Software Manual

DigiBoard PC/Xe and PC/Xi
Intelligent Serial Communications Boards

Solaris x86, Version 2.x

DBI 92000192A

Digi International Inc.,
d. b. a. DigiBoard
6400 Flying Cloud Drive
Eden Prairie, MN 55344

Phone (800) 344-4273
(612) 943-9020
FAX (612) 943-5398

Table of Contents

Introduction

The DigiBoard device driver software for Solaris x86, Version 2.x is an installable device driver; the software development system is not required to create a new kernel.

This manual also contains instructions for the following features:

- **ditty**, a utility program that sets and displays the terminal options for DigiBoard intelligent serial products (page 18)
- DigiPRINT transparent printing (page 22)
- DigiBoard Port Authority, a port monitoring and diagnostic tool (page 26)

Once you have completed the Hardware Installation instructions in the main DigiBoard hardware *Installation Guide*, you may proceed with the software device driver installation instructions starting on the following page.

Important! Be sure to read the *Release Notes* that may be included with this software device driver. The *Release Notes* contain information not available at this manual's press time.

Hardware Installation

This section provides the information needed to set the DIP switches on the board(s) prior to installation of the device driver software. See your board's *Installation Guide* for complete installation instructions.

Supported Boards

This device driver software supports the DigiBoard PC/Xe and PC/Xi families of intelligent serial communications boards. These boards are sometimes collectively referred to as PC/X* (read "PC/X-all") boards.

PC/Xi Boards and Older PC/Xe Boards

PC/Xi boards and older PC/Xe boards have two banks of DIP switches located on the top edge of the board. These switches are used to set the board's I/O port address, the dual ported memory starting address and the interrupt request (IRQ) line selection.

I/O Port Selection

PC/Xi boards and older PC/Xe boards can be configured to use one of seven I/O port addresses: **100h**, **110h**, **120h**, **200h**, **220h**, **300h** or **320h**. Each board must have its own I/O port address, which must not be used by any other device in the system, including other DigiBoard products. Refer to the *Installation Guide* for your board for switch setting information.

Memory Start Address Selection

PC/Xi boards and older PC/Xe boards can be configured to use any of a wide range of memory start addresses for their dual ported memory. This device driver software supports eleven of these addresses: **080000h**, **090000h**, **0C0000h**, **0D0000h**, **0E0000h**, **D00000h**, **D80000h**, **E00000h**, **E80000h**, **F00000h** and **F80000h**. Note that the first five addresses (080000h-0E0000h) are below the 1 megabyte boundary; the remaining addresses are in the sixteenth megabyte.

If your system has sixteen megabytes or more of RAM, you must use one of the addresses below 1 megabyte. Also, if your system has a memory cache, you may need to use one of the addresses below 1 megabyte. This is because some cache controllers interfere with the correct operation of the dual ported memory when it is addressed above 1 megabyte.

All DigiBoard intelligent boards (including C/X, EPC/X and Xem host adapters, but excluding COM/Xi boards) may share the same memory start address.

IRQ Selection

This device driver software does not use interrupts; set all eight switches in DS2 to the OFF position (away from the board) to disable interrupts.

Preset Configuration Options

If you are installing a single board, four preset memory and I/O address combinations are available to streamline the software installation process. If you are installing multiple boards, or if none of the preset options can be used (due to conflicts with other devices or memory), you can custom configure the device driver for any combination of I/O and memory start addresses. The preset options are:

Option 1: Memory Start Address: 0D0000h; I/O Address: 320h
Option 2: Memory Start Address: F00000h; I/O Address: 300h
Option 3: Memory Start Address: 0D0000h; I/O Address: 220h
Option 4: Memory Start Address: F00000h; I/O Address: 120h

Newer PC/Xe Boards

Newer PC/Xe boards have one bank of DIP switches on the end of the board. Only the I/O port address is set via switches. The dual ported memory start address and IRQ are set by the device driver when the system is booted up.

