Software Manual

AccelePort EPC/X, C/X, Xem, Xr and Xr 920

Digital UNIX V3.x & 4.x

92000186 Rev D

AccelePort and the Digi logo are either trademarks or registered trademarks of Digi International Inc. in the United States and/or other countries.

DigiPRINT, **Digi Port Authority**, **DPA**, **Xem**, **Xr**, **Xr 920**, **EPC/X**, **EPC/CON-16**, **C/X**, **C/CON-16**, **C/CON-8**, **PORTS/16em** and **PORTS/8em** are trademarks of Digi International Inc.

All other brand and product names may be the trademarks of their respective holders.

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

This manual contains instructions for installing and configuring the Digital UNIX device driver for Digi AccelePort® intelligent serial communications products.

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)
- Digi Port Authority™, a port monitoring and diagnostic tool (page 44)

Once you have installed the adapter according to the instructions in the *Installation Guide* for your AccelePort adapter, you may proceed with the software device driver installation instructions starting on the following page.

Supported Products

The device driver package provides support for Digi AccelePort EPC/X™, C/X™, Xem™, Xr™ and Xr 920™ adapters.

Note: 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 Notes

The hardware installation procedure is different depending on the system bus architecture and the type of adapter you are installing. For complete information, refer to the *Installation Guide* for your adapter. This section deals with hardware installation issues that relate particularly to Digital UNIX installations.

PCI Adapters

PCI adapter installation is the simplest, because the machine can do much of the configuration work for you. Simply choose an empty PCI slot and insert the adapter into this slot. The system will detect the adapter and assign it the necessary memory.

Note: When you configure multiple Digi PCI adapters, you *must* do so in the order that they appear on the PCI bus, so that the adapter in PCI slot 1 is configured before the adapter in PCI slot 5.

ISA adapters in ISA Machines

Installing an ISA adapter in an ISA based machine requires much more care than does the PCI installation. The system will not negotiate parameters for you, so you must first select values for four parameters which will not conflict with those chosen for other adapters installed in the system:

- I/O port address (set by DIP switches on the adapter — see the adapter's *Installation Guide*): the **`isacfg`** (see below) parameter for I/O port is **`-iobase0`**.

- Dual ported memory address (see the adapter's *Installation Guide*): there are two **`isacfg`** (see below) parameters associated with the dual ported memory address: **`-membase0`** is used to specify the base address of the memory region; **`-memlen0`** specifies the size of the region and should *always* be set to **`8000`**.

- Slot number (since ISA systems don't address adapters by slot, the slot number doesn't have to match the physical slot in the machine, but the **`isacfg`** utility still requires each adapter to be assigned a unique slot number, which will also be used by the device driver): the **`isacfg`** (see below) parameter for slot number is "**`-slot`**".

- Device handle (in order for the Digital UNIX driver to correctly detect your Digi adapter, you must supply the text string "DGIBRD" for the handle): the **`isacfg`** (see below) parameter for the device handle is **`-handle`**.

To determine existing parameters and to set the parameters for the new adapter, use the **`isacfg`** utility, available at the SRM console prompt (**`>>>`**). This console command is used to maintain ISA information for the system. For more information on this utility, see the user information and hardware manuals for your system.

Note: All numeric values supplied to the **`isacfg`** utility are interpreted as hexadecimal digits, and should be entered as such.

Examples:

To view all existing ISA configuration information:

```
>>>isacfg -all
```

To delete the entry for "slot 1", "dev 0":

```
>>>isacfg -rm -slot 1 -dev 0
```

To add an entry for an AccelePort Xr ISA adapter for "slot 5" with an I/O port address of 0x104 and a memory address of 0x90000:

```
>>>isacfg -slot 5 -dev 0 -mk -handle DGIBRD -iobase0 104
   -etyp 1 -enadev 1 -membase0 90000 -memlen0 8000
```

Note that the above must be entered on a single command line.

Note:

`show config` is an SRM command which will attempt to display information about adapters installed in all of the system buses. It can be a useful tool for finding conflicts in the dual ported memory or I/O address.

Important!

Before rebooting, be sure to use the **`init`** command at the console prompt to save any changes you have made to the ISA configuration.

EISA or ISA adapters in EISA Machines

Installation of Digi AccelePort adapters in EISA machines is aided by the EISA Configuration Utility (ECU) provided by DEC with your computer. To access the ECU, load your working copy of the EISA Configuration Utility diskette in the diskette drive and enter **ecu** at the console prompt.

