



# RealPort Installation

Microsoft Windows and Linux operating systems

---

User Guide

## Revision history—90000630

Revision	Date	Description
E	May 2021	Updated to include only supported operating systems.

## Trademarks and copyright

Digi, Digi International, and the Digi logo are trademarks or registered trademarks in the United States and other countries worldwide. All other trademarks mentioned in this document are the property of their respective owners.

© 2021 Digi International Inc. All rights reserved.

## Disclaimers

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, 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.

## Warranty

To view product warranty information, go to the following website:

[www.digi.com/howtobuy/terms](http://www.digi.com/howtobuy/terms)

## Send comments

**Documentation feedback:** To provide feedback on this document, send your comments to [techcomm@digi.com](mailto:techcomm@digi.com).

## Customer support

**Digi Technical Support:** Digi offers multiple technical support plans and service packages to help our customers get the most out of their Digi product. For information on Technical Support plans and pricing, contact us at +1 952.912.3444 or visit us at [www.digi.com/support](http://www.digi.com/support).

# Contents

---

## RealPort Installation User Guide

Device support .....	6
About this guide .....	6
Purpose .....	6
Audience .....	6
Conventions: special fonts .....	6
Conventions: square brackets .....	6
Conventions: italics .....	6
Conventions: vertical bar .....	6

## Get started: Install RealPort for Microsoft Windows and Microsoft Windows Server

Configure the Digi device for RealPort .....	8
Install RealPort on the computer .....	9
Download and run Digi Device Discovery Utility .....	9
Configure Encrypted RealPort .....	9
Optional configuration .....	10
Configure a printer .....	11
Configure a modem .....	11
Manage the RealPort driver .....	11
Upgrade the RealPort driver .....	11
Remove RealPort .....	11
Access and change the RealPort configuration .....	12
Start and stop the RealPort driver .....	12

## Get started: Install RealPort for LINUX

Before you begin: RealPort requirements for Linux .....	13
Requirements and considerations .....	13
Information to gather .....	14
Configure the Digi device for RealPort .....	14
Install the RealPort driver .....	15
Install the Driver: RPM methods .....	15
Install the driver: TGZ method .....	19
RealPort devices created for each Digi device port .....	21
Manage a Digi device with Digi RealPort Manager .....	22
Add a Digi device .....	22
Delete a Digi device .....	23

Start a RealPort daemon .....	24
Stop a RealPort daemon .....	24
Monitor port status .....	24
Digi RealPort Manager screen .....	25
Ports Window information .....	25
Manage a Digi Device with dgrp_cfg_node .....	26
init operation .....	26
uninit operation .....	27
stop operation .....	28
start operation .....	28
Uninstall the driver .....	28
Uninstall with RPM .....	29
Uninstall from a TGZ archive .....	29
Linux configuration and technical information .....	29
Device configuration overview .....	30
Configure a device for a terminal .....	31
Configure a device for a printer .....	32
Configure a device for a modem .....	32
Configuring transparent printers .....	33
Setting TTY options .....	35
Fastbaud data rate mapping .....	37

## Troubleshooting

Troubleshooting requirements .....	39
Make sure device is properly configured for use with RealPort .....	39
Verify the configuration and operation of the driver .....	40
Verify in Windows .....	40
Verify in Linux .....	41
Test communications .....	41
Advanced configuration .....	42
Serial port problems .....	42
Make sure the port server is configured correctly .....	42
Check for stuck processes .....	43
Make sure the RealPort host can reach the Digi device .....	44
Conduct a loopback test to test ports outside RealPort (all UNIX and Microsoft Windows operating systems) .....	44
Loopback plug pin-out information .....	45
Trouble accessing port .....	46
Cabling .....	46
Troubleshoot issues in Windows .....	46
None of the ports work .....	46
Some of the ports do not work .....	47
Key concept: port mapping offsets .....	48
Troubleshoot port issues in Linux .....	48
If none of the ports work .....	48
If some of the ports do not work .....	49

# RealPort Installation User Guide

---

Digi's patented RealPort software provides serial connectivity over IP Networks, no matter where the devices reside. The software is installed directly on the server and allows applications to talk to devices across a network as though the devices were directly attached to the server. In actuality, the devices are connected to a Digi device server or terminal server somewhere on the network.

RealPort is unique among COM port re-directors. It is the only implementation that allows multiple connections to multiple ports over a single TCP/IP connection. Other implementations require a separate TCP/IP connection for each serial port. Unique features also include full hardware and software flow control, as well as tunable latency and throughput.

Device support .....	6
About this guide .....	6

## Device support

Digi's RealPort software works with most of Digi's products including ConnectPort TS and LTS, PortServer, Digi One Connect ES and SP as well as the majority of Digi's Cellular routers and Intelligent gateways.

## About this guide

### Purpose

Use this guide to install and configure RealPort on a variety of operating systems.

### Audience

This guide is intended for the person responsible for installing and configuring RealPort. This person should have experience configuring network devices and be familiar with networking concepts.

### Conventions: special fonts

This font is used for any input or examples you need to enter:

---

```
set config
```

---

### Conventions: square brackets

Optional parameters are displayed within square brackets.

---

```
set config [dhcp=on]
```

---

**Note** The square brackets themselves are not actually part of the command, and should not be entered.

---

### Conventions: italics

Variables are displayed in italics.

---

```
set config ip=ip-address
```

---

**Note** Substitute an appropriate IP address for *ip-address* in the preceding command.

---

### Conventions: vertical bar

A vertical bar character (|) is used to denote a choice (logical "or").

---

```
set flow={on|off}
```

---

The preceding command would be entered as either:

---

```
set flow=on
```

---

or

---

set flow=off

---

# Get started: Install RealPort for Microsoft Windows and Microsoft Windows Server

---

**Note** For information about currently supported versions, see [www.digi.com/support/realport](http://www.digi.com/support/realport).

---

To use RealPort software, you must configure your Digi devices to use RealPort and then install RealPort on a server.

1. [Configure the Digi device for RealPort](#)
2. [Install RealPort on the computer](#)

If your data is sensitive, you should enable the Encrypted RealPort feature. Encrypted RealPort is a security measure to maintain data integrity.

- [Configure Encrypted RealPort](#)

When installation is complete, you can configure peripherals on RealPort ports.

- [Configure a printer](#)
- [Configure a modem](#)

You can upgrade, remove, or change the RealPort configuration. See [Manage the RealPort driver](#).

If you have RealPort installation or management questions, see [Troubleshoot issues in Windows](#).

## Configure the Digi device for RealPort

Use this procedure to configure the Digi device for use with Digi RealPort drivers.

---

**Note** See the appropriate *Command Reference* or *Configuration and Administration Guide* for the Digi device for device-specific information on how to perform the steps below.

---

For the ConnectPort TS/LTS and PortServer/Digi One product lines:

1. Access a root prompt on the Digi device.
2. Set the Digi device's IP address.
3. Verify that the RealPort TCP port number is set to **771**. If you need to change the default value, you need to change it on both the device and on the computer where you installed the driver.



4. For all ports that will use RealPort, set the device type to **realport**. The syntax below sets ports 2 through 16 to RealPort:

- For the Portserver TS and Digi One product lines:

---

```
#> set profile ra=2-16 profile=realport
```

---

- For the Connectport TS/LTS and Digi Connect product lines:

---

```
#> set profile profile_type=realport port=2-16
```

---

For other products not listed, please refer to the appropriate product hardware manual for the exact steps required.

## Install RealPort on the computer

You must download and run the RealPort utility. You can use one of the following methods.

### Download and run Digi Device Discovery Utility

1. Navigate to [www.digi.com/support/realport](http://www.digi.com/support/realport).
2. From the **Operating System Specific Drivers** list box, select the appropriate Microsoft Windows option. A list of driver options appears.
3. Click the desired **RealPort for Windows** version. The file is downloaded, and a Windows Explorer window launches, showing the RealPort files.
4. When the download is complete, open the .zip file and click the **setup.exe** file. The **Digi RealPort Setup Wizard** appears.
5. Select **Add a New Device**.
6. Follow the steps in the wizard to install RealPort.
7. Click **Finish** to close the wizard.

## Configure Encrypted RealPort

Encrypted RealPort is a security feature that maintains data integrity. It prevents unauthorized changes in data, including intentional destruction or alteration, tampering, duplication, or accidental loss. Encrypted RealPort also prevents disclosure to unauthorized individuals or processes. If your data is sensitive, Encrypted RealPort is strongly recommended.

