



**Digi International Inc.**

11001 Bren Road  
Minnetonka, MN 55343  
(800) 344-4273  
(612) 912-3444

**Digi International GmbH**

Domkloster 1  
50667 Köln  
Germany  
+49 221 920520

**Digi International Asia Pte Ltd**


13-06 Tower 'A'  
391A Orchard Road  
Ngee Ann City  
Singapore 0923  
+65 732 1318

# Software Manual

**MenuPort Interface™ (MPI)**

**SCO UNIX  
UNIX System V Release 4**

DBI 92000154 Rev H

The Digi logo  is a registered trademark of Digi International Inc.

**MenuPort Interface™**, **Avanstar™**, **Digi Port Authority™**, **DigiWARE™**,  
**DigiPRINT™**, **DigiSCREEN™**, **C/X™**, **EPC/X™**, **Xem™**, **C/CON-8™**,  
**C/CON-16™**, **EPC/CON-16™**, **PORTS/16em™**, **PORTS/8em™**,  
**PORTS/8emp™**, **Modem/8em™**, **Modem/4em™**, **PC/Xr™**, **PC/4r™**, **PC/8r™**,  
**PC/Xi™**, **PC/8i™**, **PC/16i™**, **PC/Xe™**, **PC/2e™**, **PC/4e™**, **PC/8e™**, **MC/Xi™**,  
**MC/4i™**, **MC/8i™**, **MC/Xe™**, **MC/2e™**, **MC/4e™**, **MC/8e™**, **SmartPort™**,  
**SmartPort Plus™** and **ClusterPort™** are trademarks of Digi International. All  
other brand and product names are the trademarks of their respective holders.

© Digi International Inc. 1995-1997

All Rights Reserved

<b>Digi International Inc.</b>	Phone	(800) 344-4273
11001 Bren Road		(612) 912-3444
Minnetonka, MN 55343	FAX	(612) 912-4952
	BBS	(612) 912-4800

<b>Digi International GmbH</b>	Phone	+49 221 920520
Domkloster 1	FAX	+49 221 9205210
50667 Köln	BBS	+49 221 9205211
Germany		

<b>Digi International Asia Pte Ltd</b>	Phone	+65 732 1318
13-06 Tower 'A'	FAX	+65 732 1312
391A Orchard Road	BBS	+65 292 7163
Ngee Ann City		
Singapore 0923		

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

Digi provides this document "as is", without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of fitness or merchantability for a particular purpose. Digi 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 .....	5
Supported Products .....	5
Software Installation.....	6
Installing the MenuPort Interface (MPI) Package.....	7
SCO UNIX.....	7
System V, Release 4.....	7
Running the MenuPort Interface (MPI) Program.....	8
Device Names.....	9
SCO UNIX.....	9
System V Release 4.....	14
Enabling the New Ports.....	19
SCO UNIX.....	19
System V Release 4.....	21
Serial Printers.....	23
Loop Back Test .....	24
Diagnostic Loop Back Plug Specifications.....	27
Xe, Xi, C/X, EPC/X, Xr, Xem.....	27
SmartPort, SmartPort Plus and ClusterPort .....	27
Avanstar .....	28
Uninstalling the MPI Software Package .....	29
Removing Device Drivers.....	29
Removing the MPI Software Package .....	29
SCO UNIX.....	29
System V Release 4 .....	30
Setting Terminal Options with ditty .....	31
Using the ditty Command with Parallel Ports .....	38
DigiPRINT Transparent Print Feature.....	40
Description & Theory of Operation .....	40
Transparent Print Activation .....	41
DigiSCREEN Multiple Screen Utility.....	44
Description and Theory of Operation.....	44
Options .....	45
Using DigiSCREEN .....	46
Dynamic Screen Assignment .....	47
DSINFO DigiSCREEN Information File .....	48
Description .....	48
Entry Format.....	48
Error Messages .....	53

Digi Support Services.....55  
    The Digi Bulletin Board System .....55  
    Internet FTP Server .....56  
    World Wide Web Server .....56  
    DigiFACTs FaxBack Server .....56  
    Technical Support .....58  
    Customer Service .....58  
    Return Procedures .....59

# Introduction

---

The Digi device driver software for UNIX is installed by the Digi MenuPort Interface package; 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 Digi intelligent serial products (page 31)
- DigiPRINT transparent printing (page 40)
- DigiSCREEN, Digi's multiple screen utility (page 44)

In addition to the actual device drivers, the MenuPort Interface package includes the Digi Port Authority tool for monitoring and testing Digi intelligent products.

## **Important!**

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

## Supported Products

The MenuPort Interface package supports the Digi PC/Xe, PC/Xi, PC/Xr, MC/Xe, MC/Xi, C/X, EPC/X, Xem, SmartPort, SmartPort Plus, ClusterPort, and the Avastar family host adapters under System V Release 4 (including SCO UnixWare and NCR MPRAS) and SCO UNIX 3.2.0-3.2.4, SCO OpenServer 5.0.

# Software Installation

---

Installation of the Digi device drivers is a two-part procedure:

First, the Digi MenuPort Interface (MPI) package is installed onto your hard disk. Second, the MPI package is run and the appropriate device drivers for your system are installed and configured.

**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 MenuPort Interface (MPI) Package

## SCO UNIX

1. Log in as super-user (root).
2. Run **custom**. Select “New Product”. Insert the Digi MenuPort Interface device driver diskette when asked to do so.
3. When the MPI installation is complete, quit **custom**.
4. Now skip to page 8 to run the MPI program.