Important!

Be sure that the appropriate Digi configuration files have been copied to this disk before use; these files all have names of the form: “!DBI*.CFG”

The ECU allows you to add, remove or configure adapters via a menu driven user interface. The information you supply to the ECU will be read by the driver during adapter initialization.

Important!

Be sure that none of the chosen I/O port and memory addresses conflict with other devices installed in your system. Use the **View or edit details** option in the ECU to verify this.

Note:

When configuring the Digital UNIX driver for multiple ISA or EISA adapters, you *must* do so in the order in which the adapters appear in the ISA or EISA bus. So an adapter in EISA slot 1 should be configured before one in slot 4.

Software Installation

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

Installing the Driver Package

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

2. To begin the installation, insert the Digi device driver diskette in the diskette drive and enter the following commands:

```
cd /
tar xf /dev/fd0a
setld -l /tmp/digi
```

This will run the software subset management utility designed to assist in add on package installation.

Refer to the *System Administrator's Guide* for more information on `setld`.

3. You will see the following (or something similar):

```
*** Enter subset selections ***

The following subsets are mandatory and will be installed automatically
unless you choose to exit without installing any subsets:

      * Digi AccelePort Device Driver Package

You may choose one of the following options:

     1) ALL of the above
     2) CANCEL selections and redisplay menus
     3) EXIT without installing any subsets

Enter your choices or press RETURN to redisplay menus.

Choices (for example, 1 2 4-6): 1
```

Choose option 1 to load the Digi AccelePort device driver package.

Note:

If a version of the Digi device driver package is already installed on the system, the installation procedure will fail.

See the system documentation of the **`setld -i`** command or *Uninstalling the Digital UNIX Device Driver* (on page 26 of this manual) for details on determining which version of the driver package is installed.

After the files have been copied, you see the following (or a similar) screen:

```
This script installs the Digi AccelePort device driver for:

        Digital Unix Rel. V4.0

This script also installs the information needed by Digital UNIX to use
these additional ports. This driver supports up to 7 asynchronous
serial adapters.

By default, the ports are installed with the basename "/dev/tty".
DigiPRINT ports are installed with the basename "/dev/pr".

        Hit <CR> to proceed or <Q> to quit:
```

Press <Enter>.

At this point in the configuration process, the driver package attempts to detect whether a saved configuration can be found. If a configuration is not found, the procedure will skip to step 4 without a prompt. If a configuration is found, you will be asked whether to use this saved configuration instead of manually entering the configuration. This option could be used to preserve a configuration during a device driver upgrade, for instance. If you choose to use the saved configuration, the installation procedure will complete automatically, and you will not be prompted for the remaining steps described in this section.

Saved configurations may be generated by the driver package during an uninstall operation, see *Uninstalling the Digital UNIX Device Driver* (on page 26) for details.

4. The system displays:

```
Searching for port names already in use...
The port names (in the /dev directory) will look like:

        /dev/ttya01s

where "tty" indicates a tty device, "a" is an ID letter
that can be different for each port box, and "01" is the port number and
the suffix "m" or "s" is "modem" or "standard" tty.

Systems with a large number of ports may need to use some letters more
than once. When this occurs, the first box to use an ID letter will
get port numbers 01-16, the second box to use that letter will get
port numbers 17-32, and so on.

You can have the install script pick ID letters for you, assigning
letters consecutively (a,b,c,d,...) and starting over from the
beginning when it runs out of letters; or you can choose them manually.
Manual ID selection is recommended for larger systems, where it can be
used to help keep port names more organized.

Do you want ID letters picked for you (y or n)?
```

If you answer “**y**”, ID letters will be automatically picked. If you answer “**n**”, you will be asked (later in the procedure) to specify an ID letter for each PORTS™ module.

5. Now the system will ask:

```
How many adapters do you wish to install (1-7)? 2
```

Enter the number of adapters you have installed in your computer.

Steps 6-8 will be repeated for each EPC/X, C/X, Xem, Xr or Xr 920 host adapter installed in your system.

6. You will see the display:

```
Configuring adapter 1.

        ISA/EISA Adapters supported:
           1)  AccelePort Xem ISA/EISA
           2)  AccelePort C/X ISA/EISA
           3)  AccelePort EPC/X ISA/EISA
           4)  AccelePort Xr ISA
           5)  AccelePort Xr 920 ISA

        PCI Adapters supported:
           6)  AccelePort Xem PCI
           7)  AccelePort C/X PCI
           8)  AccelePort EPC/X PCI
           9)  AccelePort Xr PCI
           10) AccelePort Xr 920 PCI

        What type is adapter 1?
```

Choose the applicable adapter type.

