# Installation Guide

**DigiBoard PC/Xe, PC/Xi, MC/Xe and MC/Xi**
**Intelligent Communications Boards**

**SCO XENIX**

DBI 92000155B

**DigiBoard**® is a registered trademark of Digi International Inc. **DigiWARE**™, **DigiPrint**™, **DigiScreen**™, **PC/Xi**™, **PC/16i**™, **PC/8i**™, **PC/Xe**™, **PC/8e**™, **PC/4e**™, **PC/2e**™, **MC/Xi**™, **MC/16i**™, **MC/8i**™, **MC/4i**™, **MC/Xe**™, **MC/8e**™, **MC/4e**™ and **MC/2e**™ are trademarks of Digi International Inc. All other brand and product names are the trademarks of their respective holders.

**Digi International Inc.**
6400 Flying Cloud Drive
Eden Prairie, MN 55344

Phone (612) 943-9020
FAX (612) 943-5398
BBS (612) 943-0550

**DigiBoard GmbH**
Domkloster 1
50667 Köln
Germany

Phone +49 221 920520
FAX +49 221 9205210
BBS +49 221 9205211

Information in this document is subject to change without notice and does not represent a commitment on the part of DigiBoard.

DigiBoard provides this document “as is,” without warranty of any kind, either expressed or implied, including, but not limited to, the particular purpose. DigiBoard may make improvements and/or changes in this manual or in the product(s) and/or the program(s) described in this manual at any time.

This product could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes may be incorporated in new editions of the publication.

# Table of Contents

Introduction .......... 4

Hardware Installation .......... 5
- Supported Boards .......... 5
  - PC/Xi Boards and Older PC/Xe Boards .......... 5
    - I/O Port Selection .......... 5
    - Memory Start Address Selection .......... 6
    - IRQ Selection .......... 6
    - Preset Configuration Options .......... 6
  - 8K PC/Xe Boards .......... 7
  - Micro Channel Boards .......... 7

Software Installation .......... 8
- Part One:  Linking a New Kernel .......... 9
  - PC/Xe and PC/Xi Boards .......... 9
  - MC/Xe and MC/Xi Boards) .......... 15
- Part Two:  Creating New Devices .......... 19
- Part Three:  Installing the New Kernel .......... 21
- Part Four:  Enabling the New Ports .......... 23
  - Serial Printers .......... 25
- Un-Installing the Device Driver .......... 26

Setting Terminal Options with ditty .......... 28

DigiPrint Transparent Print Option .......... 32
- Description & Theory of Operation .......... 32
- Transparent Print Activation .......... 33

DigiScreen Multiple Screen Utility .......... 36
- Description and Theory of Operation .......... 36
- Options .......... 37
- Using DigiScreen .......... 38
- Dynamic Screen Assignment .......... 39

DSINFO DigiScreen Information File .......... 40
- Description .......... 40
- Entry Format .......... 40

Error Messages .......... 45

# Introduction

The DigiBoard device driver software for SCO XENIX is an installable device driver; the software development system is *not* required to create a new kernel. The link kit, however, must be installed.

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 31)
- DigiPrint transparent printing (page 36)
- DigiScreen, DigiBoard's multiple screen utility (page 40)

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, PC/Xi, MC/Xe and MC/Xi families of intelligent serial communications boards. These boards are sometimes collectively referred to as PC/Xx and MC/Xx 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.

**Note:** 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.

**Important!** 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.

### *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

## 8K 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.

## Micro Channel Boards

MC/Xi and MC/Xe boards are configured by the setup program on the IBM reference diskette. The XENIX device driver automatically takes the board parameters from the POS (Programmable Option Select) registers.

**Important!** *Be sure to write down the I/O and memory start addresses chosen for your board(s). You will be asked for them during the software installation.*

# Software Installation

Installation of the device driver software for SCO XENIX is a four-part procedure. In **Part One**, a new kernel is linked. In **Part Two**, the new devices are created. In **Part Three**, a back-up copy of the current kernel is made and the new kernel is copied into the root directory. Finally in **Part Four** you boot the new kernel and enable the communications ports for use with terminals, modems, etc.

**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.

# Part One:  Linking a New Kernel

## PC/Xe and PC/Xi Boards

(For MC/Xe and MC/Xi boards, skip to page 17)

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

**2.** Insert the DigiWARE diskette and enter:

```
custom
```

**3.** The system will now display the Custom menu.  If the device driver has been previously installed, the Custom menu lists **DigiWARE**.  Select this. If not, select **Option 4** to add a supported product.  The system will display:

```
Installing custom data files...

Insert distribution volume 1,
and press <Return> or enter q to quit:
```

**4.** Press <Enter> and the system will display:

```
1.      Install one or more packages
2.      Remove one or more packages
3.      List the available packages
4.      List the files in a package
5.      Install a single file
6.      Select a new set to customize
7.      Display current disk usage
8.      Help

Select an option or press q to quit:
```

Select Option 1 to install the DigiWARE software.  The system will display:

```
Name  Inst  Size  DigiWARE for PC/X* Board packages
-----------------------------------------------------------------
PCXX  NO    992   DigiBoard PC/X* async driver V5.2.0


Enter the package(s) to install
or enter q to return to the menu:
```

