ACCES I/O PRODUCTS INC
10623 Roselle Street, San Diego, CA 92121
TEL (858)550-9559 FAX (858)550-7322

MODEL PCI-DIO-24S

USER MANUAL

FILE: MPCIDIO24S.Da

Notice

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.

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.

Table of Contents

List of Figures

List of Tables

Chapter 1: Introduction

Features

- 24 Bits of Digital Input/Output.
- Interrupt Generation on Input Change of State.
- Change-of-state Interrupt Software Enabled in three 8-Bit Ports.
- All 24 I/O Lines Buffered on the Card.
- 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.

Applications

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

These cards support 24 bits of parallel digital input/output on the PCI bus. They can be programmed to accept inputs or to provide outputs on three eight bit ports. Further, one of these ports can be further divided into four-bit nibbles.

The feature that distinguishes the "24S" model from a standard 24-bit digital I/O card is that the state of all inputs can be monitored and, if any 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 each control an eight-input port at a type 8255-5 Programmable Peripheral Interface chip. The change-of-state interrupt latch can be cleared by a software write.

Also, bit C3 can be used as an external interrupt to the computer if the IEN0 jumper is installed. When bit C3 goes high (edge triggering), an interrupt is requested. The Interrupt level is assigned by the system.

These cards were designed for industrial applications and can be installed in 7", or longer, PCI slots of IBM or compatible computers. Each I/O line is buffered and capable of sourcing 15 mA or sinking 24mA (64 mA on request). The card contains a Programmable Peripheral Interface chip type 8255-5 (PPI) to provide computer interface to 24 I/O lines. The PPI supports three 8-bit ports A, B, and C. Each 8-bit port can be software configured to function as either inputs or output latches. Port C can also be configured as four inputs and four output latches. Pull-ups on the card assure that there are no erroneous outputs at power up until the card is initialized by system software.

Tristate I/O line buffers (74LS245) are configured automatically by hardware logic for input or output use according to direction assignment from a control register in the 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 50-pin header on the card. Flat insulation-displacement ribbon cables can be used to connect these cards to termination panels. This provides compatibility with OPTO-22, Gordos, Potter & Brumfield, Western Reserve Controls, et al module mounting racks. Every second conductor of the flat cable is grounded to minimize the effect of crosstalk between signals. If needed for external circuits, +5 VDC power is available on each I/O connector pin 49. If you use this power, we recommend that you include a 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.

Specifications

Digital Inputs (TTL Compatible)

- Logic High: 2.0 to 5.0 VDC.
- Logic Low: -0.5 to +0.8 VDC.
- Input Load (Hi): 20 uA.
- Input Load (Lo): -200 uA.

Digital Outputs

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

- Power Output: +5 VDC from computer bus (ext. 1A fast-blow fuse recommended).
- Power Required: +5 VDC at 200 mA typical.
- Size: 7.15" Long.

Environmental

- Operating Temperature: 0 °C. to 60 °C.
- Storage Temperature: -50 °C. to +120 °C.
- Humidity: 0 to 90% RH, non-condensing.

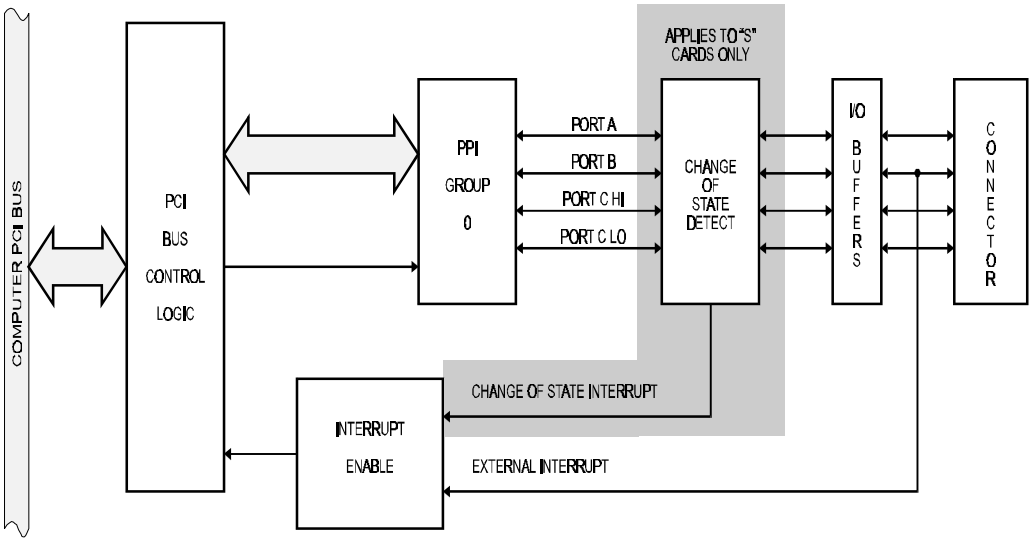


Figure 1-1: PCI-DIO-24S Block Diagram

Chapter 2: 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:K to change the active drive to the CD-ROM drive.
3. Type installK to run the install program.
4. Follow the on-screen prompts to install the software for this card.