## System V, Release 4

1. Log onto the console as super-user (root).
2. To begin the installation, insert the Digi MenuPort Interface device driver diskette into drive A and enter:

```
pkgadd -d diskette1
```

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

Alternatively, you can use the **pkgask** program to create a response file (useful for configuring multiple systems). Enter:

```
pkgask -d diskette1 -r filename
```

where *filename* is the name of the response file. Follow the online help screens for more information on configuring boards. If you already have a response file from a previous **pkgask** session, you can use it by entering:

```
pkgadd -d diskette1 -r filename
```

MPI will be installed along with the drivers. MPI will not have to be run to configure the boards.

3. Now proceed to page 8 to run the MPI program.

## Running the MenuPort Interface (MPI) Program

To run the MenuPort Interface program, log in as root and enter:

```
mpi
```

After the restricted rights legend, you will see the Main menu. Items in this menu are:

- **Config**, which lets you install, change and remove device drivers for specific boards and operating systems.
- **System**, which lets you build a new kernel, update installed device drivers and create an image of the kernel.
- **Monitor**, which runs Digi Port Authority (DPA), a software diagnostic tool that allows users to easily monitor the status of the Front End Processor/Operating System (FEP/OS), which runs on all Digi intelligent serial port boards. The DPA software also shows the status of the individual ports. With DPA, system managers can tell instantly if a board is no longer working, if its performance level decreases, or if a port is non-functional, enabling them to work more productively and solve system errors more quickly. DPA also provides a loop back test to confirm that the board can transmit and receive data.
- **Port Configuration**, which provides an easy way to set up all Digi ports in your system without having to be an expert in UNIX system administration commands. Port Configuration is available in MPI version 2.3.0 or later.
- **Notes**, which displays the release notes for the supported device drivers.

### **Note!**

MPI features context sensitive help screens; press <F1> at any time for an explanation of the currently displayed information and choices.

# Device Names

## SCO UNIX

### PC/Xe, PC/Xi, MC/Xe and MC/Xi Boards

Format used for port names:

`/dev/ttyibp`, `/dev/pribp`

**ttyi** Base name.

**pri** Base name for DigiPRINT transparent printing (see page 40).

*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. Primarily used with 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. Primarily used with modems.

For example, Port 4 Board 2 will be `/dev/ttyi2d`. The same port with modem control enabled would be named `/dev/ttyi2D`. The corresponding DigiPRINT devices are `/dev/pri2d` and `/dev/pri2D`.

## SCO UNIX (continued)

### SmartPort and SmartPort Plus Boards

Format used for port names:

`/dev/ttybp`, `/dev/prtbp`

**tty** Base name.

**prt** Base name for DigiPRINT transparent printing (see page 40).

*b* Board number plus 2 (3-6).

*p* Port identifier:

**a-z**, **aa-ff** (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. **a-z** identify ports 1-26, and **aa-ff** identify ports 27-32. Primarily used with directly connected terminals and printers.

**A-Z**, **AA-FF** (upper case) are used for modem control ports. These ports require DCD (Data Carrier Detect) to be asserted before they can transmit data. **A-Z** identify ports 1-26, and **AA-FF** identify ports 27-32. Primarily used with modems.

For example, Port 4 Board 2 will be `/dev/tty4d`. The same port with modem control enabled would be named `/dev/tty4D`. The corresponding DigiPRINT devices are `/dev/prt4d` and `/dev/prt4D`.

## SCO UNIX (continued)

### Xr Boards

Format used for port names:

`/dev/ttyrbp`, `/dev/prrbp`

**ttyr** Base name.

**pr**r**** Base name for DigiPRINT transparent printing (see page 40).

*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. Primarily used with 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. Primarily used with modems.

For example, Port 4 Board 2 will be `/dev/ttyr2d`. The same port with modem control enabled would be named `/dev/ttyr2D`. The corresponding DigiPRINT devices are `/dev/prr2d` and `/dev/prr2D`.

## SCO UNIX (continued)

### C/X, EPC/X, Xem and ClusterPort Systems

Format used for port names:

`/dev/ttycp`, `/dev/prcp`

**tty** Base name.

**pr** Base name for DigiPRINT transparent printing (see page 40).

**c** Concentrator or EBI module letter ID:

**a-z** (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. Primarily used with directly connected terminals and printers.

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

**p** Port number on the individual concentrator or EBI module (**01-64**).

Systems with a large number of ports may have to use some concentrator/EBI ID numbers more than once. When this occurs, the first module to use an ID will be assigned port numbers 01-16; the second module will be assigned 17-32, etc.

For example, Port 4 on concentrator or PORTS module “**b**” will be `/dev/ttyb04`. The same port with modem control enabled would be named `/dev/ttyB04`. The corresponding DigiPRINT devices are `/dev/prb04` and `/dev/prB04`.

## SCO UNIX (continued)

### Avanstar Boards

Format used for port names:

`/dev/ttytp`, `/dev/ttytp`

**tty** Base name.

**ttp** Base name for DigiPRINT transparent printing (see page 40).

*t* Type of port (**a** or **A**). Indicates whether port is a “standard” (**a**) or “modem control” (**A**) port.

Standard (non modem control) ports do *not* require DCD (Data Carrier Detect) to be asserted before they can transmit data, and RTS and CTS pacing (see **ditty rtSPACE** and **ditty ctSPACE**, on page 34) are not active by default. Primarily used with directly connected terminals and printers.

