KEITHLEY

KPCI-PIO32IOA & KPCI-PDISO8A

PCI Bus Isolated I/O Board

User's Manual

A GREATER MEASURE OF CONFIDENCE

WARRANTY

Hardware

Keithley Instruments, Inc. warrants that, for a period of three (3) years from the date of shipment, the Keithley Hardware product will be free from defects in materials or workmanship. This warranty will be honored provided the defect has not been caused by use of the Keithley Hardware not in accordance with the instructions for the product. This warranty shall be null and void upon: (1) any modification of Keithley Hardware that is made by other than Keithley and not approved in writing by Keithley or (2) operation of the Keithley Hardware outside of the environmental specifications therefore.

Upon receiving notification of a defect in the Keithley Hardware during the warranty period, Keithley will, at its option, either repair or replace such Keithley Hardware. During the first ninety days of the warranty period, Keithley will, at its option, supply the necessary on site labor to return the product to the condition prior to the notification of a defect. Failure to notify Keithley of a defect during the warranty shall relieve Keithley of its obligations and liabilities under this warranty.

Other Hardware

The portion of the product that is not manufactured by Keithley (Other Hardware) shall not be covered by this warranty, and Keithley shall have no duty of obligation to enforce any manufacturers' warranties on behalf of the customer. On those other manufacturers' products that Keithley purchases for resale, Keithley shall have no duty of obligation to enforce any manufacturers' warranties on behalf of the customer.

Software

Keithley warrants that for a period of one (1) year from date of shipment, the Keithley produced portion of the software or firmware (Keithley Software) will conform in all material respects with the published specifications provided such Keithley Software is used on the product for which it is intended and otherwise in accordance with the instructions therefore. Keithley does not warrant that operation of the Keithley Software will be uninterrupted or error-free and/or that the Keithley Software will be adequate for the customer's intended application and/or use. This warranty shall be null and void upon any modification of the Keithley Software that is made by other than Keithley and not approved in writing by Keithley.

If Keithley receives notification of a Keithley Software nonconformity that is covered by this warranty during the warranty period, Keithley will review the conditions described in such notice. Such notice must state the published specification(s) to which the Keithley Software fails to conform and the manner in which the Keithley Software fails to conform to such published specification(s) with sufficient specificity to permit Keithley to correct such nonconformity. If Keithley determines that the Keithley Software does not conform with the published specifications, Keithley will, at its option, provide either the programming services necessary to correct such nonconformity or develop a program change to bypass such nonconformity in the Keithley Software. Failure to notify Keithley of a nonconformity during the warranty shall relieve Keithley of its obligations and liabilities under this warranty.

Other Software

OEM software that is not produced by Keithley (Other Software) shall not be covered by this warranty, and Keithley shall have no duty or obligation to enforce any OEM's warranties on behalf of the customer.

Other Items

Keithley warrants the following items for 90 days from the date of shipment: probes, cables, rechargeable batteries, diskettes, and documentation.

Items not Covered under Warranty

This warranty does not apply to fuses, non-rechargeable batteries, damage from battery leakage, or problems arising from normal wear or failure to follow instructions.

Limitation of Warranty

This warranty does not apply to defects resulting from product modification made by Purchaser without Keithley's express written consent, or by misuse of any product or part.

Disclaimer of Warranties

EXCEPT FOR THE EXPRESS WARRANTIES ABOVE KEITHLEY DISCLAIMS ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION, ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. KEITHLEY DISCLAIMS ALL WARRANTIES WITH RESPECT TO THE OTHER HARDWARE AND OTHER SOFTWARE.

Limitation of Liability

KEITHLEY INSTRUMENTS SHALL IN NO EVENT, REGARDLESS OF CAUSE, ASSUME RESPONSIBILITY FOR OR BE LIABLE FOR: (1) ECONOMICAL, INCIDENTAL, CONSEQUENTIAL, INDIRECT, SPECIAL, PUNITIVE OR EXEMPLARY DAMAGES, WHETHER CLAIMED UNDER CONTRACT, TORT OR ANY OTHER LEGAL THEORY, (2) LOSS OF OR DAMAGE TO THE CUSTOMER'S DATA OR PROGRAMMING, OR (3) PENALTIES OR PENALTY CLAUSES OF ANY DESCRIPTION OR INDEMNIFICATION OF THE CUSTOMER OR OTHERS FOR COSTS, DAMAGES, OR EXPENSES RELATED TO THE GOODS OR SERVICES PROVIDED UNDER THIS WARRANTY.

KEITHLEY

Keithley Instruments, Inc. 28775 Aurora Road • Cleveland, Ohio 44139 • 440-248-0400 • Fax: 440-248-6168
1-888-KEITHLEY (534-8453) • www.keithley.com

Sales Offices:
BELGIUM: Bergensesteenweg 709 • B-1600 Sint-Pieters-Leeuw • 02-363 00 40 • Fax: 02/363 00 64
CHINA: Yuan Chen Xin Building, Room 705 • 12 Yumin Road, Dewai, Madian • Beijing 100029 • 8610-6202-2886 • Fax: 8610-6202-2892
FINLAND: Tietäjäntie 2 • 02130 Espoo • Phone: 09-54 75 08 10 • Fax: 09-25 10 51 00
FRANCE: 3, allée des Garays • 91127 Palaiseau Cédex • 01-64 53 20 20 • Fax: 01-60 11 77 26
GERMANY: Landsberger Strasse 65 • 82110 Germering • 089/84 93 07-40 • Fax: 089/84 93 07-34
GREAT BRITAIN: Unit 2 Commerce Park, Brunel Road • Theale • Berkshire RG7 4AB • 0118 929 7500 • Fax: 0118 929 7519
INDIA: Flat 2B, Willocrissa • 14, Rest House Crescent • Bangalore 560 001 • 91-80-509-1320/21 • Fax: 91-80-509-1322
ITALY: Viale San Gimignano, 38 • 20146 Milano • 02-48 39 16 01 • Fax: 02-48 30 22 74
JAPAN: New Pier Takeshiba North Tower 13F • 11-1, Kaigan 1-chome • Minato-ku, Tokyo 105-0022 • 81-3-5733-7555 • Fax: 81-3-5733-7556
KOREA: 2FL., URI Building • 2-14 Yangjae-Dong • Seocho-Gu, Seoul 137-888 • 82-2-574-7778 • Fax: 82-2-574-7838
NETHERLANDS: Postbus 559 • 4200 AN Gorinchem • 0183-635333 • Fax: 0183-630821
SWEDEN: c/o Regus Business Centre • Frosundaviks Allé 15, 4tr • 169 70 Solna • 08-509 04 679 • Fax: 08-655 26 10
SWITZERLAND: Kriesbachstrasse 4 • 8600 Dübendorf • 01-821 94 44 • Fax: 01-820 30 81
TAIWAN: 1FL., 85 Po Ai Street • Hsinchu, Taiwan, R.O.C. • 886-3-572-9077• Fax: 886-3-572-9031

4/02

KPCI-PIO32IOA & KPCI-PDISO8A
PCI Bus Isolated I/O Board
User's Manual

Cleveland, Ohio, U.S.A.
Second Printing, August 2002
Document Number: KPCIPIO32IOA-901-01B

Manual Print History

The print history shown below lists the printing dates of all Revisions and Addenda created for this manual. The Revision Level letter increases alphabetically as the manual undergoes subsequent updates. Addenda, which are released between Revisions, contain important change information that the user should incorporate immediately into the manual. Addenda are numbered sequentially. When a new Revision is created, all Addenda associated with the previous Revision of the manual are incorporated into the new Revision of the manual. Each new Revision includes a revised copy of this print history page.

Revision A (Document Number KPCIPIO32IOA-901-01A) ..... February 2002
Revision B (Document Number KPCIPIO32IOA-901-01A) ..... August 2002

KEITHLEY

Safety Precautions

The following safety precautions should be observed before using this product and any associated instrumentation. Although some instruments and accessories would normally be used with non-hazardous voltages, there are situations where hazardous conditions may be present.

This product is intended for use by qualified personnel who recognize shock hazards and are familiar with the safety precautions required to avoid possible injury. Read and follow all installation, operation, and maintenance information carefully before using the product. Refer to the manual for complete product specifications.

If the product is used in a manner not specified, the protection provided by the product may be impaired.

The types of product users are:

Responsible body is the individual or group responsible for the use and maintenance of equipment, for ensuring that the equipment is operated within its specifications and operating limits, and for ensuring that operators are adequately trained.

Operators use the product for its intended function. They must be trained in electrical safety procedures and proper use of the instrument. They must be protected from electric shock and contact with hazardous live circuits.

Maintenance personnel perform routine procedures on the product to keep it operating properly, for example, setting the line voltage or replacing consumable materials. Maintenance procedures are described in the manual. The procedures explicitly state if the operator may perform them. Otherwise, they should be performed only by service personnel.

Service personnel are trained to work on live circuits, and perform safe installations and repairs of products. Only properly trained service personnel may perform installation and service procedures.

Keithley products are designed for use with electrical signals that are rated Installation Category I and Installation Category II, as described in the International Electrotechnical Commission (IEC) Standard IEC 60664. Most measurement, control, and data I/O signals are Installation Category I and must not be directly connected to mains voltage or to voltage sources with high transient over-voltages. Installation Category II connections require protection for high transient over-voltages often associated with local AC mains connections. Assume all measurement, control, and data I/O connections are for connection to Category I sources unless otherwise marked or described in the Manual.

Exercise extreme caution when a shock hazard is present. Lethal voltage may be present on cable connector jacks or test fixtures. The American National Standards Institute (ANSI) states that a shock hazard exists when voltage levels greater than 30V RMS, 42.4V peak, or 60VDC are present. **A good safety practice is to expect that hazardous voltage is present in any unknown circuit before measuring.**

Operators of this product must be protected from electric shock at all times. The responsible body must ensure that operators are prevented access and/or insulated from every connection point. In some cases, connections must be exposed to potential human contact. Product operators in these circumstances must be trained to protect themselves from the risk of electric shock. If the circuit is capable of operating at or above 1000 volts, **no conductive part of the circuit may be exposed.**

Do not connect switching cards directly to unlimited power circuits. They are intended to be used with impedance limited sources. NEVER connect switching cards directly to AC mains. When connecting sources to switching cards, install protective devices to limit fault current and voltage to the card.

Before operating an instrument, make sure the line cord is connected to a properly grounded power receptacle. Inspect the connecting cables, test leads, and jumpers for possible wear, cracks, or breaks before each use.

When installing equipment where access to the main power cord is restricted, such as rack mounting, a separate main input power disconnect device must be provided, in close proximity to the equipment and within easy reach of the operator.

For maximum safety, do not touch the product, test cables, or any other instruments while power is applied to the circuit under test. ALWAYS remove power from the entire test system and discharge any capacitors before: connecting or disconnecting cables or jumpers, installing or removing switching cards, or making internal changes, such as installing or removing jumpers.

Do not touch any object that could provide a current path to the common side of the circuit under test or power line (earth) ground. Always make measurements with dry hands while standing on a dry, insulated surface capable of withstanding the voltage being measured.

The instrument and accessories must be used in accordance with its specifications and operating instructions or the safety of the equipment may be impaired.

Do not exceed the maximum signal levels of the instruments and accessories, as defined in the specifications and operating information, and as shown on the instrument or test fixture panels, or switching card.

When fuses are used in a product, replace with same type and rating for continued protection against fire hazard.

Chassis connections must only be used as shield connections for measuring circuits, NOT as safety earth ground connections.

If you are using a test fixture, keep the lid closed while power is applied to the device under test. Safe operation requires the use of a lid interlock.

If  or  is present, connect it to safety earth ground using the wire recommended in the user documentation.

The  symbol on an instrument indicates that the user should refer to the operating instructions located in the manual.

The  symbol on an instrument shows that it can source or measure 1000 volts or more, including the combined effect of normal and common mode voltages. Use standard safety precautions to avoid personal contact with these voltages.

The **WARNING** heading in a manual explains dangers that might result in personal injury or death. Always read the associated information very carefully before performing the indicated procedure.

The **CAUTION** heading in a manual explains hazards that could damage the instrument. Such damage may invalidate the warranty.

Instrumentation and accessories shall not be connected to humans.

Before performing any maintenance, disconnect the line cord and all test cables.

To maintain protection from electric shock and fire, replacement components in mains circuits, including the power transformer, test leads, and input jacks, must be purchased from Keithley Instruments. Standard fuses, with applicable national safety approvals, may be used if the rating and type are the same. Other components that are not safety related may be purchased from other suppliers as long as they are equivalent to the original component. (Note that selected parts should be purchased only through Keithley Instruments to maintain accuracy and functionality of the product.) If you are unsure about the applicability of a replacement component, call a Keithley Instruments office for information.

To clean an instrument, use a damp cloth or mild, water based cleaner. Clean the exterior of the instrument only. Do not apply cleaner directly to the instrument or allow liquids to enter or spill on the instrument. Products that consist of a circuit board with no case or chassis (e.g., data acquisition board for installation into a computer) should never require cleaning if handled according to instructions. If the board becomes contaminated and operation is affected, the board should be returned to the factory for proper cleaning/servicing.

Table of Contents

5 Troubleshooting

A Specifications

B I/O Address Mapping

C Glossary

List of Illustrations

3 Installation

5 Troubleshooting

List of Tables

2 General Description

3 Installation

4 External Interrupts

5 Troubleshooting

B I/O Address Mapping

1
Overview

This manual contains descriptive information and installation and use instructions for the KPCI-PIO32IOA and KPCI-PDISO8A isolated I/O boards.

The manual is intended for data acquisition system designers, engineers, technicians, scientists, and other users responsible for setting up, cabling, and wiring signals to KPCI-PIO32IOA and KPCI-PDISO8A boards. To follow the information and instructions contained in this manual, you must be familiar with the operation of Windows 98, Me, NT, 2000, or XP with basic data-acquisition principles, and with your application. However, if you find unfamiliar terms in this manual, check the Glossary in Appendix C. To locate topics discussed in this manual, search the index.

To use this manual effectively, review the remaining brief topics in this preface:

- The organization of the manual.
- The special font/typeface conventions used in the manual.
- Moving quickly to cross-referenced parts of the manual (in the electronic [PDF] version).

How the manual is organized

In addition to this Overview, the KPCI-PIO32IOA and KPCI-PDISO8A User's Manual is organized as follows:

- Section 2 briefly describes features and characteristics of the KPCI-PIO32IOA and KPCI-PDISO8A.
- Section 3 first describes software options and installation notes and then the following operations (which must be done after the software is installed):
  - Installing the board.
  - Configuring the combined board and software installations.
  - Checking the combined board and software installation.
  - Installing accessories, and connecting signals.
- Section 4 briefly describes the ability to configure two of the I/O lines as external interrupt enable and external interrupt request lines and identifies the connector pins used.
- Section 5 describes how to troubleshoot your system and obtain technical support.
- Appendix A contains KPCI-PIO32IOA and KPCI-PDISO8A specifications.
- Appendix B describes memory-mapping information for special situations. Appendix B can be disregarded under normal conditions. Use the DriverLINX driver provided with your board for virtually all programming situations.
- Appendix C is a glossary of some terms used in this manual.
- An Index completes the manual.

How to distinguish special text items

Italic, bold, and upper-case letters, the Courier font, and quotation marks distinguish certain text items from the general text. The following text conventions are used (exclusive of headings):

- **10 point Times Bold** distinguishes the following:
  - All Windows 98/Me/NT/2000/XP user-interaction items: commands, screen messages, menu names, menu options, and dialog-box items—including captions, user selections, and typed user inputs (but not including dialog box names, which are in regular text).
- *10 point Times Italic* distinguishes the following:
  - Emphasis in general.
  - Cross-references to other documents, such as other manuals or books.
  - *NOTE* statements.

- 10 POINT TIMES UPPER CASE distinguishes the following:
  - Switches, such as ON and OFF.
  - Keyboard keys, such as ENTER.
- `10 point Courier` distinguishes software code statements.
- "Double quote marks" distinguish the following:
  - Cross references to other manual sections, such as "Troubleshooting."
  - Literals, such as when referring to the "5V" labels on I/O connectors.

