

Software Options for Programming the KPCI-PDISO8A or KPCI-PIO32IOA

Unlike most PCI cards, the KPCI-PDISO8A and KPCI-PIO32IOA are I/O mapped PCI cards. Use of simple inp or outp statements from 16-bit DOS compilers is still an option. Additionally, a DriverLINX driver is also provided for operation in Windows.

Options when using DriverLINX:

DriverLINX provides two different API for programming these boards (see the section titled "Programming the KPCI-ISO Series" in the kpciiso.pdf software guide in the drive:\drvlinx4\docs\notes folder for full information on these API).

First recommendation is to use the Direct I/O COM object. This object allows use of simple read and write commands to interact with the digital I/O boards. The Direct I/O object is compatible with any ActiveX hosting language (VB, MSVC++, Delphi, Excel, CVI, etc.). The methods of this control provide 8-bit port I/O access to the hardware. The kpciiso.pdf document gives a good step by step process of how to use this object from the Visual Basic environment. Additional examples are available in the download center of this web site and Chapter 7 of the "DriverLINX Tutorial Manual" deals with this API.

Second recommendation is to use the full DriverLINX API. The advantage of the Service Request API is that it provides a hardware independent means of programming common functions of plug-in data acquisition boards. For example, the same DriverLINX code that controls a port of a KPCI-PDISO8A board can be reassigned to the corresponding port of the ISA version of this same board. The calls are the same because the underlying driver files handle the hardware differences. Thus the required code does not change, and time invested in application development is easily ported to new hardware when using the DriverLINX API. Example programs for use of the full DriverLINX API for digital I/O are available in the download center.

Register Level Control (not an API of DriverLINX): If you choose to make use of inp and outp for direct register level control of the card (instead of Direct I/O COM API of DriverLINX), finding the base address of the card will be an issue.

Unlike the ISA cards which have a hardware switch to control the I/O addresses the card will use, the address regions used by the KPCI card will be controlled by your motherboard's PnP BIOS. Once the address has been assigned, inp and outp statements can be used to program the card's registers. The hardware manual (Appendix B) has the register map of the card.

A PCI Bus Interrogation tool (available on www.keithley.com) could be used to determine the addresses assigned to the card. Another way to find the base address would be to install the DriverLINX driver for the card, and then look in Device Manager to see the address.

Also, for register level control in WinNT, Windows 2000 or Windows XP, be sure to use a Port I/O utility that has the kernel mode privileges required by these versions of Windows (the inherent port commands of QuickBasic, C/C++ or Delphi will work only in DOS or Win95 or Win98). A suitable Port I/O utility is available on www.keithley.com.