WIN95/98/NT

a. Place the CD into your CD-ROM drive.
b. The CD should automatically run the install program after 30 seconds. If the install program does not run, click START | RUN and type d:install, click OK or press K.
c. Follow the on-screen prompts to install the software for this card.

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:

diskcopy a: a:K

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

diskcopy a: b:K

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:K.
3. Type installK 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 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 a 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 addresses and IRQs 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 nth 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 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-24S card can be installed in a five-volt PCI slot of an IBM or compatible computer. Before , 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-24S. 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.

If you are using Windows95, your operating system should detect the new hardware and prompt you for the installation disk. The simplest option is to insert the diskette provided into the A drive and allow the operating system to look for the files that it wants.

Alternatively, if you have already installed software to your computer, you can specify the installation directory instead of allowing Windows95 to look around. Either option will work acceptably. Please note that even though Windows95 looks for and finds software to install, no driver is actually installed, used, or needed. This step merely tells Windows95 where to place the card in the registry of devices. You can find the installed card in Device Manager, under the class Data Acquisition. Use this method to determine the card's base address as PCIFind.EXE may or may not work under your specific installation of Windows95.

If you are not using Windows95, or if your installation of Windows95 does not detect the new hardware you installed, run PCIFind.EXE to determine the base address that the computer assigned.

The base address assigned by BIOS or the operating system can change each time new hardware is installed into or removed from the computer. Please re-check PCIFind or Device Manager if the hardware configuration is changed,

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 wiring, etc) be used for the input/output wiring.

Chapter 3: Option Selection

Refer to the setup programs on the diskette provided with the card. Also, refer to the Figure 1-1, PCI-DIO-24S Block Diagram and Figure 3-1, PCI-DIO-24S Option Selection Map when reading this section of the manual.

External Interrupts are accepted on the I/O connector pin 9 (bit C3). The Interrupt signal is positive true. External Interrupts are enabled if the IEN0 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 this card. Input/Output selection and the change-of-state Interrupt Enable is done via software by writing to a control register in the PPI as described in the Programming chapter of this manual.

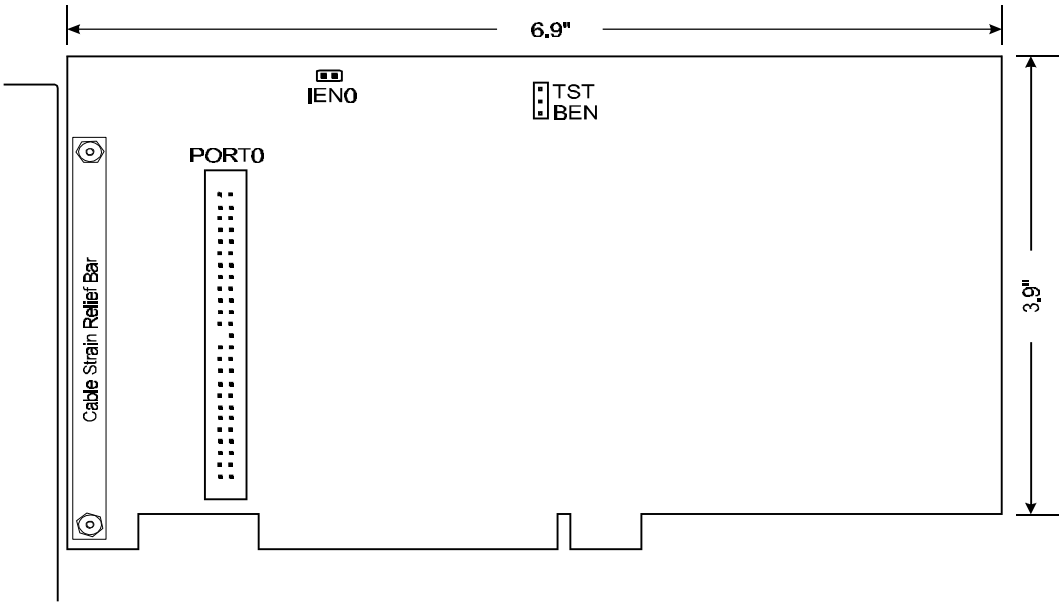


Figure 3-1: PCI-DIO-24S Option Selection Map

Chapter 4: Address Selection

These cards use one address space and occupy sixteen consecutive register locations.