Safety symbols and terms

The following symbols and terms may be found on the board or used in this manual.

The  symbol indicates that the user should refer to the operating instructions located in the manual.

The  symbol shows that high voltage may be present on the terminal(s). Use standard safety precautions to avoid personal contact with these voltages.

The ***WARNING*** heading used in this manual explains dangers that might result in personal injury or death. Always read the associated information very carefully before performing the indicated procedure.

The ***CAUTION*** heading used in this manual explains hazards that could damage the board. Such damage may invalidate the warranty.

How to move around the electronic version of the manual

When reading the electronic PDF version of this manual, use Acrobat Reader **View** and **Tools** menu selections to move generally through the manual. Additionally, mouse-click on special links in the manual to jump directly to the page of a referenced item, as follows:

- Mouse-click the top margin of any page to jump to the Table of Contents.
- Mouse-click on any Index or Table of Contents (TOC) page number to jump to the page.
- Mouse-click on any of these cross references to jump to the cross-referenced figure, table, section, or subsection. Cross references are not framed in red—in contrast to page numbers in the Index and Table of Contents.
  - Figure number headings, such as Figure 3-1.
  - Table number headings, such as Table 3-1.
  - Section and subsection headings that are enclosed in quotes, such as "How the manual is organized."

To *return* from the referenced item to what you were reading *before* you jumped to the referenced item—the Index, TOC, top page margin, or cross reference—do either of the following:

- Hold down the CONTROL key and press the [-] key (i.e. press CONTROL + -).
- In the Acrobat Reader **View** menu, click **Go Back**.

2
General Description

Specifications

General specifications are listed in "Specifications," Appendix A. I/O connections are identified in Section 3, and I/O addresses (needed by advanced programmers, only) are defined in "I/O Address Mapping," Appendix B.

System requirements

The system capabilities required to run the KPCI-PIO32IOA and KPCI-PDISO8A board, and to use the DriverLINX software supplied with the board, are listed in Table 2-1.

Table 2-1
System requirements

CPU Type	Pentium or higher processor on motherboard with PCI bus version 2.1.
Operating System	Windows® 95, 98, or Me.
	Windows® NT (version 4.0 or higher), 2000, or XP.
Memory	16 MB or greater RAM when running Windows® 95, 98, or Me.
	32 MB or greater RAM when running Windows® NT, 2000, or XP.
Hard Disk Space	4 MB for minimum installation.
	50 MB for maximum installation.
Other	A CD-ROM drive*.
	A free PCI bus expansion slot.
	Enough reserve computer power supply capacity to power the board, which draws 500mA from the 5V supply (maximum).

* Any CD-ROM drive that came installed with the required computer should be satisfactory. However, if you have post-installed an older CD-ROM drive or arrived at your present system by updating the microprocessor or replacing the motherboard, be aware that some early CD-ROM drives do not support the long file names often used in 32-bit Windows files.

Functional description

The KPCI-PIO32IOA and KPCI-PDISO8A each contain isolated control inputs and isolated solid-state relay switching outputs on a PCI interface board. Both cards work in a Windows environment and take advantage of the 32-bit bus width.

Solid-state relay switching

The KPCI-PIO32IOA's 32 isolated I/O lines and KPCI-PDISO8A's 16 isolated I/O lines include:

- Control inputs are solid-state, opto-isolated (16 for the KPCI-PIO32IOA; 8 for the KPCI-PDISO8A). Maximum control input voltages are: ±60V DC (logic high); ±1V DC (logic low).
- Relay outputs are solid-state, opto-isolated, N.O. form A (16 for the KPCI-PIO32IOA; 8 for the KPCI-PDISO8A). Outputs can switch both DC voltages (±60V DC maximum) and AC voltages (30V RMS maximum) with a maximum load current of 350mA. A relay output will close when a logic high is applied to the corresponding control input.

Under Windows 98/Me, most existing port I/O application programs and data acquisition packages designed for ISA boards work with the KPCI-PIO32IOA and KPCI-PDISO8A board. The PCI-BIOS-assigned base address is entered in place of a user-assigned base address. (Use of such programs under Windows NT/2000/XP, not recommended, is much more involved.)

Other I/O characteristics

Additional I/O port characteristics are summarized below:

- Two of the KPCI-PIO32IOA and KPCI-PDISO8A general-purpose I/O lines can be alternatively configured via software to be external interrupt enable input, $\overline{\text{INT\_ENN}}$, and external interrupt request input, INT_REQ.
- The KPCI-PIO32IOA and KPCI-PDISO8A can switch currents up to 350mA. Output current capabilities allow the boards to control some motors, solenoids, and relays.
- +5V power from the computer power supply is made available at the I/O connector, for use in external circuits.

NOTE *The maximum current on each +5V and digital ground pin is 1A. The +5V supply is not isolated and is available for use only when isolation is not required by the application.*

- Control inputs and relay outputs interface via standard 36-pin connector(s). The KPCI-PIO32IOA has two 36-pin connectors, and the KPCI-PDISO8A has one 36-pin connector.
- Output ports can always be read-accessed without switching the output states, allowing the output states to be read at any time by the controlling program.

Software

The user can select a fully integrated data acquisition software package (e.g., TestPoint or LabVIEW) or write a custom program supported by DriverLINX. DriverLINX software is included with the hardware.

DriverLINX supports programmers who wish to create custom applications using Visual C/C++, Visual Basic, or Delphi. DriverLINX accomplishes foreground and background tasks to perform data acquisition. TestPoint is a fully featured, integrated application package with a graphical drag-and-drop interface, which can be used to create data acquisition applications without programming. LabVIEW is a fully featured graphical programming language used to create virtual instrumentation.

Refer to Section 3, "Installation," for more information about software.

3
Installation

Section 3 describes the following operations, in the order in which they should be performed:

- Reviewing software options and installing software (installation notes that supplement the *Read this first* sheet).
- Preparing for and doing physical installation of the board.
- Configuring the installed software and board.
- Identifying I/O connector pins, connecting the board to interface accessories, and wiring the I/O to external circuits.

NOTE *Install the DriverLINX software before installing the KPCI-PIO32IOA or KPCI-PDISO8A board. Otherwise, the device drivers will be more difficult to install.*

Installing the software

Software options

The KPCI-PIO32IOA and KPCI-PDISO8A boards have two software options. The user can select a fully integrated data acquisition software package (e.g., TestPoint or LabVIEW). The user can also run a custom program in Visual C/C++, Visual Basic, or Delphi using DriverLINX (included with the hardware). A summary of the pros and cons of using integrated packages or writing custom programs is provided in the Keithley Full Line Catalog. The KPCI-PIO32IOA and KPCI-PDISO8A have fully functional driver support for use under Windows.

NOTE *DriverLINX must be installed to run any applications for the board, whether they are custom-programmed applications or integrated software packages, such as TestPoint or LabVIEW.*

DriverLINX driver software for Windows

DriverLINX software, supplied by Keithley with the KPCI-PIO32IOA and KPCI-PDISO8A board, provides convenient interfaces to configure and set I/O bits without register-level programming. DriverLINX supports Windows 98, Me, NT, 2000, and XP.

Most importantly, however, DriverLINX supports those programmers who wish to create custom applications using Visual C/C++, Visual Basic, or Delphi. DriverLINX accomplishes foreground and background tasks to perform data acquisition. The software includes memory and data buffer management, event triggering, extensive error checking, and context sensitive online help.

More specifically, DriverLINX provides application developers a standardized interface to over 100 services for creating foreground and background tasks for the following:

- Analog input and output
- Digital input and output
- Time and frequency measurement
- Event counting
- Pulse output
- Period measurement

In addition to basic I/O support, DriverLINX also provides:

- Built-in capabilities to handle memory and data buffer management.
- A selection of starting and stopping trigger events, including pre-triggering, mid-point triggering, and post-triggering protocols.
- Extensive error checking.
- Context-sensitive on-line help system.

Refer to your DriverLINX documentation to determine which services are supported by your KPCI-PIO32IOA or KPCI-PDISO8A board.

DriverLINX applications can be independent of some differences in data acquisitions boards because its portable APIs work across various operating systems. This capability eliminates unnecessary programming when changing operating system platforms.

TestPoint

TestPoint is a fully featured, integrated application package that incorporates many commonly used math, analysis, report generation, and graphics functions. TestPoint's graphical drag-and-drop interface can be used to create data acquisition applications for IEEE-488 instruments, data acquisition boards, and RS232-485 instruments and devices.

TestPoint includes features for controlling external devices, responding to events, processing data, creating report files, and exchanging information with other Windows programs. It provides libraries for controlling most popular GPIB instruments. OCX and ActiveX controls plug directly into TestPoint, allowing additional features from third party suppliers.

LabVIEW

LabVIEW is a fully featured graphical programming language used to create virtual instrumentation. It consists of an interactive user interface, complete with knobs, slides, switches, graphs, strip charts, and other instrument panel controls. Its data driven environment uses function blocks that are virtually wired together and pass data to each other. The function blocks, which are selected from palette menus, range from arithmetic functions to advanced acquisition, control, and analysis routines. Also included are debugging tools, help windows, execution highlighting, single stepping, probes, and breakpoints to trace and monitor the data flow execution. LabVIEW can be used to create professional applications with minimal programming.

Virtual instruments (VIs) compatible with LabVIEW are available for the KPCI-PIO32IOA and KPCI-PDISO8A.

Installing DriverLINX

Refer to the *Read this first* document that accompanies your board for installation instructions.

NOTE *Always install DriverLINX before installing the board or LabVIEW support. Both TestPoint and LabVIEW support require DriverLINX to access the board's hardware resources.*

Installing application software and drivers

Installing the TestPoint support

TestPoint provides an object for control of digital I/O channels based on the DriverLINX driver. In the directory where TestPoint is installed (C:\TESTPT), find and open DLDIO.TST. The DLINX.DIO object from this example can be copied and pasted into your custom application.

Check with the following concerning driver updates with TestPoint support for the KPCI-PIO32IOA and KPCI-PDISO8A:

- The TestPoint technical support page of the Keithley web site (www.keithley.com).
- The TestPoint manufacturer, Capital Equipment Corporation (CEC).

Installing the LabVIEW software and driver

If you plan to use only run-time LabVIEW applications with your KPCI-PIO32IOA or KPCI-PDISO8A board, you need not install DriverLINX's LabVIEW support package. LabVIEW run-time support is automatically installed when you install and configure DriverLINX.

However, if you plan to develop custom applications using the LabVIEW Integrated Development Environment, you must install the DriverLINX Virtual Instruments (VIs). These VIs are provided on your DriverLINX CD-ROM but do not install automatically when you install DriverLINX and your board. You must first install the LabVIEW application program and DriverLINX, then install the DriverLINX VIs.

If you do not install the VIs during the same session in which you install DriverLINX (refer to your *Read this first* document), you may install them later as follows:

1. Start setup.exe on the DriverLINX CD-ROM. The DriverLINX Browser introduction screen appears.
2. Click **Next** on the DriverLINX Browser introduction screen (or wait a few seconds). The DriverLINX CD Navigator screen appears.
3. On the DriverLINX CD Navigator screen, click **Install DriverLINX**. An Install These DriverLINX Components screen appears.
4. On the Install These DriverLINX Components screen, click **LabVIEW™ Support**.
5. Follow the series of on-screen instructions that appear.

Installing the KPCI-PIO32IOA and KPCI-PDISO8A boards

This subsection helps you to do the following:

- Prepare for the board installation.
- Physically install the board.

WARNING **Turn off the computer power and disconnect all external circuits from the board to avoid a possible shock hazard.**

CAUTION **Handle the board in a static-controlled workstation; wear a grounded wrist strap. Discharge static voltage differences between the wrapped board and the handling environment before removing the board from its protective wrapper. Failure to discharge static electricity before and during handling may damage semiconductor circuits on the board.**

Handle the board using the mounting bracket. Do not touch the circuit traces or connector contacts when handling the board.

Checking resources for the board

Ensure that your computer has sufficient resources, particularly power resources, to run your KPCI-PIO32IOA or KPCI-PDISO8A board. Check the capacity of the computer power supply and the power requirements of your computer and presently installed boards. Then check the additional power requirements for this board. (Refer to Appendix A, "Specifications.") If necessary, free resources by uninstalling other boards.

System responsibility

The system integrator has final responsibility for the EMC of a system containing Keithley KPCI-PIO32IOA and KPCI-PDISO8A boards. The user must not assume that installation of the CE-marked KPCI-PIO32IOA or KPCI-PDISO8A in a CE-marked PC will result in a system with acceptable RFI emissions.

Unwrapping and inspecting the board

NOTE *Install the DriverLINX software before installing the KPCI-PIO32IOA or KPCI-PDISO8A board. Otherwise, the device drivers will be more difficult to install.*

After you remove the wrapped board from its outer shipping carton, unwrap and inspect it as follows:

1. Your board is packaged at the factory in an anti-static wrapper. Do not remove the anti-static wrapper until you have discharged any static electricity voltage differences between the wrapped board and the environment. Wear a grounded wrist strap. A grounded wrist strap discharges static electricity from the wrapped board as soon as you hold it. Keep the wrist strap on until you have finished installing the board.
2. Remove the board from its anti-static wrapping material. (Store the wrapping material for future use.)
3. Inspect the board for damage. If damage is apparent, arrange to return the board to the factory. Refer to Section 5, "Troubleshooting."
4. Check the remaining contents of your package against the packing list and immediately report any missing items.
5. If the inspection is satisfactory, proceed to "Installing the board."

Installing the board

Install a board in a PCI expansion slot on your computer as follows:

1. Turn power OFF to the computer and to any external circuits attached to the board.
2. Remove the computer chassis cover.
3. Select an unoccupied PCI expansion slot in the rear panel, and remove the corresponding dummy mounting plate.
4. Insert the PCI connector of the board into the selected PCI slot of the computer. Take care not to interfere with neighboring boards. Ensure that the board is properly seated in the slot.
5. Secure the mounting bracket of the board to the chassis, using the retaining screw that you removed when you removed the dummy mounting plate.
6. Continue with the next subsection, "Configuring and checking the board and DriverLINX installations."

Configuring and checking the board and DriverLINX installations

This subsection helps you to do the following:

- Configure the combined board and DriverLINX installations, using the DriverLINX Plug and Play Wizard.
- Check the combined board and DriverLINX installations by checking the ability to start the DriverLINX PIO Control Panel.

Configuring the combined board and DriverLINX installations

After physically installing the board, turn on and reboot the computer. The DriverLINX Plug and Play Wizard screen appears automatically. Run the Wizard immediately by following the progressive instructions on the screen.

If you do not run the Wizard immediately (you cancel the Wizard after rebooting) it will normally appear automatically on subsequent reboots until you complete the final page of the Wizard. However, it is also possible to manually restart it later during the same computer session, as follows:

1. Open the Windows Explorer.
2. Double click on X:\DrvLINX4\Help\kcpi.bat, where X = the letter of the drive on which you installed DriverLINX. The Wizard appears.

NOTE *You can also start this batch file directly from the CD-ROM by double clicking on Y:\DrvLINX4\Help\KPCIISO.BAT, where Y = the letter of your CD-ROM drive.*

3. On the Windows 98/Me Plug and Play Wizard, click Wizard and follow the series of on-screen instructions that appear. The wizard will first lead you through the steps of installing your hardware—from a software viewpoint—and configuring it.
4. After you have finished configuring the board and DriverLINX, continue with the next subsection "Checking the combined board and DriverLINX installations."

Checking the combined board and DriverLINX installations

Before making any connections to the board, check the combined board and DriverLINX installations. The ability to start the AIO Panel utility, which is available after you install DriverLINX, verifies that DriverLINX and the board are installed and configured satisfactorily and are working together.

