ACCES I/O PRODUCTS INC
10623 Roselle St. San Diego, CA. 92121-1506
Tel. (858)550-9559 FAX (858)550-7322

DIGITAL INPUT/OUTPUT CARD

Models PCI-DIO-48J and PCI-DIO-48JS

USER MANUAL

File: MPCI-DIO-48J.Ab

NOTICES

The information in this document is provided for reference only. ACCES does not assume any liability arising out of the application or use of the information or products described herein. This document may contain or reference information and products protected by copyrights or patents and does not convey any license under the patent rights of ACCES, nor the rights of others.

IBM PC, PC/XT, and PC/AT are registered trademarks of the International Business Machines Corporation.

TABLE OF CONTENTS

INSTALLATION

The software provided with this card is contained on either one CD or multiple diskettes and must be installed onto your hard disk prior to use. To do this, perform the following steps as appropriate for your software format and operating system. Substitute the appropriate drive letter for your CD-ROM or disk drive where you see D: or A: respectively in the examples below.

CD INSTALLATION

DOS/WIN3.x

1. Place the CD into your CD-ROM drive.
2. Type D : Enter to change the active drive to the CD-ROM drive.
3. Type I N S T A L L Enter to run the install program.
4. Follow the on-screen prompts to install the software for this card.

WIN95/98/NT

1. Place the CD into your CD-ROM drive.
2. The CD should automatically run the install program after 30 seconds. If the install program does not run, click START | RUN and type D : I N S T A L L, click OK or press Enter.
3. Follow the on-screen prompts to install the software for this card.
4. Click the “Go to ACCES Web” button to check for software updates.

3.5-INCH DISKETTE INSTALLATION

As with any software package, you should make backup copies for everyday use and store your original master diskettes in a safe location. The easiest way to make a backup copy is to use the DOS DISKCOPY utility.

In a single-drive system, the command is:

D I S K C O P Y A : A : Enter

You will need to swap disks as requested by the system.
In a two-disk system, the command is:

D I S K C O P Y A : B : Enter

This will copy the contents of the master disk in drive A to the backup disk in drive B.

To copy the files on the master diskette to your hard disk, perform the following steps.

1. Place the master diskette into a floppy drive
2. Change the active drive to the drive that has the diskette installed. For example, if the diskette is in drive A, type A : Enter.
3. Type I N S T A L L Enter and follow the on-screen prompts.

DIRECTORIES CREATED ON THE HARD DISK

The installation process will create several directories on your hard disk. If you accept the installation defaults, the following structure will exist.

[CARDNAME] Root or base directory containing the SETUP.EXE setup program used to help you configure jumpers and calibrate the card.

DOS\PSAMPLES: A subdirectory of [CARDNAME] that contains Pascal samples.

DOS\CSAMPLES: A subdirectory of [CARDNAME] that contains “C” samples.

WIN32*language* Subdirectories containing samples for Win95/98 and NT.

WinRisc.exe: A Windows dumb-terminal type communication program designed for RS422/485 operation. Used primarily with REMOTE ACCES Data Acquisition Pods and our RS422/485 serial communication product line. Can be used to say hello to an installed modem.

ACCES32: This directory contains the Windows 95/98/NT driver used to provide access to the hardware registers when writing 32-bit Windows software. Several samples are provided in a variety of languages to demonstrate how to use this driver. The DLL provides four functions (InPortB, OutPortB, InPort, and OutPort) to access the hardware.

This directory also contains the device driver for Windows NT, ACCESNT.SYS. This device driver provides register-level hardware access in Windows NT. Two methods of using the driver are available, through ACCES32.DLL (recommended) and through the DeviceIOControl handles provided by ACCESNT.SYS (slightly faster).

SAMPLES: Samples for using ACCES32.DLL are provided in this directory. Using this DLL not only makes the hardware programming easier (MUCH easier), but also one source file can be used for both Windows 95/98 and WindowsNT.

One executable can run under both operating systems and still have full access to the hardware registers. The DLL is used exactly like any other DLL, so it is compatible with any language capable of using 32-bit DLLs. Consult the manuals provided with your language's compiler for information on using DLLs in your specific environment.

VBACCES: This directory contains sixteen-bit DLL drivers for use with VisualBASIC 3.0 and Windows 3.1 only. These drivers provide four functions, similar to the ACCES32.DLL. However, this DLL is only compatible with 16-bit executables. Migration from 16-bit to 32-bit is simplified because of the similarity between VBACCES and ACCES32.

PCI: This directory contains PCI-bus specific programs and information. If you are not using an ACCES PCI card, this directory will not be installed.

SOURCE: A utility program is provided with source code you can use to determine allocated resources at run-time from your own programs in DOS.