Enter:

```
pcxx
```

The system will then display:

```
Insert DigiWARE package volume 1 and press <return> or enter q to return to
the menu:
```

Volume 1 should still be in the diskette drive. Press <Enter>, and the system will display:

```
Extracting files...
```

**5.** You will see the following (or similar) screen:

```
           Copyright (c) 1988-93 DigiBoard Inc.     All Rights Reserved
RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the Government
is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the
Rights in Technical Data and Computer Software clause at DFARS 252.227-7013.
Digi International Inc. d/b/a DigiBoard, 6400 Flying Cloud Dr., Eden
Prairie, MN  55344.

This script installs the DigiBoard PC/X* async driver for:
        SCO XENIX System V Releases 2.2.3 - 2.3.4

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)?
```

Select your configuration from the above list.

*Note that options 1-4 are for installing single boards only*. If you are installing two, three or four boards, or wish to set up one board with parameters different from those above, choose option #5.

For any of the options selected, the corresponding DIP switches on the board must be set to match the parameters listed in the configurations above.

**Note:** If you selected option #5, the next prompts you see will be for Custom Configuration. If you answered 1-4, skip to step 9 on page 18.

**6.** The system 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 DigiBoard PC/X* boards do you wish to install (1-4)?
```

Enter the number of boards you are installing.

**7.** You will now see:

```
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)?
```

Enter the address that is set on the DIP switches on board #1.

**8.** Next, you need to specify a starting address for the dual-ported memory on your board. If you are installing multiple boards, the same address may be used for all boards that are supported by this driver. You will see the screen:

```
      The memory address choices, in hexadecimal, are:

Below 1 Meg:                              |   Above 1 Meg:
                                          |
  08)  0x00080000      0D)  0x000D0000    |      D0) 0x00D00000
  09)  0x00090000      0D2) 0x000D2000 *  |      D8) 0x00D80000
  0C)  0x000C0000      0D4) 0x000D4000 *  |      E0) 0x00E00000
  0C2) 0x000C2000 *    0D6) 0x000D6000 *  |      E8) 0x00E80000
  0C4) 0x000C4000 *    0D8) 0x000D8000 *  |      F0) 0x00F00000
  0C6) 0x000C6000 *    0DA) 0x000DA000 *  |      F8) 0x00F80000
  0C8) 0x000C8000 *    0DC) 0x000DC000 *  |
  0CA) 0x000CA000 *    0DE) 0x000DE000 *  |
  0CC) 0x000CC000 *    0E)  0x000E0000    |
  0CE) 0x000CE000 *

   (Addresses marked with * are for use
    only with newer 8K window PC/Xe boards.)

             Please enter selection code (08-F8)?
```

If you are installing a PC/Xi board or an older PC/Xe board (two banks of DIP switches on the top edge of the board), enter the address that you set on the board’s DIP switches during hardware installation.

If you are installing a newer PC/Xe board (8K boards—these have one bank of DIP switches on the end of the board), choose an address from the list, and the device driver will program the board for that address during system initialization.

**Important!**

If you are installing a combination of PC/Xi or older PC/Xe boards and PC/Xe 8K boards, choose an address that is *not* marked with an asterisk (*). This way, all boards can use the same address.

**9.** You will now be asked for the number of ports on the board. Enter “**2**”, “**4**”, “**8**” or “**16**”, as appropriate.

*If you are installing multiple boards, steps 7, 8 and 9 will be repeated for each board.*

**10.** You will now be shown the configuration you have chosen. Your screen will look similar to this:

```
You have selected the following configuration:

        Board   I/O Address   Memory Address   Ports
        --------------------------------------------
          1       0x100        0x000D0000       2
          2       0x110        0x000D0000       16
          3       0x300        0x000D0000       8
          4       0x320        0x000D0000       4

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

If the information is correct, answer “**y**”. Otherwise, answer “**n**”, and you will be returned to step 5.

**11.** Next, you need to select the **altpin** setting. This allows alternate wiring of the RJ-45 modular connectors. The default for **altpin** is OFF, giving you the standard DigiBoard RJ-45 pinouts. Setting **altpin** to ON enables the alternate RJ-45 pinouts. (This is useful for 8-pin RJ-45 connectors. See the *Installation Guide* for your board, and page 34 of this manual for more details.)

The software asks:

```
The default setting for altpin is off.
Is this acceptable (y or n)?
```

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

Note that answering “**n**” here activates the **altpin** configuration for *all* ports. Individual ports can be set with **ditty** (see page 33).

**Note:** **altpin** should be activated *only* on boards equipped with RJ-45 connectors.  If your board is equipped with DB-25 or DB-9 connectors, **altpin** should be left inactive (off).

### 12. Installing DigiScreen

DigiScreen is DigiBoard’s multiple screen utility that allows a single physical terminal to be connected to several virtual terminal sessions (*screens*) at one time. The software installation script gives you the option of installing it.

*If you have already installed DigiScreen with another DigiBoard product, there is no need to install it again, unless you wish to change the number of DigiScreen devices.*

The software asks:

```
Do you wish to install DigiScreen (y or n)?
```

If you answer “**n**”, proceed to page 21, *Creating New Devices*.