1. Try starting the AIO Panel as follows:
   a. Click the Windows Start tab.
   b. In the Start menu, click Programs.
   c. Under DriverLINX ▸ Test Panels, find the AIO Panel entry.
   d. Click on the AIO Panel entry. The AIO Panel should appear.
   e. Based on the results of step 1d, do one of the following:
      - If you are able to start the AIO Panel, skip to the next subsection, "Connecting the KPCI-PIO32IOA and KPCI-PDISO8A boards."
      - If you are not able to start the AIO Panel, then continue with step 2.

2. If you are unable to start the AIO Panel utility, then check to ensure that the installation is properly configured and DriverLINX is properly installed. Refer to the subsection "Configuring the KPCI-PIO32IOA and KPCI-PDISO8A" in the DriverLINX manual *Using DriverLINX with your Hardware—Keithley KPCI-PIO32IOA and KPCI-PDISO8A*. To access this manual on-line, do the following:
   a. Click the Windows Start tab.
   b. In the Start menu, click Programs.
   c. Find the DriverLINX folder, under which you should find the On-line Manuals entry.
   d. Click on the On-line Manuals entry. The DriverLINX Printable Documentation table of contents opens via Acrobat reader.
   e. Scroll through the DriverLINX Printable Documentation table of contents and find Configuration.
   f. Under Configuration, click Hardware References. A list of documents appears.
   g. In the list of documents, click on Using DriverLINX with Your Hardware—Keithley KPCI-PIO32IOA and KPCI-PDISO8A. The manual opens via Acrobat Reader.
   h. Print the manual now, if possible, to make reference easier.

NOTE *Acrobat Reader must already be installed. If necessary, you can first install Acrobat Reader directly from the CD-ROM by double clicking X:\Acrobat\setup.exe.*

If your data acquisition system has no printer, you can display and print the manual from the CD-ROM at another system. From Windows Explorer, select the CD-ROM drive (drive "X"). To display the Using DriverLINX with Your Hardware—Keithley PIO Series manual, double click on X:\Drvlinx4\Docs\Notes\kpci.pdf.

3. Reconfigure the installation as required, and if necessary, also reinstall DriverLINX.

NOTE *If you reinstall DriverLINX, refer also to your* Read this first *sheet (which is also available on the CD-ROM that came with your board). Make sure that DriverLINX installs smoothly and completely.*

4. Try again to start the AIO Panel (refer to step 1).
5. Based on the results of step 4, do one of the following:
   - If you can now start the AIO Panel, DriverLINX and the board are installed and configured properly and are working together. Proceed to the next subsection, "Connecting the KPCI-PIO32IOA and KPCI-PDISO8A boards."
   - If you are still unable to start the AIO Panel, refer to Section 5, "Troubleshooting," for additional procedures that may help you isolate the problem.

Connecting the KPCI-PIO32IOA and KPCI-PDISO8A boards

This subsection helps you to do the following:

- Identify pin assignments on the board.
- Connect interface accessories.
- Wire the KPCI-PIO32IOA or KPCI-PDISO8A I/O to your external circuits.

Identifying I/O connector pin assignments

The KPCI-PIO32IOA board has two 36-pin mini-D type I/O connectors for I/O, while the KPCI-PDISO8A board has one 36-pin mini-D type I/O connector for I/O. Figure 3-1 shows the configuration of the KPCI-PIO32IOA board, and Figure 3-2 shows the KPCI-PDISO8A board. Table 3-1 defines connector pin assignments for both boards. Note that those terminals labeled "IP" are control inputs, while "OP" terminals are switching outputs.

Figure 3-1
KPCI-PIO32IOA board configuration

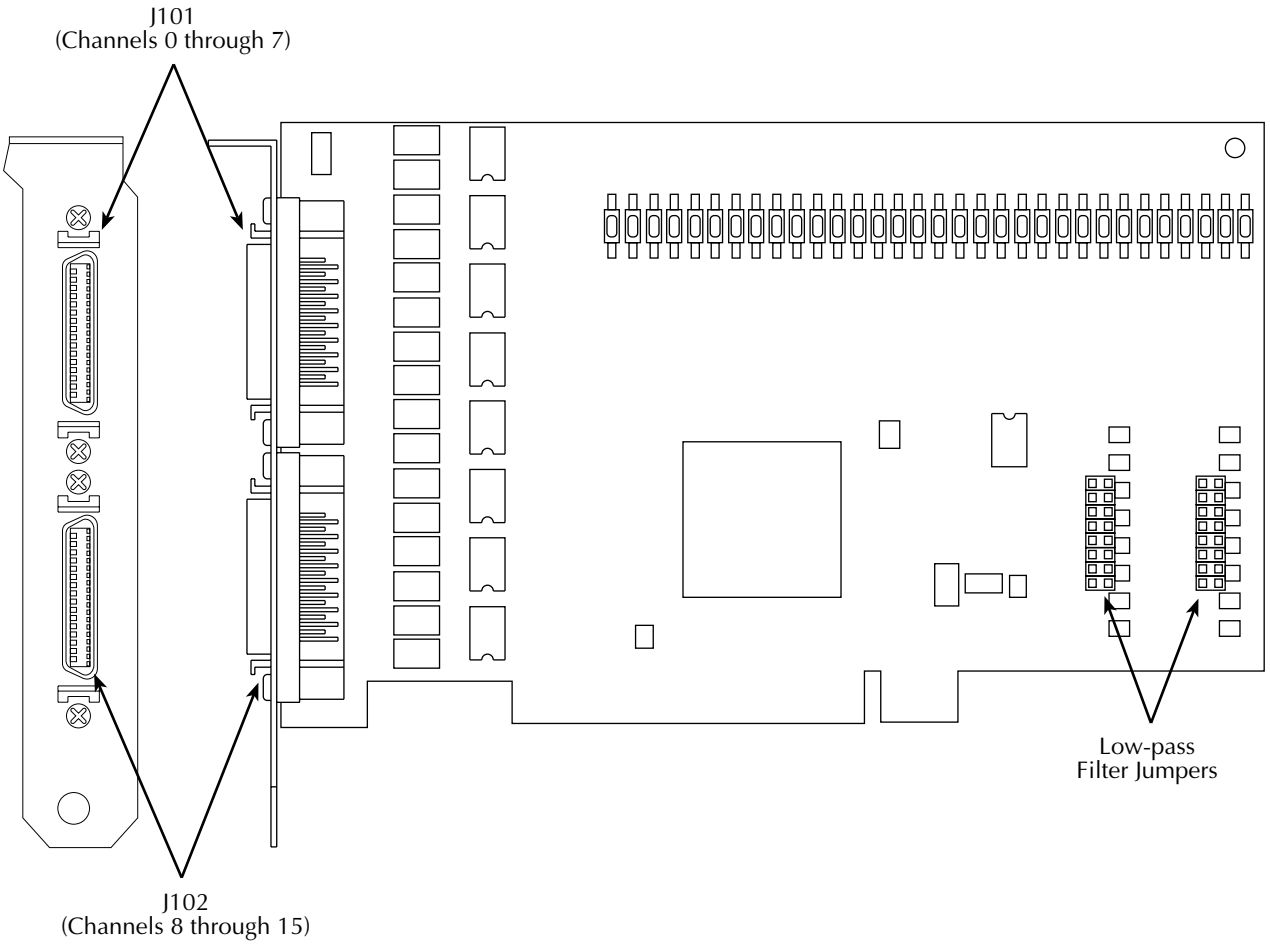


Figure 3-2
KPCI-PDISO8A board configuration

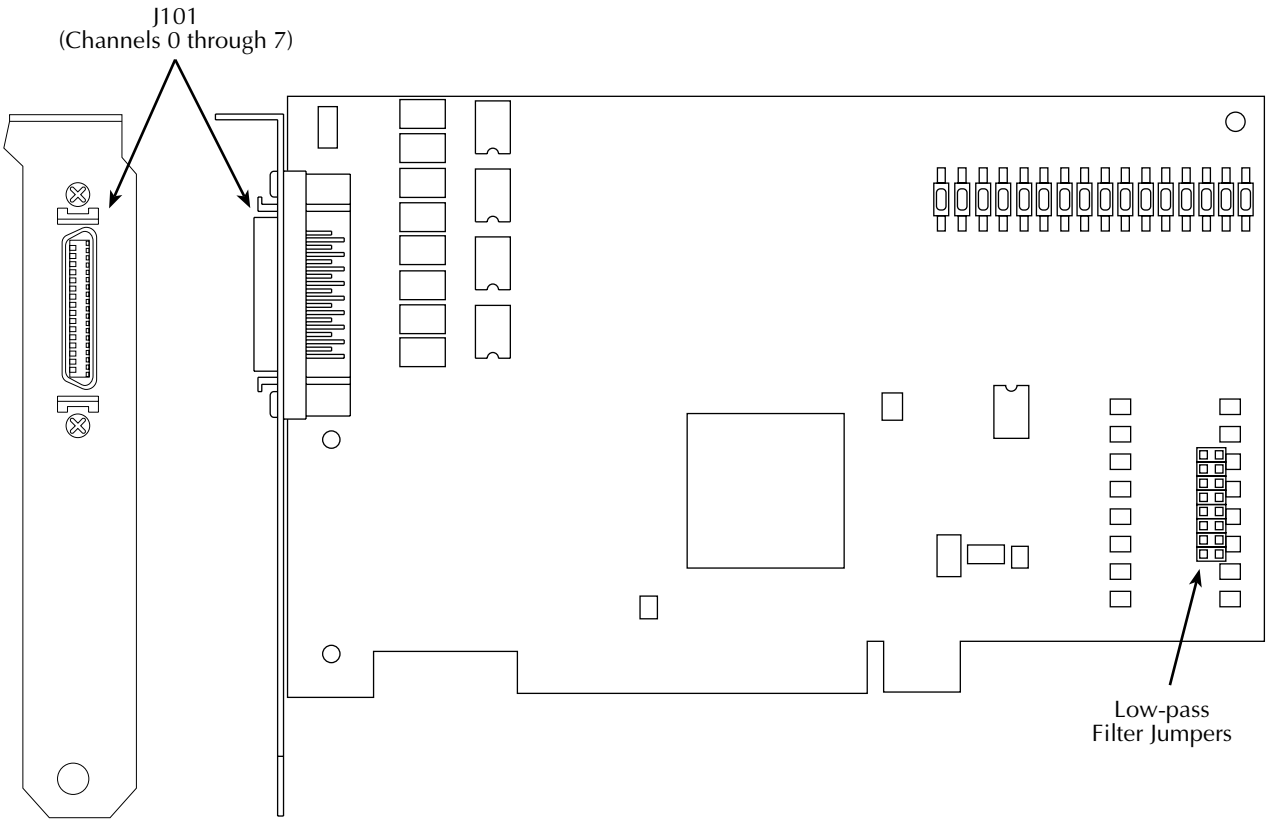


Table 3-1
KPCI-PIO32IOA and KPCI-PDISO8A I/O connector pin assignments

Top Connector (J101)				Bottom Connector (J102. KPCI-PIO32IOA Only)			
Pin Number	**Signal**	**Pin Number**	**Signal**	**Pin Number**	**Signal**	**Pin Number**	**Signal**
1	+5V	19	GND	1	+5V	19	GND
2	-	20	-	2	-	20	-
*3	IP7	21	IP7	*3	IP15	21	IP15
*4	IP6	22	IP6	*4	IP14	22	IP14
5	IP5	23	IP5	5	IP13	23	IP13
6	IP4	24	IP4	6	IP12	24	IP12
7	IP3	25	IP3	7	IP11	25	IP11
8	IP2	26	IP2	8	IP10	26	IP10
9	IP1	27	IP1	9	IP9	27	IP9
10	IP0	28	IP0	10	IP8	28	IP8
11	OP7	29	OP7	11	OP15	29	OP15
12	OP6	30	OP6	12	OP14	30	OP15
13	OP5	31	OP5	13	OP13	31	OP13
14	OP4	32	OP4	14	OP12	32	OP12
15	OP3	33	OP3	15	OP11	33	OP11
16	OP2	34	OP2	16	OP10	34	OP10
17	OP1	35	OP1	17	OP9	35	OP9
18	OP0	36	OP0	18	OP8	36	OP8

*Pin 3 is interrupt enable, and Pin 4 is interrupt request.

Connecting interface accessories

WARNING **Make sure power to all external circuits is turned OFF before connecting to the KPCI-PIO32IOA or KPCI-PDISO8A board to avoid a possible shock hazard.**

CAUTION **Handle the board at the mounting bracket, using a grounded wrist strap. Do not touch the circuit traces or connector contacts. If you do not have a grounded wrist strap, periodically discharge static electricity by placing one hand firmly on a grounded metal portion of the computer chassis.**

Using cable assemblies

The KPCI-PIO32IOA and KPCI-PDISO8A I/O connectors can be mated directly to your external circuits using locally fabricated cable assemblies. To mate a locally fabricated cable to an I/O connector, install an IEEE-1284 (1994) connector on the cable. (Mating plug 3M part number: 10136-6000EC.)

NOTE *The 36-pin mating connector is NOT available in a solder-cup version; use of the specified connector will require appropriate cable and tooling to make a reliable insulation displacement connection.*

Using the STP-36 screw terminal connector accessory

The screw terminals of the STP-36 connector accessory allow you to make easy I/O connections to the KPCI-PIO32IOA and KPCI-PDISO8A boards with screw terminals. The STP-36 connects to the board I/O connector (J101 and J102) using a CAB-1284CC cable.

As summarized in Table 3-2, the screw terminal designations (1-36) correspond to the designations of the J101 and J102 I/O connectors. For example, STP-36 screw terminal 3 corresponds to pin 3 (IP7) on the board I/O connector J101.

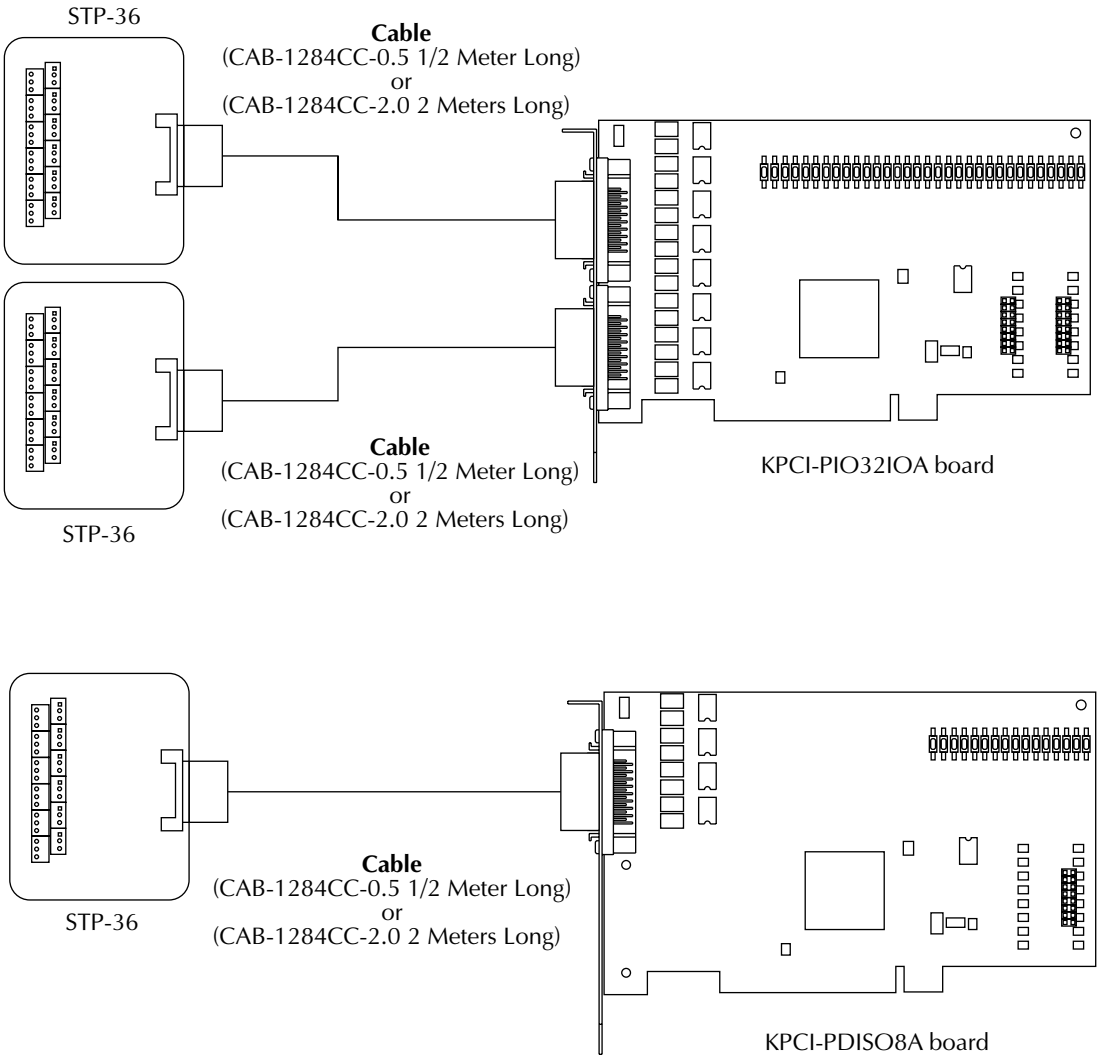
WARNING **When using the STP-36, the maximum allowed voltage is 30V RMS, 42.4V peak, or 60V DC. Exceeding this limit could cause an insulation breakdown and a shock hazard.**

Table 3-2
STP-36 screw terminal designations

STP-36 Terminal	J101/J102 Pin	STP-36 Terminal	J101/J102 Pin	STP-36 Terminal	J101/J102 Pin
1	1	13	13	25	25
2	2	14	14	26	26
3	3	15	15	27	27
4	4	16	16	28	28
5	5	17	17	29	29
6	6	18	18	30	30
7	7	19	19	31	31
8	8	20	20	32	32
9	9	21	21	33	33
10	10	22	22	34	34
11	11	23	23	35	35
12	12	24	24	36	36

Figure 3-3 shows typical STP-36 connections to the KPCI board using the designated cables.