PCIFind.exe A utility for DOS and Windows to determine what base address and IRQ are allocated to installed PCI cards. This program runs two versions, depending on the operating system. Windows 95/98/NT displays a GUI interface, and modifies the registry. When run from DOS or Windows3.x, a text interface is used. For information about the format of the registry key, consult the card-specific samples provided with the hardware. In Windows NT, NTioPCI.SYS runs each time the computer is booted, thereby refreshing the registry as PCI hardware is added or removed. In Windows 95/98/NT PCIFind.EXE places itself in the boot-sequence of the OS to refresh the registry on each power-up.

This program also provides some COM configuration when used with PCI COM ports. Specifically, it will configure compatible COM cards for IRQ sharing and multiple port issues.

WIN32IRQ: This directory provides a generic interface for IRQ handling in Windows 95/98/NT. Source code is provided for the driver, greatly simplifying the creation of custom drivers for specific needs. Samples are provided to demonstrate the use of the

generic driver. Note that the use of IRQs in near-real-time data acquisition programs requires multi-threaded application programming techniques and must be considered an intermediate to advanced programming topic. Delphi, C++ Builder, and Visual C++ samples are provided.

Findbase.exe DOS utility to determine an available base address for ISA bus , non-Plug-n-Play cards. Run this program once, before the hardware is installed in the computer, to determine an available address to give the card. Once the address has been determined, run the setup program provided with the hardware to see instructions on setting the address switch and various option selections.

Poly.exe A generic utility to convert a table of data into an n^{th} order polynomial. Useful for calculating linearization polynomial coefficients for thermocouples and other non-linear sensors.

Risc.bat A batch file demonstrating the command line parameters of RISCTerm.exe.

RISCTerm.exe A dumb-terminal type communication program designed for RS422/485 operation. Used primarily with REMOTE ACCES Data Acquisition Pods and our RS422/485 serial communication product line. Can be used to say hello to an installed modem. RISCTerm stands for Really Incredibly Simple Communications TERMinal.

INSTALLING THE CARD

The PCI-DIO-48H and PCI-DIO-48HS cards can be installed in a five-volt PCI slot of an IBM or compatible computer. Before installing the card, carefully read the **Option Selection** section of this manual and configure the card according to your requirements. Finally, our SETUP.EXE program will lead you through the process of setting the options on the PCI-DIO-48H. The setup program does not set the options. These must be set manually by jumpers on the card.

To install the card:

1. Turn OFF computer power.

2. Remove the computer cover.

3. Install jumpers from either the **Option Selection** section of this manual or the suggestions of our SETUP.EXE software program.

4. Install the card in an available PCI-bus slot. You may need to remove a backplate first.

5. Inspect for proper fit of the card and tighten screws. *Make sure that the card mounting bracket is properly screwed into place and that there is a positive chassis ground.*

6. Replace the computer cover and turn the computer ON.

7. Enter the CMOS setup program of your system and verify that the PCI plug-and-play option is set appropriately for your system. Systems running Windows95 (or any other PNP-compliant Operating System) should set the CMOS option to OS. Systems running under DOS, WindowsNT 3.51, Windows 3.1, or any other non-PNP-compliant Operating System should set the PNP CMOS option to BIOS or Motherboard. Save the option and continue booting the system.

INPUT/OUTPUT CONNECTIONS

To ensure that there is minimum susceptibility to EMI and minimum radiation, it is important that the card mounting bracket be properly screwed into place and that there be a positive chassis ground. Also, proper EMI cabling techniques (cable connect to chassis ground at the aperture, shielded twisted-pair wires, etc) should be used for the input/output wiring.

FUNCTIONAL DESCRIPTION

FEATURES

48 Bits of Digital Input/Output.
Interrupt Generation on Input or under program control (Model “48H”)
Change-of-state Interrupt Software Enabled in Six 8-Input Ports.(Model “48HS”)
All 48 I/O Lines Buffered on the Board.
I/O Buffers Can Be Enabled/Disabled under Program Control
Four and Eight Bit Ports Independently Selectable for I/O.
Pull-Ups on I/O Lines.
+5V Supply Available to the User.
Compatible with Industry Standard I/O Racks like Gordos, Opto-22, Potter & Brumfield, Western Reserve Controls, etc.

APPLICATIONS

Automatic Test Systems.
Laboratory Automation.
Robotics
Machine Control.
Security Systems, Energy Management.
Relay Monitoring and Control.
Parallel Data Transfer to/from the PC.
Sensing Switch Closures or TTL, DTL, CMOS Logic
Driving Indicator Lights or Recorders

DESCRIPTION

These PCI bus cards provide 48 bits of parallel digital input/output capability and can be installed in seven-inch (178 mm) or longer PCI-bus slots. Each I/O line is buffered and capable of sourcing 15 mA or sinking 24 mA (64 mA on request). The cards contain two type 8255 Programmable Peripheral Interface chips (PPI) to provide computer interface to 48 lines. Each PPI provides three 8-bit ports A, B, and C. Each 8-bit port can be software configured to function either as inputs or as latched outputs. Port C can also be software configured for four inputs and four latched outputs. Pull-ups on the cards assure that there are no erroneous outputs at power-up until the card is initialized by system software.