These boards can be configured to use one of seven I/O port addresses: **100h, 110h, 120h, 200h, 220h, 300h** or **320h**. Each board must have its own I/O port address, which must not be used by any other device in the system, including other DigiBoard products. Refer to the *Installation Guide* for your board for switch setting information.

Software Installation

Note: Software changes more rapidly than printed documentation can keep up. For this reason, some of the screens or prompts may not appear exactly as shown.

Installing the Driver Package

1. Log onto the console as super-user (root).

2. To begin the installation, load the device driver diskette into drive A and type:

   `pkgadd -d diskette1`

   (If you wish to load the driver from drive B, substitute "**`diskette2`**" for "**`diskette1`**" in the above command.)

3. Next, you must answer a series of questions about the installation. In each case, options will be presented and/or examples will guide you in answering the questions. The driver supports up to four PC/X* boards, with up to sixteen ports each. You will see the screen:

```
The DigiBoard PC/X* driver may be installed using one of the four
suggested configurations listed below, or you may choose to create your
own custom configuration.

        1) I/O Port=0x320, Memory Address=0x000D0000
        2) I/O Port=0x300, Memory Address=0x00F00000
        3) I/O Port=0x220, Memory Address=0x000D0000
        4) I/O Port=0x120, Memory Address=0x00F00000
        5) Custom or multiple board installation.
        6) Exit installation procedure.

        Please enter selection (1-6)?
```

   When asked, select your configuration from the above list (1-5), or choose option 6 to abort the device driver installation.

Note that options 1 through 4 are for installing *single boards only!* If you are installing more than one board, or wish to set up one board with parameters different from those above, choose **#5**.

For any of the versions selected, the corresponding DIP switches on the PC/X* board *must* be set to match the parameters listed in the configurations above.

If you selected **Option 5**, the next prompts you will see will be for **Custom Configuration Instructions.** If you answered 1-4, skip to **Step 10** on page 11.

4. The software displays:

```
Each DigiBoard PC/X* board requires an I/O port address and an 8K to 512K
block of memory.  Please select these to avoid conflicts with other boards
you may have installed in your system.

How many PC/X* boards do you wish to install (1-7)?
```

Enter in the number of PC/X* boards you are installing.

Steps 5-8 are repeated for each of the boards you specified in Step 4.

5. You must select a *Port address* for each board. The software displays the Port address options.

```
Ready to configure board #1.
============================

        The I/O port address choices, in hexadecimal, are:
                100     200     300
                110     220     320
                120
        Please enter selection (100-320):
```

Make your selection by entering one of the numbers shown above.

6. The next item requires the selection of the *host* starting address of the board's dual-ported memory (all boards can operate at the same Memory Start address, if desired). The software lists the available Memory Start address options:

```
The memory address choices, in hexadecimal, are:
                08) 0x00080000   0E) 0x000E0000   E8) 0x00E80000
                09) 0x00090000   D0) 0x00D00000   F0) 0x00F00000
                0C) 0x000C0000   D8) 0x00D80000   F8) 0x00F80000
                0D) 0x000D0000   E0) 0x00E00000
        Please enter selection code (08-F8):
```

When asked, enter your selection (08-F8).

7. The software then asks:

```
Please indicate number of ports on board #1 (2, 4, 8, or 16):
```

Answer 2, 4, 8 or 16 as appropriate for the number of channels on the DigiBoard product you are installing.

8. Next, you need to select the **altpin** setting. This allows alternate wiring of the RJ-45 modular connectors. Setting **altpin** to "OFF" gives you the standard DigiBoard RJ-45 pinouts. Setting **altpin** to "ON" enables *alternate* RJ-45 pinouts. (This is useful for 8-pin RJ-45 connectors. See the *Installation Guide* for your board for more details, as well as **`[-]altpin`** on page 19.)

The software asks:

```
Do you want to set altpin for this board? (y or n)
```

Answer **y** or **n** as appropriate.

Note that this sets *all* ports to **altpin**. Individual ports can be set with **ditty**. (See page 18.)

Steps 5-8 will be repeated for each PC/X* board you specified in Step 4.