---

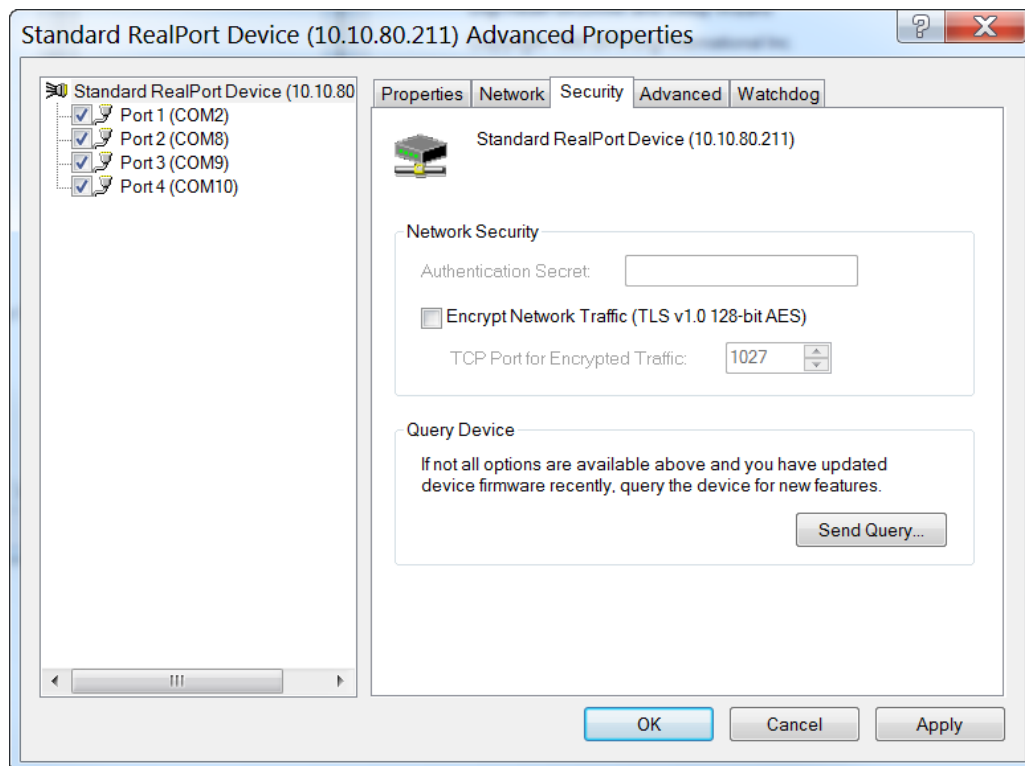
**Note** These instructions are for ConnectPort TS/LTS or PortServer/Digi One. The instructions for other products are similar, and for detailed instructions consult your product's user guide.

---

To implement Encrypted RealPort use the following procedure:

1. Follow the standard Windows process to access the **Device Manager** from your computer's operating system.
2. Select **Multi-port Serial Adapters**.
3. Right-click on your device. Click the Properties menu option. The **Properties** dialog appears.

4. Click the **Advanced** tab.
5. Click **Properties**. The **Advanced Properties** dialog appears
6. Click the **Security** tab.



7. Select the **Encrypt Network Traffic** check box to enable encrypted network traffic. When you select this option, the **TCP Port for Encrypted Traffic** field becomes available.
8. The **TCP Port for Encrypted Traffic** field has a default value of **1027**. The entry must match the device's TCP port setting. You can verify the setting on the device.
  - a. Open browser window.
  - b. Enter the IP address in the URL address bar to access the web interface.
  - c. Choose **Network > Network Services Settings**.
  - d. Select the **Enable Encrypted RealPort** option and verify that the port number is **1027**.
  - e. Click **Apply**.
9. Click **OK**.
10. Click **Apply**.

## Optional configuration

These devices can be optionally configured.

## Configure a printer

Use the standard Windows procedures to configure a printer on supported versions of Microsoft Windows running RealPort. Make sure you select the printer attached to the RealPort port.

## Configure a modem

Use the standard Windows procedures to configure a modem on supported versions of Microsoft Windows running RealPort. Make sure you select the modem attached to the RealPort port.

## Manage the RealPort driver

You can upgrade, remove, or change the RealPort configuration.

### Upgrade the RealPort driver

Upgrading the RealPort driver follows the same process as the initial installation. You should select the RealPort driver version that is newer than the currently installed version.

For detailed installation instructions, see [Install RealPort on the computer](#).

1. Navigate to [www.digi.com/support/realport](http://www.digi.com/support/realport).
2. From the **Operating System Specific Drivers** list box, select the appropriate Microsoft Windows option. A list of driver options appears.
3. Click the desired **RealPort for Windows** version. The file is downloaded, and a Windows Explorer window launches, showing the RealPort files.
4. When the download is complete, open the .zip file and click the **setup.exe** file. The **Digi RealPort Setup Wizard** appears.
5. Select **Update Digi RealPort Software**.
6. Follow the steps in the wizard to install RealPort.
7. Click **Finish** to close the wizard.

### Remove RealPort

Complete the following steps to remove RealPort software.

1. Navigate to [www.digi.com/support/realport](http://www.digi.com/support/realport).
2. From the **Operating System Specific Drivers** list box, select the appropriate Microsoft Windows option. A list of driver options appears.
3. Click the desired **RealPort for Windows** version. The file is downloaded, and a Windows Explorer window launches, showing the RealPort files.
4. When the download is complete, open the .zip file and click the **setup.exe** file. The **Digi RealPort Setup Wizard** appears.
5. Select **Remove an Existing Device**.
6. Follow the steps in the wizard to uninstall RealPort.
7. Click **Finish** to close the wizard.

## Access and change the RealPort configuration

Use these procedures to access and/or change the RealPort configuration. Configuration changes might include adding or removing ports or changing port attributes.

---

**Note** These instructions are for ConnectPort TS/LTS or PortServer/Digi One. The instructions for other products are similar, and for detailed instructions consult your product's user guide.

---

1. Use the standard Windows process to access the **Device Manager**.
2. Select **Multi-port Serial Adapters** and expand the list if necessary.
3. Right-click on the Digi terminal server device and select **Properties**. The **Properties** dialog appears.
4. Click the **Advanced** tab.
5. Click **Properties**. The **Advanced Properties** dialog appears.
6. Make changes as needed. Click **Help** in the **Properties** tab for help in making configuration changes.

## Start and stop the RealPort driver

Use this procedure to start or stop the RealPort driver on the supported Microsoft Windows Server.

1. Use the standard Windows process to access the **Device Manager**.
2. Expand the **Multi-port Serial Adapters** list if necessary and select the terminal server.
3. Right-click on the terminal server to display the menu options.
  - Select **Disable** to stop the service.
  - Select **Enable** to start the RealPort driver.

## Get started: Install RealPort for LINUX

---

To install RealPort software, you must first configure your Digi devices to use RealPort, and then install RealPort on the Linux server.

1. [Before you begin: RealPort requirements for Linux](#)
2. [Configure the Digi device for RealPort](#)
3. [Install the RealPort driver](#)
4. Add and configure your Digi device in Linux. Use one of the following methods:
  - Graphical interface: [Manage a Digi device with Digi RealPort Manager](#)
  - Command-line interface: [Manage a Digi Device with dgrp\\_cfg\\_node](#)
5. Configure your ports for operation. See [Device configuration overview](#).

### Before you begin: RealPort requirements for Linux

Before you begin the installation process, review the RealPort requirements for Linux.

#### Requirements and considerations

Read this section before beginning the RealPort driver installation.

- To determine if a particular version of the Linux RealPort driver supports a particular version of Linux, check the release notes for that driver. To find the release notes,
  - a. Navigate to [www.digi.com/support/realport](http://www.digi.com/support/realport).
  - b. From the **Operating System Specific Drivers** list box, select the **Linux** option.
  - c. Click the **RealPort Driver for Linux, release notes** option.
- RealPort utilities are located in `/usr/bin/dgrp`.
- Once the RealPort driver is installed, man pages are available for a number of the utilities associated with the package, including:

Utility	Description
<b>ditty-rp</b>	Sets and displays RealPort TTY device options.

Utility	Description
<b>dgrp_cfg_node</b>	Used to add and remove Digi devices from the RealPort driver.
<b>drpd</b>	Digi RealPort network daemon.
<b>dgrp.o</b>	RealPort driver module. These are placed in <b>/usr/bin/dgrp</b> . The actual man page entry associated with the <b>dgrp.o</b> module is <b>dgrp</b> .
<b>dgrp_gui</b>	Digi RealPort Manager for x-windows.

## Information to gather

Before you install RealPort, determine:

- The hostname or IP address to assign each unconfigured Digi device.
- The number of ports for each Digi device. Include any PORTS/em Modules that are attached to the Digi device.
- Determine the link speed between the host machine and the Digi device if a slow WAN link (for example, a cellular or a legacy leased line connection) connects them.

## Configure the Digi device for RealPort

Use this procedure to configure the Digi device for use with Digi RealPort drivers.

---

**Note** See the appropriate *Command Reference* or *Configuration and Administration Guide* for the Digi device for device-specific information on how to perform the steps below.

---

For the ConnectPort TS/LTS and PortServer/Digi One product lines:

1. Access a root prompt on the Digi device.
2. Set the Digi device's IP address.
3. Verify that the RealPort TCP port number is set to **771**. If you need to change the default value, you need to change it on both the device and on the computer where you installed the driver.
4. For all ports that will use RealPort, set the device type to **realport**. The syntax below sets ports 2 through 16 to RealPort:

- For the Portserver TS and Digi One product lines:

---

```
#> set profile ra=2-16 profile=realport
```

---

- For the Connectport TS/LTS and Digi Connect product lines:

---

```
#> set profile profile_type=realport port=2-16
```

---

For other products not listed, please refer to the appropriate product hardware manual for the exact steps required.

## Install the RealPort driver

Digi supports two distribution methods for the RealPort device driver package that is installed on the Linux server: RPM and TGZ.

When installation is complete, the Linux RealPort driver creates three different devices for each physical port of each Digi device: a [standard TTY device](#), a [callout device](#), and a [transparent print device](#). See [RealPort devices created for each Digi device port](#).

The installation methods are described in the table below.

Method	Description
RPM	This method uses a source RPM package. The source RPM method depends on rpm tools. If your Linux distribution does not support RPM packages, or if your system does not have the RPM tools installed, you must either locate and install the RPM manipulation tools or use the TGZ method. To install with this method, see <a href="#">Install the Driver: RPM methods</a> . Digi does not support the distribution of the RealPort software via binary RPMs.
TGZ	This method uses a compressed archive of the source. The TGZ method, while relatively simple, does not have the advantage of the common package management operations of RPM-based packages. It does, however, have the advantage that far more systems will support the TGZ method out-of-the-box. To install with this method, see <a href="#">Install the driver: TGZ method</a> .