PCI architecture is Plug-and-Play. This means that the BIOS or Operating System determines the resources assigned to PCI cards rather than you selecting 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 PCIFind.EXE utility program provided. 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 (if any) allotted.

Alternatively, some operating systems (Windows95/98/2000) 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. These 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.

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.

PCIFind uses the Vendor ID and Device ID to search for your card, then reads the base address and IRQ. If you want to determine the base address and IRQ without using PCIFind, use the following information:

The Vendor ID for these cards is 494F. (ASCII for "IO")
The Device ID for the PCI-DIO-24S is 0E50h.

An example of how to locate PCI card resources is provided with in the PCI\SOURCE directory, under your installation directory. This code runs in DOS, and uses the PCI defined interrupt BIOS calls to query the PCI bus for card specific information. You will need the Device ID and Vendor ID listed above to use this code.

Chapter 5: Software

Three programs are supplied to support this Digital I/O card and, also, to help you develop your applications software. These programs are on diskettes that come with your card and are as described on page 2-1 of this manual. The following paragraphs describe the setup program and 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 jumpers on the card. It is menu-driven and provides pictures of the card on the computer monitor. You make simple keystrokes to select functions. The picture on the monitor then changes to show how the jumper should be placed to effect your choices.

The setup program is a stand-alone program that can be run at any time. It does not require that the card be plugged into the computer for any part of the setup. The program is self-explanatory with operation instructions and on-line help.

To run this program, at the DOS prompt, enter SETUP.EXE followed by K.

VisualBASIC Utility Driver

Extensions are provided to the VisualBASIC language on the diskette provided with your card. The extensions are in a directory named VBACCES. These extensions are in the form of a .DLL, a .GBL, and a VisualBASIC sample. Together these files allow you to access the port and main memory space in a fashion similar to BASIC, QuickBASIC, Pascal, C/C++, Assembly, and most other standard languages.

To use these files in a VisualBASIC program, you must create a .MAK file (File l New Project) similar to the sample provided (or else, modify your existing project file) and include the .GBL file (File | Add File). Once this has been done, VisualBASIC will be
enhanced with the addition of the following functions.

InPortb

Function: Reads a byte from a hardware port. Due to limitations of VisualBASIC, the number is returned in an integer.

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

InPort

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, byval 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.

Before the program will execute, the .GBL file must be modified to include the path to the VBACCES.DLL as appropriate for your system. Merely replace the statement "VBACCES.DLL" with "drive:path\VBACCES.DLL".

As an alternative to changing the source code, you can copy the VBACCES.DLL file into your Windows directory. This will allow multiple programs to find the same .DLL without having to know where it is located. Just leave off all references to a path in the .GBL file as shown in the sample.

Chapter 6: Programming

The PCI-DIO-24S 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 cards. The PPI is addressed consecutively with Address bits A3 through A0 as follows:

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

Table 6-1: Address Assignment Table

These cards use an 8255-5 PPI to provide a total of 24 bits input/output capability. The card is designed to use the PPI 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 port can be configured as an input or an output.
c. Outputs are latched.
d. Inputs are not latched.

The PPI contains a Control Register. This write-only, 8-bit register is used to set the mode and direction of the ports. At Power-Up or Reset, all I/O lines are set as inputs. The PPI should be configured during initialization by writing to the Control Registers even if the ports are only going to be used as inputs. Output buffers are automatically set by hardware according to the Control Register states. Note that Control Register is located at base address +3 and bit assignments are as follows:

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

Table 6-2: Control Register Bit Assignment

Note

Mode 1 cannot be used by these without modification (Consult factory.). Thus, bits D2, D5, and D6 should always be set to "0". If your card has been modified for use in Mode 1, then there will be an Addendum sheet covering that in the front of this manual. These cards cannot be used in PPI Mode 2.

Note

In Mode 0, do not use the control register byte for the individual bit control feature. The hardware uses the I/O bits to control buffer direction on this card. The control register should only be used for setting up input and output of the ports and enabling the buffer.