7. This step is different depending on the type of adapter you are installing. Follow the appropriate procedure for your adapter. *If you are installing an AccelePort Xr or Xr 920, skip this step, and go to Step 8, on page 17.*

Xem—for C/X and EPC/X, follow the procedure on the next page.

If you are installing an Xem host adapter, you will now be asked for the number of PORTS modules to be connected to the adapter:

```
 Configuring adapter 1.
 Adapter type is AccelePort Xem ISA/EISA Host Adapter.

PORTS modules:
 This adapter can have 1 to 4 EBI modules connected, for 8 to 64 ports.

    How many PORTS modules are connected to this adapter (1 - 4)?
```

Select the appropriate number for your installation.

If you answered "**`n`**" to the question "**`Do you want ID letters picked for you?`**" back in Step 6, you will also be prompted at this time for the ID letters to be assigned to each of the PORTS modules.

You will also be asked for the number of ports on each PORTS module. Answer “**8**” for PORTS/8em™ modules or “**16**” for PORTS/16em™ modules.

Now proceed to Step 8, on page 17.

C/X and EPC/X host adapters—for Xem host adapters, follow the procedure on the previous page.

If you are installing an AccelePort C/X or EPC/X host adapter, you will be asked for configuration information for the two synchronous lines. This determines how the host adapter will communicate with the attached concentrator(s).

There are two methods for setting the line configuration: *basic* and *custom*. In the basic configuration, the sync lines are set to 8-wire direct and either 1228K bps or 10M bps (10M bps is applicable only to the EPC/X host adapter). The custom configuration allows you to select the wiring method and data rate for each sync line. You will see:

```
 Configuring adapter 1.
 Adapter type is AccelePort EPC/X ISA/EISA Host Adapter.

Line Configuration:
 The sync lines may be configured two ways. The basic method has all
 connections wired "8 wire direct" at a speed of either 1228K bps or
 10M bps, and can be used for quick set-up for most situations. The
 custom method allows you to choose speeds and wiring methods
 individually.

        Do you want to use the basic installation method (y/n)?
```

Basic Configuration *(for Custom Configuration, see page 16)*

7a. If you are installing an EPC/X host adapter, you will see the following screen (if you are installing a C/X host adapter, skip to Step 7b):

```
 Configuring adapter 2.
 Adapter type is AccelePort EPC/X ISA/EISA Host Adapter.

Basic Configuration:

 One of two line speeds must be chosen for the basic configuration
 method. The choices are 1228K bps and 10M bps. Running the sync
 lines at 1228K bps maintains compatibility with C/CON-16
 concentrators. Running the sync lines at 10M bps allows higher
 performance, but is not compatible with C-CON/16 concentrators and
 restricts the length of the sync cable. For more information, see
 the manual.

        1) 1228K bps
        2) 10M bps

 Which speed do you wish to run the sync lines at?
```

Important!

Select 10M bps *only* if:

- all attached concentrators are EPC/CON-16 concentrators, and
- the *cumulative* length of all daisy chain cables on either sync line is 30 feet or less.

See the EPC/CON-16 *Installation Guide* for more detailed information about data rates versus cable lengths.

7b. Now you will be asked how many concentrators are attached to each sync line. For EPC/X host adapters, you will also be asked how many EBI modules are attached to each concentrator (this refers to PORTS/16em or PORTS/8em modules connected to the “EBI Out” port of EPC/CON-16 concentrators), and how many ports are on each EBI module (answer “**8**” for PORTS/8em modules, or “**16**” for PORTS/16em modules).

Important!

EPC/CON-16 and newer C/CON-16 Concentrators

EPC/CON-16 and newer C/CON-16 concentrators (which have a plastic case) must have their node numbers set manually at installation (see the concentrator *Installation Guide* for instructions). When asked for the number of concentrators, enter the *node number* of the last concentrator in each sync line (which may be different from the actual number of concentrators). Normally concentrator node numbers are assigned sequentially, beginning with 1; however, if you are planning to add more concentrators later, and want to configure the driver and create the devices in `/dev` to support the future concentrators, you may skip node numbers to allow for later insertion of concentrators. For example, if you have three concentrators attached to a sync line, but are planning to add three more, the node number of the last concentrator in the chain should be set to “6”, and you should enter “6” for the number of concentrators (it is important that these numbers match, because the driver uses them to set the communication mode of the last concentrator to enable it to return data to the host adapter).