Modem ports require DCD (Data Carrier Detect) to be asserted before they can transmit data, and RTS and CTS pacing (see **ditty rtSPACE** and **ditty ctSPACE**, on page 34) are active by default. Primarily used with modems.

*b* Board number:

**00** = board 1  
**01** = board 2  
**02** = board 3  
**03** = board 4

*p* Port identifier:

<b>0</b> = port 1	<b>4</b> = port 5	<b>8</b> = port 9	<b>c</b> = port 13
<b>1</b> = port 2	<b>5</b> = port 6	<b>9</b> = port 10	<b>d</b> = port 14
<b>2</b> = port 3	<b>6</b> = port 7	<b>a</b> = port 11	<b>e</b> = port 15
<b>3</b> = port 4	<b>7</b> = port 8	<b>b</b> = port 12	<b>f</b> = port 16

NOTE: Ports 9 through 16 apply to 16-port boards only.

For example, standard tty port 4 on board 2 would be `/dev/ttya13`. The same port with modem control enabled would be named `/dev/ttyA13`. The corresponding DigiPRINT devices would be `/dev/ttypa13` and `/dev/ttyPA13`.

## System V Release 4

### PC/Xe, PC/Xi, MC/Xe and MC/Xi Boards

Format used for port names:

`/dev/term/ibp, /dev/priibp`

**i** Base name.

**pri** Base name for DigiPRINT transparent printing (see page 40).

**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, and RTS and CTS pacing (see **ditty rtSPACE** and **ditty ctSPACE**, on page 34) are not active by default. Primarily used with 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, and RTS and CTS pacing (see **ditty rtSPACE** and **ditty ctSPACE**, on page 34) are active by default. Primarily used with modems.

For example, Port 4 of Board 2 will be `/dev/term/i2d`. The same port with modem control enabled would be named `/dev/term/i2D`. The corresponding DigiPRINT devices are `/dev/pri2d` and `/dev/pri2D`.

## System V Release 4 (continued)

### SmartPort and SmartPort Plus Boards

Format used for port names:

*/dev/term/sbp, /dev/prsbp*

- s** Base name.
- prsb** Base name for DigiPRINT transparent printing (see page 40).
- b** Board number plus 2 (3-6).
- p** Port identifier:
- a-z, aa-ff** (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, and RTS and CTS pacing (see **ditty rtspac** and **ditty ctspac**, on page 34) are not active by default. **a-z** identify ports 1-26, and **aa-ff** identify ports 27-32. Primarily used with directly connected terminals and printers.
  - A-Z, AA-FF** (upper case) are used for modem control ports. These ports require DCD (Data Carrier Detect) to be asserted before they can transmit data, and RTS and CTS pacing (see **ditty rtspac** and **ditty ctspac**, on page 34) are active by default. **A-Z** identify ports 1-26, and **AA-FF** identify ports 27-32. Primarily used with modems.

For example, Port 4 of Board 2 will be */dev/term/s4d*. The same port with modem control enabled would be named */dev/term/s4D*. The corresponding DigiPRINT devices are */dev/prs4d* and */dev/prs4D*.

## System V Release 4 (continued)

### Xr Boards

Format used for port names:

**/dev/term/rbp**, **/dev/prrbp**

- r** Base name.
- pr** Base name for DigiPRINT transparent printing (see page 40). Transparent printer devices are placed in **/dev**.
- 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, and RTS and CTS pacing (see **ditty rtSPACE** and **ditty ctSPACE**, on page 34) are not active by default. Primarily used with 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, and RTS and CTS pacing (see **ditty rtSPACE** and **ditty ctSPACE**, on page 34) are active by default. Primarily used with modems.

For example, Port 4 of Board 2 will be **/dev/term/r2d**. The same port with modem control enabled would be named **/dev/term/r2D**. The corresponding DigiPRINT devices are **/dev/prr2d** and **/dev/prr2D**.

## System V Release 4 (continued)

### C/X, EPC/X, Xem and ClusterPort Systems

Format used for port names:

`/dev/term/cpt, /dev/prcpt`

`/dev/term/`

Directory path for tty devices.

**pr** Base name for DigiPRINT transparent printing (see page 40). Transparent printer devices are placed in `/dev`.

**c** Concentrator or EBI module letter ID. Legal letters are **a-z**.

**p** Port number on the individual concentrator or EBI module (**01-64**). Systems with a large number of ports may have to use some concentrator/EBI ID numbers more than once. When this occurs, the first module to use an ID will be assigned port numbers 01-16; the second module will be assigned 17-32, etc.

**t** Type of port (**s** or **m**). Indicates whether port is a “standard” (**s**) or “modem control” (**m**) port.

Standard (non modem control) ports do *not* require DCD (Data Carrier Detect) to be asserted before they can transmit data, and RTS and CTS pacing (see **ditty rtSPACE** and **ditty ctSPACE**, on page 34) are not active by default. Primarily used with directly connected terminals and printers.

Modem ports require DCD (Data Carrier Detect) to be asserted before they can transmit data, and RTS and CTS pacing (see **ditty rtSPACE** and **ditty ctSPACE**, on page 34) are active by default. Primarily used with modems.

For example, standard tty port 4 on concentrator or EBI module “**b**” will be `/dev/term/b04s`. The same port with modem control enabled would be named `/dev/term/b04m`. The corresponding DigiPRINT devices are `/dev/prb04s` and `/dev/prb04m`.