9. You will now be asked to confirm your selections. Depending upon the version of your operating system, you will see one of the following screens (or the equivalent, reflecting your configuration):

```
You have selected the following configuration:

        Board   I/O Address   Memory Address   Devices
        ------------------------------------------------------------
         1         0x100         0x000D0000      dty/i1a - dty/i1h
                                 Modem devices:  dty/i1A - dty/i1H
         2         0x200         0x000D0000      dty/i2a - dty/i2h
                                 Modem devices:  dty/i2A - dty/i2H
         3         0x300         0x000D0000      dty/i3a - dty/i3p
                                 Modem devices:  dty/i3A - dty/i3P
         4         0x110         0x000D0000      dty/i4a - dty/i4p
                                 Modem devices:  dty/i4A - dty/i4P

        Is this configuration acceptable (y or n)?
```

If the information is correct, answer "y." Otherwise, answer "n" and the software will let you start over again with Step 4.

Next, the installation script creates devices in **/dev**, giving Solaris the information to use these additional ports. The DigiBoard driver will support up to seven PC/X* boards, for a maximum of 112 asynchronous ports.

The devices are named with the following conventions:

Format used for port names

/dev/dty/
: Directory path for tty devices.

/dev/dpr/
: Directory path for DigiPRINT transparent printer devices.

i
: Prefix for tty ports. Devices are in **/dev/dty**.

1-7
: Board ID—Driver supports up to four boards.

a-p
: Port ID of standard (not modem-controlled) port on the specified PC/X* board.

A-P
: Port ID of "modem-controlled port" on the specified PC/X* board. Modem ports can only be opened when Data Carrier Detect (DCD) is active.

For example, **/dev/dty/i1d** specifies a standard device on the fourth port on board 1. **/dev/dty/i1D** specifies the same port, but defines it as a modem device. The DigiPRINT transparent printer device for that port is **/dev/dpr/i1d**.

Loading the Driver

When the driver loads, you will see the following (or similar) message:

```
pcxx:   port: 0x00000320   mem:0x000D8000   8 ports   PC/Xi   Ver 1.2.0
```

If there is a problem with the board or the driver configuration, you may see error messages after this screen—see page 27 for explanations of the error messages that may occur.

It is not necessary to reboot after the initial installation, but it is encouraged, to be certain that the driver is properly installed. When the system is rebooted, the driver will be force loaded and the above message will be seen if all is well.

Important!

ditty options set form an **rc** script must be set after the download program has run, and must wait for the board to boot before running.

ditty options should be added to the end of **/etc/rc2.d**. This can be done by giving it a name such as "**S99epc**" or "**S99dittystuff**", where the "**S**" must be a capital letter, and the greater the number, the later the script is run.

For example, to set **altpin** for port **i1a**, add the following line to **/etc/rc2.d/S99dittystuff**:

```
/usr/bin/ditty altpin /dev/dty/i1a
```

Enabling the New Ports

Please refer to your Solaris System Administrator's Guide, under the *Peripherals Setup* section, for the details on how to enable serial ports. The following example shows how to set up serial ports for use with terminals:

1. Connect terminals to the ports (using a null modem, if necessary) and test the connections to each terminal by entering the following command for *each* port added:

   ```
   date > /dev/dty/i1a
   ```

   (Assuming the terminal is connected to **i1a**.)

   Please note that in the above, the "date" command is used as a simple test, to provide text output that can be redirected; there is no other significance to "date" in this test.

   - If the date appears on the terminal's screen, the device is properly connected.
   - If the date *does not* appear on the terminal's screen, then that terminal is not receiving data; check the power, cables, connections, etc.
   - If nonsense characters are printed on the terminal's screen, check the baud rates, data bits, stop bits, and parity setting on your terminal.

   Once you can redirect output to a terminal with the test above, perform the following steps to enable that port.

2. Log onto the console as super-user (root).

3. Enter (on a single command line):

   ```
   pmadm -a -p ttymonx -s i1a -fu -i root -v `ttyadm -V`\
   -m "`ttyadm -d /dev/dty/i1a -l 9600 -s /usr/bin/login`"
   ```

   where **ttymon***x* is the port monitor tag. This will enable the port suitably for a terminal. For more information, see **ttymon** and **ttyadm** in the *Answer Book*.