### Install the Driver: RPM methods

There are two methods of installing with RPM:

- [Standard installation procedure](#): This method is automated, but there is user control at every step.
- [Custom installation procedure](#): This method is intended for users who wish to customize the installation; for example, to change the destination directories of certain tools. This method is the most complex and should be used only if necessary.

The procedures in this topic should be used only in Linux environments that support the installation and building of packages from an RPM repository. The procedures require that the following RPM directories exist and are used by RPM on your Linux system:

- **/usr/src/redhat/BUILD**
- **/usr/src/redhat/RPMS**
- **/usr/src/redhat/RPMS**
- **/usr/src/redhat/SOURCES**
- **/usr/src/redhat/SPECS**
- **/usr/src/redhat/SRPMS**

### **Package version and revision**

Certain commands used in the following procedures must be entered with the correct package version and revision number.

To determine package version and revision numbers, use this command:

---

```
#> rpm -qp filename
```

---

**Example:**

If the package version of the Linux driver is 1.9 and the revision is 38, enter the version and revision level as follows:

For this command variable:	Use this value:
<i>version</i>	1.9
<i>revision</i>	38

For this command:	Enter:
<code>rpm -ivv dgrp-version revision.src.rpm</code>	<code>rpm -ivv dgrp-1.9-38.src.rpm</code>

For more information about current RealPort drivers, see <https://www.digi.com/realport>.

**Standard installation procedure**

Use this procedure to install the RealPort Linux driver. See [Install the Driver: RPM methods](#) for information on entering commands in this procedure. See the [release notes](#) for additional information.

- Download the source RPM package located on the Digi Support site, [www.digi.com/support](http://www.digi.com/support) .  
For consistency, consider downloading the source RPM to the directory **/usr/src/redhat/SRPMS**.
  - Navigate to [www.digi.com/support/realport](http://www.digi.com/support/realport).
  - From the **Operating System Specific Drivers** list box, select the **Linux** option.  
A list of driver options appears.
  - Select the **RealPort Driver for Linux, src.rpm version** option to download the file.
- Install the source code in the **SOURCES** directory used by RPM (**/usr/src/redhat/SOURCES**) with this command:

---

```
#> rpm -ivv dgrp-version-revision.src.rpm
```

---

This command also copies the specification file

(**/usr/src/redhat/SPECS/dgrp-v.v.spec**) to the **SPECS** directory.



3. Create an RPM specific to your platform by executing the appropriate commands for your platform, as shown in the following table.

OS version	Commands
Red Hat	<pre># rpmbuild --rebuild --define DISTRO=REDHAT_XX (package) # rpm -ivv /usr/src/redhat/RPMS/i386/dgrp-#.#.#.i386.rpm # insmod dgrp</pre> <p>Actual DISTRO Values:                      Red Hat AS/ES/WS 2.1 = <b>REDHAT_21</b>                      Red Hat AS/ES/WS 3.0 = <b>REDHAT_ES3</b>                      Red Hat Fedora Core 1,2, or 3 = <b>FEDORA</b>                      Red Hat 9.0 = <b>REDHAT_90</b>                      Red Hat 8.0 = <b>REDHAT_80</b>                      Red Hat 7.3 = <b>REDHAT_73</b>                      Red Hat 7.2 = <b>REDHAT_72</b></p> <p>There is no DISTRO flag required for Red Hat AS/ES/WS 4</p>
SuSE 8.x	<pre># rpm --rebuild (package) # rpm -i /usr/src/packages/RPMS/i586/dgrp-#.#.#.i586.rpm # insmod dgrp</pre>
SuSE 9.x	<pre># rpmbuild --rebuild (package) # rpm -i /usr/src/packages/RPMS/i586/dgrp-#.#.#.i586.rpm # insmod dgrp</pre>
Mandrake	<pre># rpmbuild --rebuild (package) # rpm -i /usr/src/RPM/RPMS/i586/dgrp-#.#.#.i586.rpm # insmod dgrp</pre>
Debian	<pre># rpmbuild --rebuild (package) # cd /usr/src/rpm/RPMS/i386 # alien -d dgrp-#.#.#.i386.rpm # dpkg -i dgrp_#.#.#_i386.deb # insmod dgrp</pre>

For example, for the Red Hat platform, the commands are:

---

```
cd /usr/src/redhat/SPECS
rpmbuild --rebuild --define DISTRO=REDHAT_XX (package)
rpm -ivv /usr/src/redhat/RPMS/i386/dgrp-#.#.#.i386.rpm
```

---

---

```
insmod dgrp  
(
```

---

4. Install the RPM created by the previous step using the following commands.

---

```
cd /usr/src/redhat/RPMS/arch  
rpm -ivv dgrp-version-revision.arch.rpm
```

---

The *arch* value should be replaced with a string representing your architecture, such as i386 or alpha. This value is system-dependent, and is chosen by your system when the RPM is built.

To install the RealPort driver on another system of the same architecture and Linux version, copy the following binary to the same location on the other system and execute the **rpm** command in this step:

---

```
/usr/src/redhat/RPMS/arch/dgrp-version-revision.arch.rpm
```

---

5. Use the following commands to verify that all files are installed correctly:

---

```
cd /usr/src/redhat/SPECS  
rpm -bl dgrp-version.spec
```

---

### **Custom installation procedure**

This custom installation procedure is reserved for users who need to change a portion of the installation procedure or its contents in their specific environment. Reasons for such changes include the need to change the directory where files are installed or to modify the source code.

---

**Note** Modifications to the source package may cause Digi to refuse support for that package in that environment.

---

See [Install the Driver: RPM methods](#) for important information on entering commands in this procedure.

1. Download the RPM package from Digi's Support site. For consistency, consider downloading the source RPM to the directory **/usr/src/redhat/SRPMS**.
  - a. Navigate to [www.digi.com/support/realport](http://www.digi.com/support/realport).
  - b. From the **Operating System Specific Drivers** list box, select the **Linux** option.  
A list of driver options appears.
  - c. Select the **RealPort Driver for Linux, src.rpm version** option to download the file.
2. Install the source code in the **SOURCES** directory used by RPM, **/usr/src/redhat/SOURCES**, with this command:

---

```
rpm -ivv dgrp-version-revision.src.rpm
```

---

This command also copies the specification file **/usr/src/redhat/SPECS/dgrp-version.spec** to the **SPECS** directory.

3. Use the RPM tools to open the source archive:

```
cd /usr/src/redhat/SPECS
rpm -bp dgrp-version.spec
```

The **-bp** option specifies that only the preparation section (**%prep**) of the specification file should be executed. This might result in the source files being uncompressed, removed from the archive, and placed in the following directory:

```
/usr/src/redhat/BUILD/dgrp-version
```

4. Make desired modifications to the source files and/or the specification file. The source files are found in the directory.

```
/usr/src/redhat/BUILD/dgrp-version
```

The specification file is named:

```
/usr/src/redhat/SPECS/dgrp-version.spec
```

Additionally:

- Save the original and final versions of any changed file after making and testing the changes. Put these backup copies somewhere other than the **/usr/src/redhat** tree.
- If you change the final location of any of the files in the package, update the **%file** list in the specification file. Otherwise, a message that the installation was incomplete is displayed, because some files were not found.
- Digi recommends that you document changes in the **%description** section of the specification file.

## Install the driver: TGZ method

The following procedure can be used in any Linux environment. It is the only choice available for environments which do not support RPM.

Certain commands used in the following procedure need to be entered with the correct package version and revision number. Refer to the [release notes](#) for version and revision (release) numbers.

For this command:	Enter:
<code>tar -xvzf dgrp-version revision.tgz</code>	<code>tar -xvzf dgrp-1.0-2.tgzrpm</code>

1. Download the **Linux RealPort TGZ** file from the Digi Support site.
  - a. Navigate to [www.digi.com/support/realport](http://www.digi.com/support/realport).
  - b. From the **Operating System Specific Drivers** list box, select the **Linux** option.  
A list of driver options appears.
  - c. Select the **RealPort Driver for Linux, tgz version** option to download the file.
2. Select a directory (such as **/usr/src**) where the source tree will reside, and unpack the compressed archive file there. For example:

---

```
cd /usr/src
tar -xvzf dgrp-version-revision.tgz
```

---

This will create a subdirectory called **dgrp-version** containing all of the RealPort source files.

3. Change directory to the root of the source directory tree.

---

```
cd /usr/src/dgrp-version
```

---

4. Create the Makefile script:

---

```
./configure
```

---

**Note** Occasionally a Linux vendor will ship a kernel that cannot be autodetected to add the various changes that might be required for that specific kernel. Because of this, there is an option for both **srpm** and **tgz** to indicate exactly what distribution you have. Currently, the recognized options are for:

- Red Hat 7.2 - **REDHAT\_72**
- Red Hat 7.3 - **REDHAT\_73**
- Red Hat 8.0 - **REDHAT\_80**
- Red Hat 9.0 - **REDHAT\_90**
- Red Hat Fedora Core 1, 2, or 3 - **FEDORA**
- Red Hat AS/ES/WS 2.1 - **REDHAT\_AS\_21, REDHAT\_ES\_21, REDHAT\_WS\_21**
- Red Hat AS/ES/WS 3.0 - **REDHAT\_AS\_3 / REDHAT\_ES\_3 / REDHAT\_WS\_3**

For example, to indicate to **srpm** that you have Red Hat 9, run this command instead:

---

```
* rpmbuild --rebuild --define DISTRO=REDHAT_9 DRIVER_PACKAGE_NAME.srpm
```

---

To indicate to **tgz** that you have Red Hat AS 3, run this during the **configure** phase:

---

```
* ./configure DISTRO=REDHAT_AS_3
```

---

5. Examine the Makefile and make modifications as required by your system environment. Common items to check would include destination directories, naming conventions, and compiler details.
6. When satisfied with the state of the Makefile and other source files, compile the driver and its support tools by entering:

---

```
make all
```

---

7. Install the package components with:

---

```
make install
```

---

8. Register your module with the system initialization scripts by entering:

---

```
make postinstall
```

---

**Note** Do not delete the source tree since the makefiles are necessary for the uninstall procedure.

---

## RealPort devices created for each Digi device port

When RealPort is installed on the Linux server, the Linux RealPort driver creates three different devices for each physical port of each Digi device: a [standard TTY device](#), a [callout device](#), and a [transparent print device](#).