## System V Release 4 (continued)

### Avanstar Boards

Format used for port names:

`/dev/term/abpt, /dev/prabpt`

**a** Base name.

**pra** Base name for DigiPRINT transparent printing (see page 40).  
Transparent printer devices are placed in `/dev`.

**b** Board number:

`00` = board 1  
`01` = board 2  
`02` = board 3  
`03` = board 4

**p** Port identifier:

`0` = port 1    `4` = port 5    `8` = port 9    `c` = port 13  
`1` = port 2    `5` = port 6    `9` = port 10   `d` = port 14  
`2` = port 3    `6` = port 7    `a` = port 11   `e` = port 15  
`3` = port 4    `7` = port 8    `b` = port 12   `f` = port 16

NOTE: Ports 9 through 16 apply to 16-port boards only.

**t** Type of port (**s** or **m**). Indicates whether port is a “standard” (**s**) or “modem control” (**m**) port.

Standard (non modem control) ports do *not* require DCD (Data Carrier Detect) to be asserted before they can transmit data, and RTS and CTS pacing (see **ditty rtSPACE** and **ditty ctSPACE**, on page 34) are not active by default. Primarily used with directly connected terminals and printers.

Modem ports require DCD (Data Carrier Detect) to be asserted before they can transmit data, and RTS and CTS pacing (see **ditty rtSPACE** and **ditty ctSPACE**, on page 34) are active by default. Primarily used with modems.

For example, standard tty port 4 on board 2 would be `/dev/term/a13s`. The same port with modem control enabled would be named `/dev/term/a13m`. The corresponding DigiPRINT devices are `/dev/pr13s` and `/dev/pr13m`.

# Enabling the New Ports

## SCO UNIX

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/tty1a
```

(Assuming the terminal is connected to **tty1a**.)

*(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 everything looks good, but you are still not getting output, run the loop back tests described on page 24.

If a 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.

- If nonsense characters are printed on the terminal’s screen, check the baud rates, data bits, stop bits, and parity setting on your terminal (UNIX default parameters are 9600 baud, 8 data bits, 1 stop bit, and no parity).

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. **tty1A**) 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 tty1A** (assuming that the port in question is **tty1A**).

Another way to bypass the DCD requirement is to use non modem control devices (where the port ID is a small letter; e.g. **tty1a**). 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. The Terminal Control Database must be updated to include the new devices. Add the Digi device information using the following SCO UNIX **sysadmsh** command sequence:

**Accounts→Terminal:Create**

Fill in the required information on the form displayed.

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

```
enable tty1a
```

The above command will cause a login prompt to be sent to the terminal connected to **tty1a**. To activate the other ports, repeat the enable command with the appropriate port name.

**Note!**

As an alternative to step 3, you can use the Port Configuration Utility of MPI to enable ports. See *Running MPI*, on page 8.

## System V Release 4

Please refer to your UNIX 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/term/ila
```

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

*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.*

*Also note that the default settings for terminals is 9600 baud, 7 data bits, even parity and 1 stop bit.*

- 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 everything looks good, but you are still not getting output, run the loop back tests described on page 24.

If a 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.

- If nonsense characters are printed on the terminal's screen, check the baud rates, data bits, stop bits, and parity setting on your terminal. (UNIX default parameters are 9600 baud, 7 data bits, 1 stop bit, and even parity.)

Once you can redirect output to your screen with the test above, perform the following steps to enable the ports for monitoring by **ttymon**.

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

3. Enter:

**sysadm**

This will bring up the **System Administration menu**. From this menu, select **“Ports”**. The **Ports** menu will now be displayed. From the **Ports** menu, select **“Quick Terminal”**. Now select **“Add”**.

You will now be shown a list of port numbers that are currently being monitored for activity (the list may be empty at this time).

4. Press <F2> (**“Choices”**). You will be shown a list of available ports. Use the arrow keys to move through the list of ports. Press <F2> to mark ports to be enabled (an asterisk will appear by the marked choices).

IMPORTANT! Do not select both modem and standard devices for the same port number (e.g. `/dev/term/ila` and `/dev/term/i1A`).

5. When you have completed your selections, press <F3> (**“Enter”**). The list you saw in Step 4 will now include the ports you just selected. To accept the list, press <F3> (**“Save”**).

6. Now press <F3> again (**“Continue”**), press <F7> (**“Cancel”**) to exit the **Ports Setup** menu. The ports are now enabled.

**Note:**

If you performed the above steps, but the ports are not enabled, you can enable the ports from the command line. The following example is for `/dev/term/ila`:

```
pmadm -a -p ttymon3 -s ila -i root -f u -v `ttyadm -V` \  
-m "`ttyadm -d /dev/term/ila -l 9600 -s /usr/bin/login`"
```

where `ttymon3` is the tag for the port monitor and `ila` is the service tag. The `-m` argument is the monitor-specific information, `/dev/term/ila` is the device with 9600 as the label in `/etc/ttydefs`. Substitute each device name for `ila` as appropriate.

**Note:**

The default UNIX SYSTEM V REL 4 settings for these ports are: **9600, ixon, cs7, e, 1** (these defaults may have been changed by your system administrator). These settings can be modified by going again to the **Ports** menu, selecting **Ports Services**, then selecting **Modify**.

**Note:**

As an alternative to steps 3-6, you can use the Port Configuration Utility of MPI to enable ports. See *Running MPI*, on page 8 (available in MPI version 2.3.0 or greater).