The feature that distinguishes the “48JS” model from the “48J" card is that the state of all inputs can be monitored and, if one or more bits change state, a latched interrupt request can be generated. Thus, it is not necessary to use software to continuously poll the inputs to detect a change of state. The change-of-state interrupt is enabled by a software write

to an interrupt-enable register. Six bits in that register control six eight-bit input ports. The change-of-state interrupt latch can be cleared by a software write.

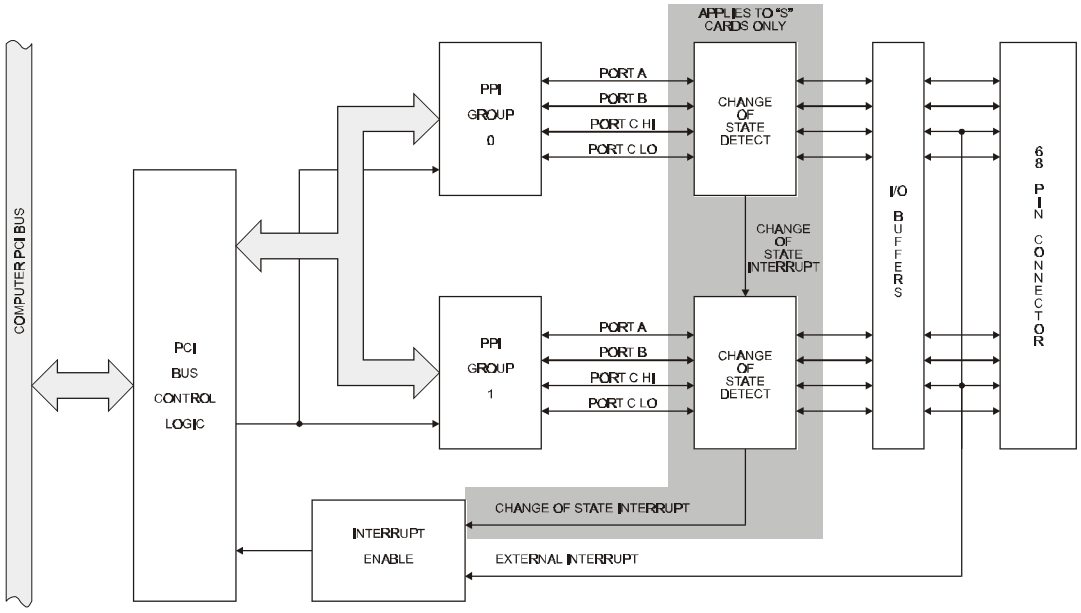
Also, Port C bit 3 at each 24-bit group can be used as an external interrupt to the computer if the IEN jumpers are installed. When Port C bit 3 goes high (edge triggering) when interrupts are enabled, an interrupt is generated. Interrupts from the ports are OR'ed together and OR'ed with the change-of-state interrupt. Interrupt levels are assigned by the system.

Tristate I/O line buffers (74LS245) are configured automatically by hardware logic for input or output according to the direction assignment programmed into a control register in the related PPI. Further, if a jumper is properly placed on the card the tristate buffers may be enabled/disabled under program control. (See the Option Selection section to follow.)

I/O wiring connections are via a 68-pin high-density connector. If needed for external circuits, +5 VDC power is available at pins 67 and 68. There is a resettable 0.5A polyfuse onboard that feeds both pins. However, if you use this power, westill recommend that you include an external 1A fast-blow fuse in your circuits in order to avoid possible damage to the host computer or cable in the event of a malfunction in those external circuits.

These cards occupy sixteen bytes of I/O address space. The base address is selected by the system. An illustrated setup program is provided on the diskette or CD shipped with your card. Interactive displays show locations and proper configuration of jumpers to set up the interrupt-enable function. Also, sample programs in Turbo-C and Turbo-Pascal are presented in the Software section of this manual.

PCI-DIO-48J and PCI-DIO-48JS BLOCK DIAGRAM



OPTION SELECTION

Refer to the setup programs on the CD or diskettes provided with the card. Also, refer to the block diagram on the previous page and the option selection map on the following page when reading this section of the manual.