Un-Installing the Device Driver

Enter the following commands to *un-install* the device driver software:

1. Log onto the console as super-user (root).

2. Enter the following command:

   pkgrm

A list of installed packages will be presented. Choose the **pcxx** package. The package containing the driver will now be removed from the system. In the event that some of the ports were open when **pkgrm** was initiated, the unloading of the driver will fail. In this case it will be necessary to reboot to complete the driver removal.

DigiBoard TTY Devices

The DigiBoard PC/X* device driver supports three different device types on each line. For example, on the first port of board number 1, there are three devices:

/dev/dty/i1A

The "modem" tty device, used for terminals, modems, printers, laboratory equipment, etc. This is also sometimes referred to as a "dial-in" device.

The **i1A** device is a traditional UNIX port with modem control. It requires Data Carrier Detect (DCD) to be high before it will operate. RTS/CTS handshaking is enabled by default.

When used with a modem, the port will wait for carrier before sending out the **login:** prompt, so the user is greeted properly upon making a connection.

When used with a terminal or other device, it is usually wise to wire the DigiBoard DCD signal to the terminal's DTR (Data Terminal Ready) line. When the terminal is turned on, the system outputs a **login:** prompt. When the terminal is turned off, any associated jobs are killed, and the user is logged out.

/dev/dty/i1a

Standard device. This is the same as **i1A** with the exception that the default handshake method is XON/XOFF, and that Data Carrier Detect need not be present to open the device. *Once a connection is established and DCD becomes active, standard devices behave in the same way as modem devices—subsequent loss of the Data Carrier Detect signal will cause the jobs to be killed and the user will be automatically logged off.*

/dev/dpr/i1a

The "Transparent Print Device" (DigiPRINT), for use with the auxiliary printer port of a terminal. Output directed to a **dpr** device goes out the auxiliary port of a terminal while the user continues to use the terminal normally.

Transparent print devices allow you to use your terminal and a local printer connected to the terminal's auxiliary port at the same time. The system assumes data you send to the **dpr** device is destined for the printer. Before sending data to the printer, the system sends a special control sequence to the terminal to activate the printer port, sends the data, then turns the printer port back off again before sending more data for the terminal.

For correct operation, several parameters must be provided to the *ditty* program. It is usually best to set these up in **/etc/rc2.d**, since they must be set each time the system is rebooted. See *Setting Terminal Options with ditty* (page 18) and *DigiPRINT Transparent Print Option* (page 22).

An example setup line in one of the **/etc/rc** files might be:

```
ditty term wyse60 maxcps 80 bufsize 2000 maxchar 50 /dev/dty/i1a
```

Setting Terminal Options with ditty

ditty is a utility program that sets and displays the terminal options for the DigiBoard concentrators and/or PORTS modules.

The **ditty** command must be run each time the machine is booted. Usually, the best way to do this is by adding **ditty** commands to your **etc/rc** system initialization file. (Put them in a text file in the directory **/etc/rc2.d**—See your System Administrators Guide for details.) Alternatively, you may include the **ditty** command sequence in your **.login** or **.profile** files. Your System Administrator can help you edit these files. The pathname for **ditty** is **/usr/bin/ditty**.

The format is:

ditty [**-a**] [**-n** *ttyname*] [*option(s)*] [*ttyname*]

With no options, **ditty** displays all DigiBoard special driver settings, modem signals, and all standard parameters displayed by **stty(1)** for the tty device referenced by standard input.

Command options are provided to change flow control settings, set transparent print options, force modem control lines, and display all tty settings. Any unrecognized options are passed to **stty(1)** for interpretation.

The options are:

-a — Display all of the unique DigiBoard option settings, as well as all of the standard tty settings reported by **stty -a**.

-n ***ttyname*** — Set and display options for the given tty device, instead of standard input. This option may be specified multiple times to perform the same operation on multiple devices.

ttyname — Set and display options for the given tty device, instead of standard input. *ttyname* can be the full pathname (e.g. **/dev/dty/i1a**), or a simplified pathname where the reference to **/dev** is implied (e.g. **dty/i1a**). This option may be used on a modem control line when no carrier is present.

