pin	Function
41	Point 3
39	Point 4
37	Point 5
35	Point 6
33	Point 7
31	Point 8
29	Point 9
27	Point 10
25	Point 11

Pin	Function
15	Point 16
13	Point 17
11	Point 18
9	Point 19
7	Point 20
5	Point 21
3	Point 22
1	Point 23
All even pins	Common

Power Supplies

We recommend that you follow these guidelines on power supplies used with your I/O system:

- Use only isolated switching or linear power supplies.
- Do not connect the negative terminal of any power supply to the computer chassis or to earth ground. Doing so can cause ground loops that may interfere with the operation of the rack, the adapter card, or the computer.
- Use a single power supply per rack, if possible. If you have to use a single power supply for more than one rack, use conservatively gauged conductors in a star configuration to distribute power.
- Do not share a rack power supply among multiple computers. Doing so connects the common and ground of all the computers together.
- On racks that have two header connectors, do not share the rack between two computers. Again, doing so connects the common and ground of both computers together.
- Always use a separate power supply for the field side of the I/O. Using the rack power supply for field actuation and monitoring defeats the isolation the I/O module provides. A sudden change of current on the field side may interfere with the operation of the rack, the adapter card, or the computer.

Rack Compatibility Information

The following table lists Opto 22 racks that are compatible with the PCI-AC5, PCIe-AC5, AC5, and G4AC5 adapter cards and racks that can be modified to work with these adapter cards.

WARNING: DO NOT USE the racks listed as not compatible; doing so may cause damage to the computer.

AC5 Compatible (Edge Connectors)	PCI-AC5, PCIe-AC5, and G4AC5 Compatible (Header Connectors)	PCI-AC5, PCIe-AC5, and G4AC5 *Compatible Only if Modified*	Not Compatible; DO NOT USE
PB8 PB16A PB16C PB24 PB24Q	G4PB8 G4PB16 G4PB24 PB24HQ SNAP-D6M SNAP-D6MC SNAP-D6MC-P SNAP-D12M SNAP-D12MC SNAP-D12MC-P	G4PB16J* G4PB16K* G4PB16L* PB4H* PB8H* PB16H* PB16HC* PB16J* PB16K* PB16L* PB16HQ* SNAP-D8M** SNAP-D8MC** SNAP-D8MC-P**	G4PB8H G4PB16H G4PB16HC

* Modification required to use these racks: Remove the jumpers to pins 1 and 49. (These jumpers are labeled JP1 and JP2 on racks G4PB16J, G4PB16K, and G4PB16L.) The jumpers can be de-soldered or clipped.
Warning: If these jumpers are not removed, then the power-on LED will be lit regardless of the actual 5-volt power status. This can result in a false power-on indication and may cause damage to the computer.

** Modification required to use these racks: Remove the JP1 and JP2 jumpers.
Warning: Failure to remove the jumpers may cause damage to the computer.

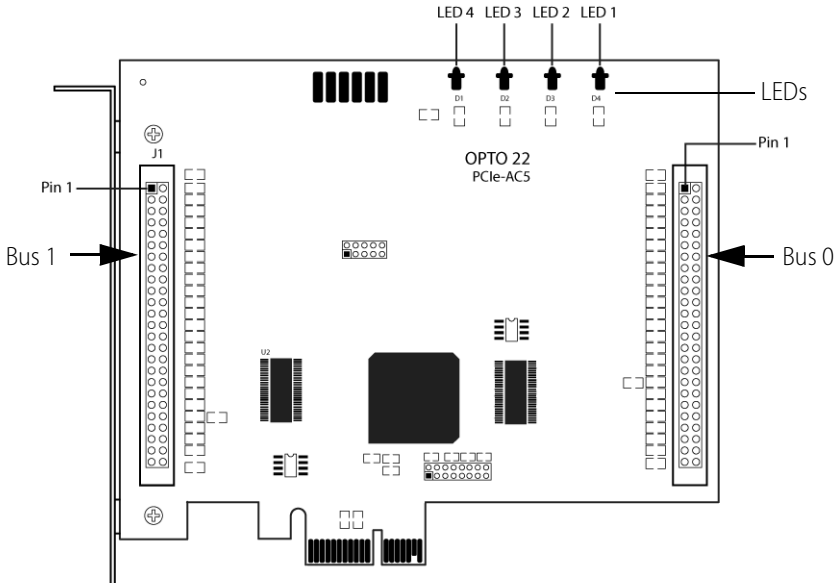
Bus and LED Locations

PCIe-AC5 and PCI-AC5 adapter cards are compatible with the AC5 50-wire ribbon cable standard. Since these cards double the point capacity from their ISA cousin (the AC5), each has two 50-wire ribbon cable interfaces, referred to as *bus 0* and *bus 1*. The PCI-AC5 and PCIe-AC5 cards include four LEDs that can be used for debugging or indicating application status. The AC5 and G4AC5 cards include one LED that flashes to indicate activity (reading from or writing to the card).