Older C/CON-16 Concentrators

Older C/CON-16 concentrators (which have a metal case) are always assigned node numbers sequentially, beginning with 1. If all of the concentrators on a sync line are older models, enter the actual number of concentrators (which will correspond to the node number of the last concentrator in the chain).

Mixed Older and Newer Concentrators

When mixing older and newer concentrators on a sync line, the older concentrators must be attached closest to the host adapter, and newer concentrators must be at the end of the chain. In this case, use the *node number* that has been set in the last concentrator in the chain (which may be different from the actual number of concentrators).

Now proceed to Step 8, on page 17.

Custom Configuration

The custom configuration requires you to specify the wiring method and the data rate for each sync line. The wiring options are:

1) 4-wire direct
2) 8-wire direct
3) 8-wire synchronous modem
4) EIA-232 synchronous modem

The supported data rates are:

4-wire direct

1) 230K
2) 460K
3) 921K*
4) 1843K*

8-wire direct

1) 2400
2) 4800
3) 9600
4) 19.2K
5) 38.4K
6) 57.6K
7) 76.8K
8) 115K
9) 230K
10) 460K
11) 921K
12) 1228K
13) 1843K*
14) 2458K*
15) 3686K*
16) 7373K*
17)10M*

8-wire synchronous modem

1) 2400
2) 4800
3) 9600
4) 14.4K
5) 19.2K
6) 38.4K
7) 57.6K
8) 76.8K
9) 115K
10) 230K
10) 460K
11) 921K
12) 1228K

8-wire EIA-232 synchronous modem

1) 2400
2) 4800
3) 9600
4) 14.4K
5) 19.2K
6) 38.4K
7) 57.6K
8) 64.1K
9) 76.8K

* Rates with an asterisk are incompatible with C/CON-16 concentrators. These data rates can only be used if *all* concentrators on a given sync line are EPC/CON-16 concentrators. Also, see *Restrictions on Cable Lengths* in the EPC/CON-16 *Installation Guide and Reference Manual*.

8. You will now be asked to confirm the configuration for the adapter:

   If you have installed an AccelePort Xem with four PORTS modules, you will see something like this:

```
You have selected the following configuration for adapter 1:
        Adapter Type: AccelePort Xem ISA/EISA host adapter.

        Module  Port Names
      ---------------------------
        1       ttya01[ms] - ttya16[ms]
        2       ttyb01[ms] - ttyb08[ms]
        3       ttyc01[ms] - ttyc08[ms]
        4       ttyd01[ms] - ttyd16[ms]

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

   If you have installed an EPC/X host adapter, you might see something like this:

```
You have selected the following configuration for adapter 2:
 AccelePort EPC/X ISA/EISA Host Adapter.

        Connections     Speed   Cabling         Port Names
      --------------------------------------------------------
          Line 1
      Concentrator 1    10M     8 wire direct   ttye01[ms] - ttye16[ms]
           EBI Module 1                         ttyf01[ms] - ttyf16[ms]
           EBI Module 2                         ttyg01[ms] - ttyg08[ms]
           EBI Module 3                         ttyh01[ms] - ttyh08[ms]
      Concentrator 2    10M     8 wire direct   ttyi01[ms] - ttyi16[ms]
          Line 2
      Concentrator 1    10M     8 wire direct   ttyj01[ms] - ttyj16[ms]
      Concentrator 2    10M     8 wire direct   ttyk01[ms] - ttyk16[ms]
           EBI Module 1                         ttyl01[ms] - ttyl08[ms]
      Concentrator 3    10M     8 wire direct   ttym01[ms] - ttym16[ms]
      Concentrator 4    10M     8 wire direct   ttyn17[ms] - ttyn32[ms]

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

   If you accept the configuration, Steps 6-8 will be repeated for each additional adapter specified in Step 5. Otherwise, you will be returned to Step 6 (page 12) to re-enter the configuration for the current adapter.

9. 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 Digi 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 adapter, and page 33 of this manual for more details.)

   The following message is displayed when setting up adapter number 1:

```
Altpin is provided to allow the DCD input signal to be exchanged with the DSR
input. This is useful on products with RJ-45 connectors as these products
have DCD on pin 10. Choosing altpin will allow the use of 8-pin connectors
while still having the benefit of full modem control. The use of altpin on
products with DB-25 connectors is not normally recommended.
```

   The software then 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.)