The devices are named according to the following convention:

---

```
[prefix][ID][port]
```

---

These elements are defined as follows.

Element	Description
<i>prefix</i>	Standard TTY devices have the prefix <b>tty</b> , callout devices have the prefix <b>cu</b> , and transparent print devices have the prefix <b>pr</b> .
<i>ID</i>	The RealPort ID for the Digi device associated with this port. A RealPort ID consists of one or two alphanumeric characters. An underscore character may be used for any of the two ID characters. See <a href="#">About the RealPort ID</a> .
<i>port</i>	The port number must consist of two digits. The ports are numbered beginning with <b>00</b> .

The following are examples of the devices that would be created for the first port of a Digi device with the RealPort ID **aa**:

Device Type	Full Path Name
Standard TTY device	<b>/dev/ttyaa00</b>
Callout device	<b>/dev/cuaa00</b>
Transparent print device	<b>/dev/praa00</b>

### Standard TTY devices

The behavior of the standard TTY devices is that of a modem controlled port. They require the Data Carrier Detect (DCD) signal to be high before they can operate. When used on a dial-in modem, the ports will wait for DCD before sending out the login prompt.

When these devices are used with a terminal or other locally connected device, you should wire the DCD signal to the remote equipment's Request To Send (RTS) line. When a terminal is then used for log-in, the system generates a prompt when the terminal is powered-on (RTS, and thus DCD is asserted). It also kills the user session if the terminal is powered-off (lowering the signals).

### Callout devices

Callout devices will be obsoleted in a future version of Linux, so they should generally be avoided. Data Carrier Detect (DCD) need not be present to open the device.

Once a connection is established and DCD becomes active, these devices behave in the same way as the standard TTY devices. Subsequent loss of the DCD signal will cause active processes on the port to be killed.

### Transparent print devices

Transparent print devices can be used with auxiliary printer ports on terminals. Output to the **pr** device goes out the auxiliary port of a terminal while you continue to use the terminal normally.

## Manage a Digi device with Digi RealPort Manager

Digi RealPort Manager can be used to perform these tasks.

---

**Note** The wish interpreter is used to support the GUI. The wish tcl interpreter needs to be in the Linux "path" before the **dgrp\_gui** command will work.

---

- [Add a Digi device](#)
- [Delete a Digi device](#)
- [Start a RealPort daemon](#)
- [Stop a RealPort daemon](#)
- [Monitor port status](#)
- [Digi RealPort Manager screen](#)
- [Ports Window information](#)

### Add a Digi device

Use this procedure to add a Digi device to your Linux System.

---

**Note** The wish interpreter is used to support the GUI. The wish tcl interpreter needs to be in the Linux "path" before the **dgrp\_gui** command will work.

---

1. Access Digi RealPort Manager by entering this command at a Linux prompt:

---

```
/usr/bin/dgrp-version/config/dgrp_gui
```

---

2. There are two ways to register a new Digi device with the RealPort package. From the RealPort Manager:
  - Press the **Add** button that appears at the bottom of the main screen.
  - Choose the **PortServer > Add New** from the menu.

The **Digi device settings** window appears.

3. Specify values for each of the fields in the window. Some of the fields have defaults. A prompt to install encrypted RealPort displays.

---

```

Would you like this RealPort session to be encrypted?
NOTE: Not all RealPort products support encrypted RealPort sessions.
Please check your RealPort product's firmware release notes
or product literature before selecting "always".
If in doubt, select "never".
(always/never) : (never): always
The following device will be configured,
0 10.1.3.68 16 aa secure (always)
Is this correct (y to add or x to abort) ? y

```

---

4. Click **Commit**. The RealPort **Command Logger** window appears, with an appropriate **dgrp\_cfg\_node** command.
5. Click **Run It** to execute the command and finish adding and initializing your Digi device. Observe any errors in the log window.

## Delete a Digi device

Use this procedure to delete a Digi device from your Linux System. The Digi devices is removed, along with all its device nodes.

**Note** The wish interpreter is used to support the GUI. The wish tcl interpreter needs to be in the Linux "path" before the **dgrp\_gui** command will work.

1. Access Digi RealPort Manager by entering this command at a Linux prompt:

---

```
/usr/bin/dgrp-version/config/dgrp_gui
```

---

2. Select the appropriate Digi device in the list box.
3. Use one of the following methods to start the deletion process:
  - Click **Delete** on the bottom of the main screen.
  - Choose **PortServer > Delete/Uninitialize** from the menu.

The **Digi PortServer Settings** window appears.

4. Confirm that the correct PortServer is being deleted and click **Remove**. The RealPort **Command Logger** window appears with an **dgrp\_cfg\_node** command.

5. Click **Run It** to execute the command and complete the removal of the Digi device, along with all of its device nodes.

## Start a RealPort daemon

Use this procedure to start a RealPort daemon for a Digi device.

---

**Note** The wish interpreter is used to support the GUI. The wish tcl interpreter needs to be in the Linux "path" before the **dgrp\_gui** command will work.

---

1. Access Digi RealPort Manager by entering this command at a Linux prompt:

```
/usr/bin/dgrp-version/config/dgrp_gui
```

---

2. Select the appropriate Digi device in the list box.
3. Choose **Daemon > Start Daemon** from the menu. The **Command Logger** screen appears, with a **dgrp\_cfg\_node** command.
4. Click **Run It** to execute the command and attempt to start the daemon. If the daemon is already running, the command silently exits.

## Stop a RealPort daemon

Use this procedure to stop a RealPort daemon or Digi device.

---

**Note** The wish interpreter is used to support the GUI. The wish tcl interpreter needs to be in the Linux "path" before the **dgrp\_gui** command will work.

---

1. Access Digi RealPort Manager by entering this command at a Linux prompt:

```
/usr/bin/dgrp-version/config/dgrp_gui
```

---

2. Select the appropriate Digi device in the list box.
3. Select **Daemon > Stop Daemon** from the menu. The **Command Logger** screen appears and a **dgrp\_cfg\_node** command displays.
4. Click **Run It** to execute the command and attempt to stop the daemon. If the daemon is not running, the command silently exits.

## Monitor port status

When monitoring ports, you can change the port monitoring delay by using the sliding scale along the bottom of the **Ports** window. Modem signal values are only correct for open ports. Ports in the **closed** or **waiting** state may not display the correct modem signals. When monitoring ports, the **Refresh** button can be selected at any time to update the port fields.

---

**Note** The wish interpreter is used to support the GUI. The wish tcl interpreter needs to be in the Linux "path" before the **dgrp\_gui** command will work.

---

Use the following procedure to monitor the status of a RealPort port.



1. Access Digi RealPort Manager by entering this command at a Linux prompt:

```
/usr/bin/dgrp-version/config/dgrp_gui
```

2. Choose the appropriate Digi device in the list box.
3. Choose **View > Ports**. The **Ports Window** appears.
4. Choose the appropriate port in the **Ports Window**.
5. Click **Modem Status** or select the **Ports > Modem Status**. The **Port Status** window appears. A signal shows a red box if the port is active or a gray box if it is inactive.

## Digi RealPort Manager screen

The main screen of the Digi RealPort Manager is used to manage the Digi devices registered with the RealPort driver. It lists all currently configured Digi devices in the central window, and provides buttons and menu options to manipulate these Digi devices.

Menu Field	Description
<b>RealPort ID</b>	The RealPort ID is used to designate your TTY devices for RealPort. It must be one or two alphanumeric characters and must be unique to your system. An underscore character can be used for an ID character. For example, a RealPort device name where the letters <b>rp</b> are used would be <b>TTYrp01</b> . See <a href="#">About the RealPort ID</a> .
<b>address</b>	The address may be specified as an IP number or IP name. No attempt is made to validate this address, nor to connect to the specified address during the installation.
<b>ports</b>	An integer indicating the number of device files which should be created for this Digi device. This does not have to match the physical number of ports, but physical ports beyond the number specified will not be available. The maximum port count allowed is 64.
<b>IP Port</b>	The Digi device IP port number. Normally this should be left at the default.
<b>Access Mode</b>	Sets the file protection mode for any device files created.
<b>Owner</b>	Sets the user ID of the file owner for any device files created. The value must be an integer.
<b>Group</b>	Sets the group ID of the file owner for any device files created. The value must be an integer.
<b>Link Speed</b>	Sets the link speed string. The default is <b>auto</b> , and can be used under most circumstances. See the <b>drpd(8)</b> man page for information on the custom speed parameters.

## Ports Window information

These are the fields of the **Ports Window**:

Fields	Description
<b>Port</b>	Port is the port number of an individual port. The first port on a Digi device is port number 0.
<b>Status</b>	The value of this field is either <b>open</b> , <b>closed</b> , or <b>waiting</b> . A port moves into the waiting state when there are devices waiting to open the port: either waiting for an event, or for another process to release the port.
<b>Speed</b>	The last known speed (in bps) of the port. If unknown (for instance, before the port is used for the first time), the value <b>0</b> is shown.
<b>Description</b>	This field is not yet active. The default value is <b>NA</b> .