The illustrations on the following pages show the locations of bus 0, bus 1, and LEDs on the cards.

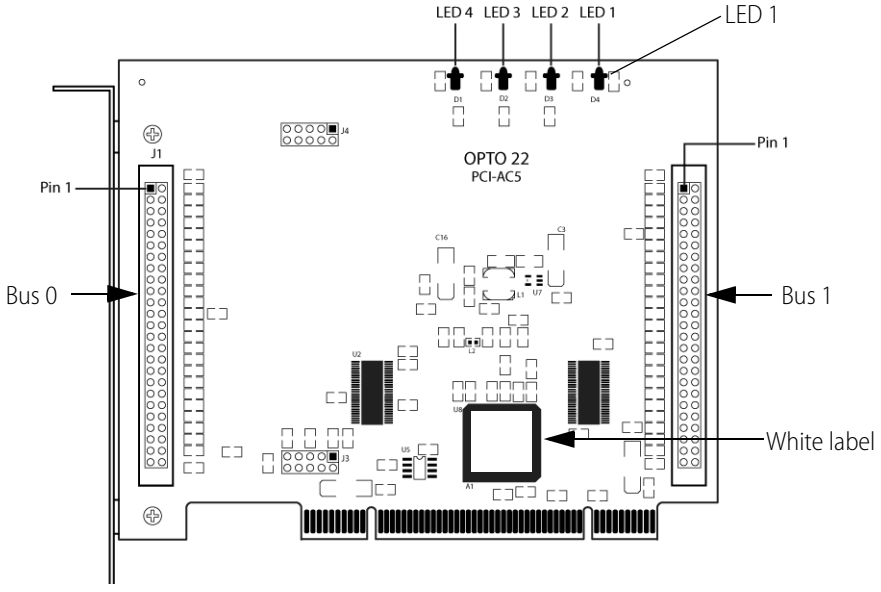
PCIe-AC5

Notice that the bus numbers on the PCIe-AC5 are reversed when compared to the PCI-AC5.



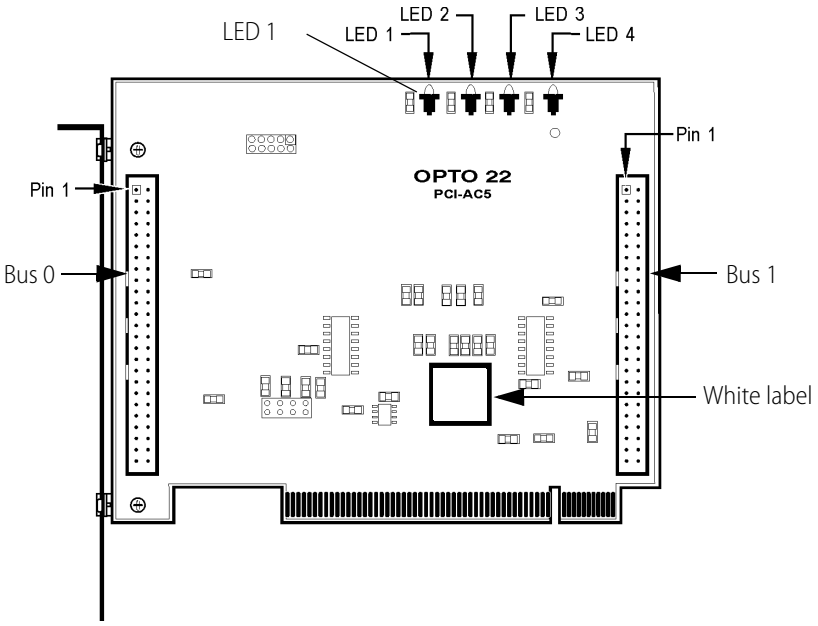
PCI-AC5 (newer version)

There are two different versions of the PCI-AC5 card. The newer version shows "9278" on the white label. Power requirements for the newer card are 5.0 VDC @ 250 mA *and* 3.3 VDC @ 250 mA. This card is not compatible with computers that provide 5.0 VDC only. Note that LEDs are numbered in the reverse order from the older card.