Creating New Devices

The devices will be created in the **`/dev`** directory during the next system boot.

The Digi driver will support up to seven AccelePort EPC/X, C/X, Xem, Xr and/or Xr 920 host adapters.

AccelePort C/X host adapters can have up to eight concentrators attached.

AccelePort EPC/X host adapters can each have up to 14 concentrators and/or PORTS. The maximum number of concentrators per synchronous line is eight. PORTS modules may be attached to the concentrators, as long as the total number of boxes (concentrators and PORTS modules) does not exceed 14.

AccelePort Xem adapters can each have four PORTS modules attached.

The devices are named according to the following conventions:

Format used for port names

/dev/tty...
Base name path for tty devices.

/dev/pr…
Default base name for DigiPRINT transparent printing devices.

a-z Concentrator or PORTS module letter I.D. Legal letters are a-z.

01-96 Port number on the individual concentrator or PORTS module.

s, m Indicates whether port is a “standard” (**s**) or “modem” (**m**) port. Modem ports can only be opened when Data Carrier Detect (DCD) is active.

By default, “standard” ports use software (XON/XOFF) flow control; “modem” ports use hardware (RTS/CTS) flow control.

For example, standard tty port 4 on concentrator or PORTS module “**b**” will be **/dev/ttyb04s**. The same port with modem control enabled would be named **/dev/ttyb04m**.

NOTE: A port may be enabled as either a standard or modem tty device, but not both.

A special device named **`epcadl`** will also be created in the **`/dev`** directory. This device is a “control port” and is used by Digi-supplied utilities (such as DPA and downld) for driver maintenance. This device SHOULD NOT be used as a serial port device.

Installing the New Kernel

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

To rebuild the kernel, issue the following command:

`doconfig -c` <*name*>

where *name* is the SCF (System Configuration File) you wish to use.

Please consult your *System Administrator's Guide* for more details about **`doconfig`**.

Be sure to replace the current kernel with the new kernel. It is generally a good idea to back up the current kernel first.

When you reboot your computer, you should see messages which indicate the adapter is functioning. Your screen will look something like this:

```
epca   port:0x00000208  mem=0x000C0000  ports=32  AccelePort Xem ISA V x.x.x
```

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 Digi hardware *Installation Guide* for more information about memory problems.

After multi-user mode has been entered, you will see this message (it may scroll by too quickly to notice):

```
Digi software download started.
```

If no errors are detected, AccelePort C/X and EPC/X host adapters will proceed to download the operating software to the C/CON-16 and/or EPC/CON-16 concentrators. Once the concentrators are running, their displays will show "**AC**".

Important!

This device driver will not function in single-user mode. The BIOS and FEPOS images are loaded by a daemon, **`/sbin/epcadl`**, which is started when the system comes up to multi-user mode.

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

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

For example, to set **altpin** for port **`a01s`**, add the following line to **`/sbin/rc2.d/S99dittystuff`**:

```
/usr/bin/ditty altpin /dev/ttya01s
```

Testing the New Ports

Please refer to your Digital UNIX *System Administrator's Guide*, under *Customizing the System Environment*, for the details on how to enable serial ports. The following example shows how to use a terminal to test the serial ports.

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/ttya01s
```

(Assuming the terminal is connected to **`ttya01s`**.)

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

Once you can redirect output to your screen with the test above, you may enable the ports with **`getty`**.

If you can tie the transmit and receive pins of a port together (with a breakout box, loopback plugs, etc.) another port testing option is available to you: the DPA (Digi Port Authority) utility, supplied by Digi, provides loopback test functionality (see page 44).

Reconfiguring an Installed Driver

Once the device driver has been installed, it may be reconfigured without having to reinstall it. To reconfigure an installed device driver, perform the following steps.

Note: Not all device driver revisions support reconfiguration. If the installed package (as determined by Step 2, below) is "**`DGIEPC100`**", the driver may not be reconfigured by this method. In this case, uninstall the driver and reinstall it with the desired configuration.

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

2. Enter the following command to list known Digi device driver packages:

   ```
   setld -i | grep DGI
   ```

   The command will produce output similar to the following:

   ```
   DGIEPC210           Digi AccelePort Device Driver Package
   DGIEPC211           Digi AccelePort Device Driver Package
   DGIEPC220 installed Digi AccelePort Device Driver Package
   ```

   The key word "installed" indicates which driver package is currently installed, and the first column of that line is the name of the installed package. This name will be used in subsequent steps.

   If the command produces no output, or if "installed" does not appear in any of the lines, then no Digi device driver package is installed and reconfiguration is not possible. If the device driver name is "**`DGIEPC100`**", reconfiguration is not available.

3. Enter the following command to reconfigure the device driver:

   `setld -c` *name* **`config`**

   Where *name* is the package name as determined in Step 2.

4. The procedure may give you the option to use the current saved configuration at this point. If you choose the saved configuration, the remainder of the process proceeds automatically and you will not be prompted for a new configuration. *If you wish to reconfigure the device driver, do not choose to use the current saved configuration*.

5. The remainder of the reconfiguration process is the same as the original configuration process, beginning with Step 3, on page 9.

Uninstalling the Digital UNIX Device Driver

Enter the following commands to *uninstall* the device driver software:

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

2. Enter the following command to list known Digi device driver packages:

```
setld -i | grep DGI
```

The command will produce output similar to the following:

```
DGIEPC210           Digi AccelePort Device Driver Package
DGIEPC211           Digi AccelePort Device Driver Package
DGIEPC220 installed Digi AccelePort Device Driver Package
```

The key word “installed” indicates which driver package is currently installed, and the first column of that line is the name of the installed package. This name will be used in subsequent steps.

If the command produces no output, or if “installed” does not appear in any of the lines, then no Digi device driver package is installed, and there is nothing to remove.

3. Enter the following command:

`setld -d` *name*

Where *name* is the package name as determined in Step 2.

4. If the driver you are removing is any revision greater than **`DGIEPC100`**, you will be asked if you would like to save the current configuration. If you choose to do so, files will be saved on your system which may be used to quickly configure the system when you install a new device driver package (only the first three steps of the installation procedure will be required).

5. Rebuild the kernel by issuing the following command:

`doconfig -c` <*name*>

where *name* is the SCF (System Configuration File) you wish to use.

Consult your *System Administrator's Guide* for more details about **`doconfig`**.

6. Replace the current kernel with the new kernel.

7. Now perform a system shutdown and reboot to activate the new kernel.

Digi TTY Devices

This device driver supports four different UNIX devices on each line. On line "**a01**," where "**a**" refers to concentrator **a**, and "**01**" refers to first line (port) on that concentrator, there are four devices:

`/dev/ttya01m`

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

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

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

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

`/dev/ttya01s`

Standard device. This is the same as **`/dev/ttya01m`** with the exceptions that the default handshake method is XON/XOFF, and that Data Carrier Detect need not be present to open the device. *Once a connection is established and DCD becomes active, standard devices behave in the same way as modem devices—subsequent loss of the Data Carrier Detect signal will cause the jobs to be killed and the user will be automatically logged off.*

/dev/pra01s
/dev/pra01m

The “Transparent Print Devices” (DigiPRINT), for use with the auxiliary printer port of a terminal *only*. Printers which are directly connected to the Digi serial ports are accessible via the tty devices. Output directed to a **pr** device goes out the auxiliary port of a terminal while the user continues to use the terminal normally. As with the tty devices, an “**s**” device and an “**m**” device are both created to allow one to select the default flow control and DCD detection behavior.

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

For correct operation, several parameters must be provided to the *ditty* program. It is usually best to set these up in **/sbin/rc2.d**, since they must be set each time the system is rebooted. See *Setting Terminal Options with ditty* (page 31) and *DigiPRINT Transparent Print Option* (page 40).

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

```
ditty term wyse60 maxcps 80 bufsize 2000 maxchar 50 /dev/ttya01s
```

The driver will also create the **/dev/epcadl** device. This device is used by Digi support utilities for driver configuration and maintenance, and is *not* to be used as a serial device port.

The following is an example directory listing of the Digi ports on a running system:

```
  crw-------   1 root   system    51,  0 Jul 10 11:02 /dev/epcadl
  crw-rw-rw-   1 root   system    52,  2 Jun 24 09:21 /dev/pra01m
  crw-rw-rw-   1 root   system    52,  3 Jun 24 09:21 /dev/pra01s
  crw-rw-rw-   1 root   system    52,  0 Jun 24 09:21 /dev/ttya01m
  crw-rw-rw-   1 root   system    52,  1 Jun 25 05:48 /dev/ttya01s
```

Terminal Port Administration

This section is broken into three parts:

- Setting Terminal Options—a description of the Digi supplied utility which allows one to configure the terminal port settings associated with a Digi adapter, as well as a description of the Digi specific terminal options
- Enabling Terminal Logins—a description of the use of the Digital UNIX **`inittab`** and **`gettydefs`** files and the **`getty`** utility to enable Digi ports for terminal login
- Setting Up Serial Printers—a description of the use of the **`lprsetup`** utility to configure a serial printer directly connected to a Digi port

Note:

This section is NOT meant to be a complete guide terminal port administration and use. For more information regarding the Digital UNIX files and utilities described below, please see your system documentation.

Setting Terminal Options with ditty

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

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

The format is:

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

With no options, **`ditty`** displays all 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.

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`**.

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

ttyname Set and display options for the given tty device, instead of standard input. *ttyname* can be the full pathname (e.g. **`/dev/ttya01s`**), or a simplified pathname where the reference to **`/dev`** is implied (e.g. **`ttya01s`**). This option may be used on a modem control line when no carrier is present.

The following options specify transient actions to be performed immediately:

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

`flush` Immediately flush (discard) tty input and output.

`flushin` Flush tty input only.

`flushout` Flush tty output only.

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

[-]fastbaud Alter the baud rate tables by the following mapping scheme.

	Xr, Xem, EPC/X	*Xr 920*	*C/X*
Specified Baud Rate:	**Baud Rate Mapped to:**	**Baud Rate Mapped to:**	**Baud Rate Mapped to:**
50	57600	57600	57600
75	76800	76800	75
110	115200	115200	115200
134	14400	14400	14400
150	57600	57600	57600
200	230400	230400	200
300	76800	76800	300
600	115200	115200	115200
1200	230400	230400	1200
1800	28800	28800	28800
2400	2400	460800	2400
4800	4800	921600	4800
9600	9600	9600	9600
19200	19200	19200	19200
38400	38400	38400	38400

Note:

It may be necessary to reduce the polling interval of the driver in order to get maximal throughput at 921600 baud with the Xr 920. *Care must be taken when changing the polling interval*. See the discussion of the **ditty poll** parameter, on page 36 for details.

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

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

`maxcps` ***n***

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

`maxchar` ***n***

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

`bufsize` ***n***

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

`onstr` ***s***

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

`offstr` ***s***

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

`term` ***t***

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

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

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

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

stopin Activate flow control to stop input.

startin Release flow control to resume stopped input.

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

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

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

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

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

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

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

Most options affect specific ports which are named as parameters. The following option is a special one, designed for use *only by advanced users of the Digi drive*r. It affects the entire driver (and consequently all ports on all controlled adapters); the port specification is required to identify the driver to be affected. This option is "sticky"—the effects continue until the system is rebooted or until the option is changed.

`poll [`*n*`]` Without a parameter, display the current driver polling interval in microseconds. With a parameter, change the current driver polling interval to the supplied number of microseconds. If the supplied value is outside of the valid range (2000 to 500000), the polling rate will not change and the current value will be displayed.

Care must be taken when changing the polling rate. A smaller rate will increase the efficiency of the driver by servicing read and write requests more often, but will also increase the time that the processor spends servicing the driver, and may cause a perceptible slowing of the system. A larger rate will affect the connections to the driver, and may result in bursty data transfer and sluggish responses in interactive sessions, but will decrease the time that the processor needs to devote to the driver and possibly result in a perceptible speeding up of the system.

Also see **`stty(1)`**, **`ioctl(2)`**, **`termios(4)`**, and **`termcap(3)`**.

Enabling Terminal Logins

When the Digi configuration procedure is completed, the software will have added a number of entries to the **/etc/inittab** file on your system. This file controls the automatic creation and recreation of system processes, such as terminal logins. For instance, the entries for port “1” on concentrator “a” are:

```
digia01s:23:off:/usr/sbin/getty /dev/ttya01s 9600
digia01m:23:off:/usr/sbin/getty /dev/ttya01m M9600
```

In order to enable either of these devices (note that only *one* of the devices that are associated with a given port may enabled at one time) edit the **/etc/inittab** file, locate the entry for the desired port, and change the “off” to “respawn”. If done, the **inittab** entries would read (if one enabled the standard device):