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 0 ports A, B, and C. If, for example, hex 80 is sent to Base Address +3, the Port 0 PPI will be configured in mode 0 with Groups 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. Now, if any of the ports are to be set as outputs, you may set the values to that 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 groups, 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. The following program fragment in C language illustrates the foregoing:

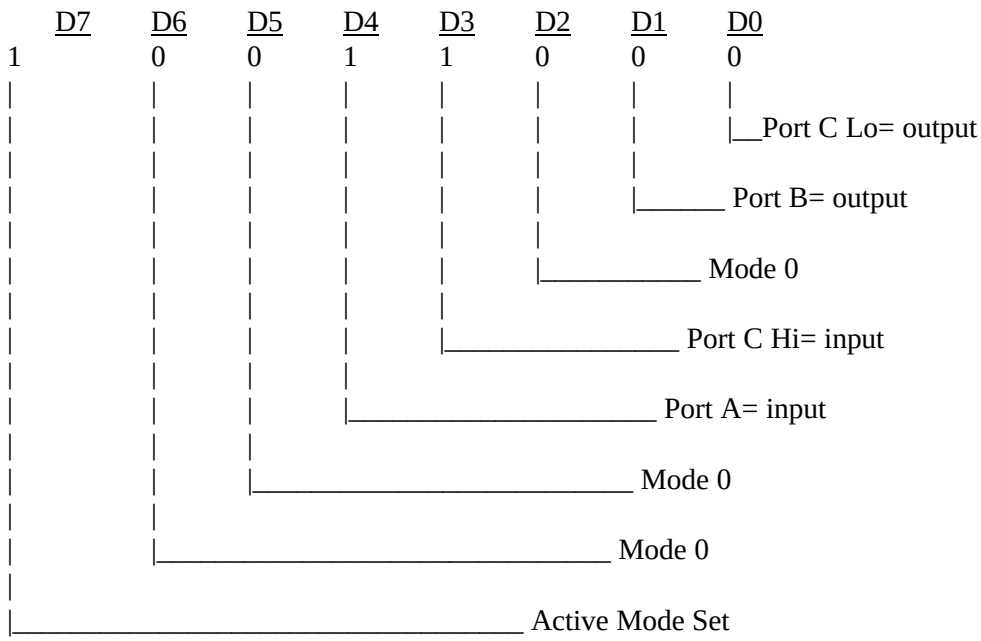
```
const BASE_ADDRESS 0x300;
outportb(BASE_ADDRESS +3, 0x89);      /*This instruction sets the mode to Mode 0, ports
                                        A and B as output, and port C as input. Since bit
                                        D7 is high, the output buffers are set to tristate
                                        condition. See item b. above.*/
outportb(BASE_ADDRESS,0);
outportb(BASE_ADDRESS+1,0);           /*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 +3, 0x09);     /*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.*/
```

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. 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 problem 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 are enabled/disabled at Base Address +8 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 the card while theTST 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 Base Address +8.

Change-of-State Interrupts

At power-up or Reset, all interrupts are disabled. To enable interrupts, read or write to any address decoded by the card. During initialization, the change-of-state interrupt enable 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 PPI as shown in Table 6-4.

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

Table 6-3: Change-of-State-Interrupt-Enable Register

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.

Chapter 7: Connector Pin Assignments

A 50-pin header is provided on these cards for I/O connections. The mating connector is an AMP type 1-746285-0 or equivalent. Connector pin assignments are listed below. Notice that every second line is grounded to minimize crosstalk between signals.

Assignment		Pin		Assignment	Pin
Port C Hi	PC7	1		Ground	2
Port C Hi	PC6	3		"	4
Port C Hi	PC5	5		"	6
Port C Hi	PC4	7		"	8
Port C Lo	PC3*	9		Ground	10
Port C Lo	PC2	11		"	12
Port C Lo	PC1	13		"	14
Port C Lo	PC0	15		"	16
Port B	PB7	17		Ground	18
Port B	PB6	19		"	20
Port B	PB5	21		"	22
Port B	PB4	23		"	24
Port B	PB3	25		"	26
Port B	PB2	27		"	28
Port B	PB1	29		"	30
Port B	PB0	31		"	32
Port A	PA7	33		Ground	34
Port A	PA6	35		"	36
Port A	PA5	37		"	38
Port A	PA4	39		"	40
Port A	PA3	41		"	42
Port A	PA2	43		"	44
Port A	PA1	45		"	46
Port A	PA0	47		"	48
+5 VDC		49		Ground	50

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

Table 7-1: Connector Pin Assignments

Appendix A: Programmable Peripheral Interface Data Sheets

The data 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 with permission of Mitsubishi Electric Corp. (Copyright 1987).

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

Customer Comments

If you experience any problems with this manual or just want to give us some feedback, please email us at: ***manuals@accesioproducts.com.***. Please detail any errors you find and include your mailing address so that we can send you any manual updates.

ACCES I/O PRODUCTS, INC.
10623 Roselle Street, San Diego CA 92121
Tel. (619)550-9559 FAX (619)550-7322
www.accesioproducts.com