The following options specify transient actions to be performed immediately:

break Send a 250 MS break signal out on the tty line.

flush Immediately flush (discard) tty input and output.

flushin Flush tty input only.

flushout Flush tty output only.

The following options are "sticky"—the effects continue until the system is rebooted or until the options are changed.

[-]fastbaud Alter the baud rate tables, so 50 baud becomes 57,600 baud, 75 baud becomes 76,800 baud and 110 baud becomes 115,200 baud.

[-]forcedcd Disable [re-enable] carrier sense, so the tty may be opened and used even when carrier is not present.

[-]altpin Switches the function of the DSR and the DCD inputs on the interface connector, so that DCD is available when using an 8-pin RJ-45 connector instead of a 10-pin RJ-45 connector.

maxcps *n* Sets the maximum Characters Per Second (CPS) rate at which characters are output to the transparent print device. The rate chosen should be just below the average print speed. If the number is too low, printer speed will be reduced. If the number is too high, the printer will resort to flow control, and user entry on the terminal will be correspondingly impaired. Default is 100 CPS.

maxchar *n* Sets the maximum number of transparent print characters the driver will place in the output queue. Reducing this number increases system overhead; increasing this number delays operator keystroke echo times when the transparent printer is in use. Default is 50 characters.

bufsize *n* Sets the driver's estimate of the size of the transparent printer's input buffer. After a period of inactivity, the driver bursts this many characters to the transparent printer before reducing to the **maxcps** rate selected above. Default is 100 characters.

onstr *s* Sets the terminal escape sequence to turn transparent printing on. The string *s* can be composed of standard ASCII printing and non-printing characters; control (non-printing) characters must be entered by their octal values, and must consist of three digits preceded by a back-slash ("\") character. For example, the "Escape" character <Esc>, 33 octal, should be entered as "**\033**". Thus, if transparent printing is turned on by the string "**<Esc>[5i**" (ANSI standard), *s* should be entered as "**\033[5i**".

offstr *s* Sets the terminal escape sequence to turn transparent printing off. See "**onstr**", above, for the format of the string *s*.

term *t* Sets the transparent printer on/off strings to values found in the internal default table. Internal defaults are used for the following terminals: **adm31**, **ansi**, **dg200**, **dg210**, **hz1500**, **mc5**, **microterm**, **multiterm**, **pcterm**, **tvi**, **vp-a2**, **vp-60**, **vt52**, **vt100**, **vt220**, **wyse30**, **wyse50**, **wyse60**, or **wyse75**. If the terminal type is not found in the internal default table, then **ditty** reads the **termcap** file and sets transparent print on/off strings to values given by the **po/pf** attributes found there.

The following options specify actions which are *not* "sticky," meaning that the changes are reset when the device is closed, and that the device will use the default values the next time it is opened.

stopout Stop output exactly as if an **xoff** character was received.

startout Restart stopped output exactly as if an **xon** character was received.

stopin Activate flow control to stop input.

startin Release flow control to resume stopped input.

[-]dtr Raise [drop] the DTR modem control line, unless DTR hardware flow control is selected.

[-]rts Raise [drop] the RTS modem control line, unless RTS hardware flow control is selected.

[-]rtspace Enable [disable] RTS hardware input flow control, so RTS drops to pause remote transmission.

[-]ctspace Enable [disable] CTS hardware output flow control, so local transmission pauses when CTS drops.

[-]dsrpace Enable [disable] DSR hardware output flow control, so local transmission pauses when DSR drops.

[-]dcdpace Enable [disable] DCD hardware output flow control, so local transmission pauses when DCD drops.

[-]dtrpace Enable [disable] DTR hardware input flow control, so DTR drops to pause remote transmission.

Also see **stty(1)**, **ioctl(2)**, **termio(7)**, and **termcap(5)**.