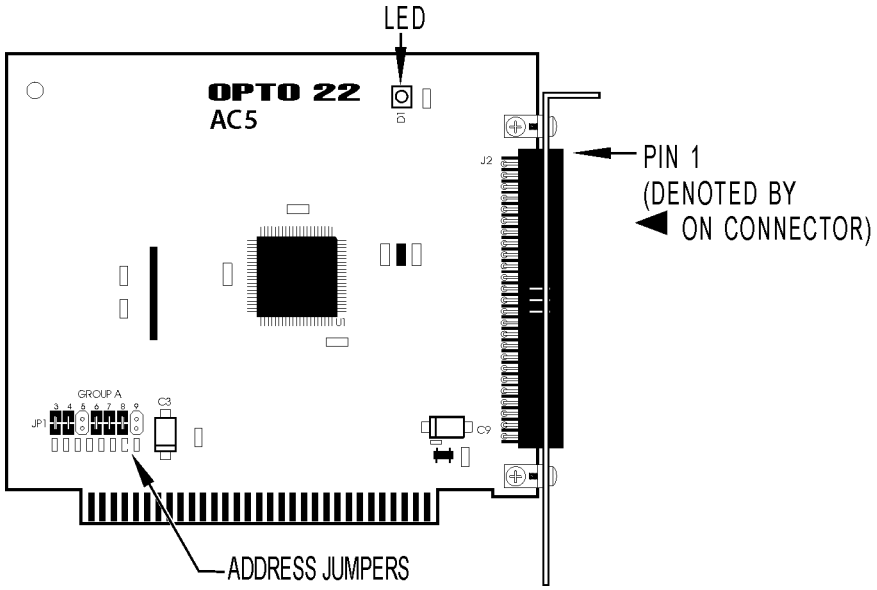
PCIe-AC5 (older version)

The older model has "8939" on its white label. Power requirements for the older card are 5.0 VDC at 600 mA. Again, note that LED numbering is reversed on the two cards. On this older board, bus 0 is labeled PCI-AC5-1 and bus 1 is labelled PCI-AC5-2.



AC5 and G4AC5

The AC5 and G4AC5 cards include one LED that flashes to indicate activity (reading from or writing to the card).



B: Converting Applications to a Newer Card

If you have current applications running on an AC5 or G4AC5 that you now need to use with a PCI-AC5 or PCIe-AC5 card, or if you are migrating AC5 applications to 32-bit Windows, you need to make two key changes to the code:

- Change the way the application accesses hardware
- Make sure data sizes are compatible with the new drivers in the PC-Based Direct I/O SDK

This appendix discusses how to make these changes.

NOTE: If you are using an unsupported operating system or a non-PC hardware platform, you will need to write your own driver. For details, see "Developing Custom Applications" on page 11 and Appendix C: PCIe/PCI-AC5 Hardware Description.

In this appendix:

Changing Hardware Access

Modern operating systems use interface drivers to prevent direct access to hardware.[1] Interface drivers access hardware via a handle.The handle also provides a simple reference method for the application. When using the PC-Based Direct I/O SDK to migrate a legacy application, the first operation your application must perform is to acquire a handle to the hardware from the operating system. At the end of the operation, the application must close the handle to release hardware

[1] In legacy applications that use the inp, outp, or PortIO functions, hardware is accessed directly and in two steps: first, you configure the adapter, then you read and write to the adapter. The direct access method relies on the assumption that no other process will attempt to access the hardware. This can cause resource conflicts, which is part of the reason Microsoft changed the driver architecture in Windows 7 (and higher).

access back to the operating system. In other words, when updating legacy applications, you must replace the direct hardware access logic, and instead, use a handle:

1. Open an adapter handle.
2. Configure the adapter.
3. Read and write to the adapter.
4. Close the adapter handle.

Updating Data Sizes

When updating a legacy application to run on a newer version of Windows, you must make sure that data sizes in the legacy code are compatible with the driver on the current version of the PC-Based Direct I/O SDK.

The PC-Based Direct I/O SDK supports both 64-bit and 32-bit versions of:

- Windows 10 Professional
- Windows 8.1 Professional
- Windows 7 Professional[2]

Converting Applications

This section provides instructions for converting applications to work with the PCI-AC5 and PCIe-AC5.

Migrating 16-bit inp and outp Applications

This section shows how to update 16-bit applications that use the functions inp, outp, PortInput (VB), and PortOutput (VB) to work with the PC-Based Direct I/O SDK.