## Manage a Digi Device with `dgrp_cfg_node`

The `dgrp_cfg_node` tool is a command-line based configuration program designed to automate a number of steps required to enable the serial ports on Digi devices for general use. The path to the `dgrp_cfg_node` executable is `/usr/bin/dgrp/config`.

The `dgrp_cfg_node` tool has four modes of operation:

Operation	Description
<b>init</b>	Can be used to add or reinitialize a Digi device in Linux. See <a href="#">init operation</a> .
<b>uninit</b>	Removes a Digi device from Linux. See <a href="#">uninit operation</a> .
<b>stop</b>	Stops a RealPort daemon. See <a href="#">stop operation</a> .
<b>start</b>	Attempts to start a RealPort daemon. See <a href="#">start operation</a> .

### init operation

The **init** operation has the following effects:

- Attempts, if necessary, to load the driver module.
- Determines whether a daemon is already running for the specified node. If it is not running, it starts the daemon.
- Creates all necessary device files in the `/dev` directory. If a device file exists, the ownership and permissions are preserved.

The standard usage requires a command such as:

```
dgrp_cfg_node init IDIPaddrports
```

Where the parameters are defined as follows:

Parameter	Definition
<b>init</b>	Indicates the operation being requested

Parameter	Definition
<i>ID</i>	Assigns the supplied ID to the Digi device. A RealPort ID consists of one or two alphanumeric characters. See <a href="#">About the RealPort ID</a> .
<i>IPaddr</i>	Either an IP address or an IP name may be assigned to the daemon via this parameter.
<i>ports</i>	Indicates the number of device files which should be created for this particular Digi device. This number does not have to match the physical number of ports, but any physical ports beyond the ports value specified will be unavailable. The maximum number of ports supported by the tool is 64.

For further details, see the **dgrp\_cfg\_node** man page, installed with this package.

### About the RealPort ID

The RealPort ID refers to an individual Digi device with an ID. You must choose the ID to use for each Digi device, and this ID must be unique within your system. The **dgrp\_cfg\_node init** operation allows you to assign an ID to a Digi device.

A RealPort ID consists of one or two alphanumeric characters.

Once a RealPort ID is assigned, the device files in the **/dev** directory, which are created by the package, encode the ID into the device name. This allows you to determine the Digi device ID for a particular device. For more information on the device naming, see [RealPort devices created for each Digi device port](#).

### uninit operation

The **uninit** operation has the following effects:

- Determines whether a daemon is running for the specified Digi device. If a daemon is running, this operation will kill it.
- Erases all information about this Digi device from its internal database. This is different from the behavior of the **stop** operation. See [stop operation](#).

The standard usage requires a command like:

---

```
dgrp_cfg_node uninit ID
```

---

The parameters are defined as follows:

Parameter	Definition
<b>uninit</b>	Indicates the operation being requested.
<i>ID</i>	Specifies which Digi device to unconfigure.

---

**Note** The command fails if any ports are in use. Be sure to kill all applications using the Digi device ports before uninitialization.

---

**Note** Even if you have removed all of the Digi devices you have previously registered with the driver, the **dgrp\_cfg\_node** utility does not unload the driver module.

---

For further details, see the **dgrp\_cfg\_node** man page, which is installed with this package.

## stop operation

The **stop** operation has the following effect:

- Determines whether a daemon is running for the specified Digi device. If a daemon is running, this operation will kill it.
- The RealPort software retains the information associated with this ID so that the daemon can be restarted with a minimum of information.

The simplest usage requires one to execute a command with a form similar to:

---

```
dgrp_cfg_node stop ID
```

---

The parameters are defined as follows:

Parameter	Definition
<b>stop</b>	Indicates the operation being requested.
<i>ID</i>	Specifies which Digi device's daemon to stop.

For example, you can stop a daemon in order to change the daemon's parameters, such as the IP address. In this case, you would use the **stop** operation to stop a daemon, change the parameters, and then use the **start** operation to restart the daemon with new parameters.

For further details, please see the **dgrp\_cfg\_node** man page installed with this package.

## start operation

The **start** operation determines whether a daemon is running for the specified node, and if not, attempts to execute a daemon based on the supplied parameters.

The simplest usage requires one to execute a command with a form similar to:

---

```
dgrp_cfg_node start ID IPAddr
```

---

The parameters are defined as follows:

Parameter	Definition
<b>start</b>	Indicates the operation being requested
<i>ID</i>	Specifies which Digi device's daemon to start.
<i>IPAddr</i>	The (possibly new) IP address or IP name to use when referring to the Digi device with the specified ID.

For further details, see the **dgrp\_cfg\_node** man page, which is installed with this package.

## Uninstall the driver

Use one of the following methods to uninstall the Linux driver.

## Uninstall with RPM

The procedures in this topic should be used only in Linux environments that support the installation and building of packages from an RPM repository. These procedures require that the following RPM directories exist and that they are used by RPM on your Linux system:

- **/usr/src/redhat/BUILD**
- **/usr/src/redhat/RPMS**
- **/usr/src/redhat/SOURCES**
- **/usr/src/redhat/SPECS**
- **/usr/src/redhat/SRPMS**

1. Use the **dgrp\_cfg\_node** command with the **uninit** option for each PortServer. This command kills each PortServer daemon and erases all information of that PortServer from the system.

---

```
dgrp_cfg_node uninit ID
```

---

Where *ID* specifies the ID of the PortServer daemon.

2. Remove the driver package using this command:

---

```
rpm -e -vv dgrp-version
```

---

## Uninstall from a TGZ archive

**Note** If you have deleted the source repository that was created when you installed this driver package, then there is no automated mechanism to remove the package files.

1. Access a command prompt at the root on the Linux system.
2. Change directories to the root of the source tree:

---

```
cd /usr/src/dgrp-version
```

---

Substitute your driver version numbers for *version* in the previous command. For example, if the driver version is 1.0, the command is:

---

```
cd /usr/src/dgrp-1.0
```

---

3. Enter these two make commands:

---

```
make preuninstall  
make uninstall
```

---

## Linux configuration and technical information

## Device configuration overview

Use these overviews of device configuration topics to decide which tool(s) you should use to configure your devices.

### *ditty-rp*

Characteristics	Actions
Features or Description	Digi's device configuration program. Run from a command prompt. The <b>ditty-rp</b> commands are normally included in a system startup file.
Uses	Manually configure Digi-specific device settings such as <b>altpin</b> or <b>forcedcd</b> . Configure printer devices. See <a href="#">Configure a device for a printer</a> .
Resources	Refer to the man page for <b>ditty-rp</b> .

### Configure a port

Device to be configured	Information location
Terminal	See <a href="#">Configure a device for a terminal</a> . This procedure describes how to configure a device for a terminal.
Printer	See <a href="#">Configure a device for a printer</a> . This procedure describes how to configure a device for a printer.
Modem	See <a href="#">Configure a device for a modem</a> . This procedure describes how to configure a device for a modem.

### Miscellaneous topics

Topic	Description
<b>altpin</b>	This option should be enabled on a port when an RJ-45 8-pin cable is used and the Data Carrier Detect (DCD) signal is required. For example, <b>altpin</b> should be enabled on a port where an RJ-45 8-pin cable is used with a modem. See <a href="#">About transparent printing</a> , and refer to the Cable Guide for your Digi product.
<b>fastbaud</b>	RealPort devices support baud rates in excess of the maximum baud rate supported by Linux. To enable the use of these fast baud rates, a <b>ditty-rp</b> parameter, <b>fastbaud</b> , has been provided. See <a href="#">Fastbaud data rate mapping</a> .

Topic	Description
Data Carrier Detect (DCD)	<p>In some cases, depending on your operating system requirements and/or your device requirements, it may be necessary for the Data Carrier Detect (DCD) signal to be active on a port before it will function. There are two ways to fulfill this requirement:</p> <ul style="list-style-type: none"> <li>■ <b>Cabling:</b> One way to fulfill the DCD requirement is to create a cable and have your device force the signal high. See the Digi One RealPort and PortServer Cable Guide.</li> <li>■ <b>ditty-rp:</b> Another way to fulfill the DCD requirement is to enable the <b>forcedcd ditty-rp</b> option. See <a href="#">About transparent printing</a>.</li> </ul>

## Configure a device for a terminal

Use this procedure to configure a Digi serial device for a terminal. See your operating system documentation for more information on configuring a serial device for a terminal.

### Procedure

1. Connect a serial cable between the port and terminal.
2. Edit the `/etc/inittab` file and add a **getty** entry for the device. The **getty** name that configures your device can vary from system to system. The following examples are only meant to serve as a guide. Substitute your device for `tyaa011` in these examples:

#### Example 1: RedHat getty\_ps

A RedHat terminal install script where **getty** is **getty\_ps** is:

---

```
dl:2345:respawn:/sbin/getty_ps tyaa11 DT9600 vt100
```

---

#### Example 2: Debian agetty

A common Debian terminal install script where **getty** is **agetty** is:

---

```
D1:23:respawn:/sbin/agetty -L tyaa11 19200 vt100
```

---

This is an **agetty** example for a hard-wired terminal:

---

```
D000:2345:respawn:/sbin/agetty -L 9600 tyaa01
```

---

The **-L** flag must be used with **agetty** for proper functionality if DCD is not wired on the cable. Please exclude the **-L** flag for DCD sensitivity.

#### Example 3: mgetty

A **mgetty** modem script which is available in both RedHat and Debian is:

---

```
T3:23:respawn: /sbin/mgetty -x0 -s 115200 tyaa11
```

---

The **mgetty** command is hardcoded to use hardware flow control, `-`, which means that a 3-wire connection will not work and that an 8-wire cable is required.