Figure 3-3
Connections to STP-36 screw terminal connector accessory



Wiring I/O to your external circuits

WARNING **Ensure that both the computer and the external circuits are turned OFF before making any connections to avoid a possible shock hazard.**

Input and output precautions

Over-current conditions

CAUTION **The KPCI output channels are not designed to handle over-current conditions. A protected source that is compatible with the KPCI specifications should always be used. If this is not possible, external over-current protection is required to protect both the KPCI and the rest of the system.**

Transient voltages

CAUTION **The KPCI card provides limited input and output protection to protect the user PC against high voltage events such as ESD strikes and voltage surges. If, however, voltage transients over 100V peak are expected to occur frequently, as with highly inductive load switching, then external protection should be provided to reduce repetitive stresses on the KPCI card, source and load.**

Non-isolated power pins

CAUTION **The KPCI card provides fuse-protected +5V and ground for convenient use in external circuits. This voltage is supplied from the bus and is not isolated from the user PC. If isolation is a requirement in the application, then this supply should either not be used at all, or be used with some external means of isolating this supply.**

Power-up state of channels

NOTE *The power-up state of the input and output channels is as follows:*

- *Input channels — For $|V| < \cong 45V$, $Z \cong 22k\Omega$. For $|V| > 45V$, $Z \cong 17k\Omega$. These channels are always configured as inputs and can be driven at all times.*
- *Output channels — These are form A, N.O. SPST style contacts. The output channels are in a high-impedance state (see specifications in Appendix A) under any of the following conditions:*
  - *The KPCI card is in the system reset state.*
  - *The system and KPCI card are unpowered.*
  - *The system and the KPCI card are powered but no software command has been given for actuation (initial state). This state is also configurable in DriverLINX.*

Input and output connections

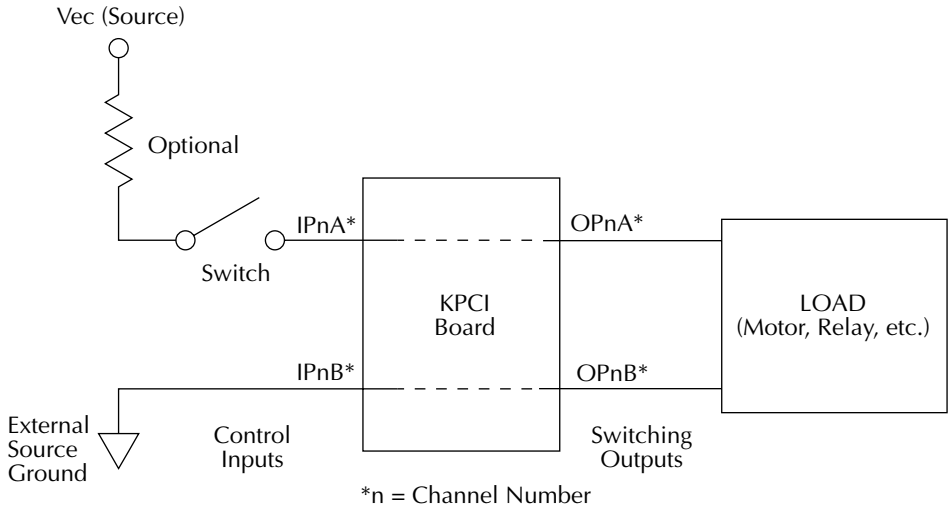
Figure 3-4 shows typical input and output connections for one channel. IPnA and IPnB are the control inputs, while OPnA and OPnB are the switching outputs (n is the channel number). See Table 3-2 for terminal designations).

Signal voltages are as follows:

- Maximum control input voltages are: ±60V DC (logic high); ±1V DC (logic low).
- Outputs can switch both DC voltages (±60V DC maximum) and AC voltages (30V RMS maximum) with a maximum load current of 350mA. An output relay is turned on (closed) when the corresponding control input is placed at a logic high.

NOTE *DC control inputs require approximately 80µA at 2.5V input and approximately 3.5mA at 60V input.*

Figure 3-4
Typical KPCI-PIO32IOA and KPCI-PDISO8A input/output connections



Using +5VDC from the computer power supply

CAUTION **Do not connect the +5V DC outputs to an external +5VDC supply. Doing so may damage the external supply, the board, and the computer.**

The +5V DC output and ground connections are fuse protected and intended for convenient use in external circuits. This voltage is supplied from the PC bus and is not isolated from the user PC. If isolation is required, do not use this supply to power external circuits, or use some external means of isolating this supply.

Do not draw more than 1A, total. Drawing excessive current may damage the KPCI-PIO32IOA or KPCI-PDISO8A board and, potentially, the computer.

The board extends fuse-protected power from the +5V DC computer supply to the I/O connector(s). This power is convenient for use in light external circuits, such as pull-up resistors. If you ensure that the conditions in the above ***CAUTION*** notice are maintained, this power may also be used to energize external accessories, provided that your computer's power supply is capable of delivering the desired current.

I/O considerations

De-bouncing inputs using the low-pass filter

The optional on-card low-pass filter can be used to de-bounce the inputs. A low pass filter for a given channel can be enabled by installing the appropriate jumper at J100 or J200. See Figure 3-1 or Figure 3-2 for jumper locations, and the specifications in Appendix A for details on filter characteristics. Note that use of the low-pass filter is recommended for AC sensing.

Power dissipation limitations

Because of variations in solid-state relay on resistance (3.5Ω typical to 4Ω maximum), the actual maximum current may be reduced from the stated 350mA maximum by the maximum power dissipation limit of 430mW. Current for a given on resistance can be calculated as follows:

$$I = \sqrt{\frac{P}{R}}$$

Where: I = current in amps
P = power in watts (430mW maximum)
R = resistance in ohms

For example, with an on resistance of 4Ω, the maximum current is reduced to 328mA as follows:

$$I = \sqrt{\frac{0.43}{4}}$$

Frequency response limitations

Because optically coupled relays are used for switching, AC signals with higher frequencies may be significantly attenuated when passing through the board. For that reason, it is recommended that you use a different board that contains suitable relays if AC performance at higher frequencies is important.

Combining output channels

In some applications, it may be desired to attempt parallel operation of multiple output channels to provide a load current that is higher than specified for one individual output channel. With some care and a good understanding of the limitations, this method can be used successfully with the KPCI board output channels.

Whenever KPCI channels are operated in parallel, due consideration should be given to each channel to ensure that each channel will remain operating within specified limits. Some items to consider while attempting this mode of operation are: KPCI PCB layout, dynamic and steady-state current unbalance, and temperature unbalances pertaining to the final circuit or system configuration. The KPCI PCB layout has been designed to minimize stray passive and reactive components where possible within each channel to the connector. These stray components are generally not dominant when low-frequency operation of the isolated output channels is used.

General guidelines

- Voltage equity of the channels is assured by the parallel configuration. However, under transient conditions, voltage differentials can appear across devices due to di/dt effects in unequal stray inductances. Strive to reduce stray inductance to values that give acceptable overshoots at the maximum operating current.
- Cables should be reasonably matched for a given connector group of channels. Try to avoid using significantly different cable lengths when paralleling channels that span cable connector groups.
- The power source that is to be switched should be as "stiff" as possible. For example, a voltage source should have very low output impedance.
- While software writes to the output channels will actuate each channel at nearly the same time, do not assume that the output channels themselves will traverse their high-to-low impedance characteristics at the same rates.

Steady-state current sharing

During the periods of time outside of the switching transitions, the current in a parallel group of output MOSFET channels will distribute itself in the individual devices in inverse proportion to their ON resistance. The channel with the lowest ON resistance will carry the highest current. This will, to an extent, be self-compensating because the power loss in this channel will be the highest. It will run the hottest, and the increase in ON resistance due to heating will be more than that of the other channels, which will tend to equalize the current.

Precautions

Once the system is wired for use in an application, it is prudent to examine the switching behavior of the paralleled channels in an offline test. This will ensure that, for the particular application, there will be no unnecessary stresses to the system with the paralleled configuration. There is no straightforward method for determining whether or not channel combining can be used successfully in a particular application. The decision should be based on how well the system channels can be matched, and how they are used with the particular sources and loads.

Channel combining cannot entirely avoid mismatches and during switching, transients and stresses are placed on the KPCI output channels. Though these stresses may be minimized, they may affect long-term reliability of the channels when high duty cycle operation is desired with combined channels.

4
External Interrupts

Using DriverLINX, you can program the driver to read or write data every time the board detects the rising edge or the falling edge of an external interrupt signal. Section 4 explains external interrupt signal connections and summarizes interrupt usage for the KPCI-PIO32IOA and KPCI-PDISO8A boards. For more information, refer to *Using DriverLINX with Your Hardware-Keithley* and other DriverLINX manuals.

External interrupt description

Through DriverLINX, you can alternatively configure two of the input lines for the following external interrupt functions:

- $\overline{\text{INT\_ENN}}$ (external interrupt enable, active low)
- INT_REQ (external interrupt request, edge triggered)

Interrupt pins

The interrupt pins on the I/O connector of the KPCI-PIO32IOA and KPCI-PDISO8A boards, and the corresponding screw terminal on the STP-36 screw terminal accessory, are identified in Table 4-1.

Table 4-1
Connector pins used for external interrupt function

Interrupt Function		Connector pins that are configurable for either interrupt or general-purpose functions		
Description	**Signal Designator**	**KPCI-PIO32IOA and KPCI-PDISO8A I/O Connector, J101**	**STP-36 Screw Terminal**	**Corresponding General-Purpose Function**
External interrupt enable	$\overline{\text{INT\_ENN}}$	3, 21	3, 21	IP7
External interrupt request	INT_REQ	4, 22	4, 22	IP6

Through DriverLINX, you can independently configure the input port so that an external interrupt request pulse causes the data at the input ports to be placed into a buffer.

Through DriverLINX, you can also configure interrupts to be edge-clocked for either of the following conditions:

- The rising (positive) edge of the external interrupt request signal
- The falling (negative) edge of the external interrupt request signal

NOTE *Positive-edge clocking and no latching are the defaults upon power-up or reset.*

Data latching

Latching external data is accomplished by first setting up the control register to latch on either a rising or falling edge. Once the appropriate external edge is received on the INT_REQ line, the current data value will be latched. The receipt of this external edge will also set the interrupt pending bit. An actual PCI interrupt will only occur at this time if the interrupts are enabled. This value will continue to be latched until the control register is written with a non-latching value. This effectively turns off the latching mode. Input data will not latch regardless of any INT_REQ edges. This state is the default power up state of the KPCI boards.

If the currently latched value was not acknowledged when the next external edge arrives, new data will be latched. See the specifications for the register maps containing the locations of the latch control register and interrupt enable and pending bits.

When data is currently latched, IP6 and IP7 will not be latched. These pins function as INT_REQ and $\overline{\text{INT\_ENN}}$ in addition to digital input lines.

In order to synchronize external data with hardware events represented on the INT_REQ data line, it is recommended that an interrupt driven programming scheme be used.

External interrupt application

Typically, you would use an external interrupt to ensure that the application program processes, at a specific time, specific data that are present at one or more ports. You can do so with a board external interrupt, subject to the following limitations:

- A KPCI-PIO32IOA and KPCI-PDISO8A external interrupt signal signals the driver to transfer data from whichever port is desired: one, several at once, or several with one per interrupt. The board provides no way to assign interrupts or intrinsically detect the interrupt status for a specific bit. (However, you could potentially hardware-set and software-detect certain I/O bits to tell a custom application program which ports to process for a given interrupt, or you could compare the current and previous data sets, and then process only changed data.)
- Conversely, if processing needs and interrupts occur too frequently, data that *does* need to be processed can be *missed*. External interrupt signals are ignored during an interrupt service routine (ISR) while an interrupt-pending bit is set high—except that the first interrupt miss, only, does set an interrupt-missed bit.

Therefore, you must 1) know how fast your host computer and application software can process data, and 2) space important data and interrupts accordingly. Of course, this requirement is not specific to the KPCI-PIO32IOA and KPCI-PDISO8A boards. It applies to any data acquisition board. See the specifications in Appendix A for hardware input response times.

You must also ensure, at the end of interrupt-initiated processing, that the application program resets the interrupt-pending bit, the interrupt-missed bit, and two other bits that recorded whether the interrupt occurred on the rising or falling edge of the interrupt request signal. DriverLINX handles this process automatically.

Refer to Table 4-2 for a list of external-interrupt responses.

For an additional application illustration, refer to the "Interrupt example scenario" subsection in Appendix B. For information on setting and configuration of external interrupts, refer to your DriverLINX software manuals.

NOTE *Do not confuse the external interrupt INT_REQ with the internal PCI bus interrupt INTA, even though INT_REQ and INTA are linked by hardware.*

Table 4-2
External-interrupt responses

Result	Interrupt Control/Status Register Interrupt Control Bits (x = don't care)		Port Group Control Register (x = don't care)		External Inputs at J101		Interrupt Control/ Status Register Status after Edge Transition*
	INT Enable Bit 12	Polarity Bit 6	Bit 6	Bit 5	Enable Pins 3.21	Edge Pins 4,22	Pending Bit Bit 17
PCI-INT; No Latch	1	0	x	0	0	↑	1
PCI-INT; No Latch	1	1	x	0	0	↓	1
No PCI-INT; No Latch (Programmed I/O Mode)	0	x	x	0	0	↑ or ↓	0
No PCI-INT; Latch Input Data	0	x	0	1	0	↑	1
No PCI-INT; No Latch	0	x	0	1	0	↓	0
No PCI-INT; No Latch	0	x	1	1	0	↑	0
No PCI-INT; Latch Input Data	0	x	1	1	0	↓	1
No PCI-INT; No Latch (Programmed I/O Mode)	x	x	x	x	1	↑ or ↓	0
PCI-INT; Latch Input Data	1	0	0	1	0	↑	1
No PCI-INT; Latch Input Data	1	1	0	1	0	↑	1
No PCI-INT; No Latch	1	0	0	1	0	↓	0
PCI-INT; No Latch	1	1	0	1	0	↓	1
PCI-INT; No Latch	1	0	1	1	0	↑	1
No PCI-INT; No Latch	1	1	1	1	0	↑	0
No PCI-INT; Latch Input Data	1	0	1	1	0	↓	1
PCI-INT; Latch Input Data	1	1	1	1	0	↓	1

*For bit 17, assume an initial clear state before edge.