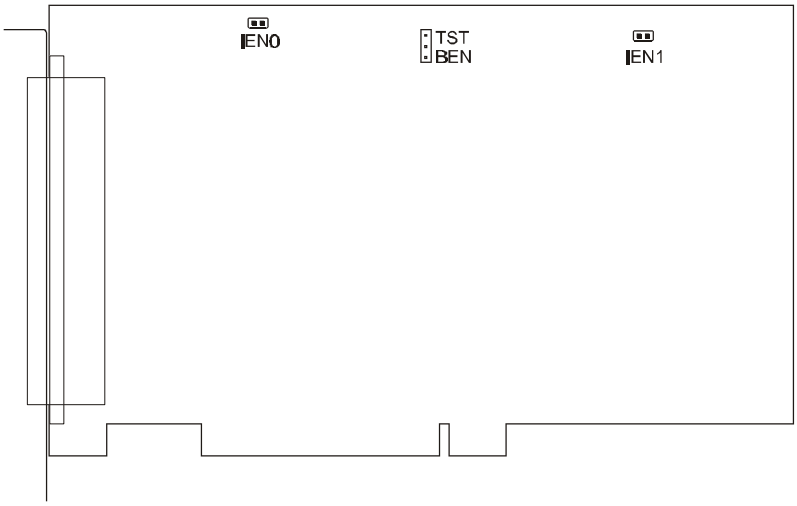
External Interrupts are accepted on the I/O connector pin 9 (bit C3 group 0) and pin 59 (bit C3 group 1). The Interrupt signal is a positive edge. External Interrupts are enabled if the IEN0 (for Group 0) and IEN1 (for Group 1) jumper is installed. Interrupts are directed to an available IRQ level by the system.

A means of enabling or disabling the 74LS245 input/output buffers under program control is provided at the jumper position labeled TST/BEN. When the jumper is in the BEN (Buffer Enable) position, the I/O buffers are always enabled. When the jumper is in the TST (Tristate) position, enabled/disabled state is controlled by a control register. (See the programming section of this manual for a description.)

> NOTE
>
> A jumper must be installed in either the TST or the BEN position for the card to function.

The foregoing are the only manual setups necessary to use either the PCI-DIO-48J or the PCI-DIO-48JS. Input/Output selection and the change-of-state Interrupt Enable is done via software by writing to a control register in each PPI as described in the PROGRAMMING section of this manual.

PCI-DIO-48J and PCI-DIO-48JS OPTION SELECTION MAP



ADDRESS SELECTION

These cards use one address space, occupying sixteen consecutive register locations.

PCI architecture is inherently plug-and-play. This means that the BIOS or Operating System determines the resources assigned to PCI cards rather than requiring the user to select those resources with switches or jumpers. As a result, you cannot set or change the card's base address or IRQ level. You can only determine what the system has assigned.

To determine the base address that has been assigned, run the ACCES-provided PCIFind.EXE utility program. This utility will display a list of all of the ACCES cards detected on the PCI bus, the addresses assigned to each function on each of the cards, and the respective IRQs allotted.

Alternatively, some operating systems (Windows95 and WindowsNT 5.0) can be queried to determine which resources were assigned. In these operating systems, you can use either PCIFind or the Device Manager utility from the System Properties Applet of the control panel. The cards are installed in the Data Acquisition class of the Device Manager list. Selecting the card, clicking Properties, and then selecting the Resources Tab will display a list of the resources allocated to the card.

PCIFind uses the your card, then reads the base address and IRQ assigned. If you want to determine these yourself, the Vendor ID is 494Fh (ASCII for "I/O") and the Device IDs are:

PCI-DIO-48J: 0C60h
PCI-DIO-48JS: 0E60h

The PCI bus supports 64K of I/O address space, so your card's addresses may be located anywhere in the 0000 to FFFF hex range.

SOFTWARE

develop your applications software.
with your card and are as described on page 1-1 of this manual.
describe the setup program and, then, the VisualBASIC utility program

SETUP.EXE

This program is supplied in the root or base directory as a tool for you to use in configuring
It is menu-driven and provides pictures of the card on the computer
You make simple keystrokes to select functions. The picture on the monitor then

The setup program is a stand-alone program that can be run at any time.
require that the card be plugged into the computer for any part of the setup.
is self-explanatory with operation instructions and on-line help.

Enter

VISUALBASIC UTILITY DRIVER

with your card. The extensions are in a directory named VBACCES.
are in the form of a .DLL, a .GBL, and a VisualBASIC sample.
you to access the port and main memory space in a fashion similar to BASIC, QuickBASIC,
++

To use these files
File l New Project) similar to the sample provided (or else, modify your existing project file)
Once this has been done, VisualBASIC will be

InPortb

port. Due to limitations of VisualBASIC,

Declaration: `function InPortb(byval address as integer) as integer`

Function: Reads an integer from a hardware port. This function returns the 16-bit value obtained from reading the low byte from *address* and the high byte from *address+1*.