Enable the port for login by rebooting the system or by entering this command at your Linux command prompt:

---

```
init q
```

---

## Configure a device for a printer

You have two options when configuring your printers with Linux drivers.

1. Connect a serial cable between the port and printer.
2. Use either the Linux **printtool** command to configure your printer or set up the printers manually. To install a printer manually, add lines similar to this example to the **/etc/inittab** file:

---

```
DG01:2345:once: cat < /dev/ttyaa11 > /dev/null &
dg01:2345:once: ditty-rp 38400 ctspace altpin -ixon -ixoff -ixany /dev/ttyaa11
```

---

The device `/dev/ttyaa11` is used as an example here. Substitute your particular device name in the previous commands. This example also sets the port speed to `38400`, enables **altpin**, enables hardware flow control (**ctspace** and **rtspace**), and disables software flow control (**-ixon**, **-ixoff**, **-ixany**) on the port. Configure the **ditty-rp** parameters as required by your specific printer and cable configuration. For more information on **ditty-rp**, see [About transparent printing](#).

---

**Note** There is a single bit in the line control register for setting stop bits. If this bit is set with 6, 7, or 8 data bits it gives 2 stop bits. With 5 data bits, however, it actually gives 1.5 stop bits. The driver and firmware simply set this line control register bit if 2 stop bits are requested. So with 5 data bits you actually get 1.5 stop bits, not 2.

---

## Configure a device for a modem

Use this procedure to configure a Digi device for a dial-in/dial-out modem connection. Configuring a device for a modem requires familiarity with both the operating system and the modem being used. The following procedure is sufficient for most cases. It may be necessary, however, to take additional steps to properly configure your modem or to set up the operating system for a specific application.

1. Connect a serial cable between the port and modem.
2. Power the modem on. The example in this section depicts hardware flow control configuration.
3. At the command prompt enter:

---

```
chown uucp:uucp /dev/ttyaa00
```

---

Here `ttyaa00` is the name of the device and **uucp** is an UNIX application.

4. Connect to the modem by entering this command at a command prompt:

---

```
cu -l /dev/ttyaa00 -s 38400
```

---

Here `ttyaa00` is the name of the non modem control device for the port.



5. Set the modem to answer after the first ring with this command:

---

```
ats0=1
```

---

6. Train the modem to the port speed with this command:

---

```
at&w
```

---

7. Enter any other desired modem commands.

8. Terminate the connection to the modem with a tilde followed by a period:

---

```
~.
```

---

9. To manually configure the port to use hardware flow control, insert this command in a Linux startup file:

---

```
ditty-rp rtspace ctspace /device
```

---

Here **/device** is the name of the Digi device. Insert the command in a startup file so it remains in effect after a reboot.

Alternately, use a **gettydef** entry that uses hardware flow control (see the next step)

10. Edit the **/etc/inittab** file and add a **getty** entry for the device. The **getty** name that configures your device can vary from system to system. An example of an **mgetty** modem script, which is available both in RedHat and Debian, is listed below (and only meant to serve as a guide):

---

```
T3:23:respawn: /sbin/mgetty -x0 -s 115200 ttyaa00
```

---

Here *ttyaa00* is the name of the device.

11. Enable the port for login by rebooting the system or by entering this command at your Linux command prompt:

---

```
init q
```

---

## Configuring transparent printers

Use this procedure to set up transparent printers on terminals.

---

**Note** This procedure sets up the communication characteristics for transparent printers. Once a transparent printer has been set up, you can use it as you would a printer connected directly to a serial port. See your UNIX documentation for information on setting up print queues.

---

At the command prompt, enter:

---

```
ditty -n ttname [options]
```

---

Here *ttname* is the name of the terminal device and *options* are selected from the list below.

Option	Description
<b>maxcps</b> <i>n</i>	Limits the maximum printer port character-per-second data rate. <b>n</b> should be set to the minimum character rate the printer can sustain in typical use.
<b>maxchar</b> <i>n</i>	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.
<b>bufsize</b> <i>n</i>	This parameter 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 <b>maxcps</b> rate.
<b>onstr</b> " <i>s</i> "	Defines the terminal escape sequence to direct subsequent data to the transparent printer. <b>s</b> is a string of ASCII characters, enclosed in quotes, that command the terminal to enter transparent printing mode. An arbitrary octal character <i>xxx</i> may be given as <code>\xxx</code> . For example, the sequence " <code>&lt;Esc&gt;[5i</code> " would be entered as: <code>"\033[5i"</code> .
<b>offstr</b> " <i>s</i> "	Defines the terminal escape sequence to stop directing data to the printer. <b>s</b> is a string of ASCII characters, enclosed in quotes, that command the terminal to enter transparent printing mode. An arbitrary octal character <i>xxx</i> may be given as <code>\xxx</code> . For example, the sequence " <code>&lt;Esc&gt;[4i</code> " would be entered as: <code>"\033[4i"</code> .
<b>term</b> <i>t</i>	Sets the transparent printer on/off strings to values found in the internal default table. Internal defaults are used for the following terminal types: adm31, ansi, dg200, dg210, hz1500, mc5, vt100, vt220, vt320, vt420, wang2x36, wyse30, wyse50, wyse60, or wyse75. If the terminal type is not found in the internal default table, then <b>ditty</b> reads the <b>terminfo</b> entry for the terminal type and sets the transparent print on/off strings to the values given by the mc5/mc4 attributes found there.

### **About transparent printing**

Most terminals have an auxiliary port that can be connected to a serial printer. When this port is configured as a transparent printer port, print jobs may be run simultaneously with normal terminal operation.

Data bound for the printer is preceded by a terminal escape sequence which turns on transparent printing. It is followed by a sequence which turns transparent printing off.

Set up a transparent printer in the same way you would set up a printer wired directly to a serial port. Data sent to a transparent printer device is automatically wrapped in the transparent print on/off command strings for the specified printer.

## Setting TTY options

The RealPort Linux device driver package includes a command, **ditty-rp**, which is a superset of **stty**. It may be used to set and display the device options for Digi RealPort devices.

The general command format is:

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

With no options, **ditty-rp** 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 **ditty-rp** commands may be executed from the command line or placed in a startup script to be run whenever the system is booted.

The options are:

Command	Results
<b>-a</b>	Display all of the unique Digi option settings, as well as all of the standard TTY settings reported by <b>stty -a</b> .
<b>-n <i>ttyname</i></b>	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.
<i>ttyname</i>	Set and display options for the specified TTY device. Replace <i>ttyname</i> with the TTY pathname (such as <i>/dev/ttya01s</i> , <i>/dev/term/a01</i> or <i>/dev/dty/a001s</i> , depending on your operating system). 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:

Command	Results
<b>break</b>	Send a 250 MS break signal out on the TTY line.
<b>flush</b>	Immediately flush (discard) TTY input and output.
<b>flushin</b>	Flush TTY input only.
<b>flushout</b>	Flush TTY output only.

The following options specify actions which are not sticky. This means that the changes are cancelled when the device is closed and that the device will use the default values the next time it is opened.

Command	Results
<b>stopout</b>	Stop output exactly as if an XOFF character were received.
<b>startout</b>	Restart stopped output exactly as if an XON character were received.

Command	Results
<b>stopin</b>	Activate flow control to stop input.
<b>startin</b>	Flush TTY output only.
<b>[-]dtr</b>	Raise [drop] the DTR modem control line, unless DTR hardware flow control is selected.
<b>[-]rts</b>	Raise [drop] the RTS modem control line, unless RTS hardware flow control is selected.

The following options are sticky. This means the effects continue until the system is rebooted or until the options are changed.

Command	Results
<b>[-]fastbaud</b>	Alter the baud rate tables to permit the use of data rates that are beyond the range supported by the operating system. See <a href="#">Fastbaud data rate mapping</a> .
<b>[-]rtspace</b>	Enable [disable] RTS hardware input flow control, so RTS drops to pause remote transmission.
<b>[-]ctspace</b>	Enable [disable] CTS hardware output flow control, so local transmission pauses when CTS drops.
<b>[-]dsrpace</b>	Enable [disable] DSR hardware output flow control, so local transmission pauses when DSR drops.
<b>[-]dcdpace</b>	Enable [disable] DCD hardware output flow control, so local transmission pauses when DCD drops.
<b>[-]dtrpace</b>	Enable [disable] DTR hardware input flow control, so DTR drops to pause remote transmission.
<b>[-]forcedcd</b>	Disable [re-enable] carrier sense, so the TTY may be opened and used even when carrier is not present.
<b>startc c</b>	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, <b>&lt;CTRL-Q&gt;</b> , can be entered as "17" (decimal), "021" (octal) or "0x11" (hexadecimal).
<b>stopc c</b>	Sets the XOFF flow control character. The character may be given as a decimal, octal, or hexadecimal number (see <b>startc</b> , above, for format of octal and hexadecimal numbers).
<b>astartc c</b>	Sets auxiliary XON flow control character. The character may be given as a decimal, octal, or hexadecimal number (see <b>startc</b> , above, for format of octal and hexadecimal numbers).
<b>astopc c</b>	Sets auxiliary XOFF flow control character. The character may be given as a decimal, octal, or hexadecimal number (see <b>startc</b> , above, for format of octal and hexadecimal numbers).