**13.** The software continues:

```
Each session under DigiScreen requires a DigiScreen device.

How many DigiScreen devices do you wish to have installed?
(max. is 256, default is 32)
```

Answer with the number of DigiScreen devices you think will be adequate for all connected terminals. Note that the default value will change according to the number of previously created DigiScreen devices you may have. See page 40 for a complete description of the DigiScreen multiple screen utility.

**14.** The software now asks:

```
Do you wish to make new device nodes (y or n)?
```

Most users will need to create the new devices at this time, and should answer y and proceed to **Part Two: Creating New Devices**.

If you are an experienced XENIX user, you have the option of skipping Part Two (you will then need to create the device nodes manually). If you answer n, you will proceed to **Part Three: Installing the New Kernel**.

## MC/Xe and MC/Xi Boards

(For PC/Xe and PC/Xi boards, see page 9)

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

**2.** Insert the DigiWARE diskette and enter:

```
custom
```

**3.** The system will now display the Custom menu. If the device driver has been previously installed, the Custom menu lists **DigiWARE**. Select this. If not, select **Option 4** to add a supported product. The system will display:

```
Installing custom data files...

Insert distribution volume 1,
and press <Return> or enter q to quit:
```

**4.** Press <Enter> and the system will display:

```
1.       Install one or more packages
2.       Remove one or more packages
3.       List the available packages
4.       List the files in a package
5.       Install a single file
6.       Select a new set to customize
7.       Display current disk usage
8.       Help

Select an option or press q to quit:
```

Select Option 1 to install the DigiWARE software. The system will display:

```
Name  Inst  Size  DigiWARE for PC/X* Board packages
-----------------------------------------------------------------
MCXX  NO    992   DigiBoard MC/X* async driver V5.2.0


Enter the package(s) to install
or enter q to return to the menu:
```

Enter:

**`mcxx`** (for MC/Xe or MC/Xi boards)

The system will then display:

```
Insert DigiWARE package volume 1 and press <return> or enter q to return to
the menu:
```

Volume 1 should still be in the diskette drive. Press <Enter>, and the system will display:

```
Extracting files...
```

**5.** You will see the following (or similar) screen:

```
Each DigiBoard MC/X* board requires an I/O port address and an 8K to 128K
block of memory. Please select these to avoid conflicts with other boards
you may have installed in your system.

How many DigiBoard MC/X* boards do you wish to install (1-4)?
```

Enter the number of boards you are installing. Note that boards are installed by slot number; the board closest to slot 0 is board number 1, and so on.

**6.** You will now be asked for the number of ports on the board. Enter “**`2`**”, “**`4`**”, “**`8`**” or “**`16`**”, as appropriate.

*If you are installing multiple boards, step 6 will be repeated for each board.*

**7.** You will now be shown the configuration you have chosen. Your screen will look similar to this:

```
You have selected the following configuration:

        Board   I/O Address   Memory Address   Ports
        -------------------------------------------
          1       0x328         0x000D0000       2
          2       0xF1F0        0x00FA0000       16
          3       0xF2F0        0x00FC0000       8
          4       0x308         0x000D2000       4

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

If the information is correct, answer "`y`". Otherwise, answer "`n`", and you will be returned to step 5.

**8.** Next, you need to select the **altpin** setting. This allows alternate wiring of the RJ-45 modular connectors. The default for **altpin** is OFF, giving you the standard DigiBoard RJ-45 pinouts. Setting **altpin** to ON enables the alternate RJ-45 pinouts. (This is useful for 8-pin RJ-45 connectors. See the *Installation Guide* for your board, and page 34 of this manual for more details.)

The software asks:

```
The default setting for altpin is off.
Is this acceptable (y or n)?
```

Answer "`y`" or "`n`" as appropriate.

Note that answering "`n`" here activates the **altpin** configuration for *all* ports. Individual ports can be set with **`ditty`** (see page 33).

**Note:** **altpin** should be activated *only* on boards equipped with RJ-45 connectors. If your board is equipped with DB-25 or DB-9 connectors, **altpin** should be left inactive (off).

### 9. Installing DigiScreen

DigiScreen is DigiBoard's multiple screen utility that allows a single physical terminal to be connected to several virtual terminal sessions (*screens*) at one time. The software installation script gives you the option of installing it.

*If you have already installed DigiScreen with another DigiBoard product, there is no need to install it again, unless you wish to change the number of DigiScreen devices.*

The software asks:

```
Do you wish to install DigiScreen (y or n)?
```

If you answer "**n**", proceed to page 21, *Creating New Devices*.

**10.** The software continues:

```
Each session under DigiScreen requires a DigiScreen device.