5
Troubleshooting

If your KPCI-PIO32IOA or KPCI-PDISO8A board is not operating properly, use the information in this section to isolate the problem before calling Keithley Applications Engineering. If you then need to contact an applications engineer, refer to "Technical support."

Identifying symptoms and possible causes

First try to isolate the problem using Table 5-1, which lists general symptoms and possible solutions for KPCI-PIO32IOA or KPCI-PDISO8A board problems.

Table 5-1
Basic troubleshooting information

Symptom	Possible Cause	Possible Cause Validation/Solution
Computer does not boot when board is installed.	Resource conflict. KPCI-PIO32IOA or KPCI-PDISO8A board is conflicting with other boards in the system.	1. Validate the cause of the conflict. Temporarily unplug boards—especially ISA boards[1]—one at a time, and try booting the computer. Repeat until a boot is attained. 2. Try resolving conflicts by reinstalling one PCI board at a time and rebooting after each reinstallation.[2] However, you may ultimately need to change ISA board resource allocations, such as base address or interrupt assignments.
	Board not seated properly.	Check the installation of the board.
	The power supply of the host computer is too small to handle all the system resources.	Check the needs of all system resources and obtain a larger power supply.
Board does not respond to the AIO Panel.	DriverLINX is not installed properly or is not configured properly.	Refer to "Configuring the combined board and DriverLINX installations" on page 3-6. Check the Windows® Device Manager and follow the installation troubleshooting instructions in the DriverLINX on-line help.
	The board is incorrectly aligned in the expansion slot.	Check the board for proper seating.
	The board is damaged.	Contact Keithley Applications Engineering.
Data appears to be invalid.	An open connection exists.	Check screw terminal wiring.
	Transducer is not connected to channel being read.	Check the transducer connections.
	One or more external circuits are not compatible.	Check external circuit schematics. Test external circuits with a DMM.
+5V power is not available at I/0 connector.	External circuit digital output and power loads exceed current capacity of the board, tripping self-resetting fuses.	Check external circuits for shorts. Check the external circuit load (must be less than 1A, total).
Intermittent operation.	Vibrations or loose connections exist.	Cushion source of vibration and tighten connections.
	The board is overheating.	Check environmental and ambient temperature. See your computer documentation.
	Electrical noise exists.	Check for AC power lines in close proximity to signal lines. Provide better shielding or reroute unshielded wiring.
System lockup during operation.	A timing error occurred.	Restart your computer. Then analyze your program by debugging and narrowing the list of possible failure locations.

[1]Plug and Play cannot tell if an ISA board already uses an address it assigns to a PCI board.
[2]Plug and Play may then assign different, nonconflicting addresses to the PCI boards.

If your board is not operating properly after using the information in Table 5-1, continue with "Systematic problem isolation" to further isolate the problem.

Systematic problem isolation

General problem isolation procedure

If you were unable to isolate the problem using Table 5-1, then follow Figure 5-1 and the accompanying written procedure. The flowchart in Figure 5-1 summarizes how to systematically check and eliminate some problem causes. The corresponding written procedure amplifies the flowchart steps with more detail.

WARNING **To avoid a possible shock hazard, always turn OFF your computer and any external circuits connected to the KPCI-PIO32IOA or KPCI-PDISO8A board before removing or replacing the board.**

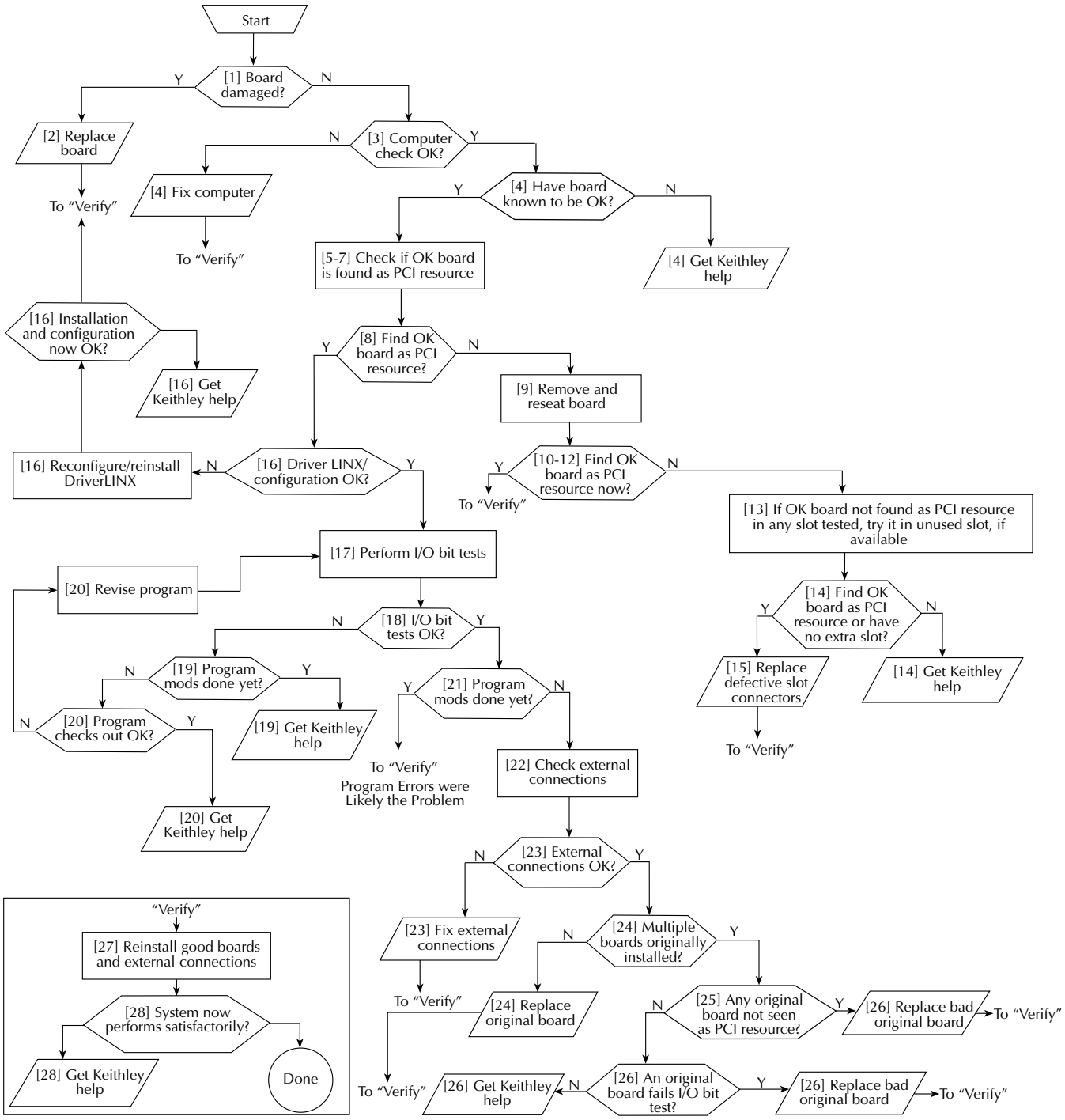
CAUTION **Handle the board at the mounting bracket, using a grounded wrist strap. Do not touch the circuit traces or connector contacts. If you do not have a grounded wrist strap, periodically discharge static electricity by placing one hand firmly on a grounded metal portion of the computer chassis.**

NOTE *In the following procedure, the term "board" always refers to a KPCI-PIO32IOA or KPCI-PDISO8A board. The procedure never directs you to install or remove any type of PCI board other than a KPCI-PIO32IOA or KPCI-PDISO8A board.*

In Figure 5-1, the number in brackets in each flowchart block (e.g. [21]*) refers to the corresponding step number in the verbal procedure. If multiple blocks in the flowchart have the same number, each of those blocks is part of a single verbal step. Conversely, if there is a range of numbers in the brackets (e.g.* [4, 5, 6]*), the block summarizes multiple verbal steps.*

The logic in this procedure assumes that the problem has only one cause. Therefore, once a cause is found and corrected, the reader is instructed to reassemble the system and verify proper operation.

Figure 5-1
Problem isolation flowchart



To further isolate the problem to the KPCI-PIO32IOA or KPCI-PDISO8A board or to the host computer, use the following steps:

1. Check if board damage is seen on inspection. Proceed as follows:

NOTE *If more than one KPCI-PIO32IOA or KPCI-PDISO8A board is installed in your computer, disconnect, remove, and check all KPCI-PIO32IOA and KPCI-PDISO8A boards.*

   a. Turn power OFF to the host computer.
   b. Turn power OFF to all external circuits and accessories connected to all installed KPC-boards.
   c. From each installed KPCI-PIO32IOA and KPCI-PDISO8A board, unplug the STP-36 accessory that is wired to external circuits.

NOTE *Do not disconnect any circuits from the STP-36 screw terminals.*

   d. Remove all KPCI-PIO32IOA and KPCI-PDISO8A boards from the computer.
   e. Visually inspect all KPCI-PIO32IOA and KPCI-PDISO8A boards for damage.
2. Based on the results of step 1, do the following:
   - If all KPCI-PIO32IOA and KPCI-PDISO8A boards are not obviously damaged, continue with step 3 and check for host computer malfunction.
   - If a board(s) is obviously damaged on inspection, repair or replace the board. Refer to "Technical support" for information on returning the board for repair or replacement. Skip to step 27.
3. Check if the computer functions satisfactorily by itself. Proceed as follows:
   a. Remove all KPCI-PIO32IOA and KPCI-PDISO8A boards from the host computer.
   b. Turn ON power to the host computer.
   c. Perform any necessary diagnostics to the computer hardware and operating system.
4. Based on the results of step 3, do one of the following:
   - If the computer functions satisfactorily, the problem must lie elsewhere; do the following steps:
     a. If you have another board that you know is OK, i.e. works properly, then continue with step 5.
     b. If you do not have another board that you know is OK, i.e. works properly, read the instructions in "Technical support." Then contact Keithley for help in isolating the cause of your problem.
   - If the computer does not function satisfactorily, do the following steps:
     a. Diagnose and fix the computer malfunction.
     b. Assume that fixing the computer malfunction has solved your problem, and skip to step 27.
5. Prepare for PCI resource checks in step 6 by clearing any KPCI-PIO32IOA and KPCI-PDISO8A board listings from the Windows 98/Me/2000/XP device manager. If any KPCI-PIO32IOA and KPCI-PDISO8A boards are listed, the computer could falsely report as PCI resources one or more of the now-empty PCI slots that previously held the KPCI-PIO32IOA and KPCI-PDISO8A boards.

   Proceed as follows for the Windows 98/Me/2000/XP operating system. (A different procedure is required for Windows NT.):
   a. Right-click the **My Computer** icon on your desktop.
   b. On the menu that appears, click **Properties**.

c. On the System Properties dialog box that appears, click the **Device Manager** tab. The Device Manager appears.

d. In the Device Manager, look for a **DriverLINX drivers** item.

- If you find a **DriverLINX drivers** item with a **+** sign to the left of the item, click the **+** sign. A second level list may appear containing one or more boards.
- If you do not find a **DriverLINX drivers** item, skip to step 6.

e. Delete any KPCI-PIO32IOA and KPCI-PDISO8A boards, *only*, from the list by highlighting the item and clicking the **Remove** button.

6. Check if the computer finds an OK board to be a PCI resource in a slot from which you removed a KPCI-PIO32IOA and KPCI-PDISO8A board.
   a. Shut down Windows 95/98/NT and turn OFF power to the host computer.
   b. Perform the procedure "Checking for board as a PCI resource" found later in Section 5.
7. If you had originally installed additional KPCI-PIO32IOA and KPCI-PDISO8A boards in other PCI slots, then repeat step 6 with the OK board in each of these other slots.
8. Based on the results of steps 6 and 7, do one of the following:
   a. If the board is recognized as a PCI component in all slots tested, then there may be software issues. Skip to step 16.
   b. If the OK board is not recognized as a PCI component in all slots tested, then a PCI slot connector(s) is suspect. Continue with step 9.
9. Make sure that the suspect slot connector and the board contacts have been wiped adequately and are properly mated. Do the following:
   a. Turn OFF power to the host computer.
   b. Remove and reseat the board a few times in the suspect PCI slot connector. This creates a wiping action to improve the probability of good contact.
   c. Make sure that the board is firmly seated in the suspect PCI slot connector.
   d. Replace the cover of the computer.
   e. Turn ON power to the host computer.
10. Check if the computer now finds the OK board to be a PCI resource in the suspect slot connector. Use step 3, only, of the procedure "Checking for board as a PCI resource."
11. If you found more than one suspect slot connector, then repeat steps 9 and 10 with the OK board in the other suspect slot connectors.
12. Based on the results of steps 9 through 11, do one of the following:
    - If the board was recognized as a PCI resource in all slots tested, then the cause of the problem was probably high contact resistance, which apparently has been corrected by the wiping action. Skip to step 27.
    - If the OK board was not identified as a PCI resource in all slots tested, then any slots in which it was not identified are still suspect. Continue with step 13.
13. Based on the history of steps 6 through 12, do one of the following actions,
    - If, at this point, *both* of the following are true, then install and test the OK board in a previously unused slot to confirm that the originally tested slot(s) (rather than some other factor) was at fault:
      – The OK board was not identified as a PCI resource in *any* slot that you tested.
      – You have a PCI slot that was unused before you started the problem isolation scheme.

Proceed as follows:

a. Turn OFF power to the host computer.

b. Move the KPCI-PIO32IOA or KPCI-PDISO8A board to the slot that was unused before you started the problem isolation scheme.

c. Reinstall the cover of the computer.

d. Perform step 3, only, of the procedure "Checking for board as a PCI resource," found later in Section 5.

e. Continue with step 14.

- If, at this point, you do not have a PCI slot that was unused before you started the problem isolation scheme, continue with step 14.

14. Based on the history of steps 6 through 13, do one of the following:
    - If, at any point in steps 6 through 13, the OK board was identified as a PCI resource in at least one slot on the host computer, then any slot in which it did not work is likely defective. Continue with step 15.
    - If you only had one slot in which to install the OK board, assume that this slot is defective. Continue with step 15.
    - However, if at this point you have tested the OK board in multiple slots and the board has not been identified as a PCI resource in *any* slot, then the cause of your problem may be outside the scope of these diagnostics. Read the instructions in "Technical support," and then contact Keithley for help in isolating the cause of your problem.
15. Replace the defective slot connector, as follows:
    a. Turn OFF the computer.
    b. Remove the OK board.
    c. Have a qualified service person replace the defective PCI slot connector.
    d. Skip to step 27.
16. Continuing from step 8, check whether DriverLINX is installed and configured properly to work with the board and, if problems are found, try to remedy them.
    a. Check whether you can start the AIO Panel. Refer to steps 1a through 1d of "Checking the combined board and DriverLINX installations" in Section 3.
    b. Based on the results of step 16a above, do one of the following:
        - If the AIO Panel starts, the board and DriverLINX are installed and configured properly. Skip to step 17 of the problem isolation scheme. Your problem may be caused by faulty I/O.
        - If the AIO Panel does not start, the combined board and DriverLINX are not configured properly, or DriverLINX is not installed properly. Continue with step 16c of the problem isolation scheme.
    c. Perform steps 2 through 4 of "Checking the combined board and DriverLINX installations" in Section 3.
    d. Based on the results of step 16c above, do one of the following:
        - If the AIO Panel now starts, then skip to step 27.
        - If the AIO Panel still does not start, then your problem may be outside the scope of these diagnostics. Read the instructions in "Technical support," and then contact Keithley for help in isolating the cause of your problem.
17. Perform the following I/O bit test to determine whether you can write and read all I/O bits to and from the OK board. This test is outlined separately under the heading "I/O test".