Command	Results
<b>[-]aixon</b>	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.
<b>maxcps</b> <i>n</i>	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 is reduced. If the number is too high, the printer resorts to flow control, and user entry on the terminal is impaired correspondingly. Default is 100 CPS.
<b>maxchar</b> <i>n</i>	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.
<b>bufsize</b> <i>n</i>	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 <b>maxcps</b> rate selected above. Default is 100 characters.
<b>onstr</b> " <i>s</i> "	Defines the terminal escape sequence to direct subsequent data to the transparent printer. <i>s</i> is a string of ASCII characters, enclosed in quotes, that command the terminal to enter transparent printing mode. An arbitrary octal character <i>xxx</i> may be given as <code>\xxx</code> . For example, the sequence <code>&lt;Esc&gt;[5i</code> would be entered as <code>"\033[5i"</code> .
<b>offstr</b> " <i>s</i> "	Defines the terminal escape sequence to stop directing data to the printer. <i>s</i> is a string of ASCII characters, enclosed in quotes, that command the terminal to enter transparent printing mode. An arbitrary octal character <i>xxx</i> may be given as <code>\xxx</code> . For example, the sequence <code>&lt;Esc&gt;[5i</code> would be entered as <code>"\033[5i"</code> .
<b>term</b> <i>t</i>	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, pterm, 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 <b>ditty</b> reads the <b>terminfo</b> entry for the terminal type. It also sets transparent print on/off strings to values given by the mc5/mc4 attributes found there.

### Fastbaud data rate mapping

Use the table below to see how setting **fastbaud** affects RealPort data rates.

Specified Data Rate	Data Rate Mapped to
50	57600
75	76800
110	115200

Specified Data Rate	Data Rate Mapped to
134	131657
150	153600
200	230400
300	460800

## Troubleshooting

---

The troubleshooting steps in this section should be done in the following order to try to eliminate your problem.

1. Review [troubleshooting requirements](#).
2. [Make sure device is properly configured for use with RealPort](#).
3. [Verify the configuration and operation of the driver](#).
4. Verify that data is flowing back and forth via the Data In and Data Out counters. See [Test communications](#).
5. Verify data packet timing. See [Advanced configuration](#).
6. Check connection issue. See [Serial port problems](#).
7. [Conduct a loopback test to test ports outside RealPort \(all UNIX and Microsoft Windows operating systems\)](#).

If, after working through these steps, your problem is not solved, try the resources listed below.

- Visit our support knowledge bases at [www.digi.com/support/knowledgebase.jsp](http://www.digi.com/support/knowledgebase.jsp) to look for articles related to your situation.
- Visit our support forums at [www.digi.com/support/forum](http://www.digi.com/support/forum) and search for possible posts from other users with similar situations.
- If the knowledge base or support forums do not have the information you need, fill out an online support request via [www.digi.com/support/eservice/eservicelogin.jsp](http://www.digi.com/support/eservice/eservicelogin.jsp). You must create a new user account.
- You may also email our support staff at [tech.support@digi.com](mailto:tech.support@digi.com).

### Troubleshooting requirements

The following software and equipment is required to troubleshoot RealPort:

- Terminal Emulation software (Hyperterm, TeraTermPro, PuTTY, or similar)
- Loopback plug (supplied with device)

### Make sure device is properly configured for use with RealPort

1. Configure the IP address for the device.
2. Configure the serial ports of the device for use with RealPort.

---

**Note** Most devices support use of the Digi Discovery Utility for discovery on a local LAN. This utility performs initial configuration of the IP address if the device is in the default configuration.

Log into the web interface and set the configuration of the serial ports for use with RealPort. You can use the following web UI path to configure the serial ports for most devices:

**Configuration > Serial Ports > [Select Serial Port] > Choose Profile > RealPort > Apply**

Complete this step for each port that supports RealPort.

---

## Verify the configuration and operation of the driver

Complete the following steps to test RealPort communications:

1. Use the Terminal Emulator to open a connection to port.
2. Use the loopback plug to confirm that data echoes when the loopback is inserted in the correct port.
3. Verify the driver is communicating with the Digi Serial to Network device.

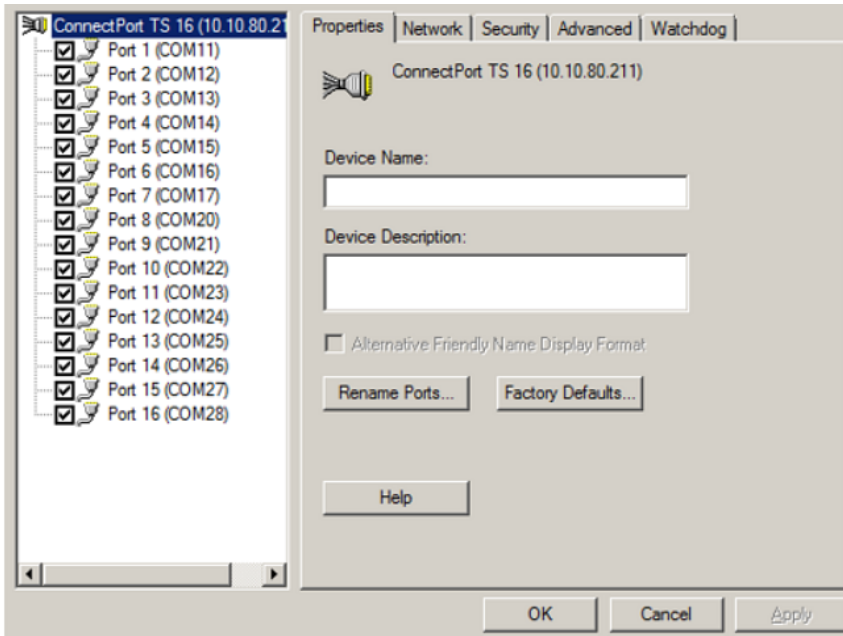
RealPort maps Windows COM ports and Linux TTY ports to physical serial ports on the Digi Serial to Network device. This mapping is defined on the computer as part of the driver installation.

### Verify in Windows

1. Click **Device Manager > Multiport Serial Adapters > Verify IP address of target Digi Serial to Network device.**
2. Double-click on the device.
3. Click the **Advanced** tab.
4. Verify the following:
  - Connection indicates **Connected**.
  - No. of Ports is correct.
5. Click **Properties**. Note the mapping of the computer's COM ports to the physical ports on the device.

In the example, Serial port 1 on the device is controlled by COM11 on the computer.





This information is also available from the command line using the **setup.com/examine** function or from the command line using **cmd** or **PowerShell**. Use **setup.com/help** to show the full command syntax for configuring RealPort via command line.

### Verify in Linux

You can view the configuration in the following directory:

**/etc/dgrp.backing.store**

```
# ID IP PortCount SpeedString IPPort Mode Owner Group Encrypt EncryptPort
#
# If any of the last seven options should use the default, the
# string "default" appears instead.
#
a 192.168.1.123 16 auto default default default default default default
```

**Where the target device is IP address 192.168.1.123 and this device has 16 ports.**

Confirm that the ports exist in the /dev directory:

```
These ports will be created as:
/dev/tty_dgrp_a_0
/dev/tty_dgrp_a_1
...
/dev/tty_dgrp_z_16
```

**Where /dev/tty\_dgrp\_a\_0 will be physical serial port 1 on the device.**

### Test communications

Verify that data is flowing back and forth via the Data In and Data Out counters.

1. Log into the web interface of the device, and select **Administration > System Information > Serial**.
2. Select a serial port.
3. Open the correct COM or TTY port that maps to this port.
4. You can see the signal state for the port change and verify that the following serial settings are reflected in the web interface:
  - Baud rate
  - Flow control
  - Data and stop bits
5. Insert the loopback plug into the port, and type a message. The message echoes back to you, and the Data In and Data Out reflects the number of characters you type.  
If successful, communication is established and the driver is installed correctly.
6. If you have established communication and your device is not communicating with your application, check the cabling and use the web interface to verify data is flowing back and forth via the Data In and Data Out counters.

## Advanced configuration

Use the Advanced configuration of RealPort to help ensure data packet timing.

## Serial port problems

Serial port problems usually involve configuration or connection issues.

### Make sure the port server is configured correctly

---

**Note** This procedure is not applicable for EtherLite, Digi CM devices, or Digi Passport devices.

---

1. Review the port settings by entering the **set port** command:

- For the Portserver TS and Digi One product lines:

```
#> set ports range=2-16 dev=rp
```

- For the Connectport TS/LTS and Digi Connect product lines:

```
#> set profile profile_type=realport port=2-16
```

- Any port that will have a RealPort device attached should be configured as follows:

Port type	Should be set to
<b>dev</b>	<b>rp</b> or <b>prn</b>
<b>auto</b>	<b>off</b>

You can use the Web Interface to verify the device type as well.

- Any port that will have a RealPort device attached should have all signals off except **ixon** and **ixoff** (which can be on or off) and **altpin** (which should be off for a 10 wire cable and on for anything else).

You can use the Web Interface to verify the port setting as well.

2. Enter a **set auth** command

```
#> set auth
```

Results will be similar to the following:

Ind	IPaddr	Mask	RealPort	Login	Unrestricted
1	0.0.0.0	0.0.0.0	0000-000	-----	00000000
			0-000000	-----	00000000

The **Realport** column should show **0** for all RealPort ports. The ports are indicated across in groups of eight dashes (--) In the example above, all ports are set correctly in the RealPort column except ports 5 and 10 (which are possibly being used for non-RealPort connections as indicated by the dash(-) in position 5 and 10). This can only be verified in the command line.

## Check for stuck processes

**Note** This procedure is not applicable for EtherLite, Digi CM devices, or Digi Passport devices.

1. At the root command prompt, type:

---

```
#> who
```

---

If a process (RealPort or otherwise) is running on the port, type:

---

```
#> kill tty=[port_number]
```

---

If this does not work, unplug the serial device and reboot the Digi device.