```
digia01s:23:respawn:/usr/sbin/getty /dev/ttya01s 9600
digia01m:23:off:/usr/sbin/getty /dev/ttya01m M9600
```

During the next reboot, when the system reaches either run level 2 or 3, a **getty** will be spawned to allow a login on **ttya01s**, and it will be re-spawned automatically if getty ever exits (for a description of **getty**, see below). To force the system to reread the **inittab** file right away, execute the command:

```
init q
```

NOTE: See *Digi TTY Devices* for a description of the differences between the “**s**” device and the “**m**” device on a given port.

The first parameter to the **getty** command is the port which should be enabled for login. The second parameter is an identifier which describes the terminal port settings to be used during the login session. This identifier corresponds to an entry in the **/etc/gettydefs** file. A sample from a **gettydefs** file could include:

```
M2400# B2400 HUPCL CS8 # B2400 SANE IXANY TAB3 HUPCL #login: #M9600
M9600# B9600 HUPCL CS8 # B9600 SANE IXANY TAB3 HUPCL #login: #M2400
9600# B9600 HUPCL CS8 CLOCAL # B9600 SANE IXANY TAB3 CLOCAL #login: #9600
```

Each entry consists of “**#**” separated columns: the first column is the identifier string, the second column are the terminal parameters to set before login, the third column are the terminal parameters to set after a successful login, the fourth column is the desired login prompt, and the last column is an identifier which specifies the next entry to use if the baud rate is incorrect when using the current entry. The terminal parameters in the second and third columns correspond to those described in the "**`termios`**" manual page.

To change the default settings for a terminal login on a Digi port:

- Select or create an entry in the **`/etc/gettydefs`** file.
- Change the second argument of **`getty`** in the corresponding entry of the **`/etc/inittab`** file to the identifier for the selected **`gettydefs`** entry.
- Issue the **`init q`** command to force the system to reread the **`inittab`** file.

For more information on **`getty`**, **`gettydefs`**, **`inittab`** and **`termios`**, please see your Digital UNIX documentation.

Setting Up Serial Printers

The Digital UNIX utility which is used to configure printers is **`lprsetup`**. To configure a printer directly connected to a Digi serial port with this utility:

- Be sure that this port is not being used for terminal logins by changing any entries set to **`respawn`** to **`off`** for the given port in **`etc/inittab`**. Be sure to execute the **`init q`** command if any changes have been made.
- Run the **`lprsetup`** utility. Be sure to set the **`ct`** parameter to **`dev`** and the **`lp`** parameter to an appropriate **`/dev/tty`** device for the given port.

Important! DO NOT use one of the **`/dev/pr`** ports, as these ports are reserved for transparent print (see page 40) purposes and will not work correctly for directly connected printers.

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 **`/etc/inittab`**, 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 **`sbin/rc`** system initialization file. (Put them in a text file in the directory **`/sbin/rc2.d`**—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/pra01s`**

ditty Examples

Example 1:

The following command configures the DigiPRINT options for a DEC VT100 terminal connected to **`/dev/ttya01s`**. (Note that the printer uses **`/dev/pra01s`**). **`maxcps`**, **`maxchar`** and **`bufsize`** are left to defaults. Type:

```
ditty term vt100 /dev/ttya01s
```

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/ttya01s
```

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/ttya01s
```

Digi Port Authority

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

The DPA software also shows the status of the individual asynchronous ports by displaying a simulated concentrator front panel. Each of the eight supported EIA-232 signals is displayed, along with input and output flow control status and the status of the host adapter to concentrator connection status.

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

Hardware:	AccelePort C/X, EPC/X, Xem, Xr and Xr 920
Software:	Digital UNIX V3.x & 4.x device driver

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

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

Using the DPA Software

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

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

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

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

To use the Digi Port Authority to perform a loopback function test on a given port, do the following:

- Be sure that no processes are currently using the port.
- Tie the read and write pins of the serial port together. This can be done, for instance, with a breakout box or a specially designed loopback plug.
- Navigate to the “port” display in DPA for the port in question. This display shows the active EIA-232 signals, and the loopback function should be mentioned in the status bar at the bottom of the display.
- Hit **`t`** to begin the loopback testing.

The loopback test works by placing test data directly into the write buffer of the adapter and testing the input results for accuracy.

The following is a list of keystrokes and their function key equivalents in Digi Port Authority:

Function Key	Action	Keystroke
F1	Help	?
Up Arrow	Move Up	k
Down Arrow	Move Down	j
Left Arrow	Move Left	h
Right Arrow	Move Right	l
Page Down	Next Page	N
Page Up	Previous Page	P