Description & Theory of Operation

Most terminals have an auxiliary port that can be connected to a serial printer. These terminals support two print modes, Auxiliary and Transparent. If both print modes are OFF, data received by the terminal is simply displayed on the screen. With Auxiliary print mode ON, data received by the terminal is displayed on the screen, and is also transmitted to the printer. With Transparent Print Mode ON, the terminal transmits data received directly to the printer, without displaying it on the screen.

DigiPRINT allows you to use your terminal in a normal manner, while information is also being sent *over the same serial connection from the host* to the printer connected to the terminal's auxiliary printer port. This is "transparent printing." The DigiPRINT software determines whether packets of data are bound for the screen or for the printer, and precedes data bound for the printer with the Transparent Print Mode ON command, and follows it with the Transparent Print Mode OFF command.

Data for the terminal screen has the highest priority, and DigiPRINT sends data to the printer only if there is a break in information being sent to the screen. If continuous data is being transmitted to the terminal device, nothing gets sent to the printer.

Whenever an auxiliary printer port is used, flow control to the printer becomes an issue. If the printer falls behind and invokes flow control, output to both the printer and the terminal is stopped: this is aggravating to the terminal user. The **ditty** command provides three parameters to limit printer output and avoid this situation. (See *Setting Terminal Options with ditty* on page 18 for a complete description of the **ditty** command.)

The parameter **maxcps** limits the maximum printer port character-per-second data rate. This number should be set to the minimum character rate the printer can sustain in typical use.

The parameter **maxchar** limits the number of characters queued to the printer ahead of terminal output. Lower numbers increase system overhead, higher numbers result in keystroke echo delays. A value of 50 is generally a good compromise at 9600 baud.

The parameter **bufsize** should be set to a value just below the printer's buffer size. After a period of inactivity, the driver will burst up to this many characters to the printer to fill the print buffer before slowing to the maxcps rate.

The printer on/off strings are also set using **ditty**.

DigiPRINT will be available for use after the DigiBoard device driver software for your operating system is installed, and the transparent print options are activated with the **ditty** program.

A cable must be connected between the auxiliary port of the terminal and the printer. The baud rate on the terminal auxiliary port and the printer must be the same, and the printer and the auxiliary port of the terminal must use the same handshaking mode. The auxiliary port must also be enabled. If your terminal is not one of those directly supported, you must know the escape sequence of your terminal.

Refer to your terminal and printer manuals for connection information, escape codes, and to see what handshaking modes are supported (i.e. XON/XOFF, busy/ready, RTS/CTS, etc.).

Printer devices (**dpr/ila**, etc.) must not be in **/etc/inittab**, and must not be enabled.

Transparent Print Activation

DigiPRINT is activated with **ditty**. The **ditty** program configures the DigiBoard tty device driver for transparent print options. (See *Setting Terminal Options with ditty* for a complete description of the **ditty** command.) The **ditty** command must be run each time the machine is booted. Usually, the best way to do this is by adding **ditty** commands to your **etc/rc** system initialization file. (Put them in a text file in the directory **/etc/rc.d/8**—See your System Administrators Guide for details.) Alternatively, you may include the **ditty** command sequence in your **.login** or **.profile** files, to ensure that DigiPRINT is activated when you log in. Your System Administrator can help you edit these files. The pathname for **ditty** is **/usr/bin/ditty**.

DigiPRINT transparent print Options are set using the **ditty** program in the following manner:

ditty [**-a**] [*option(s)*] *port*

The command line options are:

maxcps *n* — Sets the maximum Characters Per Second (CPS) rate at which characters are output to the transparent print device. See *Setting Terminal Options with ditty* (page 18) for more information.

maxchar *n* — Sets the maximum number of transparent print characters the driver will place in the output queue. See *Setting Terminal Options with ditty* for more information.

bufsize *n* — Sets the driver's estimate of the size of the transparent printer's input buffer. See *Setting Terminal Options with ditty* for more information.

onstr *s* — Sets the terminal escape sequence to turn the transparent printer on. An arbitrary octal character *xxx* may be given as *****xxx*.

offstr *s* — Sets the terminal escape sequence to turn the transparent printer off. An arbitrary octal character *xxx* may be given as *****xxx*.

term *t* — Specifies the terminal type. See *Setting Terminal Options with ditty* for more information.

port — Specifies the tty device.