## Serial Printers

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

```
ditty printer [baud rate and flow control options] ttyila
```

This command forces the specified port to stay open, thus preserving any “non-sticky” serial line characteristics (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 a text file in the directory `/etc/rc.d/8` if you are running SCO UNIX—or in `/etc/rc2.d` if you are running System V Release 4—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 ttyila
```

### **Note:**

As an alternative to the above procedure, you can use the Port Configuration Utility of MPI to set up serial printers. See *Running MPI*, on page 8 (available in MPI version 2.3.0 or greater).

## Loop Back Test

Individual ports can be tested for correct operation by running the Digi Port Authority (DPA) program. To perform port tests with the DPA program, perform the following steps:

1. Run the MenuPort Interface program by entering the following command:

```
mpi
```

2. From the MenuPort Interface program's Main menu, select **Monitor**. The DPA program will be run.
3. From the DPA program's Main menu, use the up and down arrow keys select the board you wish to test.

NOTE: If the arrow keys do not map correctly, the DPA program will display the following message:

```
Invalid Key. (Key Number)
```

If this occurs, substitute the standard UNIX cursor movement keys for the up and down arrows: Down = **J** or **j**; Up = **K** or **k**.

4. If the board you selected was a PC/Xe, PC/Xi, MC/Xe, MC/Xi, PC/Xr, Avastar, SmartPort or SmartPort Plus board, you will now see the Channel Monitoring screen. Proceed to Step 5.

If the board you selected was a C/X, EPC/X, Xem or ClusterPort host adapter, you will now see the Adapter Configuration screen. Use the arrow keys to select the concentrator or EBI module you wish to test, and press <Enter> to get to the Channel Monitoring screen.

5. From the Channel Monitoring screen, use the left and right arrow keys to select the channel to test (or press **1-9** for channels 1-9, respectively, and **a-g** for channels 10-16, respectively).

A loop back diagnostic test can now be run against the selected channel by pressing the **T** (or **t**) key from the channel monitoring screen. A loop back plug (see page 27 for specifications and part numbers) must be installed in the channel to be tested.

The loop back test consists of five phases:

**Phase 1** Verify that the channel is not currently in use. All signals must be low.

If the port is busy, a message similar to the following will appear:

```
**** Port is Busy :DTR:CD:DSR:RTS
```

Probable cause of failure:

- Port is open by some process (e.g. getty).
- Incorrect loop back connector or cable.

**Phase 2** 128 bytes of test data (the letter A is used) are put into the transmit buffer.

If the write operation fails, a message similar to the following will appear:

```
**** Loop Back Test Failure #1.
```

After the write operation is complete, the FEP/OS is notified that data is available for transmission. At this time, the 128 bytes of data is transmitted. If the transmission fails, the following message will appear:

```
**** Loop Back Failure Sending 128 Bytes from Buffer.
```

Probable cause of failure:

- Hardware failure.
- FEP/OS or driver failure.

**Phase 3** At this point, the FEP/OS should have received 128 bytes of data. If not, the following message will appear:

```
**** Loop Back Failure Receiving 128 Bytes.  
**** Data Not Transferred to RX.
```

Probable cause of failure:

- Hardware failure.
- Loop back connector not installed, or incorrectly wired.
- FEP/OS or driver failure.

**Phase 4** 128 bytes of data are read from the receive buffer. If the read operation fails, the following message will appear:

```
**** Loop Back Failure Reading 128 Bytes from Buffer.
```

Probable cause of failure:

- Hardware failure.
- FEP/OS or driver failure.

**Phase 5** The data that was transmitted is compared with the data that has been read from the receive buffer. If the comparison fails, the following message will appear:

```
**** Loop Back Failure Verifying 128 Bytes.  
**** Data Incorrectly Transferred
```

Probable cause of failure:

- Hardware failure.
- FEP/OS or driver failure.

If all test phases pass, the following message will appear:

```
Loop Back Test Passed.
```

Press the **Q** key to return to the channel monitoring screen.

## Diagnostic Loop Back Plug Specifications

### *Xe, Xi, C/X, EPC/X, Xr, Xem*

The loopback plug for RJ-45 versions (Digi P/N 60000401) consists of a single 10-pin RJ-45 plug wired as follows:

- Pin 3 connected to pins 1 and 8 (RTS to RI and CTS)
- Pin 5 connected to pin 6 (TxD to RxD)
- Pin 9 connected to pins 2 and 10 (DTR to DSR and DCD)

For EIA-232 DB-25 versions, the loopback plug (Digi P/N 60000442) consists of a female DB-25 connector wired as follows:

- Pin 4 connected to pins 22 and 5 (RTS to RI and CTS)
- Pin 2 connected to pin 3 (TxD to RxD)
- Pin 20 connected to pins 6 and 8 (DTR to DSR and DCD)

For EIA-422 DB-25 versions, the loopback plug (Digi P/N 60000638-01) consists of a female DB-25 connector wired as follows:

- Pin 2 connected to pin 3 (TxD+ to RxD+)
- Pin 4 connected to pin 5 (RTS+ to CTS+)
- Pin 14 connected to pin 16 (TxD- to RxD-)
- Pin 19 connected to pin 13 (RTS- to CTS-)

### *SmartPort, SmartPort Plus and ClusterPort*

The loopback plug for RJ-45 versions (Digi P/N 62500031) consists of a single 8-pin RJ-45 plug wired as follows:

- Pin 3 connected to pin 4 (TxD to RxD)
- Pin 2 connected to pin 6 (DTR to CTS)
- Pin 7 connected to pin 8 (DCD to RTS)

For DB-25 versions, the loopback plug (Digi P/N 62500029) consists of a female DB-25 connector wired as follows:

- Pin 4 connected to pins 6 and 8 (RTS to DSR and DCD)
- Pin 2 connected to pin 3 (TxD to RxD)
- Pin 20 connected to pin 5 (DTR to CTS)

*Avanstar*

The loopback plug (Digi P/N 799001-03) consists of a male DB-25 connector wired as follows:

- Pin 4 connected to pins 5 and 8 (RTS to CTS and DCD)
- Pin 2 connected to pin 3 (TxD to RxD)
- Pin 20 connected to pins 6 and 22 (DTR to DSR and RI)
- Pin 11 connected to pin 13 (EIA-422 TxD+ to RxD+)
- Pin 14 connected to pin 16 (EIA-422 TxD- to RxD-)

## Uninstalling the MPI Software Package

Uninstalling the MPI software package is a two-part procedure. First, you must remove any device drivers that were installed via the MPI software, then you remove the MPI software package itself.

### Removing Device Drivers

1. To run the MenuPort Interface program, log in as root and enter:  
`mpi`
2. From the Main menu, select **Config**.
3. Now select **Remove**, from the Config menu.
4. From the Pick List, select the device driver you wish to remove. Repeat this step for all installed device drivers.

### Removing the MPI Software Package

#### *SCO UNIX*

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 `mpi`, then select `all`, to remove the entire package.

Removing Digi MenuPort Interface Package.

4. Quit `custom`, and perform a system shutdown and reboot to activate the new kernel.

### *System V Release 4*

1. Log onto the console as super-user (root).
2. Enter the following command:

```
pkgrm -n mpi
```

It will take some time for the new kernel to be built. When it is finished, perform a system shutdown and reboot to activate the new kernel.

# Setting Terminal Options with `ditty`

---

`/usr/bin/ditty` is a utility program that sets and displays the terminal options for Digi intelligent serial products.

The format is:

```
ditty [-a] [-p] [-n ttyname] [option(s)] [ttyname]
```

With no options, `ditty` displays all Digi 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.

## Note:

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` if you are running SCO UNIX, or `/etc/rc2.d` if you are running System V, Release 3 or 4—see your System Administrator's Guide for details.) Alternatively, you may include the `ditty` command sequence in your `.login` or `.profile` files, to ensure that the options you want are activated when you log in. Your system administrator can help you edit these files.

The options are:

- a** Display all of the unique Digi option settings, as well as all of the standard tty settings reported by `stty -a`.
- p** Interpret the port as a parallel printer port. This flag should only be used for the ninth port of a PORTS/8emp module.
- 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** (SCO) or **term/xxx** (SVR4). 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.  
**clear** Clear the **tbusy** flag in the channel structure.  
**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. *This option is not available in System V Release 4.*

**[ - ]fastbaud** Alter the baud rate tables by the following mapping scheme.  
*This option is not available for the SmartPort and the SmartPort Plus.*

	<i>MC/Xi with EPROM version 1.2 or greater, PC/Xi, PC/Xe, MC/Xe</i>	<i>MC/Xi with EPROM version 1.1</i>	<i>PC/Xr, Xem, EPC/X</i>	<i>C/X</i>	<i>Cluster-Port</i>	<i>Avanstar</i>
<b>Specified Baud Rate:</b>	<b>Baud Rate Mapped to:</b>	<b>Baud Rate Mapped to:</b>	<b>Baud Rate Mapped to:</b>	<b>Baud Rate Mapped to:</b>	<b>Baud Rate Mapped to:</b>	<b>Baud Rate Mapped to:</b>
50	57600	19200	57600	57600	57600	57600
75	76800	19200	76800	75	75	76800
110	115200	19200	115200	115200	110	115200
134	14400	134	14400	14400	134	14400
150	28800	150	57600	57600	150	57600
200	200	200	230000	200	200	200
300	300	300	76800	300	300	76800
600	600	600	115200	115200	600	115200
1200	1200	1200	230000	1200	1200	1200
1800	1800	1800	28800	28800	1800	28800
2400	2400	2400	2400	2400	2400	2400
4800	4800	4800	4800	4800	4800	4800
9600	9600	9600	9600	9600	9600	9600
19200	19200	19200	19200	19200	19200	19200
38400	38400	19200	38400	38400	38400	38400

- [-]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. *This option is not available on SmartPort, SmartPort Plus or ClusterPort.*

**eia232|eia422**

By default, each port on an Avastar configurable connection panel powers up in EIA-232 mode. This option sets the EIA mode for the given port to either EIA-232 or EIA-422. *This option is available only for Avastar boards with configurable connection panels.*

**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). *This option is not available in System V Release 4.*
- 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). *This option is not available in System V Release 4.*
- [-]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. *This option is not available in System V Release 4.*
- [-]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). *This option is not available in System V Release 4.*
- [-]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. *This option is not available in System V Release 4.*
- edelay** *n* **edelay** is a tunable parameter used to set the number of milliseconds of delay between the time the first character arrives after a period of no characters and notification of its arrival to the host. This is also referred to as the wakeup rate between the host adapter software (FEPOS) and the host device driver. This has the advantage of reducing host overhead by allowing the host to process larger blocks of incoming data.
- Larger **edelay** values result in more characters being sent in a given time period. This will reduce host processor utilization and character response time and increase overall system throughput.

Smaller **edelay** values result in fewer characters being sent in a given time period. This will increase character response time and increase host processor utilization.

The default value for **edelay** is 100. This is a good value for normal tty activity including typing and *uucp*. For applications receiving continuous input at high speeds, increasing **edelay** will result in lowering host overhead and an increase in overall system throughput. A value of 250 is reasonable.

- 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**, **vt320**, **vt420**, **wang2x36**, **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)**.

## Using the `ditty` Command with Parallel Ports

The `ditty` command features a special mode for use with the parallel port on Digi PORTS/8emp modules. When the `-p` flag is included in the `ditty` command, certain flags and functions are interpreted differently than for serial ports. Following is an example of the output of the command `ditty -p ttye09`:

```
parallel output port
pulse width 0.8 us -noack hupcl
BUSY- IP0+ IP1- IP2- OP+
-opost -olcuc -onlcr -ocrnl -onocr -onlret
```

**Pulse Width** defines how long the data is strobed; the default is 0.8 microseconds, which is the fastest supported strobe. To change the strobe length, use the `ditty` command as follows:

```
(SCO UNIX)  ditty 1 ttya09
(SVR4)      ditty 1 term/a09s
```

The above example sets a 1.0 microsecond pulse width.

Valid pulse widths are: **0.8, 1, 1.2, 1.6, 2, 3, 4, 5, 10, 20, 30, 40** and **50**.

The “baud rate” of the port is used to control the pulse width—9600 baud or faster translates to 0.8 microseconds; 4800 baud is 1 microsecond, 2400 is 1.2 microseconds, etc. The `-p` flag is implied, and need not be included in the command.

### `[-]noack`

When `noack` is specified (`ditty noack ttya09`), the device driver will not wait for an ACK pulse from the receiving device before transmitting another character; only the BUSY signal will stop data transmission.

When `noack` is turned off (`ditty -noack ttya09`), an ACK pulse is expected after each character is transmitted.

The `dsrpace` control is mapped to `noack` for parallel ports. The `-p` flag is implied, and need not be included in the command.

`hupcl` has the usual meaning: “hang up on close”. Since DTR is mapped to the parallel port’s SELECTIN line (`OP`; see below), dropping and raising DTR (which happens when a port is closed and then opened again) will have the effect of strobing the RESET line of the parallel device connected to the port. `-hupcl` prevents the device from being reset when it is reopened.

**BUSY** shows the state of the BUSY line. **BUSY** is not asserted (-) when the input device can accept data, and asserted (+) when it cannot. **BUSY** is mapped to DSR for parallel ports.

**IP0**, **IP1** and **IP2** are device dependent inputs. They provide information only, and do not affect the flow of data. They are typically used for signals like Paper Out, On/Off Line, etc. **IP0**, **IP1** and **IP2** are mapped to CTS, DCD and RI, respectively.

**OP** is the SELECTIN line, which is mapped from DTR. When **OP** changes from LOW to HIGH, the parallel port's RESET line is strobed. Normally, the RESET function is handled by the driver; however, a RESET can be forced by issuing the command "**ditty OP- ttya09**" followed by "**ditty OP+ ttya09**", or by the command "**ditty reset ttya09**". The **-p** flag is implied, and need not be included in the command.

**[-]opost**, **[-]olcuc**, **[-]onlcr**, **[-]ocrnl**, **[-]onocr**, **[-]onlret**

These flags have the usual meaning (see **stty(1)**).

# DigiPRINT Transparent Print Feature

---

## 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 Digi 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 Digi 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** if you are running SCO UNIX, or **/etc/rc2.d** if you are running System V, Release 3 or 4—see your System Administrator's 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:

<b>maxcps</b> <i>n</i>	Sets the maximum Characters Per Second (CPS) rate at which characters are output to the transparent print device. See <i>Setting Terminal Options with ditty</i> (page 31) for more information.
<b>maxchar</b> <i>n</i>	Sets the maximum number of transparent print characters the driver will place in the output queue. See <i>Setting Terminal Options with ditty</i> for more information.
<b>bufsize</b> <i>n</i>	Sets the driver's estimate of the size of the transparent printer's input buffer. See <i>Setting Terminal Options with ditty</i> for more information.
<b>onstr</b> <i>s</i>	Sets the terminal escape sequence to turn the transparent printer on. An arbitrary octal character <i>xxx</i> may be given as <b>\xxx</b> .
<b>offstr</b> <i>s</i>	Sets the terminal escape sequence to turn the transparent printer off. An arbitrary octal character <i>xxx</i> may be given as <b>\xxx</b> .
<b>term</b> <i>t</i>	Specifies the terminal type. See <i>Setting Terminal Options with ditty</i> for more information.
<i>port</i>	Specifies the tty device (e.g. <b>tty11a</b> ).

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

**Example:**     **cat filename > /dev/pr11a**

## ditty Examples for Transparent Printing

### Example 1:

The following command configures the DigiPRINT options for a DEC VT100 terminal connected to `/dev/tty1a`. (Note that the printer will be called `/dev/pr1a`). `maxcps`, `maxchar` and `bufsize` are left to defaults. Enter:

```
ditty term vt100 /dev/tty1a
```

### 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/tty1a
```

### 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/tty1a
```

# DigiSCREEN Multiple Screen Utility

---

(Also see `dsinfo`, on page 48.)

## 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 ] [ -b ]
```

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.

The **-b** option prevents background screens from blocking output, and provides each screen with a 2K buffered output window as long as the programs/tasks don't do any type of ioctl that will change the terminal settings (i.e. [ **TC** | **XC** | **O\_XC** ] **SET** [ **A** | **AF** | **AW** ] ).