Declaration: `function InPort(byval address as integer) as integer`

OutPortb

Function: Writes the lower eight bits of *value* to the hardware port at *address*. This function returns the value output.

Declaration: `function OutPortb(byval address as integer, byval value as integer) as integer`

OutPort

Function: Writes all 16 bits of *value* to the hardware port at *address*. This function returns the value output.

Declaration: `function OutPort(byval address as integer, byval value as integer) as integer`

Peek

Function: Reads a byte from main memory (DRAM).

Declaration: `function Peek(byval segment as integer, byval offset as integer) as integer`

Poke

Function: Writes the lower eight bits of *value* to *segment*:*offset.*

Declaration: `function Poke(byval segment as integer, byva offset as integer, byval value as integer) as integer`

Note that in all of the above functions, an inherent limitation of BASIC in general and VisualBASIC in particular makes the values sent less intuitive. All integers in BASIC are signed numbers, wherein data are stored in two's complement form. All bit patterns must be converted to-and-from this two's complement form if meaningful display is required. Otherwise, values returned from the InPortb function will be -128 to 127, rather than 0 to 255. An alternative is to perform all assignments in hexadecimal, rather then decimal form.

the program execute, the file must modified to the path the as appropriate for your system. Merely replace the statement "-*drive:path*\VBACCES.DLL".

an alternative changing the code, you copy the file into your Windows directory. This will allow multiple programs to find the same .DLL without to know it is Just leave off all to a in the file

PROGRAMMING

These cards are I/O-mapped devices that are easily configured from any language and any language can easily perform digital I/O through the card's ports. This is especially true if the form of the data is byte or word wide. All references to the I/O ports would be in absolute port addressing. However, a table could be used to convert the byte or word data ports to a logical reference.

DEVELOPING YOUR APPLICATION SOFTWARE

If you wish to gain a better understanding of the programs on diskette, then the information in the following paragraphs will be of interest to you. Refer to the data sheets and 8255-5 specification in Appendix A.

A total of 16 register locations are used by these. The PPIs are addressed consecutively with Address bits A3 through A0 as follows:

TABLE 2.ADDRESS ASSIGNMENT TABLE

Address	Port Assignment	Operation
Base Address	Port A Group 0	Read/Write
Base Address +1	Port B Group 0	Read/Write
Base Address +2	Port C Group 0	Read Write
Base Address +3	Control Group 0	Write Only
Base Address +4	Port A Group 1	Read/Write
Base Address +5	Port B Group 1	Read/Write
Base Address +6	Port C Group 1	Read/Write
Base Address +7	Control Group 1	Write Only
Base Address +8	Enable/Disable Buffer, Grp 0	Write Only
Base Address +9	Enable/Disable Buffer, Grp 1	Write Only
Base Address +B	Enable Chg-of-St. Interrupt	Write Only
Base Address +F	Clear Chg-of-St. Interrupt	Write Only

These cards use two type 8255-5 PPIs to provide a total of 48 bits input/output capability. The cards are designed to use each of these PPI's in Mode 0 wherein:

a. There are two 8-bit groups (A and B) and two 4-bit groups (C Hi and C Lo).
b. Any group can be configured as an input or an output.
c. Outputs are latched.
d. Inputs are not latched.

Each PPI contains Control Register. This write-only, 8-bit register is used to set the and direction the ports. At Power-Up or Reset, all I/O lines are set as inputs.

Each PPI should be configured during initialization by writing to the Control Registers even if the groups are only going to be used as inputs. Output buffers are automatically set by hardware according to the Control Register states. Note that Control Registers are located at base address +3 and base address +7. Bit assignments in each of these Control Registers are as follows:

TABLE 3. CONTROL REGISTER BIT ASSIGNMENT

Bit	Assignment	Code
D0	Port C Lo (C0-C3)	1=Input, 0=Output
D1	Port B	1=Input, 0=Output
D2	Mode Select	1=Mode 1, 0=Mode 0
D3	Port C Hi (C4-C7)	1=Input, 0=Output
D4	Port A	1=Input, 0=Output
D5,D6	Mode Select	00=Mode 0, 01=Mode 1, 1X=Mode 2
D7	Mode Set Flag	1=Active

Note: Mode 1 cannot be used by these cards without modification. Thus, bits D2, D5, and D6 should always be set to "0". If your card has been modified to operate in Mode 1, then there will be an Addendum page in the front of this manual. These cards cannot be used in Mode 2 of the PPI.