Now data can be sent to your printer via the printer devices.

Example: **cat** *filename* **> /dev/dpr/i1a**

ditty Examples

Example 1:

The following command configures the DigiPRINT options for a DEC VT100 terminal connected to **/dev/dty/i1a** (note that the printer uses **/dev/dpr/i1a**). **maxcps**, **maxchar** and **bufsize** are left to defaults. Enter:

```
ditty term vt100 /dev/dty/i1a
```

Example 2:

The following example uses **onstr** and **offstr** arguments (this sets the terminal to use ANSI Standard). Again **maxcps**, **maxchar**, and **bufsize** are defaults. Enter:

```
ditty onstr "\033[5i" offstr "\033[4i" /dev/dty/i1a
```

Example 3:

This example command sets the DigiPRINT option for a WYSE30 terminal, with **maxcps** of 75, a **maxchar** of 100, and a printer buffer size, **bufsize**, of 1000. Enter:

```
ditty term wyse30 maxcps 75 maxchar 100 bufsize 1000 /dev/dty/i1a
```

DigiBoard Port Authority

Introduction

The DigiBoard Port Authority (DPA) program is a software tool which provides a means to monitor the status of the Front End Processor/Operating System (FEP/OS), which is the on-board software run by DigiBoard intelligent serial communications products.

The DPA software also shows the status of the individual ports on a DigiBoard intelligent serial board. Each of the eight supported RS-232 signals is displayed, along with input and output flow control status.

The DPA program can be run from any terminal on the system, and supports the following DigiBoard hardware and device drivers:

Hardware: DigiBoard PC/Xe and PC/Xi boards
Software: Solaris X86 device driver, version 1.0.0 and up

The DPA software is installed automatically when you install this device driver.

Note: Be sure to read the *Release Notes* that may be included with this software. The *Release Notes* contain information not available at this manual's press time.

Using the DPA Software

To run the DigiBoard Port Authority software, enter the following command from any terminal, or the system console:

`dpa` [**`-l`** *logfile*]

The **`-l`** *logfile* option specifies the file path for screen dumps. If this option is not specified, the default log file path is **`/tmp/dpalog`**.

DigiBoard Port Authority is fully documented in context-sensitive help screens.

Error Messages

The following error messages are generated by the driver:

WARNING: pcxx: No board found at port 0x00000320 <1>.

or

WARNING: pcxx at port 0x00000320 not resetting <3>.

What it means:

The board doesn't respond to reset.

Action to take:

Make sure the board is fully seated in the computer's slot.
Make sure the board's jumper settings and the device driver's settings match.
Potential hardware problem.

WARNING: pcxx: memory allocation error: port 0x00000320 <2>.

What it means:

The operating system would not allocate memory to the driver.

Action to take:

Adjust kernel resources.

WARNING: pcxx: No memory at 0x000D0000 <4>.

What it means:

The driver cannot read the board's dual ported memory; or the board failed a test of its memory.

Action to take:

Make sure the board is fully seated in the computer's slot.
Make sure the board's settings and the device driver's settings match.
Make sure the jumpers or DIP switches are in their proper positions.
Make sure that there is no memory contention with other devices.
Potential hardware problem.

WARNING: pcxx: port 0x00000320 failed diagnostics <5>.

or

WARNING: pcxx: port 0x00000320: BIOS failed <6>.

What it means:

The driver encountered an error executing on-board BIOS.

Action to take:

Make sure the jumpers or DIP switches are in their proper positions.
Make sure there is no memory contention with other devices.
Potential software problem.
Potential hardware problem.

WARNING: pcxx: FEPOS failed <7>.

What it means:

The driver encountered an error executing on-board FEP BIOS.

Action to take:

Make sure the jumpers or DIP switches are in their proper positions.
Make sure there is no memory contention with other devices.
Potential software problem.
Potential hardware problem.