How many DigiScreen devices do you wish to have installed?
(max. is 256, default is 32)
```

Answer with the number of DigiScreen devices you think will be adequate for all connected terminals. Note that the default value will change according to the number of previously created DigiScreen devices you may have. See page 40 for a complete description of the DigiScreen multiple screen utility.

**11.** The software now asks:

```
Do you wish to make new device nodes (y or n)?
```

Most users will need to create the new devices at this time, and should answer y and proceed to **Part Two: Creating New Devices**.

If you are an experienced XENIX user, you have the option of skipping Part Two (you will then need to create the device nodes manually). If you answer n, you will proceed to **Part Three: Installing the New Kernel**.

# Part Two:  Creating New Devices

This portion of the installation procedure creates devices in **`/dev`** and edits **`/etc/ttys`**, giving the operating system the information it needs to use the additional ports.  By default, these ports are installed with device names **`ttyi1a`** through **`ttyi1p`**, **`ttyi2a`** through **`ttyi2p`**, **`ttyi3a`** through **`ttyi3p`** and **`ttyi4a`** through **`ttyi4p`**.

The device driver will support up to 4 boards, for a maximum of 64 asynchronous ports.

The software displays:

```
This script also installs the information needed by XENIX to use these
additional ports.  The DigiBoard PC/X* async driver will support up to
four boards for a maximum of sixty-four asynchronous serial ports.

By default, the ports are installed as ttyi1a through ttyi1p.
Ports with modem control are installed as ttyi1A through ttyi1P.
DigiPrint ports are installed as pri1a through pri1p.
DigiPrint ports with modem control are installed as pri1A through pri1P.

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

Answer "**`y`**" if the default device names are acceptable.  You may then proceed to **Part Three: Installing the New Kernel**.

If you answer "**`n`**", you will see the following:

```
Device names will be a base name with the board number (1-4)
and the port number (a-p) appended.

Please enter base name to use:
```

Enter the new base name.

You will then be asked for an alternative DigiPrint printer base name (the default is **`pri1a`** through **`pri4p`**).  Enter the new DigiPrint base name.

Now proceed to **Part Three: Installing the New Kernel**.

The devices are named according to the following conventions:

Format used for port names:

**ttyi***bp*, **pri***bp*

**ttyi** Default base name. You may define a different one.

**pri** Default base name for DigiPrint transparent printing.

*b* Board number (1-4).

*p* Port identifier:

**a-p** (lower case) are used for non modem control ports. These ports do *not* require DCD (Data Carrier Detect) to be asserted before they can transmit data. These ports are used primarily for directly connected terminals and printers.

**A-P** (upper case) are used for modem control ports. These ports require DCD (Data Carrier Detect) to be asserted before they can transmit data. These ports are used primarily for modems.

For example, if “**ttyi**” is the base name, Port 4 Board 2 will be **ttyi2d**. The same port with modem control enabled would be named **ttyi2D**.

# Part Three:  Installing the New Kernel

The driver installation is now complete, and the new kernel must be installed.

**1.** You will see:

```
The new kernel with driver modifications is in /usr/sys/conf/xenix.

Do you want this kernel to boot by default (y or n)?
```

If you enter “**y**”, you will see:

```
The old kernel is installed in /xenix.old

The new kernel is installed in /xenix

Enter the following command to reboot your system and
activate the new kernel:

shutdown 0   <cr>
```

If you enter “**n**”, you will see:

```
Changes will not be reflected unless

/usr/sys/conf/xenix is copied to /xenix.
```

**2.** The system displays:

```
1.       Install one or more packages
2.       Remove one or more packages
3.       List the available packages
4.       List the files in a package
5.       Install a single file
6.       Select a new set to customize
7.       Display current disk usage
8.       Help

Select an option or press q to quit:
```

Enter “**q**”.

**3.** If you chose to have the new kernel boot by default, enter the following command to boot the new kernel:

```
shutdown 0
```

**4.** When the computer reboots, you should see messages which indicate the board is functioning. Your screen will look something like this:

```
device    address     vector   dma       comment
------------------------------------------------------------
%pcxx     0x320         -       -     mem=0x000D0000
                                      Ports=8
                                      PC/Xe V x.x.x
```

If there is a problem with the board or the driver configuration, you may see error messages after this screen—see page 51 for explanations of the error messages that may occur.

Watch for error messages for up to a minute after this screen appears—if there is a memory contention with some other device, such as a SCSI controller, it may not manifest immediately (it will only show up when both devices attempt to access the same memory addresses at the same time). See the appendix on memory conflicts in the appropriate Board hardware *Installation Guide* for more information about memory problems.

Now proceed to Part Four, beginning on the next page, to enable the new ports.

# Part Four:  Enabling the New Ports

**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/ttyi1a
```

(Assuming the terminal is connected to **ttyi1a**.)

*(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 (the default parameters are 9600 baud, 8 data bits, 1 stop bit, and no parity).

Another test you can perform will verify that a port can both transmit and receive data:

Connect the port's transmit and receive lines together (pins 2 and 3 of a DB-25 connector, or the middle two pins of an RJ-45 connector), and enter the following commands:

```
cat </dev/ttyi1a &
cat /etc/termcap > /dev/ttyi1a
```

The first command runs in the background, and directs all input from **ttyi1a** to **stdout** (your screen).  The second command transmits the **termcap** file to **ttyi1a**.  If the port is working, the data is sent out the port, received back by the same port and displayed on your screen.

If the port fails, check another port.  If more than one port fails, the problem is probably due to memory contention, rather than defective hardware.  Try reinstalling the device driver with a different memory start address.

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

**Important!**

If the port tests above work, but you still can’t respawn a **`getty`**, it is probably because DCD is not asserted.

Remember that modem controlled ports (where the port ID is a capital letter; e.g. **`ttyi1A`**) must have Data Carrier Detect (DCD) asserted before they can function. If DCD is not connected to an active output line of the peripheral device, it must be defeated, either by “tying it high” (e.g. connecting it to the same port’s DTR line) or by issuing the command **`ditty forcedcd ttyi1A`** (assuming that the port in question is **`ttyi1A`**).

Another way to bypass the DCD requirement is to use non modem control devices (where the port ID is a small letter; e.g. **`ttyi1a`**). Non modem control devices function identically to modem control devices, except that they do not require DCD to be asserted in order to function.

**2.** To activate a port for use with a terminal, enter the following command:

```
enable ttyi1a
```

(If you’ve assigned a new base name, replace the “tty” above with the name you chose.)

The above command will cause a login prompt to be sent to the terminal connected to **`ttyi1a`**. To activate the other ports, repeat the enable command with the appropriate port name.

## Serial Printers

To prepare a port for connection to a printer, enter the following command:

**`ditty printer`** [*baud rate and flow control options*] **`ttyi1a`**

This command forces the specified port to stay open, thus preserving any "non-sticky" options (see *Setting Terminal Options with ditty*, beginning on page 31, for complete descriptions of all **`ditty`** parameters) for that port.

*This command is recommended for any port that has a printer connected to it.*

The command may be placed in **`/etc/rc.d/8/userdef`** so that it is automatically executed whenever the system is booted.

EXAMPLE:

The command shown below results in 9600,8,1,N, with software flow control.

```
ditty printer ixon opost onlcr ttyi1a
```

# Un-Installing the Device Driver

The DigiBoard device driver software can be *removed* from the kernel by using the program custom. Again, the Development Kit is not required, although the link kit must be installed.

Removing the driver from the kernel is a two-part process. First, the driver references are removed from the system configuration files. Secondly, a new kernel is linked, and copied to the root directory.

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

**1.** Log onto the console as super-user (root), and enter the following command:

```
custom
```

**2.** When the operating system menu asks, choose **REMOVE**.

**3.** Select **`pcxx`** or **`mcxx`**.

**4.** Select the DigiBoard driver. The software will respond with:

```
Removing DigiBoard PC/XX intelligent async serial driver.
```

**5.** Next, the software asks:

```
Removing DigiScreen devices will affect ALL DigiBoard Drivers.
Do you wish to remove DigiScreen devices (y or n)?
```

Answer "**`y`**" or "**`n`**".

**6.** The software asks:

```
Do you wish to remove DigiBoard PC/XX entries from /etc/ttytype (y or n)?
```

Answer "**`y`**" if these entries are no longer needed (or if you are in doubt), or answer "**`n`**" if you wish to preserve the old entries.

**7.** You will see:

```
Do you want this kernel to boot by default (y or n)?
```

Answer “**y**”. The system displays:

```
Do you want the kernel environment rebuilt (y or n)?
```

Answer “**y**”. You will see:

```
The kernel has been successfully linked and installed.  To activate it,
reboot your system.
```

**8.** Remove the **perms** file:

```
rm /etc/perms/pcxx
```

**9.** Enter the following command to boot the new kernel:

```
shutdown 0
```

# Setting Terminal Options with ditty

**`ditty`** is a utility program that sets and displays the terminal options for DigiBoard intelligent serial products.

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 ttys. |
| ***ttyname*** | Set and display options for the given tty device, instead of standard input. This form can be used with a tty pathname prefixed by **`/dev/`** or with a simple tty name beginning with **`tty`** (or the basename you selected during installation, if different). 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 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.

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

**`[-]printer`** Forces port to stay open so that non-sticky parameters such as baud rate and flow control are not reset to default values.

**`[-]fastcook`** Perform cooked output processing on the intelligent card to reduce host CPU usage, and increase raw mode input performance.

**`[-]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.

**`[-]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.

**[-]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.

**startc** ***c*** Sets the XON flow control character. The character may be given as a decimal, octal or hexadecimal number. Octal numbers are recognized by the presence of a leading zero, and hexadecimal numbers are denoted by a leading “0x”. For example, the standard XON character, <CTRL-Q>, can be entered as “17” (decimal), “021” (octal) or “0x11” (hexadecimal).

**stopc** ***c*** Sets the XOFF flow control character. The character may be given as a decimal, octal, or hexadecimal number (see **startc**, above, for format of octal and hexadecimal numbers).

**astartc** ***c*** Sets auxiliary XON flow control character. The character may be given as a decimal, octal, or hexadecimal number (see **startc**, above, for format of octal and hexadecimal numbers).

**astopc** ***c*** Sets auxiliary XOFF flow control character. The character may be given as a decimal, octal, or hexadecimal number (see **startc**, above, for format of octal and hexadecimal numbers).

**[-]aixon** Enables auxiliary flow control, so that two unique characters are used for XON and XOFF. If both XOFF characters are received, transmission will not resume until both XON characters are received.

**[-]2200flow** Use 2200 style flow control on the port. The 2200 terminals support an attached printer and use four flow control characters: terminal XON (0xF8), printer XON (0xF9), terminal XOFF (0xFA) and printer XOFF (0xFB).

**[-]2200print**

The **2200print** flag determines how these flow control characters (see **2200flow**, above) are interpreted. If **2200print** is set, run independent flow control for terminal and transparent print devices. Otherwise, terminal and printer flow control are logically tied together, so if either XOFF character is received, all output is paused until the matching XON character is received.

**`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 **`terminfo`** entry for the terminal type and sets transparent print on/off strings to values given by the **mc5/mc4** attributes found there.

Also see **`stty(1)`**, **`ioctl(2)`**, **`termio(4)`**, and **`terminfo(5)`**.

# DigiPrint Transparent Print Option

## 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 31 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 (**pra01**, etc.) must not be in either the **/etc/inittab** or **/etc/ttys** files, 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 31) 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/pri1a**

## ditty Examples for Transparent Printing

**Example 1:**

The following command configures the DigiPrint options for a DEC VT100 terminal connected to **/dev/ttyi1a**. (Note that the printer uses **/dev/pri1a**). **maxcps**, **maxchar** and **bufsize** are left to defaults. Type:

```
ditty term vt100 /dev/ttyi1a
```

**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. Type:

```
ditty onstr "\033[5i" offstr "\033[4i" /dev/ttyi1a
```

**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. Type (all on one line, with a carriage return at the end only):

```
ditty term wyse30 maxcps 75 maxchar 100 bufsize 1000 /dev/ttyi1a
```

# DigiScreen Multiple Screen Utility

(Also see **`dsinfo`**, on page 45.)

## Description and Theory of Operation

DigiScreen is a utility that allows a single physical terminal to be connected to several virtual terminal sessions (*screens*) at one time. It is mainly intended for use with terminals that have two or more pages of screen memory. With such terminals, switching between virtual screens will also switch between physical terminal screen pages, allowing each virtual screen's image to be saved and restored. On terminals without multiple pages of screen memory, DigiScreen can still be used to switch among virtual screen sessions, although the appearance of the screen will not be maintained when switching screens.

**Note:** For full support of DigiScreen, your terminal must be able to switch internal screen pages on command *and must remember the cursor position for each page*. While DigiScreen will work on both smart and dumb terminals, screen images are not saved during screen changes on dumb terminals. DigiScreen also supports terminals connected to two or more computers through separate serial ports.

# Options

DigiScreen is called with the following format:

**`dscreen`** [ **`-i`** *infofile* ] [ **`-t`** *termtype* ]

If *infofile* is specified, it will be used as the source of terminal configuration information. Otherwise, if the environment variable **`DSINFO`** is defined (e.g. by placing a command like **`DSINFO=/etc/my_infofile`** in **`.profile`**), it specifies the name of the file to be used as the source of terminal configuration information. If neither *infofile* nor **`DSINFO`** is specified, the configuration information is read from the file **`/etc/dsinfo`**. This option is used to define a different set of keys to be used with DigiScreen, e.g. when the originally defined DigiScreen keys conflict with an application one wishes to use.

The terminal type is used to select which entry in the *infofile* (default **`/etc/dsinfo`**) is used to describe the terminal. If the desired terminal type does not match the setting of the **`TERM`** environment variable (again, for alternate key mappings), it can be specified as *termtype* with the **`-t`** option.

**Command Line Examples:**