These cards provide a means to enable/disable the tristate I/O buffers under program control. If the TST/BEN jumper on the card is installed in the BEN position, the I/O buffers are permanently enabled. However, if that jumper is in the TST position, enable/disable of the buffers is *software controlled* via the control register as follows:

a. The card is initialized in the receive mode by the computer reset command.

b. When bit D7 of the Control Register is set high, direction of the three groups of the associated PPI chip as well as the mode can be set. For example, a write to Base Address +3 with data bit D7 high programs port direction at group 0 ports A, B, and C. If, for example, hex 80 is sent to Base Address +3, the group 0 PPI will be configured in mode 0 with ports A, B, and C as outputs.

At the same time, data bit D7 is also latched in a buffer controller for the associated PPI chip. A high state disables the buffers and, thus, all four buffers will be put in the tristate mode; i.e. disabled.

c. If any of the ports are to be set as outputs, you may set the values to the respective port with the outputs still in the tristate condition. (If all ports are to be set as inputs, this step is not necessary.)

d. If data bit D7 is low when the control byte is written, ONLY the associated buffer controller is addressed. If, for example, a control byte of hex 80 has been sent as previously described, and the data to be output are correct, and it is now desired to open the three ports, then it is necessary to send a control byte of hex 00 to base address +3 to enable the port 0 buffers. When you do this, the buffers will be enabled.

NOTE

Note that all data bits except D7 must be the same for the two control bytes

Those buffers will now remain enabled until another control byte with data bit D7 high is sent to base address +3.

Similarly, the group 1 ports can be enabled/disabled via the control register at base address +7. The following program fragment in C language illustrates the foregoing:

```
const BASE_ADDRESS 0x300;
/*Set the mode to Mode 0, ports A and B as output, and port C as input. Since bit D7 is high, the output buffers are set to tristate condition. See item b. above.*/
outportb(BASE_ADDRESS +3, 0x89);
outportb(BASE_ADDRESS,0);
/*These instructions set the initial state of ports A and B to all zeroes. Port C is not set because it is configured as an input. See item c. above.*/
outportb(BASE_ADDRESS+1,0);
/*Enable the tristate output buffers by using the same control byte used to configure the PPI, but now set bit D7 low. See item d. above.*/
outportb(BASE_ADDRESS +3, 0x09);
```

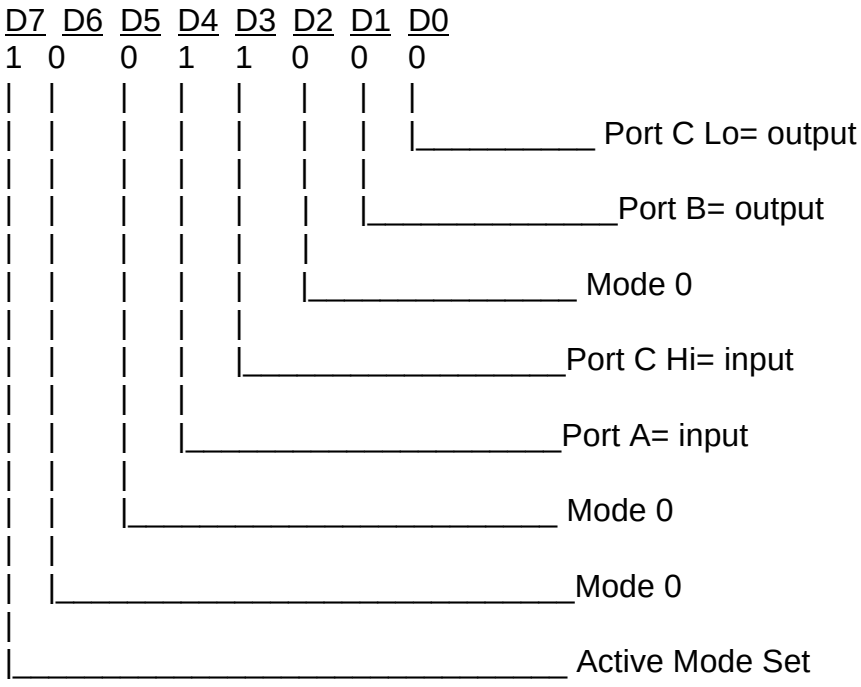
(Continued on next page)

PROGRAMMING EXAMPLE (BASIC)

The following example in BASIC is provided as a guide to assist you in developing your working software. In this example, the card base address is 2D0 hex and the I/O lines of group 0 are to be setup as follows:

Port A = Input
Port B = Output
Port C Hi= Input
Port C Lo= Output

The first step is to configure the control register. Configure bits of the control register as:



This corresponds to 98 hex. If the card address is 2D0 hex, use the BASIC OUT command to write to the control register as follows:

```
10  BASEADDR=&H2D0
20  OUT BASEADDR+3,&H98
```