Check the LEDs on the port to see if OFC is on. This can be monitored using our DPA Remote software available for download on our website. Go to [www.digi.com/support](http://www.digi.com/support), enter your device and operating system, and then search for DPA Remote.

2. If still not working, try typing **control-Q** from a terminal (or PC running terminal emulation software) attached to the port. Otherwise the port will need to be flushed from the operating system level.

---

## Make sure the RealPort host can reach the Digi device

---

**Note** This procedure is not applicable for EtherLite, Digi CM devices, or Digi Passport devices.

---

Verify that there are no processes on the port. Use the **who** command process in [Check for stuck processes](#) to verify. Once the **who** command shows no processes on the port, from the UNIX RealPort host root prompt, type:

---

```
telnet [IP_of_digi_unit] 771
```

---

From a Windows PC, go to **Start > Run** and enter the above command.

If the host cannot **telnet** to the **771** socket, RealPort cannot connect to it. This indicates that the IP address or the RealPort (**771**) socket is in use, is being blocked, or is firewalled. If the connection succeeds, disconnect using **<Ctrl> ] <Enter>**, followed by typing **quit** at the **telnet** prompt.

If you are unable to communicate with the individual ports on the Digi device, first confirm the port dip switches are configured to match your peripheral devices communication protocol for RS-232 (1-up; 2-, 3-, and 4-down), RS-485, or RS-422.

---

## Conduct a loopback test to test ports outside RealPort (all UNIX and Microsoft Windows operating systems)

---

**Note** This procedure is not applicable for EtherLite, Digi passport, or Digi CM devices. Refer to [Test communications](#) for procedure for testing ports within RealPort.

---

1. Plug the RJ-45 loopback plug into the serial port to be tested.
2. From the command prompt (or **Start > Run** in Windows), **telnet** to the IP address of the Digi device using the following command from your UNIX prompt, replacing `###.###.###.###` with the actual IP address of the unit:

```
telnet ###.###.###.###
```

3. Login as root (default password of dbps).

From the Digi device root prompt:

```
#> set line baud=(baud_rate) range=(port_number)
```

```
#> set port dev=prn auto=off range=(port_number)
```

To configure testing at 9600 baud:

```
#> set serial port=(port_number) baud=9600
```

4. Kill any residual processes on the port:

```
#> kill tty=(port_number)
```

5. Connect directly to the port:

```
#> connect (port_number)
```

You will connect directly to the port and see any incoming data from any device attached to the port. By inserting the loopback plug shipped with the Digi product into the appropriate port, you will be able to view any data typed on your **connect** session. Removing the loopback plug will not display data typed on the **connect** session. If the loopback test worked, you have successfully tested the integrity of the port.

6. Close the **connect** session:

```
<Ctrl> [ . <Enter>
```

Type **<ctrl>** [ followed by a period and the Enter key to close the connect session

## Loopback plug pin-out information

More information can be found at [www.digi.com/support](http://www.digi.com/support).

## Trouble accessing port

1. Verify that there are no processes currently running on the port from the Digi device root prompt:

---

```
#> who
```

---

2. If there appear to be processes on the port, the following command will kill non-RealPort processes. If the **Connected To** field shows RealPort, this will need to be killed at the operating system level of the RealPort server/host (see your operating system documentation for information about killing a process).

---

```
#> kill tty=#
```

---

Rebooting the unit can also clear the process.

## Cabling

If the loopback test performed in the previous sections passed but you are unable to communicate to your device, you will need to verify:

- You are using the proper cabling: [www.digi.com/support](http://www.digi.com/support).
- Altpin settings: When **altpin** is enabled, DCD becomes available on pin 1 of an 8-pin RJ-45 connector (set flow **altpin=on range=(port\_number)**).

If you have not resolved your problem at this point, please write down how far you were able to get in the above procedures (and the results) to help describe your problem to Digi support personnel.

## Troubleshoot issues in Windows

RealPort problems in Windows usually involve all or some of the ports not working. Before you begin, make sure device firmware is correctly configured on your product.

### None of the ports work

Follow this process to verify that the Digi Device/Terminal Server is running and configured properly.

1. Verify that the Digi Device/Terminal Server is running the latest firmware version available from the Digi Support site, [www.digi.com/support](http://www.digi.com/support).
2. Verify that the Digi Device/Terminal Server is configured properly. From a command prompt (or **Start > Run**), type:

---

```
telnet [IP_of_digi_unit] 20xx
```

---

Where **20XX** would be replaced by **20+[port number]**. For example, to connect to port 1, type:

---

```
telnet [IP_of_digi_unit] 2001
```

---

3. Insert the loopback plug shipped with the Digi Device/Terminal Server into the port. This plug allows you to view any data being typed. You will not be able to view any data typed if you remove the loopback plug.
  - If the loopback test fails, the Digi Device/Terminal Server is not configured properly.
  - If the loopback test works, the Digi Device/Terminal Server is configured properly.
4. Verify that the RealPort host can reach the Digi Device/Terminal Server. Type the following from a command prompt (or **Start > Run**):

---

```
telnet [IP_of_digi_unit] 771
```

---

A blank window with a cursor should be displayed. If it is not displayed, RealPort cannot connect to the Digi Device/Terminal Server because the socket it uses to communicate (771) is in use, being blocked, or firewalled.

---

**Note** You will not be able to type anything in this window, but two lines || may be displayed. These characters are the driver’s built-in keepalive and can be ignored.

---

5. If you still cannot communicate after verifying that the RealPort host can reach the Digi Device/Terminal Server, try reinstalling the driver. See [Install RealPort on the computer](#). If this does not resolve the problem, see [Troubleshooting](#).

## Some of the ports do not work

Follow this process to verify that the Digi Device/Terminal Server is running and configured properly.

1. Check for active processes that may be running on that port by logging into the Digi Device/Terminal Server via **telnet**.
  - a. To configure testing at 9600 baud:
 

---

```
#> set serial port=(port_number) baud=9600
```

---
  - b. At the root prompt, type:
 

---

```
#> who
```

---
  - c. If there appears to be an active process on the port, try killing it with the following command:
 

---

```
#> kill tty=#
```

---

---

**Note** Rebooting the unit can also clear the process.

---

2. Test the non-working port with **Hyperterminal** by choosing **Start > Accessories > Communications > Hyperterminal**.
3. Type in a name and select an icon of your choice. Click **OK** when finished.

4. In the **Connect To** dialog box, go to the connect using the drop-down list and choose the com port you wish to test. Click **OK** when finished.
5. For testing with the loopback, the default port settings are fine. Click **OK**.
6. Insert the loopback plug, which ships with the Digi Device/Terminal Server, into whichever port you decided to test. You should see a blinking cursor on the main **Hyperterminal** screen. If you get an **Unable to open comX** message, this means the port is already open. You should then make sure all applications that might be accessing the port are closed, or their services have been stopped (faxing applications, RAS, Citrix MetaFrame, etc.).
7. Type **Hello!** on the **Hyperterminal** screen. The characters should appear as you type them. If the loopback test does not pass, see [Troubleshooting](#).
8. Test your serial device on a standard COM port (i.e. COM1/COM2).

If the loopback test passed and the serial device works on a standard COM port (that is, COM1/COM2, verify the following:

- Cabling: See [www.digi.com/support](http://www.digi.com/support).

If you have not resolved your problem at this point, record how far you were able to get in the above procedure—and the results—to describe the problem. See [Troubleshooting](#) for contact information and additional troubleshooting recommendations.

### Key concept: port mapping offsets

The number of local ports on the supported Windows Server host offsets the port number on the Digi device. For example, if you have two local COM ports, Port 1 of the Digi device will be installed by the RealPort software as COM 3. Once the RealPort driver has been installed, you can modify the COM port numbering scheme.

## Troubleshoot port issues in Linux

RealPort problems in Linux usually involve all or some of the ports not working.

### If none of the ports work

Follow this process to verify that the Digi Device/Terminal Server is running and configured properly.

1. Check the status of the RealPort Daemon. From the Linux root prompt, type:

---

```
ps -ef | grep drpd
```

---

You should see an entry for each Digi device similar to the following. Please verify the respective IP address.

---

```
root 2254 1 0 Dec 13 ? 0:03/usr/bin/dgrp-version/daemon/drpd a 192.168.2.2
```

---

There are entries for the IP address and node name. Use only one or the other, not both. Make sure that it is the correct IP address or node name for the particular unit. If using the node name, make sure that the host can **ping** or **telnet** to that node name (**ping nodename**).



- To restart the daemon, type the following from the Linux root prompt using an Xterm capable session:

---

```
/usr/bin/dgrp-version/config/dgrp_gui
```

---

This graphical utility allows you to highlight the desired entry, then to select the **Daemon Menu** item from the top to be able to **check**, **stop**, and **start** RealPort daemons.

- To stop and start the RealPort daemons from a non-graphical interface, use the following commands:

- To stop the RealPort daemon, where *x* represents the ID letter of the unit, type the following from the Linux root prompt:

---

```
/usr/bin/dgrp-version/config/dgrp_cfg_node -v -v stop x
```

---

- To start the daemon, type the following from the Linux root prompt:

---

```
/usr/bin/dgrp-version/config/dgrp_cfg_node -v -v start x ###.###.###.###
```

---

Here *x* represents the unit ID letter and *###.###.###.###* represents the IP address.

## If some of the ports do not work

Check for carrier detect. Check the serial signals by typing **display port range=[port\_number]** from the Device root prompt. For a graphical utility to display port signals, please download and install DPA Remote. Go to [www.digi.com/support](http://www.digi.com/support) and enter your device and operating system then search DPA Remote.

If an 8-pin RJ45 cable is being used and DSR is asserted but not DCD, use the following command:

---

```
#> set flow altpin=on ra=[port_number]
```

---