18. Based on the results of step 17, do one of the following:
    - If you can write and read all I/O bits to and from the OK board, skip to step 21.
    - If you cannot write and read all I/O bits to and from the OK board, there may be data acquisition program errors or defective slot contacts corresponding to I/O bits; continue with step 19.
19. You have arrived at this step because you cannot write and read all I/O bits to and from the OK board. You may have already tried some data acquisition program source-code debugging. Do one of the following:
    - If *either* of the following statements are true, then continue with step 20.
      – You have arrived at step 19 before debugging the source code.
      – You have arrived at step 19 after debugging the source code once, but you have not yet tried to find more code bugs after two or more I/O test failures.
    - If *both* of the following statements are true, then the cause of your problem may be outside the scope of these diagnostics. Read the instructions in "Technical support," and then contact Keithley for help in isolating the cause of your problem:
      – You have arrived again at step 19 after having debugged the source code at least once and after having failed the I/O bit tests at least a second time.
      – You have tried to find more code bugs after two or more I/O test failures and cannot find any more bugs.
20. Thoroughly check if all data acquisition program source-code lines are OK. Check the program documentation and/or use a debugger to look for programming errors that may be causing the problem.
    - If programming errors are found, do the following:
      a. Debug and fix all known data acquisition program errors.
      b. Repeat steps 17 and 18.
    - If no programming errors are found after thorough debugging, then the cause of your problem may be outside the scope of these diagnostics. Read the instructions in "Technical support," and then contact Keithley for help in isolating the cause of your problem.
21. If you can write and read all I/O bits to and from the OK board, do the following:
    - If you have arrived at step 21 after program corrections have been made, and you can now write and read all I/O bits to and from the OK board, then the problem has apparently been resolved. Skip to step 27.
    - If you have arrived at step 21 without making data acquisition program corrections and can write and read all I/O bits to and from the OK board, then faulty external I/O connections may have caused your problem. Continue with step 22.
22. Check for external wiring faults as follows:
    a. Into the OK KPCI-PIO32IOA or KPCI-PDISO8A board, plug back one of the STP-36 accessories (or an in-house-wired 36-pin mating connector) that is wired to external circuits. You unplugged one or more of these in step 1.
    b. Check each external I/O connection, one at a time, for short circuits and open circuits and immediately correct any faults as you find them.
    c. If KPCI-PIO32IOA and KPCI-PDISO8A boards were installed in more than one PCI slot, repeat steps 22a and 22b until all external circuits have been checked and all faults have been corrected.
23. Based on the results of step 22, do the following:
    - If any external I/O connection faults were found and corrected, assume that your problems were caused by the faulty connections. Skip to step 27.
    - If all external I/O connections were found to be normal, then, by process of elimination, the KPCI-PIO32IOA or KPCI-PDISO8A board(s) originally installed in the computer is likely the cause of the problem. Continue with step 24.

24. Replace the faulty board(s). Do one of the following:
    - If only one KPCI-PIO32IOA or KPCI-PDISO8A board was installed when the problem occurred, proceed as follows:
        a. Leave the OK board in the expansion slot as a replacement. To repair or replace the faulty board, contact Keithley as described in "Technical support."
        b. Skip to step 27.
    - If more than one board was installed when the problem occurred, determine which one is faulty, starting with step 25.
25. Check whether an original board—a board that you removed in step 1—is detected as a PCI resource. Proceed as follows for the Windows 95/98/Me operating system (a different procedure is required for Windows NT/2000/XP):
    a. Turn OFF the computer.
    b. Remove the KPCI-PIO32IOA or KPCI-PDISO8A board.
    c. Reinstall the computer cover.
    d. Turn ON the computer.
    e. Clear KPCI-PIO32IOA and KPCI-PDISO8A boards from Windows 95/98 device manager, if any are listed. (Otherwise, when you reboot the computer, the computer may still say that empty PCI slots that once held the OK KPCI-PIO32IOA and KPCI-PDISO8A board are PCI resources.) Proceed as follows:
        1. Right-click the **My Computer** icon on your desktop.
        2. On the menu that appears, click **Properties**.
        3. On the System Properties dialog box that appears, click the **Device Manager** tab. The Device Manager appears.
        4. In the Device Manager, look for a **DriverLINX drivers** item.
        5. If you find a **DriverLINX drivers** item with a **+** sign to the left of the item, click the **+** sign. A second level list may appear containing one or more boards.
        6. Delete any KPCI-PIO32IOA and KPCI-PDISO8A boards, *only*, from the list by highlighting the item and clicking the **Remove** button.
    f. Perform step 1, only, of the procedure "Checking for board as a PCI resource," found later in this section.
    g. Turn OFF the computer.
    h. Install one of the original KPCI-PIO32IOA or KPCI-PDISO8A boards in a PCI slot known to be satisfactory.
    i. Reinstall the computer cover.
    j. Perform step 3, only, of the procedure "Checking for board as a PCI resource," found later in this section.
    k. Based on the results of step 25j, do one of the following:
        - If, in step 25j, the original board that you checked is not recognized as a PCI resource, you have located the faulty board. Do the following:
            1. Replace it with the OK board. (To repair the faulty board or obtain a new one, contact Keithley as described in "Technical support.")
            2. Skip to step 27.
        - If, in step 25j, the original board that you checked is recognized as a PCI resource, but you have not yet checked all original KPCI-PIO32IOA and KPCI-PDISO8A boards as PCI resources, repeat steps 25e through 25k for each additional original board.
        - If you have reached this point after checking all original KPCI-PIO32IOA and KPCI-PDISO8A boards as PCI resources, and each board has been recognized as a PCI resource, then continue with step 26. Probably one of the boards has faulty I/O.

26. Determine which original board has faulty I/O. If the faulty board survived the PCI resource test in step 25, it should fail the I/O test.
    a. If an original board remains installed following step 25, then skip to step 26e.
    b. Turn OFF the computer.
    c. Install one of the original boards in a slot known to be satisfactory.
    d. Turn ON the computer.
    e. Perform the test outlined in the "I/O test" subsection. Then return to this step (26e).
    f. Based on the results of step 26e, do one of the following:
       - If, in step 26e, you *cannot* write and read all I/O bits to and from the first original KPCI-PIO32IOA or KPCI-PDISO8A board that you test, you have located the faulty board; replace it with the OK board. (To repair the faulty board or obtain a new one, contact Keithley as described in "Technical support.") Then skip to step 27.
       - If, in step 26e, you *can* write and read all I/O bits to and from the first original board that you test, repeat steps 26b through 26e for additional original KPCI-PIO32IOA and KPCI-PDISO8A boards until you find a faulty board. Replace the faulty board with the OK board. (To repair the faulty board or obtain a new one, contact Keithley as described in "Technical support.") Then continue with step 27.
       - However, if you have reached this point after trying all original boards, and the I/O on each board has been found satisfactory, then the cause of your problem may be outside the scope of these diagnostics. Read the instructions in "Technical support," and contact Keithley for help in isolating the cause of your problem.
27. Assuming the problem has been resolved, do the following:
    a. Turn OFF the computer.
    b. Install the good KPCI-PIO32IOA and KPCI-PDISO8A boards in good slots.
    c. Reinstall the computer cover.
    d. Reconnect all external circuits as discussed in Section 3, "Installation."
    e. Turn ON the computer.
    f. Verify that the system now performs satisfactorily.
28. Based on the results of step 27f, select one of the following:
    - If the system now performs satisfactorily, you have successfully isolated and corrected the problem.
    - If the system still does not perform satisfactorily, then the cause of your problem may be outside the scope of these diagnostics. Read the instructions in "Technical support," and then contact Keithley for help in isolating the cause of your problem.

Specified tests

Checking for board as a PCI resource

WARNING **To avoid a possible shock hazard, always turn OFF your computer and any external circuits connected to the KPCI-PIO32IOA or KPCI-PDISO8A board before removing or replacing the board.**

CAUTION **The test outlined in this subsection involves handling of the circuit board. Handle the board at the mounting bracket, using a grounded wrist strap. Do not touch the circuit traces or connector contacts. If you do not have a grounded wrist strap, periodically discharge static electricity by placing one hand firmly on a grounded metal portion of the computer chassis.**

Always reinstall the cover before turning the computer ON.

NOTE *The following procedure depends on the Plug and Play capability of Windows. If your computer operating system is Windows NT, you must use a different procedure to detect the board as a PCI resource. Consult your Windows NT documentation.*

The following procedure provides details to help you determine whether Windows Plug and Play finds your KPCI-PIO32IOA or KPCI-PDISO8A board as a PCI resource.

1. Determine the PCI resources that Plug and Play detects before any KPCI-PIO32IOA and KPCI-PDISO8A boards are installed. Proceed as follows:
   a. Insert a blank diskette, or any diskette that you are sure is unbootable, into the A: drive.
   b. Turn ON the computer and allow it to start the boot cycle.

      The boot cycle stalls at a text screen listing system characteristics and resources and saying at the bottom: **Non-system disk or disk error. Replace and press any key when ready.**

NOTE *This system characteristics and resources screen is normally displayed only fleetingly during the boot cycle. Having an unbootable diskette in your computer automatically stops the boot cycle at this screen, allowing for convenient viewing. This is not harmful to your computer. The more common approach—using the* PAUSE *key to pause the boot cycle at this screen—requires fast reflexes with some systems.*

   c. Note the displayed list of PCI devices under a heading something like **PCI device listing…** If you have a printer, print the screen by pressing the PRINT SCREEN key.
   d. Remove the diskette and allow the boot cycle to finish.
2. Install a good board—a KPCI-PIO32IOA and KPCI-PDISO8A board that you know is fully functional—as follows:
   a. Shut down Windows, and turn OFF power to the host computer.
   b. Install the good board in a slot from which you removed a potentially faulty board. Refer to Section 3 for board installation instructions.
   c. Reinstall the cover of the computer.

NOTE *If you removed more than one board in step 1, install only one good board in one expansion slot.*

Do not connect any external circuits to the board at this point.

3. Again determine the PCI resources detected by your computer after the KPCI-PIO32IOA and KPCI-PDISO8A board is installed. Windows Plug and Play should find and configure the new board as a PCI resource if all of the following are true:
   - The board functions properly as a PCI device.
   - The contacts of the expansion slot in which the OK board is installed are in good condition.
   - The OK board is seated properly in the expansion slot.

Do the following, as you did in step 1:

a. Insert an unbootable diskette.
b. Turn ON the computer and allow the boot cycle to stall at the **Non-system disk or disk error…** message.
c. Again, note the displayed list of PCI devices. A new device should be listed, likely as an unidentified peripheral. If your resource listing includes PCI slot numbers, the slot number for the new device should match the number of the slot in which your board is installed.
d. Remove the diskette and allow the boot cycle to finish.

I/O test

WARNING **To avoid a possible shock hazard, always turn OFF your computer and any external circuits connected to the KPCI-PIO32IOA or KPCI-PDISO8A board before removing or replacing the board.**

CAUTION **The tests outlined in this subsection involve handling of the KPCI-PIO32IOA or KPCI-PDISO8A circuit board. Handle the board at the mounting bracket using a grounded wrist strap. Do not touch the circuit traces or connector contacts. If you do not have a grounded wrist strap, periodically discharge static electricity by placing one hand firmly on a grounded metal portion of the computer chassis.**

Always reinstall the cover before turning the computer ON.

Channel assignments

Table 5-2 summarizes DriverLINX logical channel assignments.

Table 5-2
DriverLINX board channel assignments

Model	DriverLINX Channel	I/O
KPCI-PDISO8A	0	Inputs (IP0-IP7)
	1	Outputs (OP0-OP7) with readback
KPCI-PIO32IOA	0	Inputs (IP0-IP7)
	1	Inputs (IP8-IP15)
	2	Outputs (OP0-OP7) with readback
	3	Outputs (OP8-OP15) with readback

Test procedure

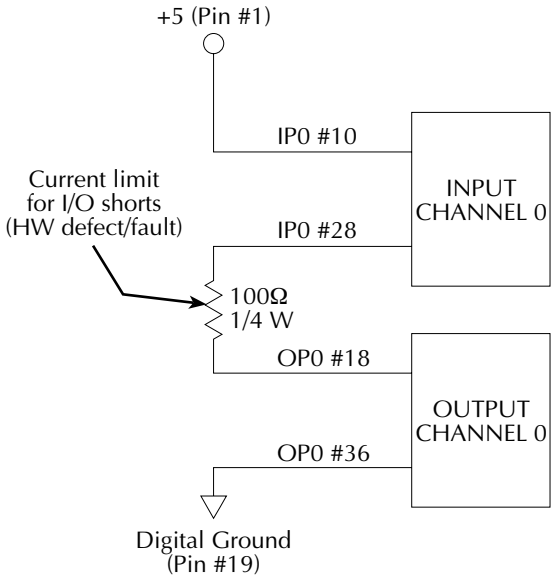
The I/O test checks input and output performance. Perform the test as follows:

1. Prepare a test connector using a female 36-pin connector or the STP-36 screw terminal accessory, as shown in Figure 5-2.

NOTE *Figure 5-2 shows wiring for only one channel. Each of the eight or 16 channels on the board should be wired in the same way to test all eight or 16 I/O channels simultaneously.*

2. Turn OFF the host computer.
3. Insert the test connector, which you prepared in step 1, into the J101 or J102 on the board, depending on which channels you wish to test.
4. Turn ON the host computer, and boot Windows.
5. Click the Windows **Start** tab, then run **DriverLINX**.
6. Use the **Special...** button in the DriverLINX Device configuration page to run the test.
7. Verify that the board passes the test as reported by DriverLINX.

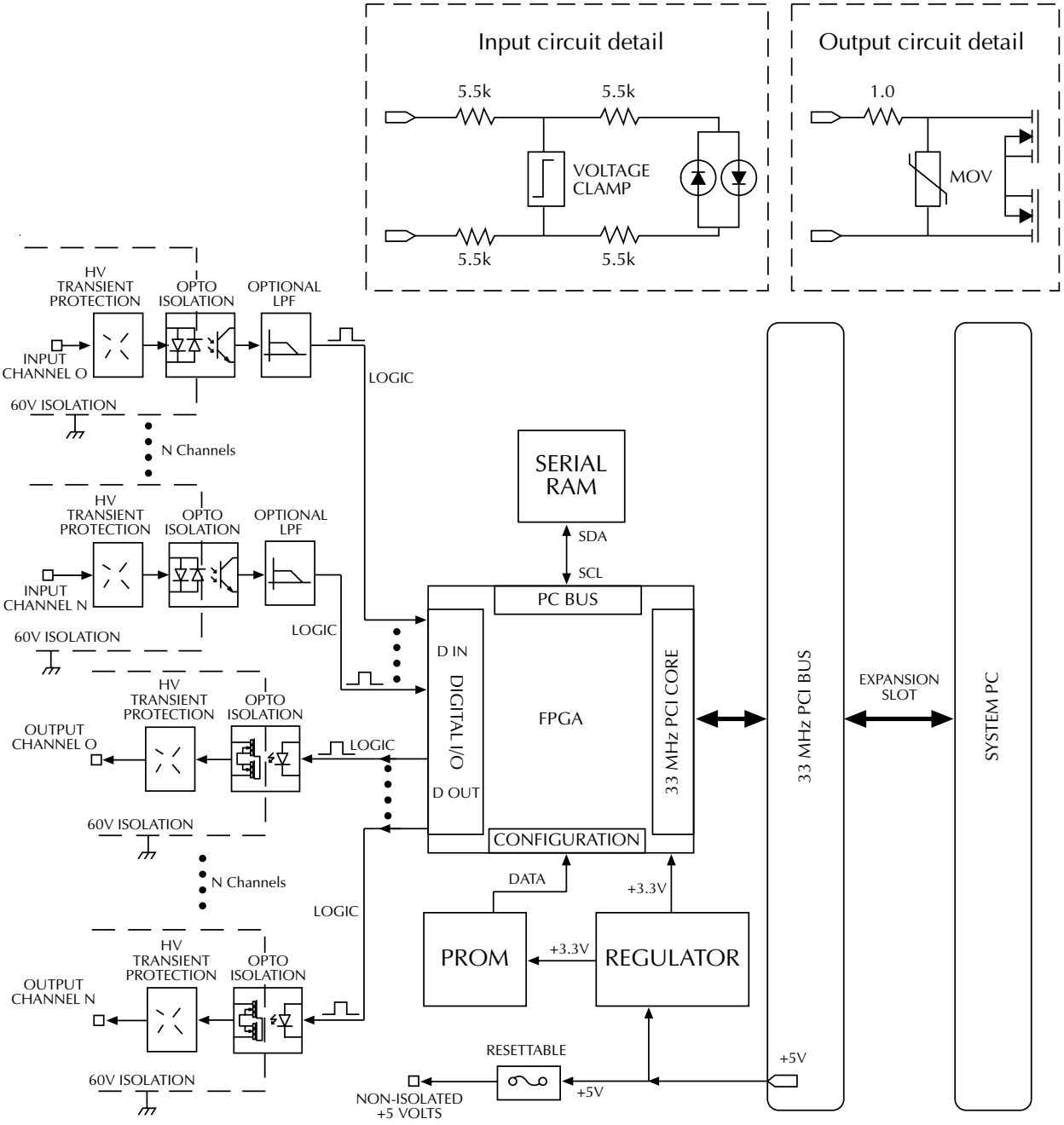
Figure 5-2
Mating connector wiring for I/O test



Block diagram

Figure 5-3 shows a block diagram of the KPCI board. Note that both input and output channels have transient protection and are optically isolated, while an FPGA IC performs all logic and interfacing functions for the board. Card configuration data is stored in a PROM, while a serial RAM performs temporary storage.