```
dscreen
dscreen -i /etc/my_infofile -t tvi925
```

# Using DigiScreen

When DigiScreen is run, it starts up one virtual screen. Some of the keys on the terminal keyboard will not be passed through to the virtual screen; instead, DigiScreen will intercept these keys and perform certain actions when they are pressed. The actions include select a specific screen, block all input and output, start a new screen, end DigiScreen (exit code 0), quit DigiScreen (exit code 1), switch to the previous screen, and list the DigiScreen keys and what they do. Which function each key performs is dependent upon the terminal and the terminal description in the **`dsinfo`** file.

When a new virtual screen is created, it is assigned to a select key. When this key is pressed, DigiScreen will switch the physical terminal to the video page associated with the particular virtual screen and direct all input and output to go between the physical terminal and the virtual screen. Each virtual screen must have a select key; once all of the select keys defined in the **`dsinfo`** file have virtual screens assigned to them, no more screens may be created. Individual screen sessions will end when the original shell process exits, and this will free the associated select key for use with another virtual screen. DigiScreen exits when there are no more active screens.

Block keys can be used to stop output (in a fashion similar to <Ctrl-S> when using "**`ixon`**" flow control). However, the true purpose of these keys is to allow for transparently setting up terminal sessions on two computers using a terminal that has two serial ports. See **`dsinfo`** (page 45) for more information.

Pressing a new screen key will create a new screen and assign it to one of the select keys, unless one of the necessary resources is exhausted. Each new screen requires a select key as defined in the **`dsinfo`** file, a DigiScreen pseudo terminal device, enough memory for the various structures used to keep track of the screen, and a process to run the shell. If any of these are not available, the new screen operation will fail and print a message indicating the reason for the failure.

Pressing an end key will send a SIGHUP signal to all the screen sessions, clean up, and exit with a status of 0. Pressing a quit key will perform the same actions, but will exit with a status of 1.

Pressing a previous key will switch the terminal to the screen that was last displayed.

Pressing a list key will cause a list of the keys recognized by DigiScreen and their actions to be displayed on the terminal. When DigiScreen starts a new screen, it will display the message “Press *KEY* for help” (where *KEY* is the name of the list key) if there is a list key defined.

# Dynamic Screen Assignment

Normally, the terminal description entry in the **`dsinfo`** file will have the same number of screen selection keys as the terminal has physical screen pages. However, if more screen selection keys are defined than the number of physical screen pages defined, DigiScreen will dynamically assign physical screen pages to virtual screens. When a virtual screen that doesn't have an associated page of screen memory in the terminal is selected, DigiScreen assigns the least recently used physical screen to the virtual screen. When this occurs, some sort of indication is given that the physical screen is connected to a different virtual screen; for instance, the screen may be cleared. Using a terminal that has only one physical screen is the simplest case of this; the one screen is shared between all virtual screens.

Notes:

- Avoid switching screens when the screen is being written to; you may interrupt an escape sequence and leave the terminal in an unknown state.
- Even if your terminal saves the cursor position for individual screens, it may not save other states such as insert mode, inverse video, etc. If this is the case in your situation, make sure you are not in any such mode when you switch screens.

# DSINFO DigiScreen Information File

## Description

**`dsinfo`** is a database of terminal descriptions used by **`dscreen`** (DigiScreen). The information in the descriptions include what keys are to be used by DigiScreen and what functions they perform, how many pages of screen memory the terminal has, and what code sequences are sent/received to use these features.

## Entry Format

Entries in **`dsinfo`** consist of a number of comma separated fields. The first field is a list of alternate names for the terminal, separated by “**`|`**” characters.

The remaining fields are strings describing the capabilities of the terminal to DigiScreen. Within these strings, the following escape codes are recognized:

| | |
|---|---|
| **`\E,\e`** | escape character |
| **`\n,\l`** | newline (a.k.a. linefeed) character |
| **`\r`** | carriage return |
| **`\t`** | tab character |
| **`\b`** | backspace character |
| **`\f`** | formfeed character |
| **`\s`** | space character |
| **`\`***nnn* | character with octal value *nnn* |
| **`^`*x*** | <Ctrl-*x*> for any appropriate *x* |

Any other character preceded by a backslash will yield the character itself. The strings are entered as *type*=*string*, where *type* is the type of string as listed below, and *string* is the string value.

### String Types

The string types are as follows:

**dsk***x* A string type that starts with "**dsk**" describes a key. The type must be four letters long, and the fourth letter *x* indicates what action is taken when the key is received. The key types are:

| Type | Action |
|---|---|
| **dsks** | Switch Screens |
| **dskb** | Block Input and Output |
| **dske** | End DigiScreen |
| **dskq** | Quit DigiScreen (non-zero exit status) |
| **dskc** | Create New Screen |
| **dskp** | Switch to Previous Screen |
| **dskl** | List Keys and Actions |

Any other key type (a string type **dsk***x* that doesn't end in **s**, **b**, **e**, **q**, **p**, or **l**) will cause no internal dscreen action, but will show up in the key listing and will be recognized and acted upon (see below). A type of **dskn** (**n** for No Operation) is guaranteed not to be used for any function in future versions; it is recommended that this be used when no internal dscreen action is desired. The value string for each key has three substrings, which are separated by "**|**" characters (use "**\|**" to include the "**|**" character in one of the substrings). The first substring is the sequence of characters that the terminal sends when the key is pressed. The second substring is a label for the key that is printed when a list of the keys is presented (for example, "Shift-F1"). The third substring is a sequence of characters that DigiScreen sends to the terminal when this key is pressed, before performing the action this key requests.

**dsp** A string type of "**dsp**" describes a physical screen in the terminal. One **dsp** string should be present for each physical screen in the terminal. The value string for each physical screen has two substrings, which are separated by a "**|**" character (again, use "**\|**" to include the "**|**" character in one of the substrings). The first substring should be the sequence of characters to send to the terminal to display and output to the particular physical page on the terminal. The second substring is sent to the terminal any time the page is used for something new. This second substring is usually set to the clear screen sequence. It is sent under two conditions. The first condition is when a new virtual terminal session is being created. The second condition occurs when the user is running more virtual terminals than there are physical screens; if the user selects a virtual terminal such that DigiScreen has to re-use one of the physical screens, it will send this sequence to the screen to indicate to the user that the screen contents don't match the output of the virtual terminal to which it is connected. Note that running with more virtual terminals than physical screens can be quite confusing and is not particularly recommended; it can be avoided by defining no more screen selection keys ("**`dsks=...`**") than physical screens ("**`dsp=...`**") in the **`dsinfo`** entry.

**`dst`** A string with a type of "**`dst`**" adjusts DigiScreen's input timeout. The value of the string should be a decimal number. The timeout value is in tenths of a second, and has a maximum value of 255. The default timeout value is 1 (or .1 seconds). When DigiScreen recognizes a prefix of an input key sequence but doesn't have all the characters of the sequence, it waits for more characters. If the timeout occurs before more characters are received, the characters are sent on to the virtual screen and DigiScreen will not consider these characters as part of an input key sequence. It may be necessary to raise this value if one or more of the "keys" DigiScreen is to trigger on is actually a number of keystrokes (i.e. assigning Ctrl-Z 1, Ctrl-Z 2, Ctrl-Z 3, etc. for screen selection, Ctrl-Z N for new screen and so on).

**Example 1**

The following example entry is for a Wyse 60 with three screens:

```
wy60|wyse60|wyse model 60,
         dsks=^A`^M|Shift-F1|,
         dsks=^Aa^M|Shift-F2|,
         dsks=^Ab^M|Shift-F3|,
         dskc=\200|Ctrl-F1|,
         dske=\201|Ctrl-F2|\Ew0\E+,
         dskl=\202|Ctrl-F3|,
         dsp=\Ew0|\E+,
         dsp=\Ew1|\E+,
         dsp=\Ew2|\E+,
```

With this entry, <Shift-F1> through <Shift-F3> are used for selecting screens 1 through 3, respectively. <Ctrl-F1> will create a new screen, <Ctrl-F2> will send "<ESC> w 0 <ESC> +" to the screen (switching to window 0 and clearing the screen) and then end dscreen, and <Ctrl-F3> will list the keys and their functions. The three physical screens are displayed by sending "<ESC> w 0", "<ESC> w 1", and "<ESC> w 2." Each time a physical screen is used for a new screen, the sequence "<ESC> +" will be sent to the terminal, which will clear the screen.

**Example 2**

This example is, again, for a Wyse 60 with three screens, but one of the screens is on a second computer communicating through the second serial port on the terminal:

```
wy60-1|wyse60-1|wyse model 60 - first serial port,
            dsks=^A`^M|Shift-F1|,
            dsks=^Aa^M|Shift-F2|,
            dskb=^Ab^M|Shift-F3|\Ed#^Ab\r^T\Ee9,
            dskc=\200|Ctrl-F1|,
            dske=\201|Ctrl-F2|\Ed#\201^T\Ew0\E+,
            dskl=\202|Ctrl-F3|,
            dsp=\Ew0|\E+,dsp=\Ew1|\E+,

wy60-2|wyse60-2|wyse model 60 - second serial port,
            dskb=^A`^M|Shift-F1|\Ed#^A`\r^T\Ee8,
            dskb=^Aa^M|Shift-F2|\Ed#^Aa\r^T\Ee8,
            dsks=^Ab^M|Shift-F3|,
            dskc=\200|Ctrl-F1|,
            dske=\201|Ctrl-F2|\Ed#\201^T\Ew0\E+,
            dskl=\202|Ctrl-F3|,
            dsp=\Ew2|\E+,
```

For this setup to work, DigiScreen must be run on both computers, with terminal type **wy60-1** on the first computer and terminal type **wy60-2** on the second computer (using the **-t** option to DigiScreen). The **wy60-1** entry will be examined first.

The first two key entries are unchanged from the original wy60 entry. The third key, however, has type "**dskb**," which means block both input and output. When this key is pressed, the sequence "<ESC> d # <Ctrl-A> b <CR> <Ctrl-T> <ESC> e 9" is sent to the terminal; after this output is blocked and DigiScreen continues scanning input for key sequences but discards all other input.

The effects caused by the sequence sent to the terminal contain the real magic here. The sequence "**<ESC> d #**" puts the terminal in "Transparent Print Mode," which echoes all characters up to a **<Ctrl-T>** out the other serial port. The characters "**<Ctrl-A> b <CR>**" are sent out the other serial port, informing the **dscreen** process on the other computer that it should activate the window associated with the **<Shift-F3>** key. The "**<Ctrl-T>**" takes the terminal out of the Transparent Print mode, and the sequence "**<ESC> e 9**" tells the terminal to switch to the other ("AUX") serial port for data communications.

At this point the other computer takes over and sends an “**<ESC> w 2**” to switch to the third physical screen, and then resumes normal communication.

The wy60-2 entry follows the same general pattern for keys **<Shift-F1>** and **<Shift-F2>**: switch to transparent print mode; send function key string to other computer; switch transparent print off; and switch to the other serial port. The end key (**<Ctrl-F2>**) works the same for both computers; it sends the end key sequence to the other computer through the transparent print mechanism, switches the terminal to window 0, clears the screen, then exits.

# Error Messages

The following error messages are generated by the driver:

```
***WARNING*** No board found at port 0x320, check switch settings.
```

or

```
***WARNING*** PC/Xi at port 0x320 not resetting, check switch settings
```

*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*** Driver memory allocation failed for PC/Xi at port 0x320.
```

*What it means:*

The operating system would not allocate memory to the driver.

*Action to take:*

Adjust kernel resources.

**`***WARNING*** No memory at 0xd8000 for PC/Xe at port 0x320, check switch settings`**

*What it means:*

The driver cannot read the board's dual ported 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.
Make sure you haven't selected an 8K address for a 64K board.
Potential hardware problem.

**`***WARNING*** PC/Xi at port 0x320 failed diagnostics. Error #(N1,N2).`**

*What it means:*

The driver encountered an error executing on-board BIOS. N1 and N2 are status bits at the time of the failure.

*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*** PC/Xi at port 0x320 failed BIOS function #X. Error #(N1,N2).`**

or

**`***WARNING*** FEPOS for PC/Xe at port 0x320 not functioning. Error #(N1,N2).`**

*What it means:*

The driver encountered an error executing on-board FEPOS. X is the BIOS function that failed, N1 and N2 are status bits at the time of the failure.

*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.