Microsoft Visual Basic Applications

Follow these instructions to migrate Microsoft Visual Basic applications:

1. Add the module OptoPCI.bas to the Visual Basic project.
2. Add an appropriate adapter open (OptoPCI_PCIAC5_Open or OptoPCI_AC5_Open) before accessing the adapter.
3. Locate the point configuration section of the code and modify this segment to contain the appropriate SDK configure function. It may help to search for occurrences of the PortInput and PortOutput functions.
4. Locate the read and write section of the code and modify this segment to contain the appropriate SDK read or update (write) function. It may be easier to search for PortInput and PortOutput functions.
5. Just before the end of the application, add the SDK function OptoPCI_Close to release each acquired handle to the operating system.

[2] For Windows 7 to properly identify the SDK's digital signatures and files, Microsoft Security Advisory 3123479 update (or higher) must be installed. To download the update, see https://support.microsoft.com/en-us/kb/3123479.

Microsoft Visual C/C++ Applications

Follow these instructions to migrate Microsoft Visual C/C++ applications:

1. Add the header OptoPCI.h to all source modules that reference the SDK functions.
2. Add the DLL link library OptoPCI.lib to the library link profile of the project.
3. Add an appropriate adapter open (OptoPCI_PCIAC5_Open or OptoPCI_AC5_Open) before accessing the adapter.
4. Locate the point configuration section of the code and modify this segment to contain the appropriate SDK configure (write) function. It may be easier to search for occurrences of the inp and outp functions.
5. Locate the read and write section of the code and modify this segment to contain the appropriate SDK read or update (write) function. It may help to search for inp and outp functions.
6. Just before the end of the application, add the SDK function OptoPCI_Close to release each acquired handle to the operating system.

Migrating OptoPMux.dll Applications

This section shows how to update applications that used Opto 22's Pamux DLL to access the AC5 or G4AC5. Applications that used the OptoPMux.dll used the functions PamuxDigBankWriteFast and PamuxDigBankReadFast as if they were equivalent to inp and outp calls.

Visual Basic Applications

To migrate Visual Basic applications, follow these steps:

1. Replace the OptoPMux.BAS module with OptoPCI.BAS module.
2. Locate the function PamuxCardOpen and replace it with the appropriate OptoPCI_AC5_Open or OptoPCI_PCIAC5_Open function.
3. Locate the code segment that configures the AC5 or G4AC5 and replace it with the appropriate OptoPCI_Direct_Config function suitable for the application. It may be easier to search for the function PamuxDigBankWriteFast in the source modules.
4. Locate the code segment that reads and writes to the AC5 or G4AC5. Replace the functions PamuxDigBankWriteFast and PamuxDigBankReadFast with appropriate read or update SDK functions.
5. To update the close function, search for the function PamuxCardClose. Replace it with the SDK function OptoPCI_Close.

Visual C/C++ Applications

To migrate Visual C/C++ applications, follow these steps:

1. Replace all locations of the header OptoPMux.h with OptoPCI.h.
2. Replace OptoPMux.lib with OptoPCI.lib (the dll link library).
3. Locate the function PamuxCardOpen and replace it with the appropriate OptoPCI_AC5_Open or OptoPCI_PCIAC5_Open function.
4. Locate the code segment that initializes the AC5 or G4AC5 and replace it with the appropriate OptoPCI_Direct_Config function suitable for the application. It may be easier to search for the function PamuxDigBankWriteFast.

5. Locate the code segment that reads and writes to the AC5 or G4AC5. Replace the functions PamuxDigBankWriteFast and PamuxDigBankReadFast with appropriate read or update SDK functions.
6. To update the close function, search for PamuxCardClose. Replace it with the SDK function OptoPCI_Close.

C: PCIe/PCI-AC5 Hardware Description

This technical reference is provided for those who are either technically curious or are interested in authoring a device driver for an unsupported operating system or a Windows kernel-mode driver.

The information is provided *as is*; Opto 22 does not support any kernel layer or drivers other than those we provide.

This appendix does not provide complete instructions for writing a driver; it provides only the specific information necessary for these Opto 22 adapter cards. For more information see *PCI Hardware & Software, 4th Edition*, by Edward Solari and George Willse (ISBN #: 092939259-0).

In this appendix:

Hardware Model

The PCI bus identifies both the PCI-AC5 and PCIe-AC5 as shown in the following table:

Description	Description	Enumeration Number
Vendor ID	Opto 22	0x148a
Device ID	PCI-AC5	0xac05

PCI-AC5/PCIe-AC5 Direct I/O Memory Map

All memory-mapped offsets are referenced from PCI base address register 0 (BAR0).

Each offset is a memory-mapped 32-bit wide register.