To read the inputs at Port A and the upper nybble of Port C:

```
30  X=INP(BASEADDR)        'Read Port A
40  Y=INP(BASEADDR+2)/16   'Read Port C Hi
```

To set outputs high (1) at Port B and the lower nybble of Port C:

```
50  OUT BASEADDR+1,&HFF  'Turn on all Port B bits
60  OUT BASEADDR+2,&HF   'Turn on all bits of Port C
                            lower nybble
```

ENABLING/DISABLING I/O BUFFERS

When using the tristate mode (Jumper in the TST position), the method to disable the I/O buffers involved writing a control word to the Control Register at Base Address +3 and Base Address +7. This control word was required to have bit D7 (the most significant bit) set. That meant that the PPI translated it as an “active mode set” and reset the output data latches to “zero” on all output ports *and* the output buffers were disabled. However, if the buffers are to be enabled at a later time, the output latches will be in a “zero” state. For example, if all the outputs were 1's, they will now be 0's and the output buffers will be disabled. This can be resolved as follows.

Two computer I/O bus addresses are available that permit you to enable or disable the I/O buffers at will, *without programming the PPI mode*. Buffers for Port 0 bits are enabled/disabled at Base Address +8 and buffers for Port 1 bits are enabled/disabled at Base Address +9. To enable the buffers and to set outputs to the desired state, you can write to the Control Register with bit D7 low. If you wish to subsequently disable the buffers, you can write to the Control Register with bit D7 high. In this way you can enable/disable the output buffers without programming the PPI mode.

Note

When writing a command byte to these cards while the TST jumper is installed, the PPI output buffers are disabled. Thus, when you desire to to change the mode, you must first set the new mode and then enable the buffers. Enabling the buffers can be done at either Base Address +3 (or +7) or Base Address +8 (or +9).

CHANGE-OF-STATE INTERRUPTS (Model PCI-DIO48JS only)

At power-up or Reset, a register that enables change-of-state interrupts is set to zero. This *enables all inputs to generate change-of-state interrupts.* During initialization this register should be programmed to prevent interrupt generation by inputs that you do not want to cause change-of-state interrupts or by ports that are programmed as outputs. To program this Change-of-State-Interrupt-Enable Register, write to it at Base Address+B. Data bits D0 through D5 control ports A, B, and C of the 8255 PPIs as shown in Table 4.

TABLE 4. CHANGE-OF-STATE-INTERRUPT-ENABLE REGISTER

Bit	Port Controlled
D0	Group 0, Port A
D1	Group 0, Port B
D2	Group 0, Port C
D3	Group 1, Port A
D4	Group 1, Port B
D5	Group 1, Port C

Writing a “one” disables the port; writing a “zero” enables it. This register is latched. and, to clear the latch, write anything at Base Address+F.

INTERRUPT EXAMPLES

A rising edge with at least a microsecond hold time will be latched if the IEN jumper is in place and if any address decoded by the board has been read or written to. Interrupts are disabled on ‘power up’. An interrupt service routine may clear all interrupts from the board by writing any value to Base Address+1Fh.

Example 1: Internal interrupt (generated by software)

The interrupt option jumper for the group must be in the IEN position. Configure port C Low as an output and set bit C3 low. Enable interrupts globally by reading any register. Toggle bit C3 high for at least 1 uS then low. Note that this pulse will show up at pin 9 or 59 of the connector.

Example 2: External interrupt

The interrupt option jumper for the group must be in the IEN position. Configure port C Low as an input and enable interrupts globally by reading any address decoded by the board. The pulse at pins 9 or 59 (bit C3) must go high for at least 1 uS then low.

Example 3: Gating the external interrupt

An open-collector (or open-drain) driver must connect to pin 9 or 59 to avoid contention (this is critical). Write 0XXX (where X is ‘don’t care’) to port C Low. To gate the external interrupt ‘off’ configure port C Low as an output (don’t change the value of the most significant bit). To gate the external interrupt ‘on’ configure port C Low as an input.

CONNECTOR PIN ASSIGNMENTS

The mating is HPBP-3168-10 IDC type ABS Hood One Touch Latches, HPBP-7068-10 S, type for vermold (Acon catalog dated August 97), 10168-6000EC (shielded with or metal shell) or 10168-8000EC (unshielded) (3M Electronic Products Div.), or equivalent. Connector pin assignments are listed below.