## Command Line Examples:

```
dscreen  
dscreen -b  
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 48) 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

`/etc/dsinfo` is a database of terminal descriptions used by `dscreen` (DigiSCREEN). The information in the descriptions includes 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:

**dskx** 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
<b>dsks</b>	Switch Screens
<b>dskb</b>	Block Input and Output
<b>dske</b>	End DigiSCREEN
<b>dsq</b>	Quit DigiSCREEN (non-zero exit status)
<b>dskc</b>	Create New Screen
<b>dskp</b>	Switch to Previous Screen
<b>dskl</b>	List Keys and Actions

Any other key type (a string type **dskx** 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 for a communicating through the second serial port on the terminal:

```
wy60-1|wyse60-1|wyse model 60 - first serial port,
      dskb=^A^M|Shift-F1|,
      dskb=^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,
      dskb=^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.

# Digi Support Services

---

## The Digi Bulletin Board System

Digi provides an electronic bulletin board service (BBS) for our customers. This bulletin board provides general and technical information about Digi's products.

The Digi BBS allows users to download software drivers as soon as they become available. There is also a feature to allow users with problems or questions about Digi products to leave messages to Digi Technical Support.

Using the Digi BBS is easy. Simply dial **(612) 912-4800**. In Europe, dial **+49 221 9205211**; in Asia, dial **+65 735 2460**.

The recommended modem communications parameters are 8 bits, no parity and one stop bit (**8 N 1**). Other settings may also work.

Download protocols include Zmodem, Xmodem, Ymodem, Kermit and others.

## Internet FTP Server

Digi has set up an Anonymous FTP server for those with access to the Internet network. The address is **ftp.dgii.com**. Log in as **anonymous**, and enter your E-mail address when asked for a password. Drivers and installation tips are located in the **/drivers** directory. A text file, **download.doc**, gives information on uncompressing the files after downloading. Tip: Be sure to enter **"bin"** before downloading, to ensure binary transfer of files.

## World Wide Web Server

Product information, manuals, new product announcements, programs, application stories and more can be obtained through the World Wide Web. Our address is **http://www.dgii.com**.

## DigiFACTs FaxBack Server

Manuals and technical information can also be obtained by FAX. To use the FaxBack server, simply call (612) 912-4990 on a touch tone phone.

## Information About Your System

Serial number of your Digi product: \_\_\_\_\_

Make, model and clock speed of your computer: \_\_\_\_\_

\_\_\_\_\_

How much RAM does your computer have? \_\_\_\_\_

**Hard disk controller:** Type: \_\_\_\_\_ Memory addressed at: \_\_\_\_\_

I/O port used: \_\_\_\_\_ IRQ: \_\_\_\_\_

**LAN card:** Type: \_\_\_\_\_ Memory addressed at: \_\_\_\_\_

I/O port used: \_\_\_\_\_ IRQ: \_\_\_\_\_

**Other:** Type: \_\_\_\_\_ Memory addressed at: \_\_\_\_\_

I/O port used: \_\_\_\_\_ IRQ: \_\_\_\_\_

**Operating system:** \_\_\_\_\_ Version: \_\_\_\_\_

**Digi device driver version:** \_\_\_\_\_

## Technical Support

At Digi, we are proud of our products, and support them. Our dealers and distributors are fully trained on our product line, so that they can help you on a technical level should assistance be needed.

Your first level of support is your Digi dealer, the place where you purchased your Digi product. Your dealer has the training to help you with any installation questions or difficulties you might have.

If you still experience difficulties (after contacting your first level of support), Digi has a staff of Technical Support Specialists that can assist you. They can be reached at **(612) 912-3456**. In Europe, call **+49 221 920520**, and in Asia, call **+65 732 1318**. FAX numbers are: (612) 912-4958 (USA), +49 221 9205210 (Europe) and +65 732 1312 (Asia).

When you call Digi Technical Support, please call from a position where you can operate your system. Also, please fill out the form on the preceding page before calling, so your Technical Support representative can have a clear picture of your system and any potential conflicts between devices.

Digi Technical Support can also be reached via Internet E-mail. Please send correspondences to **support@dgii.com**, and include your voice and FAX phone numbers.

## Customer Service

Digi also has a staff of Customer Service representatives to help you with software and documentation update requests, as well as Returned Merchandise Authorizations (RMAs) in case you need to return your Digi product for repair (see page 59). They may be reached at **(612) 912-3456**. Digi Customer Service may be contacted by FAX at (612) 912-4959.

Digi Customer Service can also be reached via Internet E-mail. Please send correspondences to **cust\_serv@dgii.com**, and include your voice and FAX phone numbers.

## Return Procedures

All Digi products have a five-year parts and labor warranty, and we are ultimately responsible for any defective parts, according to the limits specified in the warranty. However, many of the reported problems are due to factors other than defects in the product itself. To save you time and possibly additional cost, Digi asks that you *first* try to resolve any difficulties by contacting our Technical Support representatives at **(612) 912-3456**.

### **Important!**

*Be sure to have the serial number of your Digi product at hand before calling Technical Support.*

Returns should be directed to the dealer or distributor from whom you purchased the product. If you need to return your Digi product for repair, it is first necessary to obtain an RMA (Returned Merchandise Authorization) number from Digi, by speaking to a Digi Customer Service representative. Authorized returns should be shipped to Digi International, 10000 West 76th Street, Eden Prairie, MN 55344. The RMA number should appear on the shipping carton, on or near the address label.

### **Note:**

*Products received without an RMA number clearly marked on the outside of the package will be returned, unopened, to the sender*