Figure 5-3
KPCI board block diagram



KPCI-PIO32IOA N = 15 In, 15 Out
KPCI-PDISO8A N = 7 In, 7 Out

Technical support

Before returning any equipment for repair, call Keithley for technical support at:

1-888-KEITHLEY
Monday - Friday, 8:00 a.m. - 5:00 p.m., Eastern Time

An applications engineer will help you diagnose and resolve your problem over the telephone. Please make sure that you have the following information available before you call:

KPCI-PIO32IOA and KPCI-PDISO8A board configuration	Model	_____
	Serial #	_____
	Revision code	_____
	Interrupt level setting	_____
	Number of channels	_____
Computer	Manufacturer	_____
	CPU type	_____
	Clock speed (MHz)	_____
	MB of RAM	_____
	Video system	_____
	BIOS type	_____
Operating system	Windows version	_____
Software package	Name	_____
	Serial #	_____
	Version	_____
	Invoice/Order #	_____
Compiler (if applicable)	Language	_____
	Manufacturer	_____
	Version	_____
Accessories	Type	_____
	Type	_____
	Type	_____
	Type	_____
	Type	_____
	Type	_____
	Type	_____
	Type	_____

If a telephone resolution is not possible, the applications engineer will issue you a Return Material Authorization (RMA) number and ask you to return the equipment. Include the RMA number with any documentation regarding the equipment.

When returning equipment for repair, include the following information:

- Your name, address, and telephone number.
- The invoice or order number and date of equipment purchase.
- A description of the problem or its symptoms.
- The RMA number on the **outside** of the package.

Repackage the equipment, using the original anti-static wrapping if possible, and handle it with ground protection. Ship the equipment to:

ATTN: RMA #_____
Repair Department
Keithley Instruments, Inc.
28775 Aurora Road
Cleveland, Ohio 44139

Telephone 1-888-KEITHLEY
FAX (440) 248-6168

NOTE *If you are submitting your equipment for repair under warranty, you must include the invoice number and date of purchase.*

To enable Keithley to respond as quickly as possible, you must include the RMA number on the outside of the package.

A
Specifications

KPCI-PDISO8A and KPCI-PIO32IOA Specifications

These two products are to achieve a PCI version of the PDISO-8 and PIO32-IO ISA cards.

Control Inputs[1]

Quantity: 8 KPCI-PDISO8A
16 KPCI-PIO32IOA

Type: Solid-State, Opto-isolated

DC Inputs without optional filter

PARAMETER	MIN.	TYP.	MAX.	UNIT
Voltage Input Logic High	+/- 2.5		+/- 60	V_{DC}
Voltage Input Logic Low			+/- 1.0	V_{DC}
Response Time		0.6	1.0	ms
Isolation[2]	60			V pk

AC Inputs with optional filtering[3]

PARAMETER	MIN.	TYP.	MAX.	UNIT
Low Pass Filter Time Constant		175		ms
AC Sense Voltage Logic High	10		42	V pp
AC Sense Voltage Logic Low			< 10	V pp
AC Line Frequency Sensing		50 and 60		Hz

Interrupt Line Assignment

PARAMETER	Signal	Location	Pin #
External Interrupt Enable	INT_ENN	IP7	3,21
External Interrupt Request	INT_REQ	IP6	4,22

Outputs

Quantity: 8 KPCI-PDISO8A
16 KPCI-PIO32IOA

Type: Solid-State, Opto-isolated N.O. Relay (Form A)

AC/DC Load Switching

PARAMETER		MIN.	TYP.	MAX.	UNIT
Load Voltage					
	DC			+/-60	V_{DC}
	AC			30	V_{RMS}
Load Current[4]					
	DC			350	mA $_{DC}$
	AC			247	mA $_{RMS}$
Off State Leakage Current[5]			316	2000	nA
On State Channel Resistance [7]			3.5	4.0	Ω
Relay Turn ON Time			1.0	3.5	ms

PARAMETER	MIN.	TYP.	MAX.	UNIT
Relay Turn OFF Time		80	200	µs
Power Dissipation				
Per Channel			430	mW
Isolation [2]	60			V pk

NON-Isolated external power[6]

PARAMETER	Maxiumum Capacity	Location	Pin #
+5 V_{DC} PC Bus Sourced	1.0 Amperes	+5V	1 Top & Bot Connector
Digital Ground Return	1.0 Amperes	GND	19 Top & Bot Connector

GENERAL SPECIFICATIONS

POWER REQUIREMENTS: +5V: 500 mA Maximum

EMC: Conforms to European Union Directive 89/336/EEC, EN61326-1

SAFETY: Conforms to European Union Directive 73/23/EEC, EN61010-1

ENVIRONMENTAL:

Operating Temperature: 0 to 50°C

Storage Temperature: -20 to 70°C

Humidity (non condensing): 0 to 80% at 35°C

DIMENSION: 18.1cm x 10.8cm x 1.9cm (7.1in x 4.25in x 0.75in)

WEIGHT: KPCI-PIO32IOA – 0.13kg (0.30lbs) KPCI-PDISO8A – 0.12kg (0.26lbs)

ACCESSORIES:

Cables: CAB-1284CC-0.5, CAB-1284CC-2

STP-36: 36-pin Screw Terminal Panel

Notes:

1. Inputs are not guaranteed to be compatible with all TTL/CMOS logic families.
2. Minimum guaranteed isolation, which includes channel to channel, channel to PC and channel to chassis ground.
3. Selectable, jumper switched LPF. Recommended for AC or slow DC inputs.
4. De-rate 3.6 mA/°C if ambient temperature is over 25°C.
5. Output channel in off-state, T_A = 50°C, 60 V_{DC} applied with 30% RH (non-condensing) for maximum and typical values.
6. For convenience only. These are not isolated from the rest of the system. Do not use directly in external circuit if isolation is required.
7. Typical measurement was achieved with an STP-36 and 2 meter cable, CAB-1284CC-2 at 25°C.

Board Address Mapping

	Offset (byte)	Content	Description
BADDRx	0	Port A	Read Only – Input Register
	1	Port B	Read Only (KPCI-PIO32IOA Only)
	2	NA	
	3	Control	Latching control (R/W)
	4	Port A'	Readback / Write – Output Register
	5	Port B'	Readback / Write (KPCI-PIO32IOA Only)
	6	NA	
	7	NA	
	…		
	34 - 37	FW Rev	ASCII format: "A0xx" (Read Only)
	38 - 3A	INT CSR	Interrupt Control Register (R/W)
	3B	NA	

BADDRx + 0x03
Control Byte Format:
-- Bit 7 - This bit to be ignored.
-- Bit 6 - Access Mode Select – For B5 and B6 see table below.
-- Bit 5 - Access Mode Select – For B5 and B6 see table below.
-- Bit 4:0 - These bits to be ignored.

B6 B5 (Control byte)
0 0 – No input latching on rising interrupt edge.
0 1 – Latch group inputs on rising interrupt edge.
1 0 – No input latching on falling interrupt edge.
1 1 – Latch group inputs on falling interrupt edge.

BADDRx + 0x38
Interrupt Status Register Format: (DWord Access)
(Taken from the AMCC Architecture – used on KPCI-PIOs and KPCI-3160)
--**Bit 23** – Interrupt missed =1 for missed interrupt. Write 1 to acknowledge/clear.
--Bit 22:18 – These bits to be ignored.
--**Bit 17** – Interrupt pending = 1 for pending interrupt. Write 1 to acknowledge/clear.
--Bit 16 – This bit to be ignored.

--Bit 15:13 – These bits to be ignored.
--**Bit 12** – Interrupt enable = 1, HW Interrupts enabled, 0 = HW Interrupts disabled.
--Bit 11:08 – These bits to be ignored.

--Bit 7 – This bit to be ignored.
--**Bit 6** – Interrupt polarity select = 1 for falling edge, Write 0 for rising edge.
--Bit 5:0 – These bits to be ignored.

(x = 0 or 1) IO mapped use BADDR0. Memory mapped use BADDR1.

KPCI-PIO32IOA and KPCI-PDISO8A Pin Assignment

Top Connector

Pin #	Signal	Pin #	Signal
1	+5V	19	GND
2	-	20	-
3	IP7	21	IP7
4	IP6	22	IP6
5	IP5	23	IP5
6	IP4	24	IP4
7	IP3	25	IP3
8	IP2	26	IP2
9	IP1	27	IP1
10	IP0	28	IP0
11	OP7	29	OP7
12	OP6	30	OP6
13	OP5	31	OP5
14	OP4	32	OP4
15	OP3	33	OP3
16	OP2	34	OP2
17	OP1	35	OP1
18	OP0	36	OP0

Bottom Connector (For KPCI-PIO32IOA only)

Pin #	Signal	Pin #	Signal
1	+5V	19	GND
2	-	20	-
3	IP15	21	IP15
4	IP14	22	IP14
5	IP13	23	IP13
6	IP12	24	IP12
7	IP11	25	IP11
8	IP10	26	IP10
9	IP9	27	IP9
10	IP8	28	IP8
11	OP15	29	OP15
12	OP14	30	OP14
13	OP13	31	OP13
14	OP12	32	OP12
15	OP11	33	OP11
16	OP10	34	OP10
17	OP9	35	OP9
18	OP8	36	OP8

B
I/O Address Mapping

NOTE *A typical user of the KPCI-PIO32IOA or KPCI-PDISO8A board does not need to read this appendix (except perhaps for the supplementary interrupt illustration under "Interrupt example scenario"). Register level programming of the KPCI-PIO32IOA or KPCI-PDISO8A board is neither practical nor necessary for most users. Register level interfacing with the PCI bus is more complex than with the ISA bus. PCI board addresses are mapped automatically in I/O space or memory, whereas ISA board addresses are assigned manually by the user in I/O space.*

The DriverLINX driver shipped with your board provides a user-friendly Application Programming Interface (API) that supports Visual C++, Visual Basic, and Delphi programming languages under Windows 98/Me and Windows NT 4.0, 2000, and XP. You are encouraged to use the capabilities of DriverLINX and ignore the rest of the information in this chapter (except perhaps the subsection "Interrupt example scenario," which may help you to understand how external interrupts work).

However, there are circumstances in which advanced users may desire or need to bypass DriverLINX entirely and write their own drivers. Alternatively, advanced users may wish to program the KPCI-PIO32IOA or KPCI-PDISO8A at the register level using an ActiveX hosting language. Finally, some users may wish to reuse an existing application program that makes port I/O calls to an ISA-bus digital I/O board.

Appendix B discusses the following:

- General PCI address assignments
- Control and data register address assignments inside the I/O space, as follows:
  - Assignments for a control and data register map
  - Assignments for bits of a special interrupt control/status register, including an example scenario showing how the bits are used
- Some general requirements for manipulating control and data registers
- Reuse of an existing port I/O application program with the KPCI-PIO32IOA or KPCI-PDISO8A board

General PCI address assignments

The PCI specification allows each PCI-bus board to be assigned up to five distinct address regions for general use at the discretion of the hardware designer. The first region, at base address BADDR0, is mandatory per the PCI specification, as published by the PCI Special Interest Group (PCISIG). The other four address regions, located at base addresses BADDR1, BADDR2, BADDR3, and BADDR4, are optional. The PCI BIOS or the Plug and Play operating system automatically allocates BADDR0 through BADDR4 at power-up, based on the PCI boards that it finds installed at that time. After power-up, computer software can read PCI configuration space to determine the location of BADDR0 through BADDR4. (The term "computer software" hereafter in Appendix B refers to the combination of the application programming interface (API)/driver—normally, DriverLINX—and the application program. For information about application programming through DriverLINX, refer to your DriverLINX documentation.)

The KPCI-PIO32IOA and KPCI-PDISO8A use both BADDR0 and BADDR1 mapped in I/O and memory space, respectively.

Register assignments

The KPCI-PIO32IOA and KPCI-PDISO8A registers are mapped shown in Table B-1.

Table B-1
Register mapping

Offset * (Byte)	Content	Description
0	Port A	Read Only - Input Register (IP0-IP7)
1	Port B	Read Only - Input Register (IP8-IP15, KPCI-PIO32IOA only)
2	N/A	
3	Control	Latching control (R/W)
4	Port A	Output Register (R/W) for OP0-OP7
5	Port B	Output Register (R/W) for OP8-OP15 (KPCI-PIO32IOA only)
6	N/A	
7	N/A	
-		
34-37	FW Rev	ASCII firmware revision, format "AOxx" (Read Only)
38-3A	INT CSR	Interrupt control register (R/W)
3B	N/A	

*BADDRx + offset, where I/O mapped is BADDR0, and memory mapped is BADDR1.

All offsets from the base address are specified as multiples of four bytes (modulo 4 addressing), because each offset specifies a four-byte (32-bit) wide register.

NOTE *The term "base" address, as used in the following subsections, does not have the same meaning for a PCI board, such as the KPCI-PIO32IOA and KPCI-PDISO8A, as for an ISA board. The base address for your KPCI-PIO32IOA or KPCI-PDISO8A board is an address BADDR0 that is mapped in I/O space or BADDR1 that is memory mapped and is assigned at power-up by the PCI BIOS or the Plug and Play operating system. It remains constant only as long as the computer is powered. It is not a fixed, user-assigned I/O address such as 0x300 or 0x310 set on a DIP switch, nor is it a fixed address such as is assigned for a printer.*

Any BADDR that is set to 0x00 has been disabled by system BIOS.

Control and data registers

The control and data register map for the input ports follows. The control port consists of bits that allow some additional features to implement interrupt control and latching. Both boards separate the I/O lines into two 32-bit registers. Up to 16-bits of input and output are available on the KPCI-PIO32IOA, while the KPCI-PDISO8A has 8-bit capabilities.

Register contents are as follows:

Bits of each control and data register	31.....24	23.....16	15.....8	7.....0
Contents	Control port (latching only)	Not used	Port B	Port A

The contents of data and control registers may be transferred either 8 bits or 32 bits at a time.

Control register bit functions

The control register bit functions for each port are identical. These functions are listed in Table B-2.