		PIN	ASSIGNMENT	
Group 0 Port C Hi	PC7	1		9
Group 0 Port C Hi			"	10
	PC5	3	"	
Group 0 Port C Hi	PC4	4		12
Group 0 Port C Lo			Ground	21
	PC2	6	"	
Group 0 Port C Lo	PC1	7		23
Group 0 Port C Lo			"	24
	PB7	13	Ground	
Group 0 Port B	PB6	14		34
Group 0 Port B				
	PB4	16		
Group 0 Port B	PB3	17		
Group 0 Port B				
	PB1	19		
Group 0 Port B	PB0	20		
Group 0 Port A				
	PA6	26		
Group 0 Port A	PA5	27		
Group 0 Port A				
	PA3	29		
Group 0 Port A	PA2	30		
Group 0 Port A				
	PA0	32		
+5 VDC		68		

* This line is an I/O port and also a User Interrupt.

	PIN	ASSIGNMENT	

Group 1 Port C Hi	PC7	37	Ground	35
Group 1 Port C Hi	PC6	38	"	36
Group 1 Port C Hi	PC5	39	"	45
Group 1 Port C Hi	PC4	40	"	46
Group 1 Port C Lo	PC3*	41	Ground	47
Group 1 Port C Lo	PC2	42	"	48
Group 1 Port C Lo	PC1	43	"	57
Group 1 Port C Lo	PC0	44	"	58
Group 1 Port B	PB7	49		
Group 1 Port B	PB6	50		
Group 1 Port B	PB5	51		
Group 1 Port B	PB4	52		
Group 1 Port B	PB3	53		
Group 1 Port B	PB2	54		
Group 1 Port B	PB1	55		
Group 1 Port B	PB0	56		
Group 1 Port A	PA7	59		
Group 1 Port A	PA6	60		
Group 1 Port A	PA5	61		
Group 1 Port A	PA4	62		
Group 1 Port A	PA3	63		
Group 1 Port A	PA2	64		
Group 1 Port A	PA1	65		
Group 1 Port A	PA0	66		
+5 VDC		67		

* This line is an I/O port and also a User Interrupt.

SPECIFICATIONS

Logic High:	2.0 to 5.0 VDC
	-0.5 to +0.8 VDC
Input Load (Hi):	
Input Load (Lo):	-200 uA

Logic High: 2.5 VDC min., source 15 mA
Logic Low: 0.5 VDC max., sink 24 mA (64 mA optional)

+5 VDC from computer bus (ext. 1A fast-blow fuse recommended)

Power Required:

Size: 6.9" Long (176 mm)

Operating Temperature: 0 degr. to 60 degr. C
Storage Temperature: -50 degr. to +120 degr. C

WARRANTY

Prior to shipment, ACCES equipment is thoroughly inspected and tested to applicable specifications. However, should equipment failure occur, ACCES assures its customers that prompt service and support will be available. All equipment originally manufactured by ACCES which is found to be defective will be repaired or replaced subject to the following considerations.

TERMS AND CONDITIONS

If a unit is suspected of failure, contact ACCES' Customer Service department. Be prepared to give the unit model number, serial number, and a description of the failure symptom(s). We may suggest some simple tests to confirm the failure. We will assign a Return Material Authorization (RMA) number which must appear on the outer label of the return package. All units/components should be properly packed for handling and returned with freight prepaid to the ACCES designated Service Center, and will be returned to the customer's/user's site freight prepaid and invoiced.

COVERAGE

First Three Years: Returned unit/part will be repaired and/or replaced at ACCES option with no charge for labor or parts not excluded by warranty. Warranty commences with equipment shipment.

Following Years: Throughout your equipment's lifetime, ACCES stands ready to provide on-site or in-plant service at reasonable rates similar to those of other manufacturers in the industry.

EQUIPMENT NOT MANUFACTURED BY ACCES

Equipment provided but not manufactured by ACCES is warranted and will be repaired according to the terms and conditions of the respective equipment manufacturer's warranty.

GENERAL

Under this Warranty, liability of ACCES is limited to replacing, repairing or issuing credit (at ACCES discretion) for any products which are proved to be defective during the warranty period. In no case is ACCES liable for consequential or special damage arriving from use or misuse of our product. The customer is responsible for all charges caused by modifications or additions to ACCES equipment not approved in writing by ACCES or, if in ACCES opinion the equipment has been subjected to abnormal use. "Abnormal use" for purposes of this warranty is defined as any use to which the equipment is exposed other than that use specified or intended as evidenced by purchase or sales representation. Other than the above, no other warranty, expressed or implied, shall apply to any and all such equipment furnished or sold by ACCES.

APPENDIX A
PROGRAMMABLE PERIPHERAL INTERFACE DATA

The ata sheets in this Appendix are provided to help your understanding of the 8255-5 PPI which is made by a number of companies. These sheets are reprinted

The diagrams, and other data are believed be correct reliable. However, no responsibility is assumed by Mitsubishi Electric Corporation for use, nor any infringements patents or rights belonging third parties which may from their Values shown on these data sheets are subject to change for product improvement.