Base Register	Bus Number	Description	Memory Offset
BAR0	0	Configuration Write Register	0x0000 0004
BAR0	0	Data Write Register	0x0000 0008
BAR0	0	Configuration Read Register	0x0000 0100
BAR0	0	Data Read Register	0x0000 0300
BAR0	1	Configuration Write Register	0x0000 0014
BAR0	1	Data Write Register	0x0000 0018
BAR0	1	Configuration Read Register	0x0000 0500
BAR0	1	Data Read Register	0x0000 0700

Register Descriptions

The PCI-AC5 and PCIe-AC5 cards feature two buses of 24-point I/O each. The two buses are functionally identical.

All registers use the least significant 24 bits for point status and point configuration. The most significant two bits are used for the status indicators (LEDs) on the PCI-AC5 and PCIe-AC5.

All the registers of the direct I/O interface are memory mapped and assume a 4-byte wide read or write PCI bus cycle. Reads and writes of other data sizes will result in unpredictable results.

NOTE: The BIOS allocates the memory map of the PCI-AC5 well above the 1 megabyte address space of DOS architecture computers. DOS applications need a "high memory" driver or a DOS extender to access the PCI address space, as well as a DOS mode PCI BIOS extension function library to access the device's PCI register set.

Using the Status Indicators

To use the LEDs, configure them as outputs. Read and write the LEDs in the same way you read and write points.

Since data writes occur at the robust PCI data rate, rapid changes to LED states may not be visible on the hardware. You may want to consider using a mechanism to delay the next update of the indicators so you can see their state change.

Configuring Points

To configure I/O points, write a 32-bit word to the bus's configuration register. A bit value of 1 configures the bus's point as an output. A bit value of 0 configures the point as an input.

Reading On/Off Status

To read the on/off status of points, read the bus's data read register. The data is negative-true logic (a zero bit means on and a 1 bit means off). The data returned from the read is in a bit-packed format. Each bit contains the status of one point of the bank. The lower 24 bits reflect the states of the outputs, and the most significant two bits reflect the status of the LEDs on the card.

Turning Outputs On and Off

To turn output points on or off, write to the bus's data write register. The data you write is negative-true logic (a 0 bit means on and a 1 bit means off). If you write to a point that is configured as an input, the write is ignored. Also, when you set up the data mask to write to the hardware, remember that the most significant two bits are the LEDs on the card.

D: AC5 and G4AC5 Technical Reference

This technical reference is provided for those who are either technically curious or are interested in authoring a device driver for an unsupported operating system or a Windows kernel-mode driver.

The information is provided *as is*; Opto 22 does not support any kernel layer or drivers other than those we provide.

This appendix does not provide complete instructions for writing a driver; it provides only the specific information necessary for these Opto 22 adapter cards. For more information see *PCI Hardware & Software, 4th Edition*, by Edward Solari and George Willse (ISBN #: 092939259-0).

In this appendix:

Hardware Model

The AC5 appears as an I/O address space device with the base address defined by the jumper settings. The six bytes starting at the base address are used for on-board functions. There are three banks of 8-bit ports. Each bank has a data register and a control register. The control register is used to toggle the configuration mode of the device. The data register is used for reading data, writing data, and writing the bit configuration, when used with the same bank's configuration register.

AC5/G4AC5 Memory Map

The adapter card uses six registers for configuration and data transfer, as shown below:

Address	Function	I/O Points
Base Address + 0	Data Register 1	Points 0 through 7
Base Address + 1	Control Register 1	N/A
Base Address + 2	Data Register 2	Points 8 through 15
Base Address + 3	Control Register 2	N/A

Address	Function	I/O Points
Base Address + 4	Data Register 3	Points 16 through 23
Base Address + 5	Control Register 3	N/A

Configuring Points

Before you can use the adapter card, you must initialize it by configuring each point to function as either an input or an output. To do so, write a configuration byte for each group of eight channels shown above:

1. Write a value of 0 to the bank's control register.
2. Write a bit mask to the bank's data register. Setting a bit to 1 configures the point as an output. Setting a bit to 0 configures the point as an input.
3. Write a value of 34 hexadecimal to the bank's control register.

Reading On/Off Status

To read on/off status for a bank of digital points, read the bank's data register. The read data is negative-true logic (a zero bit means on and a 1 bit means off). The data returned from the read is in a bit-packed format. Each bit contains the status of one point of the bank.

Turning Outputs On and Off

To turn output points on or off, write to the bank's data register. The data you write is negative-true logic (a 0 bit means on and a 1 bit means off). If you write to a point that is configured as input, the write is ignored.