Table B-2
Control register bit functions for each port

Bit Number	Function	Variable Name	I/O Status for this Bit
Bit 6	Access Mode Select	N/A	Bit 6 Bit 5 0 0 = No input latching. 0 1 = Latch group on rising edge of INT_REQ signal. 1 0 = No input latching. 1 1 = Latch group on falling edge of INT_REQ signal.
Bit 5	Access Mode Select	N/A	

Interrupt control/status register

The firmware of the KPCI-PIO32IOA and KPCI-PDISO8A boards implement a 32-bit interrupt control/status register. This register is located at BADDR0 + 0x38 offset, where 0x designates hexadecimal. A control bit of the interrupt control/status register is used to configure the board for interrupt-based data transfer and processing. Two status bits are used to determine whether one or more interrupts are pending or were missed during data processing. Two control bits provide for the selection of interrupt polarity (falling or rising edge) and the enabling/disabling of the interrupt signal.

Interrupt control/status bit descriptions

The four interrupt control/status register bits are described in detail in Table B-3.

Table B-3
Bit functions for interrupt control/status register

Interrupt Control/Status Register Bit			Status for this Bit[1]	
Bit Number	**Bit Function**	**Where the Bit is Set and Cleared**	**When the Bit Value = 0**	**When the Bit Value = 1**
Bit 06	Selects polarity for the external interrupt signal.	Set and cleared by computer software.	Sets rising edge for external interrupt signal.	Sets falling edge for external interrupt signal.
Bit 12	Configures the board for external interrupt service.	Set and cleared by computer software.	Interrupts disabled. Data transfer and processing via polling or upon software command, only. All inputs are general purpose inputs.	Interrupts enabled. Data transfer and processing in response to an external signal, only. The highest two inputs of port A are INT_REQ and $\overline{\text{INT\_ENN}}$ instead of IP6 and IP7.
Bit 17	Interrupt-pending. Indicates whether or not an external interrupt signal has been received at the board INT_REQ input.	Automatically set high when board firmware detects an interrupt. Must acknowledge (write 1) to clear.	Register status awaits detection of interrupt signal by firmware. Computer CPU is presently doing other tasks (not processing KPCI-PIO32IOA or KPCI-PDISO8A data).	Interrupt signal has been received. Computer CPU is processing, or is about to process, KPCI-PIO32IOA or KPCI-PDISO8A data via an ISR.[3]
Bit 23	Interrupt-missed. Indicates whether or not at least one KPCI-PIO32IOA or KPCI-PDISO8A external interrupt signal has been sent and missed while interrupt-pending bit 17 is high.	Automatically set high when board firmware detects missed interrupt.[2] Must acknowledge (write 1) to clear.	Interrupts have not been missed OR register status awaits missed-interrupt detection by firmware.[1]	One or more interrupts have been missed.[3]

[1] All bits listed in this table are cleared to logic-low on power-up.
[2] Although this bit has both software read and software write capability, computer software writes should only be used to *clear* the bit.
[3] This status is correct only if bit 17 is deliberately cleared by computer software at the conclusion of each board ISR (each ISR that is used to process board data).

Interrupt example scenario

The following example is one possible scenario that may help you to understand and use the KPCI-PIO32IOA and KPCI-PDISO8A interrupt feature. (This example is simplified, and some details may not apply to your specific system or to your requirements.) It illustrates the workings of the interrupts and bits 6, 12, 17, and 23 of the interrupt control/status register. (Refer also to Section 4, "External Interrupts.")

1. At some point, computer software sets interrupt-enable bit 12 of the interrupt control/status register to logic-high. (The term "software" here refers to the combination of the application programming interface (API)/driver—normally, DriverLINX—and the application program. To understand how to program interrupt-triggered data acquisition through DriverLINX, refer to your DriverLINX documentation.) This status, detected by board firmware, enables the board to process data using external interrupts. It changes general-purpose inputs PC6 and PC7 of port group 3 into external interrupt request and external interrupt enable inputs INT_REQ and $\overline{\text{INT\_ENN}}$.
2. At some point, computer software sets latching polarity bit 6 for port group 0 to determine whether that data at port group 0 latches on the rising or falling edge of INT_REQ.

3. Prior to sending data, a user circuit sets $\overline{\text{INT\_ENN}}$ to logic low. Board firmware detects that $\overline{\text{INT\_ENN}}$ is low and allows the edges of interrupt signals at INT_REQ to be detected.
4. When data is ready to be transferred and processed, the user circuit sends an external interrupt request signal to INT_REQ.
5. Board firmware detects the rising or falling edge of the INT_REQ signal, depending on interrupt polarity settings in the port-group control registers (Table B-2).
6. Software sets bit 6 of the interrupt control/status register high or low depending on whether the user wants a PCI interrupt to be triggered by the falling or rising edge of INT_REQ.
7. If inputs are configured to latch, the current data value will be latched. (See bits 5 and 6 in Table B-2.)
8. The interrupt-received status in board firmware causes interrupt-pending bit 17 of the interrupt control/status register to be set to logic high.
9. The interrupt-received status in board firmware causes a computer CPU interrupt to start, stopping execution of the current CPU task.
10. Computer hardware detects an interrupt request signal and transfers control to an interrupt service routine (ISR).
11. Computer software starts the ISR, which takes control of the CPU and starts processing the KPCI-PIO32IOA or KPCI-PDISO8A input data.
12. The ISR proceeds.
13. Ideally, for a well-planned data acquisition session, the conditions are met while the ISR is in progress, and the interrupt-pending bit is set. No new external interrupt requests occur during this time.

    However if the above conditions are not met while the interrupt-pending bit is set, the following occurs:

    a. The rising or falling edges of interrupt signals have no effect; these interrupts are missed.
    b. When the *first* rising or falling edge is missed, the problem is recorded as follows:
        1. Board firmware detects the missed interrupt, causing interrupt-missed bit 23 of the interrupt control/status register to be set to logic high.
        2. Computer software, if appropriately programmed, detects that bit 23 has been set and notifies the user of the missed interrupt.
    c. If *additional* rising or falling edges are missed (edges 2, 3, ..., n), the problems are *not* recorded, as follows:
        1. No additional interrupt-missed bits are set (there is only one interrupt-missed bit).
        2. Computer software cannot further notify the user.
14. The ISR, if appropriately programmed, writes ones (Acknowledge) to clear the interrupt-pending bit 17 and interrupt-missed bit 23 in the interrupt control/status register to logic low, which clears interrupt-pending status and interrupt-missing status in board firmware.
15. The ISR, if appropriately programmed, clears the rising/falling edge of bit 6 in the interrupt control/status register to logic low.
16. The ISR finishes.
17. The ISR dispatcher of the operating system detects that the ISR has finished and sends an end-of-interrupt instruction to the CPU.

NOTE *The end-of-ISR behavior depends on the operating system being used.*

18. The CPU returns to the task that it was executing at the time of the interrupt.
19. At some point, the user circuit may disable interrupts and latching by setting $\overline{\text{INT\_ENN}}$ to logic high.
20. Board firmware detects that $\overline{\text{INT\_ENN}}$ is high and rejects interrupt signals at INT_REQ.

Manipulating control and data registers

Be aware that software programming for a PCI board, such as the KPCI-PIO32IOA and KPCI-PDISO8A, is more involved than for an ISA board. As mentioned in the Appendix B introduction, DriverLINX eliminates the need for user interaction with control and data registers. However, control and data registers can be manipulated in the following special situations:

- You are an advanced user needing to use the KPCI-PIO32IOA or KPCI-PDISO8A with an operating system other than Microsoft Windows 98/Me or Windows NT 4.0, 2000, or XP or greater. In this situation, you must write a new driver. This task requires an in-depth knowledge of PCI-bus interfacing and your development operating system.
- You want to program the KPCI-PIO32IOA or KPCI-PDISO8A at the register level using an ActiveX hosting language. In this situation, you may use the "Direct I/O ActiveX Automation Object" that comes with DriverLINX. The Direct I/O ActiveX Automation Object allows you to set the control and data registers directly and bypass the DriverLINX API, yet avoids the full complexities of PCI bus interfacing. Refer to your DriverLINX manual for more information.
- You want to reuse, with the KPCI-PIO32IOA or KPCI-PDISO8A, an existing application program that makes port I/O calls to an ISA-bus digital I/O board such as the PIO-96 or PIO-24.

General approach to manipulating control and data registers

This subsection outlines some general program tasks needed to use the data and control registers of the KPCI-PIO32IOA and KPCI-PDISO8A board.

- If the board is to transfer and process data upon receipt of external interrupts, the following are set:
  - Bit 12 of the interrupt control/status register, located at BADDR0 + 0x38, is set to 1 to configure the board for interrupt service.
  - The interrupt-pending bit and the interrupt missed bit in BADDR0 are initialized to the cleared condition by writing 1s to interrupt control/status register status bits 17 and 23 at BADDR0 + 38. The polarity select bit is initialized to the clear condition by writing a 0 to the interrupt control/status register control bit 6.
- If you retrieve data using interrupts, the interrupt-pending bit and interrupt-missing bit must be cleared by software at the conclusion of each interrupt-service (ISR) by writing ones (Acknowledge) to interrupt control/status register bits 17 and 23 at base address 0 + 0x38. The interrupt enable bit and interrupt polarity select bit of the interrupt control/status register are cleared by writing zeros to bits 12 and 6.
- Input control registers are read only.
- Output registers have read/write capabilities. Writing to an output register changes the states of the solid-state relays. Reading the output registers returns the states of the solid-state relays.
- All input and output registers are static, and no configuration ability exists.

C
Glossary

Address[1]
A number specifying a location in memory where data is stored.

API
See application programming interface.

Application programming interface[1]
A set of routines used by an application program to direct the performance of a procedure by the computer's operating system.

Bus mastering
On a microcomputer bus such as the PCI bus, the ability of an expansion board to take control of the bus and transfer data to memory at high speed, independently of the CPU. Replaces direct memory access (DMA).

Bus
An interconnection system that allows each part of a computer to communicate with the other parts.

Byte
A group of eight bits.

Contact bounce
The intermittent and undesired opening of relay contacts during closure, or closing of relay contacts during opening.

Crosstalk
The coupling of a signal from one input to another (or from one channel to another or to the output) by conduction or radiation. Crosstalk is expressed in decibels at a specified load and up to a specific frequency.

Darlington
A high gain current amplifier composed of two bipolar transistors, typically integrated in a single package.

DLL
See Dynamic Link Library.

Direct memory access
See DMA mode.

DMA mode
Direct memory access mode. Mode in which data transfers directly between an I/O device and computer memory. In the most general sense, PCI bus mastering is a DMA mode. More commonly, however, DMA mode refers to data transfers across the ISA bus, using a special circuitry on the computer motherboard. *See also* bus mastering.

Driver
Software that controls a specific hardware device, such as a data acquisition board.

Dynamic Link Library (DLL)
A software module in Microsoft Windows containing executable code and data that can be called or used by Windows applications or other DLLs. DLL functions and data are loaded and linked at run time when they are referenced by a Windows application or other DLLs.

Expansion slot[1]
A socket in a computer designed to hold expansion boards and connect them to the system bus (data pathway).

Foreground task
An operation, such as a task that occurs in the single or synchronous mode, that cannot take place while another program or routine is running.

FIFO
First-in/first-out memory buffer. The first data into the buffer is the first data out of the buffer.

GPIB
Abbreviation for General Purpose Interface Bus. It is a standard for parallel interfaces.

IEEE-488
See GPIB.

Input/Output (I/O)
The process of transferring data to and from a computer-controlled system using its communication channels, operator interface devices, data acquisition devices, or control interfaces.

Input/output port[1]
A channel through which data is transferred between an input or output device and the processor.

ISA Bus
Industry Standard Architecture. The 16-bit wide bus architecture used in most MS-DOS and Windows computers. Sometimes called the AT bus.

Map[1]
Any representation of the structure of an object. For example, a memory map describes the layout of objects in an area of memory, and a symbol map lists the association between symbol names and memory addresses in a program.

OCX
Abbreviation for OLE Custom Control.

Pass-through operation
See target mode.

PCI
Abbreviation for Peripheral Component Interconnect. It is a standard for a local bus.

Port
See input/output port.

Port I/O call
A software program statement that assigns bit values to an I/O port or retrieves bit values from an I/O port. Examples include a C/C++ statement containing an `inp` or `outp` function or a Basic statement containing a `peek` or `poke` function.

Register[1]
A set of bits of high speed memory within a microprocessor or other electronic device, used to hold data for a particular purpose.

Shielding
A metal enclosure for the circuit being measured or a metal sleeve surrounding wire conductors (coax or triax cable) to lessen interference, interaction, or current leakage. The shield is usually grounded.

Target mode
A PCI bus mode in which data from a data acquisition board is transferred indirectly to the computer memory in the foreground, via the host computer CPU, instead of directly, via Bus mastering. Sometimes referred to as pass-through operation. *See also* bus mastering and foreground task.

Trap[1] **(*verb*)**
To intercept an action or event before it occurs, usually in order to do something else.

TTL
Abbreviation for transistor-transistor-logic. A popular logic circuit family that uses multiple-emitter transistors. A low signal state is defined as a signal 0.8V and below. A high signal state is defined as a signal +2.0V and above.

[1]*Microsoft Press® Computer Dictionary, Third Edition*. Refer to "Sources" below.

Sources:

Keithley Instruments, Inc., *Catalog and Reference Guide* (full line catalog), glossary, 1998

Index

D

E

F

G

H

I

L

M

O

P

R

S

T

U

V

W

KEITHLEY

Keithley Instruments, Inc. 28775 Aurora Road • Cleveland, Ohio 44139 • 440-248-0400 • Fax: 440-248-6168
1-888-KEITHLEY (534-8453) • www.keithley.com

Sales Offices: BELGIUM: Bergensesteenweg 709 • B-1600 Sint-Pieters-Leeuw • 02-363 00 40 • Fax: 02/363 00 64
CHINA: Yuan Chen Xin Building, Room 705 • 12 Yumin Road, Dewai, Madian • Beijing 100029 • 8610-6202-2886 • Fax: 8610-6202-2892
FINLAND: Tietäjäntie 2 • 02130 Espoo • Phone: 09-54 75 08 10 • Fax: 09-25 10 51 00
FRANCE: 3, allée des Garays • 91127 Palaiseau Cédex • 01-64 53 20 20 • Fax: 01-60 11 77 26
GERMANY: Landsberger Strasse 65 • 82110 Germering • 089/84 93 07-40 • Fax: 089/84 93 07-34
GREAT BRITAIN: Unit 2 Commerce Park, Brunel Road • Theale • Berkshire RG7 4AB • 0118 929 7500 • Fax: 0118 929 7519
INDIA: Flat 2B, Willocrissa • 14, Rest House Crescent • Bangalore 560 001 • 91-80-509-1320/21 • Fax: 91-80-509-1322
ITALY: Viale San Gimignano, 38 • 20146 Milano • 02-48 39 16 01 • Fax: 02-48 30 22 74
JAPAN: New Pier Takeshiba North Tower 13F • 11-1, Kaigan 1-chome • Minato-ku, Tokyo 105-0022 • 81-3-5733-7555 • Fax: 81-3-5733-7556
KOREA: 2FL., URI Building • 2-14 Yangjae-Dong • Seocho-Gu, Seoul 137-888 • 82-2-574-7778 • Fax: 82-2-574-7838
NETHERLANDS: Postbus 559 • 4200 AN Gorinchem • 0183-635333 • Fax: 0183-630821
SWEDEN: c/o Regus Business Centre • Frosundaviks Allé 15, 4tr • 169 70 Solna • 08-509 04 679 • Fax: 08-655 26 10
SWITZERLAND: Kriesbachstrasse 4 • 8600 Dübendorf • 01-821 94 44 • Fax: 01-820 30 81
TAIWAN: 1FL., 85 Po Ai Street • Hsinchu, Taiwan, R.O.C. • 886-3-572-9077• Fax: 886-3-572-9031

4/02